Agriculture Automation and Control

Manoj Karkee
Qin Zhang *Editors*

Fundamentals of Agricultural and Field Robotics

Springer

Agriculture Automation and Control

Series Editor

Qin Zhang
Biological Systems Engineering
Washington State University
Prosser, WA, USA

The ultimate goal of agricultural research and technology development is to help farmers produce sufficient foods, feeds, fibers, or biofuels while at the same time, minimize the environmental impacts caused by these large scale activities. Automation offers a potential means by which improved productivity, resource optimization, and worker health and safety, can be accomplished. Although research on agricultural automation can be found in the published literature, there lacks a curated source of reference that is devoted to the unique characteristics of the agricultural system. This book series aims to fill the gap by bringing together scientists, engineers, and others working in these areas, and from around the world, to share their success stories and challenges. Individual book volume will have a focused theme and will be guest-edited by researchers/scientists renowned for their work within the respective sub-discipline.

More information about this series at http://www.springer.com/series/15728

Manoj Karkee • Qin Zhang
Editors

Fundamentals of Agricultural and Field Robotics

Springer

Editors
Manoj Karkee
Biological Systems Engineering
Washington State University
Prosser, WA, USA

Qin Zhang
Biological Systems Engineering
Washington State University
Prosser, WA, USA

ISSN 2731-3492 ISSN 2731-3506 (electronic)
Agriculture Automation and Control
ISBN 978-3-030-70399-8 ISBN 978-3-030-70400-1 (eBook)
https://doi.org/10.1007/978-3-030-70400-1

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Where are We Coming From?

When we joined Washington State University (WSU) around 10 years ago, we were faced with the new form of agriculture that we were not used to. Both of us were trained on mechanization and automation technologies for row crops (such as rice, wheat, corn, and soybean) and most of our career experience was also around the same domain. Dr. Manoj Karkee completed his PhD in agricultural engineering and human computer interaction at Iowa State University working on dynamic systems modeling, control, and navigation/guidance of tractor and towed implement systems. He graduated in 2009 and continued researching more in this area before making the move to Washington. Dr. Qin Zhang graduated with PhD in agricultural Engineering from University of Illinois Urbana-Champaign (UIUC) in agricultural automation and worked at Caterpillar Inc. and UIUC for more than 15 years researching and developing various automation technologies for agriculture, including auto-guidance and intelligent field machinery technologies that have now been widely adopted around the world.

In Washington and the Pacific Northwest (PNW) region of the USA, our work revolves around a completely different farming environment. Contrary to a reasonable level of homogeneity we find in the Midwest in terms of major commercial crops, the PNW region presented one of the most diverse forms of agriculture focusing heavily on high-value fruit and vegetable crops, which are major parts of a cluster of crops called specialty crops. Washington produces more than 300 different commercial crops, presenting unique challenges and opportunities for making farming more efficient and sustainable. Each type of crop is grown in comparatively small acreage. In addition, there are many different crop cultivars and cropping systems planted within a given crop type. Using apples as an example, there are a few dozen different cultivars planted in Washington State alone, and these cultivars are planted in many different crop architectures. In addition to variability in the crop architecture, color, geometric (shape, size), and physiological (e.g., surface toughness) parameters of produce also vary widely. These unique combinations of crop

types, cropping systems, and cultivars present unique situations requiring specialized mechanization, automation, and robotic solutions.

Nevertheless, our experience from both row crop and specialty crop agriculture tells us that robotic solutions for all types of agricultural and field applications share a wide range of fundamental theories and principles as well as a fair share of challenges such as difficult field conditions, variable and unstable environment, and biological variability of plant and produce. To address these challenges, as discussed more widely in various chapters of this book, automation and robotics for agricultural and field applications have been in the fore front of research and development in recent years. With the advent of novel, affordable, and more powerful sensing technologies, sensors (e.g., red–green–blue depth) and sensing platforms (e.g., UAVs and ground robots), novel and advanced robotic technologies (e.g., soft robotics), robust machine learning techniques (e.g., deep learning), and increasingly powerful and affordable computational tools (e.g., graphical processing units), we can now envision a world where automating even the most specific/unique field operations (e.g., red raspberry pruning and bundling) is imaginable.

In this context, both public and private (big and small) enterprises around the world are actively involved in research and development of wide variations of robotic technologies for farming and other field applications. As new researchers are attracted to the field every day and as there is an increasing need for training the next generation of workforce for development, operation, and maintenance of smart, robotic technologies for farming and other field applications, a book covering the fundamental principles that can be applicable to a wide swath of applications in various types of agricultural industries was deemed crucial. With this context in the background, this book was conceptualized around 3 years ago, and have been in writing for about 2 years. In this process, we got the full, unconditional support from experts all around the word contributing to the book with their long experience, unparalleled insights, and thoughtful ideas. It was a privilege to read the contributions of 33 professors, researchers, scholars, engineers, scientist, and students from across the globe who are world leaders in their respective fields.

We believe this book fills the gap of a good, comprehensive reference for processors, scientists, engineers, and scholars working actively in robotics, in general, and agricultural and field robotics, in particular. We also believe that this book can provide a great text or primary reference for students who are developing their knowledge and experience in robotics and for early career researchers who are trying to build their research and scholarship programs around agricultural and field robotics. Theories, assumptions, and hypothesis are good starting points and can provide strong motivations. What we just discussed in this paragraph are our assumptions and hypothesis. As we are the scientists who always have doubts on our hypothesis and conduct rigorous research to validate or dis-validate our assumptions and hypothesis, we are now out of control in terms of what we could do differently in the book, and it is up to fellow researchers, engineers, students, and scholars like you to prove us right or wrong in terms of what, if any, values this book brings to you and to the profession.

Organization of the Book

The book has been organized into 3 distinctive parts and 16 chapters. After presenting an introductory discussion on the importance and fundamentals of agricultural and field robotics in Chap. 1, 5 chapters have been presented to describe various sensing and machine vision systems (Part I of the book) as it applies to agricultural and field robotics. Chapter 2 presents color sensing and image processing systems whereas Chap. 3 presents the fundamentals of 3D sensing approaches and systems. Basics on spectral sensing is presented in Chap. 4. Chapters 5 and 6 present various ways crop sensing and scouting can be performed in the field and control environment farming including new research and development efforts in crop phenotyping. Part II of the book, starting with Chap. 7, focuses on mechanisms, dynamics, and control of agricultural and field robotic systems. First, robotic manipulation systems (robotic arms) and their optimization for agricultural applications will be presented in Chap. 7, and end-effector (robotic hand) systems are discussed in Chap. 8. Chapter 9 presents the fundamentals of control techniques with specific focus on robotic harvesting. Chapter 10 presents various aspects of guidance and autosteering systems whereas Chap. 11 describes technologies for in-field sorting of fruit crops and Chap. 12 presents the basics of modeling and simulation techniques for robotic systems. Third and final part of this book focuses on emerging topics in agricultural and field robotics. In this part, advanced learning and classification techniques (Chap. 13) and digital farming techniques such as the Internet of Things (IoT) and big data (Chap. 14) are discussed. Similarly, two additional emerging topics are covered in Chap. 15 (Human-machine interactions) and Chap. 16 (Plant-machine interactions). All these chapters, generally, begin with fundamental concepts and algorithms followed by specific case studies demonstrating the ways the concepts and algorithms are applied to solve specific agricultural and field robotic challenges. Finally, all chapters present a brief summary and concluding thoughts with authors' insights into the topic area covered.

It is to be noted that this book primarily addresses the fundamentals of agricultural robotics, thus most of the examples, case studies, and cited literature are borrowed from agricultural industries, specifically crop production agriculture. Agriculture, being one of the most diverse, variable, uncertain, and biologically driven field production environments, focus on agriculture provided, in our opinion, the best example to discuss the fundamentals of robotics for field applications. The concepts, algorithms, and tools discussed in this book, though the examples come from efforts in crop production systems, are equally applicable to robotics beyond production agriculture, particularly for outdoor and field applications such as those common in animal farming, military, mining, and construction industries.

Summary and Concluding Thoughts

To summarize, automation and robotics is an increasingly important area of research, innovation, development, and commercial adoption in agricultural and field applications. The overall success in developing novel robotic solutions for complex agricultural and field problems requires advancement and innovative integration of various tools, techniques, and concepts including machine vision systems, other sensors and sensing systems, navigation and guidance techniques, modeling, simulation and control methods, manipulation and end-effector technologies, and robust machine learning techniques. This book has been developed to cover some of these important areas of robotics as it applies to field environments. The following are three most important features of the book:

(i) The first book discussing the fundamentals of this emerging technology in agriculture, which is suitable for senior-level undergraduate students and graduate students (as a textbook or reference book) and for researchers, engineers, policy makers, farmers, and other stakeholders as a reference book.
(ii) Use of a systematic approach to discuss the fundamentals of automation and robotics as it relates to agricultural and field applications supported by unique and emerging examples from cutting-edge research and development programs around the world.
(iii) The book has presented basic principles of generic concepts and technologies in agricultural robotics that is applicable to all areas of agricultural and field operations, including field crops, special crops, and green house and vertical farming, among others.

Last but not least, as we are drafting this preface, we are in the most unprecedented time of our generation, the COVID-19 pandemic. We are under complete lockdown (about two-thirds of the world population is in the same condition), and most of the latest editing and final polishing of this book occurred at small corners of our homes. This pandemic has reminded us how helpless we are, as individuals, under the vast, mighty force of nature. We hope, by the time this book comes out, we will be in a much better situation in relation to COVID-19. We also hope, however, that this pandemic is a timely reminder for us to play our roles on living a life that maintains a harmony with nature and a life that strives to optimize resource utilization for sustainable civilization. We would feel proud if this book, directly or indirectly, helps make the tiniest of impacts on advancing agricultural and field operations to be more efficient in conserving crucial natural resources.

Prosser, WA, USA

Manoj Karkee
Qin Zhang

Contents

Contributors

Geng Bai University of Nebraska Lincoln, Lincoln, NE, USA

Harjatin Baweja Carnegie Mellon University, Pittsburgh, PA, USA

Santosh Bhusal Harvest Croo Robotics, Tampa, FL, USA

Jose Blasco The Valencian Institute of Agricultural Research, Valencia, Spain

Abhilash K. Chandel Washington State University, Prosser, WA, USA

Joseph Davidson Oregon State University, Corvallis, OR, USA

Qingchun Feng Beijing Research Center of Intelligent Equipment for Agriculture, National Research Center of Intelligent Equipment for Agriculture, Beijing, China

Jingyao Gai Iowa State University, Ames, IA, USA

Zongmei Gao Washington State University, Prosser, WA, USA

Yufeng Ge University of Nebraska Lincoln, Lincoln, NE, USA

Hans W. Griepentrog University of Hohenheim, Stuttgart, Germany

Cameron Hohimer Harvard University, Cambridge, MA, USA

Manoj Karkee Washington State University, Prosser, WA, USA

Lav R. Khot Washington State University, Prosser, WA, USA

Won Suk Lee University of Florida, Gainesville, FL, USA

Renfu Lu U.S. Department of Agriculture, Agricultural Research Service, East Lansing, MI, USA

Danny Mann University of Manitoba, Winnipeg, MB, Canada

Siddhartha Mehta University of Florida, Gainesville, FL, USA

Changki Mo Washington State University Tri-Cities, Richland, WA, USA

Timo Oksanen Technical University of Munich, München, Germany

Dimitrios S. Paraforos University of Hohenheim, Stuttgart, Germany

Tanvir Prahar Carnegie Mellon University, Pittsburgh, PA, USA

Francisco Rovira-Más Universitat Politècnica de València, Valencia, Spain

Maciej Rysz University of Florida, Gainesville, FL, USA

Verónica Saiz-Rubio Universitat Politècnica de València, Valencia, Spain

Abhisesh Silwal Carnegie Mellon University, Pittsburgh, PA, USA

Rajeev Sinha Washington State University, Prosser, WA, USA

Riikka Soitinaho Technical University of Munich, München, Germany

Brian L. Steward Iowa State University, Ames, IA, USA

Lie Tang Iowa State University, Ames, IA, USA

Mehari Z. Tekeste Iowa State University, Ames, IA, USA

Matthew D. Whiting Washington State University, Prosser, WA, USA

Qin Zhang Washington State University, Prosser, WA, USA

Xin Zhang Washington State University, Prosser, WA, USA

Zhao Zhang North Dakota State University, Fargo, ND, USA

About the Editors

Manoj Karkee Dr. Manoj Karkee is an Associate Professor in the Biological Systems Engineering Department at Washington State University (WSU) and is the co-director of WSU-University of Technology Sydney joint center on #AgRobotics. He was born and raised in eastern hills of Nepal. He received his diploma in Civil Engineering and undergraduate in Computer Engineering from Tribhuvan University, Nepal. His MS was in remote sensing and GIS from Asian Institute of Technology, Thailand and his PhD was in Agricultural Engineering and Human Computer Interaction from Iowa State University. Dr. Karkee leads a strong research program in the area of sensing, machine vision and #AgRobotics at the WSU Center for Precision and Automated Agricultural Systems (CPAAS). He has published more than 65 peer-reviewed journal articles, and more than 25 referred conference papers. He has also been awarded two US patents, and has published one book (edited) and eight book chapters. Dr. Karkee is currently serving as an elected chair for International Federation of Automatic Control (IFAC) Technical Committee on *'Control in Agriculture'*, as an associate editor for *'Computers and Electronics in Agriculture' and 'Transactions of the ASABE'*, and as a guest editor for *'Sensors'* Dr. Karkee was recognized as *'2019 Pioneer in Artificial Intelligence and IoT'* by Connected World magazine and was featured as the *'Western Innovator'* by Capital Press. More about his research and scholarship can be found at https://labs.wsu.edu/karkee-ag-robotics/.

Qin Zhang Dr. Qin Zhang is the Director of the Center for Precision and Automated Agricultural Systems (CPAAS) of Washington State University (WSU), and a Professor of Agricultural Automation in the Department of Biological Systems Engineering, WSU. His research interests are in the areas of agricultural automation, agricultural robotics, and off-road equipment mechatronics. Prior to his current position, he was a faculty member at the University of Illinois at Urbana-Champaign, worked at Caterpillar Inc., and taught at Zhejiang Agricultural University in China. Based on his research outcomes, he has authored/edited seven books, written more than a dozen separate book chapters, edited three conference

proceedings, published over 180 peer reviewed journal articles, and been awarded 11 U.S. patents. He is currently serving as the Editor-in-Chief for *Computers and Electronics in Agriculture*. Dr. Qin Zhang received his B.S. degree in engineering from Zhejiang Agricultural University, China; M.S. degree from the University of Idaho and Ph.D. degree from the University of Illinois at Urbana-Champaign. Dr. Qin Zhang is a member of Washington State Academy of Science and an ASABE Fellow and is serving or served as a guest or an adjunct professor for 9 other universities.

Chapter 1
Agricultural and Field Robotics: An Introduction

Qin Zhang and Manoj Karkee

1.1 Background

The primary purpose of agriculture is to produce sufficient high-quality food for human being to sustain and enhance life. Commonly accepted population growth models predict that there will be more than nine billion people by 2050 in the world, and the increasing population will significantly increase the demand for food, fiber, and fuel. People have historically improved and kept evolving farming technologies to meet the needs for feeding continuously growing human population by increasing productivity and production efficiency and enhancing food safety and nutrition while protecting the environment and conserving natural resources. One big challenge the agricultural industry of the United States (and so do many other countries) facing today is the shortage of human labors to conduct field operations, and the trend is expected to continue and become even worse.

One solution to address the field labor shortage challenge is the adoption of mechanized and automated farming technologies. Over the past century, mechanization technologies have made revolutionary changes in field crop production, making it possible to achieve high yields using minimal farm labor. Attributed to its great impact to societal advancement, agricultural mechanization was recognized as the seventh greatest engineering achievements of the twentieth century by the National Academy of Engineering of the United States. Continuing up on this success, mechanized farming has been advancing through adoption of increased level of automation and intelligence to further improve the precision management of crops (including input resources), increase productivity, and reduce farm labor dependency in field operations beyond what has been possible with conventional mechanization technologies. For example, farmers have widely adopted

Q. Zhang (✉) · M. Karkee
Washington State University, Prosser, WA, USA
e-mail: qinzhang@wsu.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_1

auto-steering technology commercialized early this century for many different field operations including tilling, planting, chemical application, and harvesting (Erickson 2019). Automated thinning and precise weeding in vegetable and other crops are other technologies that have recently been commercialized. Mechanization and automation/robotics have played similar roles in other field applications such as those common in construction, mining, and military industries. For agricultural and other field machinery to be capable of performing those automated field operations, machinery needs hold the abilities of (i) being aware of actual operation condition, (ii) determining corrections suitable for changed conditions, and (iii) implementing the corrections during field operation.

These three basic abilities required for automated or intelligent agricultural/field machinery actually are the same as those needed for robots which include the capabilities of (i) perceiving the situation of an operation with surrounding conditions, (ii) making appropriate decisions for smartly performing the operation under the condition, and (iii) automatically implementing the desired operation. Such similarity between desired abilities of intelligent agricultural and field machinery and robots makes a logical sense to call such machinery robotic machines. It implies that agricultural and field robots do not have to be in a form of human-like machines in appearance, but keep their conventional configuration for most effective, efficient, and robust field operations. Such a definition allows us to inherit the accomplishments of century-long development of agricultural and field machinery technology in creating robots for various agricultural and other field applications.

1.2 Fundamental Technologies for Agricultural and Field Robotics

1.2.1 Sensing and Situation Awareness

As mentioned earlier, the first capability of a robotic machine has to possess is its ability to perceive an awareness of the operational situation, which is acquired using sensors and/or sensing systems integrated with those machines. As a machine designed to mimic humans performing various tasks, ideally a robot should possess all the "senses" of human being, namely, vision, hearing, feel/touch, smell, and taste. However, robotic agricultural/field machines are designed to perform some specific tasks in some specific operational sites/conditions, and therefore, they often are not needed to have the full ability to sense to gain the needed awareness of the situation to conduct appropriate operations. In a wide range of agricultural and field applications, the ability to see is often sufficient, and therefore, visual sensing plays a critical role in many robotic machines designed and developed for agricultural and other fields.

The first fundamental sensing function requested by any mobile robotic machinery is the capability to gain an awareness of its surrounding and find its ways to

move on the desired paths to perform the designated operations. Composed of image acquisition hardware and image processing software to automatically inspect the environment and objects of interest based on the visual characteristics, computer vision could provide the required capability (Reid et al. 2000). One of the widely used and simplest computer vision systems could probably be the monocular vision. Similar to one eye vision of human being, a monocular vision is capable of providing a two-dimensional (2D) visual perception on the relative positions of objects of interest within a field of view. This technique has found its application in detecting a guidance directrix on crop rows or the edges along harvested crops to guide robotic machinery performing different operations in row crop fields (Rovira Más et al. 2005). Various methodologies of image processing have been developed for extracting the guidance information for providing a steering signal for navigating the mobile robotic machinery (Reid et al. 2000). More discussion on various techniques used to acquire and process images can be found in Chap. 2. Although the 2D image approach is computationally efficient, it lacks reliable means to locate the actual position of an object of interest. From this point of view, binocular vision can be used to obtain a stereo view (3D image) of the scene and therefore to provide more robust perception on perspective pathways because of its capability of providing depth information. Furthermore, stereovision has two important advantages for navigating robotic machinery: (i) moderate insensitivity to shadows and changes in lighting conditions and (ii) its capability to provide useful state information to the tracking phase as the localization of potential obstacles in front of unmanned vehicles (Rovira Más et al. 2009).

Because agricultural and field robots are used on outdoor natural environment, it requires that surrounding awareness sensing on these machines need to have high robustness, high reliability, acceptable accuracy, high mechanical and temperature stability, and low cost. There are a few other sensing methods, such as global positioning systems (GPS), and laser scanning/LIDAR systems, which are widely used (stand-alone or in combination with other sensors using sensor fusion techniques) on robotic agricultural/field machinery to provide reliable positioning information for navigating such machinery operating in the field autonomously. Detailed description on stereovision and other 3D sensing techniques and systems can be found in Chap. 3.

Other than providing navigation information, vision sensors are also used in detecting other characteristics of object of interest either to support robotic operations or to scout crop growth/health conditions. To provide such functionalities, machine vision-based sensing techniques use different types of modalities to acquire appropriate information. Standard imaging sensors can be used to detect monochrome or color responses for determining physical properties of object of interest, such as relative location, shape, and/or size; and spectral imaging sensors can be used to detect responses in various bands of spectrum for measuring biological properties of plants, such as nutrient, water, and disease stresses of a plant. A few examples of standard image sensing applications (other than navigation) include weed detection for robotic weed control (Blasco et al. 2002), branch detection for robotic pruning of apple trees (Karkee et al. 2014), and apple detection for robotic

picking (Silwal et al. 2016). Some examples of spectral image sensing include crop nitrogen stress detection (Kim et al. 2000; Noh et al. 2006), soybean disease detection (Cui et al. 2010), and blueberry fruit maturity detection (Yang et al. 2014). More discussion on spectral sensing techniques can be found in Chap. 4 and their applications on crop scouting can be found in Chap. 5.

One specific application of various sensing capabilities of a robot would be in automated phenotyping, which aims at performing high-throughput screening of genotypes for more effective breeding selection of crops and has gain high attention in recent years. As plant phenotyping attempts to measure plant growth, architecture, and composition of organs to canopies with a certain degree of accuracy and precision at different scales, both standard and spectral imaging methods have found their applications in measuring phenotyping parameters (Li et al. 2014). The commonly measured plant phenotype parameters include plant architectural data, such as plant height, stem diameter, color, leaf area, and leaf angle, and abiotic stress, such as drought and salinity adaptation, disease resistance, and yield (Berger et al. 2010; Arvidsson et al. 2011). One advantage of image-based sensing is its ability to acquire high-resolution data, which allows an ability to analyze and visualize plants/objects often using multidimensional, multiparameter, or sometimes multispectral information. Therefore, imaging sensors have been increasingly used to quantify plant phenotyping parameters both in controlled environments and in open fields (Walter et al. 2012; White et al. 2012; Sankaran et al. 2018). More discussion on various sensing systems and their applications in precision agriculture and plant phenotyping can be found in Chap. 6.

In addition to visual sensing (and spectral imaging, as an extension of visual sensing), some other sensing methods can also be used to mimic human's capability to gain a comprehensive situation awareness to support robotic agricultural/field machinery performing some specific tasks. A few examples include the use of electronic nose or electronic tongue to determine the maturity or quality of some produces based on their smell or taste (Gómez et al. 2006; Ulloa et al. 2013) and the use of acoustic sensor to measure the canopy density in orchards and vineyards in terms of reflectance of ultrasonic sound (Palleja and Landers 2015). Many operations in agricultural production require some interaction between the machinery and the crops or animals which are often very sensitive to magnitude of mechanical impacts, and some types of touching force/impact sensing could also be necessary for some situations.

It needs to be pointed out that other than situation awareness sensing, robotic machinery may require to have other types of sensors for measuring operational parameters to achieve accurate controls of automated implementation. For example, an effective robotic apple picking may require the robotic machine to be equipped with position, speed, and/or force sensors on its manipulators and end-effectors for controlling the picking actuator quickly and accurately in reaching the target fruit and effectively removing it from the tree.

1.2.2 Intelligent Decision-Making

After obtaining the required perception ability, an essential ability to distinguish robotic machinery from conventional ones is its ability in making intelligent operational decisions in response to the perceived operation situations. One of the major challenges in making robotic agricultural/field machinery work properly and autonomously in different kinds of field is mostly caused by the high level of randomness in both the biological and physical properties, wide variations in geometric features of task/target targets, as well as the high level of uncertainty in operation conditions. To overcome these challenges, it demands the robotic machinery possesses the ability in making intelligent decision in terms of detected operation situation. One approach is to combine human workers and robots synergistically and allow the robot to mimic human experts' making intelligent operational decision by using examples from human experts making the decision in similar scenarios. One example of this approach is a farmer-assisted fertilizing robot developed by Vakilian and Massah (2017) for precise nitrogen management in greenhouse crops. Such an approach requires the robot to detect the operation scenario using onboard sensors, such as using a visual sensor to acquire textural features indicating crop growth condition, and check the detected indicators against a set of reference scenarios. After a matching scenario is found, the robot will then apply an adequate rate of fertilization similar to how a human worker will do for this reference scenario. This approach requires the availability of a set of reference scenarios with human workers' reaction for the case in similar scenarios. Another approach is to separate the sensing system from the robot (Zion et al. 2014), which is frequently proposed especially for harvesting robots. By mapping the harvesting targets in the field using an adequate coordinating system prior to harvest, the robot could reach a bank of targets according to the recorded coordinates. It could speed up the operation substantially as the sensing method is no longer a limiting factor to draw the robotic harvesting efficiency down, which is one of the major remaining challenges for agricultural and field robots.

In many field operations in agricultural production (and other similar applications), it is often desirable that the robotic machinery possesses substantial level of intelligence to work properly and effectively under highly uncertain and changing operation conditions. One way to solve such a problem is via learning from sample data. As a computational method that involves progressively improving the performance on a specific task through data-based learning, machine learning (ML) algorithms have been adopted in supporting decision-making on robotic machinery to avoid or minimize using explicitly developed programs or models. A few examples include the naïve Bayes (NB), k-mean clustering, support vector machines (SVMs), and k-Nearest Neighbor (kNN)-based ML algorithms (Rehman et al. 2019). Deep learning (DL), a class of extended classical machine learning methods created by adding more "depth" (or the complexity), is one of the newest and most robust ML techniques that enhances the capability of automated feature extraction from raw data (Kamilaris and Prenafeta-Boldú 2018). Therefore, DL is suitable for solving

complicated intelligent control problems in agricultural and field applications. The advancement of machine learning (including DL) technologies has been and will continue to offer more useful tools in making intelligent operational decisions for agricultural and field robots.

1.3 Challenges and Opportunities

The main function of agricultural and field robots is to perform designated tasks automatically or autonomously on designated production/operation sites. As mentioned previously, tremendous progress has been made in the past century in developing and adopting mechanization technologies for agricultural and other field operations. Many modern agricultural machines that have been widely used in various production operations today were matured from decades of continuing improvement for achieving the best possible performance on doing the specific tasks. Similar improvements have been made in other field operations over the last several decades. These matured machines and machinery systems provide a rich resource and a strong foundation for developing actuation technologies for many agricultural and field robots.

There are still challenges for creating capable and effective robotic actuation technologies for doing the work today's agricultural and field machinery are not able to perform or could not effectively perform due to their low levels of intelligence. For example, the production of high-value specialty crops, which the US Department of Agriculture (USDA) defines as fruits and vegetables, tree nuts, dried fruits, horticulture, and nursery crops, is still largely dependent on manual labor. This dependence is mainly attributed to the lack of mature mechanization/automation technologies for various field operations such as fresh fruit and vegetable harvesting, tree training and pruning, crop pollination and thinning, and weed control, among others (Davidson et al. 2016). As a specific application example of robotic fruit harvesting, the key bottlenecks for commercial development of such a machinery are sensitivity of the produce quality to mechanical impact during harvesting and the extensive variability that exists in the unstructured orchard environment. The actuation technology (including the manipulator and end-effector) optimization for specialty crop harvesting applications is still an active area of research (Sivaraman and Burks 2007; Van Henten et al. 2009; Lehnert et al. 2015; Chap. 7 of this book). Hoeing actuators for intra-row mechanical weeding (Gobor et al. 2013) and string tying actuators for hops' production (He et al. 2012) are two examples of other actuation technologies need to be developed especially for robotic applications in the agricultural and field environments as there are either no existing mechanical devices available for performing the work or the existing devices are inadequate for performing robotic operations.

Powered by the recent technological advancement in machine learning, sensing and data processing techniques, as well as parallel computing, agricultural and field robots have never been so close to be practically used in field for commercial

productions/operations. It creates an urgency for starting a new study on management of robotic field operations, consisting of robotic equipment selection, efficient utilization in the field, and optimization for economic returns.

Agricultural and field robots are developed, to a large extent, for solving the challenges of the human field labor shortage and improve worker health and safety, and these robotic machines have to be uniquely designed, normally for specific field operations. It forms an important feature to distinguish agricultural and field applications of robots from industrial operation: while industry applications could adopt robotic operation to a few selective tasks in a factory, agricultural applications would make sense only if the entire operation were robotized to solve the challenge of field labor shortage. Another basic consideration in equipment selection is proper sizing of robots. Proper sizing of robots for every field operation to optimally match their capacities plays an essential role for achieving productive, efficient, and profitable operations. Coordination strategies and control of multiple robots also play a critical role for productive, efficient, and safe operations.

One critical obstacle, specifically for agricultural applications, for effective utilization of robotic equipment is the insufficient skills of farmers to effectively manage, operate, and supervise robots as they could in using conventional machinery. Effective robot managing and supervision may include work-plan creation for individual robot and the entire robot fleet, system initialization, the operation-specific data/information management and utilization, and the override control for abnormal conditions, and many of those tasks require human-robot interaction. Such tasks normally require special skills to perform and are often beyond ordinary farmer's capability to manage. One possible solution for such a skill challenge might be professional services, either through robot management and maintenance services or through robotic operation services. The former is to provide technical support to help end users (e.g., farmers) managing and operating their robotic equipment, and the latter is to provide custom robotic field operation services for the end users.

1.3.1 Economics: A Critical Dimension

Like in any other commercial operations, economic performance of robotic farming (and other field operations) is one of the most important measures in robotic field operation management. An ideal robotic farming system, for example, should be able to perform a most productive operation at the lowest total cost. As agricultural production is often measured by the yield, one way to measure economic performance is by the total cost per unit of yield. The total costs for a robotic production should include the initial costs, the operational and maintenance costs, and the error (e.g., crop damage) costs. The initial costs are one-time expenses for purchasing, delivery, and maybe initial integration and calibration when applicable of the robotic equipment. As the lifespan of a robot is usually over multiple years (production seasons), it could be divided into per season costs in assessing its economic performance. The operational costs are more complicated to determine and can be

calculated in annual (or per season) basis. This category of costs should include service costs (such as maintenance and repair costs), material costs (such as fuels and other applicable materials), and labor costs (such as for monitoring and supervision). The use of robot in agricultural production does not mean to completely eliminate human labor in the operation, but to replace human labors from tedious field work by using less but skillful operators to manage and supervise the robotic operation. The error cost is relatively difficult to calculate as there is no sufficient information supporting the estimation of what error in robotic operation would cause what yield reduction, but impact heavily to the overall economic performance of robotic production as any error could result in a substantial yield loss and the entire economic performance is based on the yield.

1.4 Concluding Thoughts

Manual operations in agriculture and other field environments are challenging: they are not only labor intensive but are laborious and pose health and safety risks. In the twenty-first century of social and technological development, people deserve and have the potential to move away from performing back-breaking and risky work, such as climbing up and down tall ladders with heavy load of fruit (e.g., 15 Kg) in manual tree fruit harvesting, by using robotic machinery. As discussed before, tremendous progress has been made over the last century in agriculture in developing and adopting mechanization and automation technologies to minimizing farming inputs such as fertilizer, water, and labor while improving crop yield and quality. Similar progresses have also been made in other relevant (field) industries such as construction and mining. Using the foundation provided by these matured machinery systems and through the integration of advanced tools and technologies, reliable, robust, and affordable robotic technologies could be developed for these industries. With the recent advancement in AI techniques such as deep learning, ever-increasing capability and decreasing cost of computational technology (including parallel computing), powerful but affordable sensing systems such as hyperspectral imagers and novel robotic solutions such as soft robotics, we can now envision a world where all labor-intensive and laborious farming/field operations are performed by autonomous machines. In this way of farming, we believe, the role of human workers will be to operate, collaborate with, supervise, and/or troubleshoot these machines (based on the nature of the autonomous machine) remotely from off-site offices. The future of farming, what we also call *smart farming* or *Ag 4.0*, we believe, will also see widespread adoption of qualitative decision-making in farming by intelligent machines using AI, IoT, and big data analytics (and in collaboration with human experts). What the Prime Minister of Canada, Justine Trudeau, recently said about general technological developments holds true in agriculture (and related industries) as well: “we have never seen this rapid advancement in agricultural technologies in the past and we will never be this slow again.”

References

Arvidsson S, Pérez-Rodríguez P, Mueller-Roeber B (2011) A growth phenotyping pipeline for arabidopsis thaliana integrating image analysis and rosette area modeling for robust quantification of genotype effects. New Phytol 191:895–907

Berger B, Parent B, Tester M (2010) High-throughput shoot imaging to study drought responses. J Exp Bot 61:3519–3528

Blasco J, Aleixos N, Roger J, Rabatel G, Moltó E (2002) Robotic weed control using machine vision. Biosyst Eng 83(2):149–157

Cui D, Zhang Q, Li M, Hartman G, Zhao Y (2010) Image processing methods for quantitatively detecting soybean rust from multispectral images. Biosyst Eng 107(3):186–193

Davidson J, Silwal A, Karkee M, Mo C, Zhang Q (2016) Hand-picking dynamic analysis for undersensed robotic apple harvesting. Trans ASABE 59(4):745–758

Erickson B (2019) CropLife-Purdue University precision agriculture dealership survey. Retrieved from https://www.croplife.com/management/2019-precision-agriculture-dealership-survey-more-moves-toward-decision-agriculture/. Accessed 20 Apr 2020

Gobor Z, Lammers P, Martinov M (2013) Development of a mechatronic in-row weeding system with rotational hoeing tools: theoretical approach and simulation. Comput Electron Agric 98:166–174

Gómez A, Hu G, Wang J, Peraira A (2006) Evaluation of tomato maturity by electronic nose. Comput Electron Agric 54(1):44–52

He L, Zhang Q, Du X, Luo R, Karkee M (2012) A twining robot for high-trellis string tying in hops production. Trans ASABE 55(5):1667–1673

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

Karkee M, Adhikari B, Amatya S, Zhang Q (2014) Identification of pruning branches in tall spindle apple trees for automated pruning. Comput Electron Agric 103(2014):127–135

Kim Y, Reid J, Hansen A, Zhang Q (2000) On-field crop stress detection system using multispectral imaging sensor. Agric Biosyst Eng 1:88–94

Lehnert C, Perez T, McCool C (2015) Optimisation-based design of a manipulator for harvesting capsicum. Workshop on robotics in agriculture at the international conference on robotics and agriculture (ICRA), pp 1–4

Li L, Zhang Q, Huang D (2014) A review of imaging technique for plant phenotyping. Sensors 14:20078–20111

Noh H, Zhang Q, Shin B, Han S, Feng L (2006) A neural network model of maize crop nitrogen stress assessment for a multispectral imaging sensor. Biosyst Eng 94:477–485

Palleja T, Landers A (2015) Real time canopy density estimation using ultrasonic envelope signals in the orchard and vineyard. Comput Electron Agric 115:108–117

Rehman T, Mahmud M, Chang Y, Jin J, Shin J (2019) Current and future applications of statistical machine learning algorithms for agricultural machine vision systems. Comput Electron Agric 156:585–605

Reid J, Zhang Q, Noguchi N, Dickson M (2000) Agricultural automatic guidance research in North America. Comput Electronics Agric 25:155–167

Rovira Más F, Zhang Q, Reid J, Will J (2005) Visual crop row detection algorithm for automated guidance. Proc Inst Mech Eng D J Automobile Eng 219:999–1010

Rovira Más F, Wang Q, Zhang Q (2009) Bifocal stereoscopic vision for intelligent vehicles. Int J Vehic Technol:123231

Sankaran S, Zhou J, Khot L, Trapp J, Mndolwa E, Miklas P (2018) High-throughput field phenotyping in dry bean using small unmanned aerial vehicle based multispectral imagery. Comput Electron Agric 151:84–92

Silwal A, Karkee M, Zhang Q (2016) A hierarchical approach to apple identification for robotic harvesting. Trans ASABE 59(5):1079–1086

Sivaraman B, Burks T (2007) Robot manipulator for citrus harvesting: configuration selection. ASABE Annual International Meeting

Ulloa P, Guerra R, Cavaco A, Rosa da Costa A, Figueira A, Brigas A (2013) Determination of the botanical origin of honey by sensor fusion of impedance e-tongue and optical spectroscopy. Comput Electron Agric 94:1–11

Vakilian K, Massah J (2017) A farmer-assistant robot for nitrogen fertilizing management of greenhouse crops. Comput Electron Agric 139:153–163

Van Henten E, Van't Slot D, Hol C, Van Willigenburg L (2009) Optimal manipulator design for a cucumber harvesting robot. Comput Electron Agric 65(2):247–257

Walter A, Studer B, Kölliker R (2012) Advanced phenotyping offers opportunities for improved breeding of forage and turf species. Ann Bot 110:1271–1279

White J, Andrade-Sanchez P, Gore M, Bronson K, Coffelt T, Conley M, Feldmann K, French A, Heun J, Hunsaker D (2012) Field-based phenomics for plant genetics research. Field Crops Res 133:101–112

Yang C, Lee W, Gader P (2014) Hyperspectral band selection for detecting different blueberry fruit maturity stages. Comput Electron Agric 109:23–31

Zion B, Mann M, Levin D, Shilo A, Rubinstein D, Shmulevich I (2014) Harvest-order planning for a multiarm robotic harvester. Comput Electron Agric 103:75–81

Part I
Sensing and Machine Vision

Chapter 2
Sensors I: Color Imaging and Basics of Image Processing

Won Suk Lee and Jose Blasco

2.1 Introduction

The human eye is geared by nature to sense the difference between colors. In nature, the perceived color is mainly determined by the different types of pigments present in plants, such as chlorophylls, carotenes, xanthophylls, and anthocyanins, that offer information on the type and status of plants and their fruits. This is very important, for example, for harvesting robots or those that act according to the state of the plants. Likewise, color allows differentiating structural elements of the scene and obtaining information from the environment that is essential, for example, in autonomous guidance systems. Color cameras are the most widely used devices in artificial vision because they produce images similar to those perceived by the human eye, and are therefore widely used to automate agricultural operations in a framework of precision agriculture (Cubero et al., 2016). The acquisition technology of these images is very advanced, and there are also numerous techniques to analyze and obtain information from this type of images. To obtain good results, it is very important to acquire high-quality images. Therefore, the selection of the cameras and the lighting conditions for the images are very important, especially in field conditions where the images are poorly structured and the lighting conditions are changing. Subsequently, it is necessary to follow a series of basic steps in the image analysis. First, a preprocessing is necessary to improve the image and eliminate noise pixels to achieve faster and more efficient subsequent processing, followed by a segmentation operation to obtain the regions of interest. Finally, a feature

W. S. Lee (✉)
University of Florida, Gainesville, FL, USA
e-mail: wslee@ufl.edu

J. Blasco
The Valencian Institute of Agricultural Research, Valencia, Spain

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_2

extraction is required to obtain the desired information. For any of these tasks, it is essential to develop efficient, robust, and accurate processing algorithms.

This chapter is an overview of the main topics related to the basics of color imaging and image processing operations applied to robotics in agriculture. Due the limited scope of this chapter, readers are encouraged to read reference books to get into details of image processing, such as Gonzalez and Woods (2018) and Russ and Neal (2017).

2.2 Basics of Color Imaging

The spectrum visible to humans goes from violet light to red light (Fig. 2.1). When light strikes an object, it absorbs part of the light and reflects the rest, which is perceived by the human eye through the retina. The retina contains two different types of light-sensing photoreceptor cells, rods and cones. The rods are activated in low light conditions, while the cones usually contain three types of pigments that are sensitive to wavelengths of light corresponding to the colors red, green, and blue. Therefore, all the colors that humans can recognize are a combination of these primary colors (Goldstein 2010). In order to determine, measure and compare colors, precise methods are needed to represent the colors with unique values.

2.2.1 Color Representation

The objective of a color model is to facilitate the expression of the colors in a standardized way. In general, a color model is the mathematical description of a coordinate system and a particular space (color space) in which each color is represented only by a single point (Ibraheem et al. 2012). Color models are used to describe the

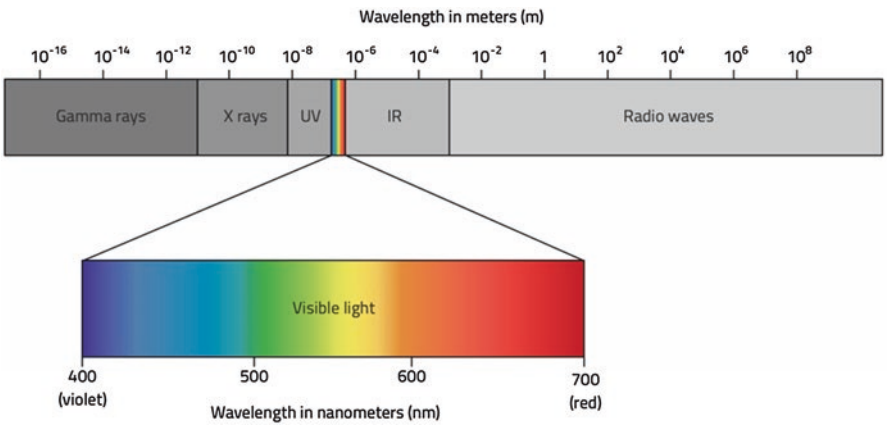


Fig. 2.1 Electromagnetic spectrum and the visible light

colors of digital images. In addition, a color space is a particular implementation of a color model that has a specific range of colors (Hastings and Rubin 2012). For example, in the RGB color model, there are different color spaces, such as Adobe RGB and sRGB. Different devices (for example digital cameras), due to the electronics and/or the software implemented on them, have their own color spaces and, therefore, can capture or represent only the colors within their range (Ford and Roberts 1998). There are a number of color spaces in common usage depending on the particular industry and/or application involved. For example, as humans we normally determine color by parameters such as brightness, hue, and colorfulness. On computers it is more common to describe color by three components, normally red, green, and blue. There are many different color spaces used in practice and each one represents a different method to describe the colors. Some of the most known models are briefly described below.

RGB Model

Red (R), green (G), and blue (B) or RGB model is the most widely used model in digital devices. It is based on an additive mixing model, where each color is formed by a combination of the three primary colors: red, green, and blue. The spatial representation is through a cube where each side measures 1 and each axis represents one of the three primary (RGB) color coordinates. In this model, the black color is represented at point (0,0,0) and the white color at (1,1,1). Figure 2.2b shows the distribution of the colors of the images in Fig. 2.2a, using the RGB color model.

A normalized variant of this model is defined as rgb, which is derived by dividing the RGB values by (R + G + B). As this color space is native for electronic devices, the RGB coordinates are commonly used in vegetation indices to assess different properties of the crops using remote sensing techniques (Meyer and Neto 2008).

HSV and HLS Models

These models were designed to be more easily understandable and interpretable since they use parameters more related with the perception of the color, such as hue, saturation, lightness (HSL), or value (HSV). Lightness (or value) of a color is the quality of being lighter or darker. Saturation means the difference of color with respect to a gray color with the same intensity. As saturation normally ranges between 0 and 1, the grey color would be 0 and the most colorful color would be 1. Hue can be defined as the dominant frequency of the spectrum. It is typically represented in a color wheel and expressed in angular degrees (°), with red being 0° (as well as 360°), green being 120°, and blue being 240°. Figure 2.2c shows the distribution of the colors of the images in Fig. 2.2a, using the HSV color model.

CIELAB and CIELUV Models

These models were defined by the CIE for industrial color applications where measurement and color comparison are important. The models separate a brightness channel (L^*) and two chrominance channels (a^*b^* and u^*v^*). The latter are defined by nonlinear transformations of the RGB model in order to achieve perceptually uniform representations of color. In these models, colors are presented such that the differences between perceived colors are related to the distance between these

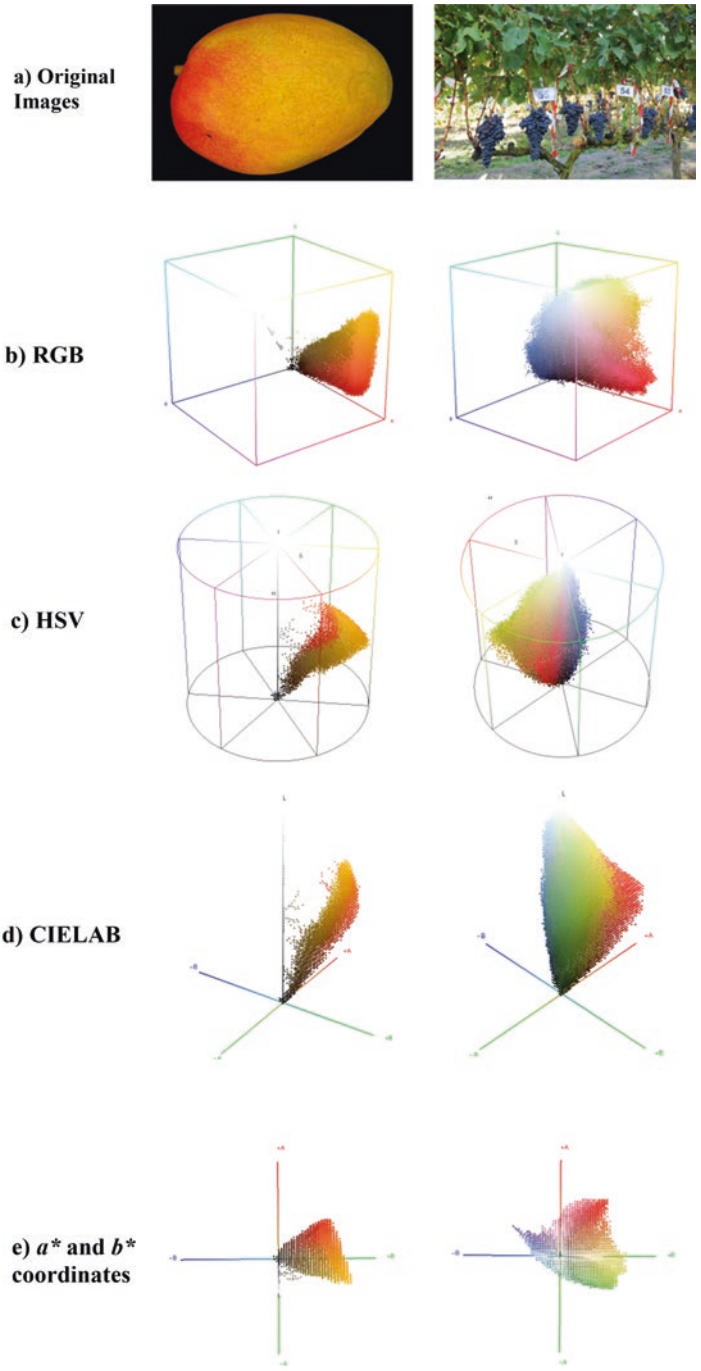


Fig. 2.2 Representation of the colors of two images, (**a**) one of a mango fruit and another of a vineyard using (**b**) RGB, (**c**) HSV, and (**d**) CIELAB color models. It can be seen how most colors are concentrated in a particular region of different color spaces, which indicates that in both fruit and vegetation images, only a relative small amount of colors is really used

colors in the color space. In the model, a^* coordinate represents changes from red to green and b^* coordinate represents changes from blue to yellow. This is one of the reasons why this color space is especially useful for measuring the color of agricultural produce such as fruit. For example, when most of fruits mature, the color changes from green to red, sometimes through orange or yellow. The fact that the coordinate varies from green to red makes this color space suitable for the creation of indices that indicate the maturity or quality of fruits, such as the citrus color index (CCI), the color index of tomatoes (Jarquín-Enríquez et al. 2013), the browning index (Cefola et al. 2012), or others (Pathare et al. 2013). Figure 2.2d shows the distribution of the colors of the images in Fig. 2.2a using the CIELAB color model, while Fig. 2.2e shows the distribution of the a^* and b^* coordinates.

This color space is represented by a chromaticity diagram that presents some interesting features (Fig. 2.3). Colors are described as perceived by the human eye in full daylight. The exterior curve is formed by pure spectral colors. The rest are composite colors. The sum of two colors is found in the line that joins them, which is very interesting and useful to determine color differences. The white color is at x = 1/3, y = 1/3, and the line joining any two complementary colors passes through that point. The diagram is complete, which means that it contains all the colors visible by humans. The notation Y_{xy} specifies colors by identifying value (Y) and the color as viewed in the chromaticity diagram (x,y) (XRITE 2018).

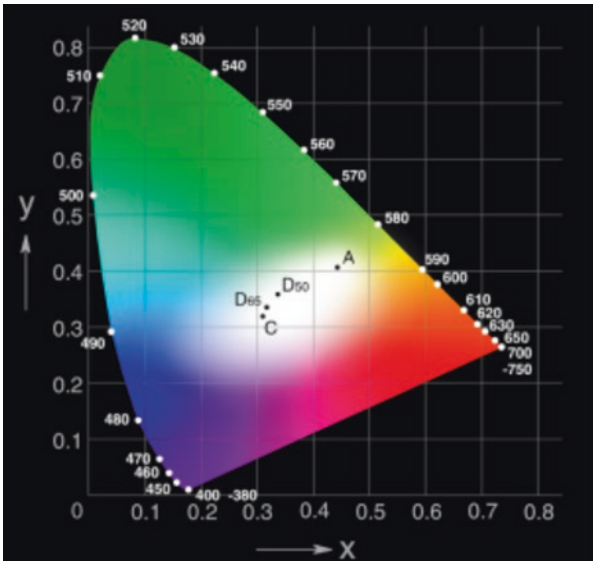


Fig. 2.3 CIE 1931 chromaticity diagram. All colors in the visible spectrum are represented, with the same brightness intensity

2.2.2 Color Space Conversion

To convert an image from one color space to another, we need to choose the method carefully because the visualization can be significantly different. As devices and color spaces represent the colors in a different way, during the conversion from one color space to another there can be colors of the original image that cannot be reproduced in the target color space, especially those located in the borders. Hence, some of the colors will be approximated to the closest one when possible and others, especially the saturated colors, will be just trimmed. This conversion can be done in several ways, such as absolute colorimetric mode, relative colorimetric mode, or perceptual conversion (Kahu et al. 2019).

In relative colorimetric mode, the original colors not reproduced during conversion are replaced by the closest and most saturated color. All other colors do not change. Therefore, if an element of a color has certain components that are outside the target space, these will be lost creating a flat color situation, since the color will be replaced by other colors inside the target space. On the other hand, in the perceptual conversion, efforts are made to keep all the perceptual shades of the color, but reducing the saturation so that they can fit within the limit of the target space. This approach, therefore, does not lead to a situation of saturation or flat coloring as the relative colorimetric technique, but the colors will change. The general perception of the image will be preserved, but the accuracy of the color will be reduced. There is not an ideal conversion method; each image and each application may desire a more suitable conversion mode and therefore depends on the colors of the image and the space to be converted.

2.2.3 Color Comparison

Two colors can look the same to the human eye if the difference is below the threshold of perception. However, these small differences can be measured and quantified. The color difference can be defined as the numerical comparison of the color of two samples, which indicates the differences in absolute color coordinates and is commonly known as Delta E (ΔE). It is normally calculated as the Euclidean distance between corresponding color locations within a color space (Sharma 2003). For instance, the difference between two color values expressed in L^*, a^* and b^* coordinates will be shown how far apart visually the two samples are. But, when this is applied to nonperceptual color spaces such as RGB, the Euclidean distance between two colors does not match the perceptual variations between these colors.

In 1976, the International Commission on Illumination (CIE) published the first formula to measure differences in CIELAB coordinates. Given two colors 1 and 2 in the CIELAB space, their distance can be calculated using Eq. (2.1). Values lower than 3 are considered as nonperceptible by the human eye, while values above 5 are considered to be clearly distinguishable.

$$\Delta E* = \left(\left(L*_1 - L*_2 \right)^2 + \left(a*_1 - a*_2 \right)^2 + \left(b*_1 - b*_2 \right)^2 \right)^{1/2} \quad (2.1)$$

Later, it was updated in 1994 and 2000 to correct and maintain the perceptual uniformity, but the original formulation is still the most widely used technique due to its simplicity (Fraser et al. 2004).

2.3 Image Acquisition

Imaging is a process by which 3D information (the scene) is projected onto a 2D plane (the image; more discussion can be found in Chap. 3). The images can be obtained from a wide range of sensors (i.e., magnetic resonance imaging, X-rays, radar, thermal or spectral cameras). This chapter focuses on images captured by standard color cameras that use CMOS (complementary metal oxide semiconductor) or CCD (charge coupled device) image sensors (photoreceptor). An image sensor is an electronic device used in digital cameras and imaging devices which converts the received light into an electronic signal. When the light passes through the camera lens, it goes through a small aperture and reaches the sensor. The photoreceptor captures that light and converts it into an electronic signal that depends on the intensity of the light and transmits it to the processor of the imaging device, which transforms the electronic signal into a digital image (Yotter and Wilson 2003). Figure 2.4 depicts this process.

To obtain the color from the electronic signals, digital color cameras generally use the so-called CFAs (color filter arrays) to interpolate the color information. Among these color filters, the Bayer filter is the most known and used. The most common Bayer matrix consists of 2 × 2 mosaics that follow an RGBG scheme: green, which provides more luminance information, is repeated twice. From here, to build the final image, it is necessary to apply a specific interpolation algorithm, which is run by the image processor. The final image result is greatly influenced by the interpolation method used (Bull 2014).

The result is a digital image represented as a matrix whose values can be defined as a two-dimensional function, f (x, y), where x and y are the spatial coordinates of the image and results in the value of the intensity or gray level at point (x, y). An

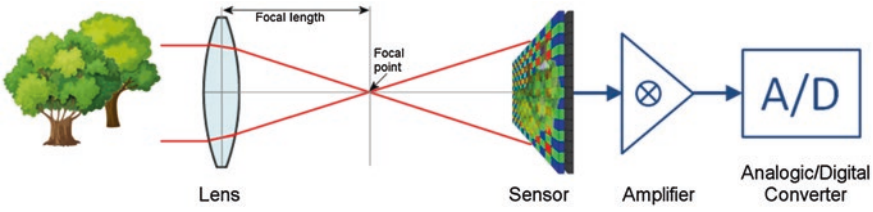


Fig. 2.4 Process of the formation of an image in a digital camera

image of m rows and n columns has a total of n x m pixels. The spatial resolution of an image is the number of pixels per unit length of the sensor. Color images usually consist of three monochromatic images that correspond to the colors obtained through the Bayer filter. Thus, the color of a pixel is represented by a vector containing the red, green, and blue values f (x, y) = {R, G, B}, within the RGB color space. However, color can also be represented in other color spaces (see Sect. 2.2).

Some acquisition parameters are important to capture quality images. For example, if the aperture is too small, very little light will arrive at the photoreceptor and therefore the image will be very dark. To prevent this, some parameters of the cameras can be adjusted. However, it is very important to know precisely how they work since the quality of the image acquired, especially in difficult field conditions where the camera is sometimes mounted on vehicles moving through uneven terrain, depends heavily on those parameters. One solution is to increase the exposure time, keeping the size of the aperture small to maintain focus. However, if the camera or object moves, the image will be blurred. Another option is to increase the size of the aperture and try to focus the image manually to improve accuracy. But, due to the geometry of the lenses, they only keep the images focused for scenes that are within a certain distance (depth of field). A third option is to maintain a reduced exposure time to freeze movement, a reduced aperture size to better maintain focus, and increase sensitivity by adjusting the gain (sensitivity, ISO value) of the camera. But increased sensitivity also leads to increased noise.

Basically, to achieve a quality image, these parameters must be adjusted so that the photoreceptor reaches a sufficient amount of light. The diaphragm is a device that opens and closes to allow more or less light to pass to the photoreceptor in imitation of the iris of the human eye. The smaller the opening, the more depth of field, but less light entering into the photoreceptor. The opening is expressed in relation to the constant f, whose typical values are between f/1.4 and f/22. The greater the number, the smaller the opening size.

The shutter speed is the time during which the light passes through the opening/aperture to the photodetector. It is measured in seconds, and together with the aperture of the diaphragm, it determines the amount of light that enters the camera. To obtain images of static objects, it is not a problem to set slow speeds. But if the objects or the camera are in motion, it will be essential to set the shutter at a high speed. For example, if an object moves at 1 m/s, it will move 1 mm for every ms of shutter speed. However, the higher the shutter speed, the less time the light takes to reach the sensor, and, therefore, the light intensity required for the sensor to properly collect the scene is greater. The sensitivity in digital is related to the gain (voltage in relation to the number of photons), which is an electronic amplification of the signal, but at the same time it can amplify the noise of the image, so it is desirable to not increase the gain unless it is necessary.

Optics also play a leading role in capturing good images (Catrysse 2015). The lens is the part of the camera that directs the light rays toward the photoreceptor. The quality of the lenses allows to obtain more or less clear, bright, and focused images. This determines the depth of field, which is defined by the closest and farthest distances that can be sharply focused, which is particularly important when working in

the field with the camera incorporated in a moving agricultural vehicle or robot that can be approached or get away from the scene due to the irregularities of the terrain. Therefore, in these cases it is very important to have an adequate depth of field. There are autofocus lenses, but they also increase the mechanical complexity, and the time needed to adjust the focus makes them unsuitable for capturing fast images while the vehicle is moving.

The distance to the scene can also be adjusted by using zoom lenses. The magnification of a lens is determined by the focal length, which is the distance between the optical center of the lens and the focus or point where the image is formed on the sensor. In a camera with a longer focal length, the objects appear larger, but the scene is smaller (greater viewing angle). There are lenses that allow to adjust the zoom manually or automatically. But it is advisable to use fixed lenses for machine vision applications because they are simpler and produce higher quality images. On the other hand, it is important to maintain the fixed and known distance to target objects if the objective is to determine some properties of the objects under analysis, such as object size without the use of 3D sensing techniques (Chap. 3).

2.4 Basic Image Processing Operations

To obtain useful information about and from the images, they must be processed. In its basic form, as discussed before, an image is composed of a set of values that can be expressed as a matrix. In the case of monochrome images, these values correspond to the gray levels of each pixel. In the case of color images, they generally consist of three matrices that correspond to the red, green, and blue (RGB) values of each pixel, or it could be a three dimensional matrix with $f(x,y)$ being the spatial dimensions and R,G,B channels stacked over each other in the third (depth) dimension. These values are spatially connected to form objects or regions of interest. However, to find and extract them from the image, it is necessary to apply a series of image processing techniques. In general, the basic steps to process an image generally include a preprocessing, which is necessary to improve certain characteristics of the image, eliminate noise, and correct deficiencies that occur during acquisition. The following is the segmentation of the image, which consists of dividing the image into regions of interest. For this step, there are numerous region-based and pixel-based techniques used (Gonzalez and Woods 2018; Russ and Neal 2017). Pixel-based methods are simpler to implement but very sensitive to noise or contrast changes in the image. On the contrary, region-based techniques look for homogeneous areas or contours in the image. These techniques can be supervised, when it is necessary (or appropriate) to include prior knowledge of the image in the segmentation algorithms, or unsupervised, otherwise. The result of the image segmentation process is the classification/categorization of pixels into different classes, grouped to form regions. Once the image has been divided into regions, it is necessary to determine which of these regions correspond to the objects of interest, extract the

characteristics of these objects, and make decisions based on the characteristics found (Gonzalez and Woods 2018; Russ and Neal 2017).

2.4.1 Image Enhancement

The images can be obtained in controlled environments, which allows establishing the most favorable conditions for acquisition depending on the problem to be addressed. However, when images are obtained outdoors, conditions cannot be controlled and unwanted effects occur in the form of brightness, shadows, movement, noise, or objects of interest that appear hidden in the images. For example, a robot that advances in the field will obtain images that will be influenced by changes in the lighting condition caused by clouds or the changing sun angle in relation to the vegetation/objects being observed. In these cases, it is especially interesting to perform image enhancement and preprocessing operations. Image preprocessing is used to enhance or highlight some image properties to facilitate the following operations of analysis and extraction of image characteristics. Some of the improvement techniques are based on specific transformation operations in pixels that do not take into account the neighborhood of the pixel. However, to improve the contrast of an image, it is also possible to use global operations, such as histogram equalization, which aims to accentuate the visual contrast, making better use of the range of available intensity values. Histogram equalization creates an image whose intensity levels are equally likely and cover the entire dynamic range. The result of this process is an image whose dynamic range has increased, which will tend to have a greater contrast.

2.4.1.1 Histogram

One of the basic tools to obtain information about the distribution of pixel intensities in the image is the histogram. A histogram is a measure of the frequency of occurrence of each intensity level value within an image. It is represented in a graph where the X-axis shows the color values or intensities and the Y-axis the frequency of these intensities. As example, Fig. 2.5 shows an image (left) and the histogram for each of the red green and blue components of the image. Equation (2.2) shows how the relative histogram value for gray level k is calculated in an image of size N × M. In the case of color images, a histogram can be obtained for each color channel.

$$H(k) = (\#\text{pixels with intensity } k) / N \times M \tag{2.2}$$

This measure evaluates the contrast of the image. If the histogram graph shows a narrow peak, a low contrast is identified since few gray levels are used and with very

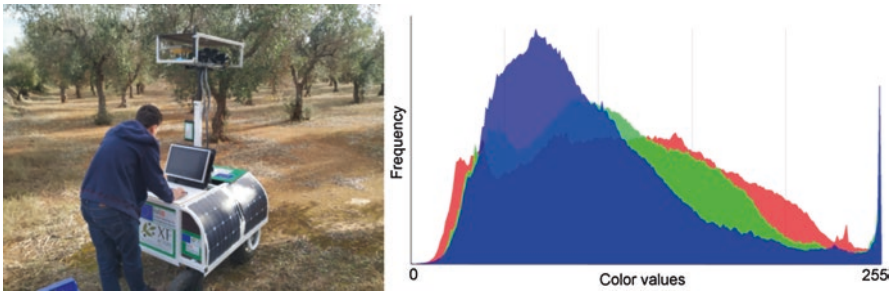


Fig. 2.5 Image and histogram of the red, green, and blue components of the image. The peak on the right side corresponds to the values in the saturated regions for each channel

close values. If the graph shows a wider distribution among all gray levels, it means more intensity differences in the image and therefore greater contrast. If several peaks are observed, they may correspond to different regions of interest, which facilitates the segmentation of the image. A special case is a bimodal distribution, in which there are only two peaks present. They can identify with two regions of interest or allow discrimination between the background and the foreground of an image. The histogram can be used to make visual improvements to the image, such as an increase in contrast using techniques such as equalization.

2.4.1.2 Morphological Operations

Various techniques can be used to identify shapes of objects of interest in images as discussed earlier in this chapter. As a postprocessing step using a structuring element and various calculation with pixels, morphological operations can remove small unnecessary objects, make objects more visible, fill holes, and enhance contrast among other things. Major morphological operations include erosion, dilation, hit-miss, convex hull, open, close, skeleton, prune, thin, and thick (Russ and Neal 2017). *Erosion* removes small objects, whereas *dilation* fills smaller voids and changes objects such that they become wider and more visible. *Hit-Miss* is used to find patterns in an image, and *Convex Hull* is used to find envelope of an object to detect its shape. *Open* is an integrated operation that combines erosion and dilation so that object shapes are not changing while small unnecessary objects are removed. *Close* is a similar operation, but it first conducts dilation and then erosion to fill holes while keeping object shapes intact. *Skeleton* (or skeletonization) is used to estimate skeleton of objects (or change objects into lines). *Thin,* similar to *skeleton* operation, removes selected foreground pixels to find skeleton, whereas *thick* is used to add selected foreground pixels. *Prune* is another operation that is used to remove unnecessary branches of an object, which is important to find key shapes.

2.4.1.3 Low-Pass Filtering

Smoothing filters (also known as low pass filters) can be used to reduce noise in an image but can cause unwanted level of blurring on the image. The most common method to apply a filter is the convolution of the image with a kernel. A template of size 3 × 3 kernel can be seen in Fig. 2.6. By placing the kernel on the pixel of interest of the image, the result is obtained by the sum of the multiplication of each element of the kernel by the value of the pixel in the image. Therefore, the response P to the kernel of any pixel in the image is given by Eq. (2.3). To obtain the effect on the whole image, the kernel moves through all the rows and columns sequentially that form the image so that it is finally applied to all the pixels.

$$P = \sum_{i=1}^{9} p_i z_i \tag{2.3}$$

where z_i is the gray level in the image at the pixel corresponding to p_i in the kernel. The pixel of interest is the central one.

Smoothing is achieved by decreasing the differences between neighboring pixels. This can be obtained, for example, using a mean filter. This filter replaces the value of each pixel with the average value of its neighbors. Normally a 3 × 3 kernel is applied, but depending on the desired effect, larger kernels can be used. The result makes the image appear blurry by reducing the contrast of the edges of the objects. The mean is sensitive to the extreme values of the neighborhood. If there are pixels with extreme values (outliers), they will significantly affect the final result. Therefore, a common option is to replace the mean filter with a median one, which is not sensitive to the presence of outliers and also guarantees that the result is a value that already exists among the neighbors. This may be of interest in some circumstances. The median filter is not a convolution and therefore cannot be expressed through a kernel. Another widely used filter, both to eliminate noise caused by isolated pixels and to soften the contours of objects in images, is the mode filter. This filter replaces the value of each pixel with the highest frequency among its neighbors. Special mention has to be paid to the Gaussian smoothing operator, which is efficient in noise removal. Its smoothing efficiency depends on the value of its standard deviation. It is similar to the mean filter, but it uses a different kernel that represents the shape of a Gaussian hump (Shapiro and Stockman 2001).

Fig. 2.6 Kernel o 3 × 3 convolution matrix of size

p_1	p_2	p_3
p_4	p_5	p_6
p_7	p_8	p_9

2.4.2 Segmentation

Segmentation divides an image into the different regions that compose it in order to find those that are of interest. There are several techniques to perform this division. Some techniques are based on classifying the pixel according to its intrinsic values (pixel-wise techniques). Basically, they are based on thresholds or classifiers that assign each pixel into a predefined class, depending on its individual value. Other techniques are based on detecting regions with similar characteristics (region-based techniques). In addition, segmentation algorithms can be supervised or unsupervised depending on whether they need previous information from the user to perform the segmentation. Supervised segmentation starts with a previous knowledge about the image that the user enters into the segmentation or classification algorithms. For example, previous knowledge might include the number of classes in which the image is to be divided and/or the assignment of some pixels with certain characteristics (e.g., color) to those classes. These pixels form the training dataset. On the contrary, in unsupervised segmentation, the algorithms extract the characteristics of the image without the use of prior knowledge, which leads to image segmentation without any user intervention. The best technique to be used depends on the problem to be addressed. Some of the most common segmentation techniques there are discussed below.

2.4.2.1 Pixel-Wise Techniques

Threshold-Based Segmentation

Within the image segmentation techniques, thresholding is one of the simplest methods. It consists of dividing the image into regions of interest based on the variation of the intensity of the pixels. Basically, in the case of monochromatic images, a comparison of each intensity value of the pixels is made with respect to a threshold value (established according to the problem to be solved). Values above that threshold are assigned to one class (object or background) and pixels that are below to another class. This is known as binarization. In more advanced cases, different thresholds can be established to increase the number of classes.

When there is a high contrast between the objects of interest and the background, the threshold can be chosen arbitrarily. In more complex cases, the study of the histogram is convenient. By analyzing the frequency of appearance of the different values of the pixels, it is possible to set appropriate thresholds. There are several techniques to establish thresholds from the study of the histogram. In the case of a bimodal distribution, in which the intensities corresponding to the objects are concentrated on one side of the histogram and the background is concentrated on the other side, the optimum threshold is usually at the lowest value between the two peaks of the histogram. To obtain the optimal threshold, the first step is to find the two highest peaks of the histogram and then find the lowest point between them. In some cases, the difficulty consists in the existence of secondary peaks that, due to

their proximity to the main one, must be ignored, so it is sometimes convenient to establish a minimum distance between the two peaks to be considered.

When the histogram is not bimodal, it is necessary to combine both the spatial information present in the image and the information about gray levels. Among the existing methods, Isodata is one of the simplest. It is an iterative technique used to obtain the correct threshold. The histogram is initially segmented into two parts using a start threshold T. Next, the average intensity of the pixels belonging to each of the parts in which the histogram $m1$, $m2$ is segmented is computed. Using these values, a new threshold value is calculated by the formula: $T = (m1 + m2)/2$. This process is repeated until the difference between the values T in successive iterations is smaller than a predefined parameter ΔT.

However, probably, the most commonly used is the Otsu method (Otsu, 1979). The algorithm calculates the threshold value so that the dispersion (distance between the values with respect to a mean value) within each segment is as small as possible, while the dispersion is the highest possible between different segments. For this, the quotient between both variances is calculated, and a threshold that maximizes this quotient is searched. Examples for the application of this method can be found in Xiong et al. (2018) and Zhuang et al. (2018).

In general, thresholding is a simple and fast method to segment images where the objects of interest are highly contrasted against the background of the image. However, this technique does not work well in images with low contrast or acquired with poor lighting conditions. In these cases, the intensities of the pixels may vary depending on their position in the image, which negatively influences the segmentation. For these cases, it is advisable to use a variable or adaptive thresholding instead of a fixed threshold. One technique for setting variable thresholds is the partition of the image. Image partition is one of the simplest methods to search for variable thresholds being useful to compensate for lighting problems in the image. It consists of dividing the image into regions of a certain size so that the lighting in each of them could be comparatively more uniform, and then applying a local threshold to each of the regions would result in more desirable segmentation results.

When the target is the segmentation of a color image, this technique can be applied to each of the three color channels. However, as the color offers more information, in these cases, other segmentation techniques are advisable (e.g., use of classifiers as discussed below).

Segmentation Using Classifiers

When images are captured with color information, it can be used to discriminate different objects and also to obtain information about each object identified. For example, in images of tree canopies, the color allows to distinguish fruit and leaves, which could be used to make a yield prediction. In addition, it also allows to estimate the stage of fruit maturity. To do this, it is necessary to associate colors in the image with the different objects found in the scene. The first step is to divide the image into regions of interest, that is, into objects that are different, which can be accomplished by classifying pixels into different groups or regions.

In images of the field captured by a moving robot outdoors, the color of a pixel can be seen as a random event, within the n-dimensional space defined by the color space used. The probability distribution of the color is determined by the type and composition of the environment, the ambient light, the camera settings, and the movement of the robot that carries the camera (Rey et al. 2019, Cubero et al. 2020).

There are a number of classification methods, some based on the intrinsic color of the pixel to be classified and others that take into account the neighborhood in which the pixel is located. As an example of a pixel-oriented classification method, suppose we know the functions of the probability distribution of the colors of the fruit tree canopies, including vegetation $p_v\ (color)$, a ripe fruit, $p_f\ (color)$, where *color* is a list of n-elements of any color space; for example, in the RGB color space, the color would be the list of the three components red, green, and blue of that pixel. Considering equiprobable the a priori probabilities of belonging to the vegetation or fruit classes, the probability that a pixel belongs to vegetation or fruit can be calculated using the Bayes rule, given by Eqs. 2.4 and 2.5 (Blasco et al. 2009):

$$P\left(fruit \mid color\right) = \frac{p_f\left(color\right)}{p_f\left(color\right) + p_v\left(color\right)} \tag{2.4}$$

$$P\left(vegetation \mid color\right) = \frac{p_v\left(color\right)}{p_f\left(color\right) + p_v\left(color\right)} \tag{2.5}$$

These functions result in the probability of a pixel belonging to one or another class. Consequently, that pixel is assigned to the class with the highest probability of belonging between vegetation and fruit. This example is shown with two classes, but it can be expanded to more, obtaining a probability function for each of the classes present in the classification problem. By classifying all the pixels of the image, the segmentation of the image is achieved. The pixel-oriented classification approach can be improved with the application of local processing operations, which take into account the characteristics of the pixels in a given neighborhood.

2.4.2.2 Region-Based Segmentation

The algorithms to segment monochrome images are generally based on finding discontinuities or similarities between gray levels of neighboring pixels.

Segmentation Based on Edge Detection

Normally, what defines the objects in the images are the borders between them. A border is defined as the discontinuity between two neighboring regions with a relatively different gray level. These discontinuities can be detected, for example, using first- and second-order derivatives. There are numerous techniques and operators to calculate these derivatives. In the case of first-order derivatives, it is common to use the gradient. The gradient operator is a first-order directional derivative that reaches its maximum value in the direction in which the variation is maximum. As an

example of a second-order derivative, the Laplacian function can be used, which is a filter whose result is 0 in the homogeneous region and positive or negative where there is contrast between regions. A commonly used discrete approximation to the Laplacian filter is shown in Fig. 2.2c.

Once the edges are obtained, it is normal that they do not completely close the contour of objects due, among other things, to the presence of angles, changes in lighting and/or presence of other noises. Therefore, after edge detection, techniques are usually used to join the pixels of different edges and turn them into the contours of objects (Gonzalez and Woods 2018). One of the simplest procedures for linking edges is to analyze the characteristics of pixels in small neighborhoods (for example, 3×3) around each point (x, y) belonging to an edge. All points in that neighborhood that are similar and share certain properties in common are linked. The properties that are mostly used are the magnitude of the gradient that has given rise to the edge pixel and the direction of the gradient at that point. As a result, an image with the objects outlined is achieved.

Segmentation Based on Aggregation Techniques

Similarity-based techniques search for those regions that are homogeneous or with a similar texture or other common characteristics. The main methods are based on region growing, or division and fusion of regions (Gonzalez and Woods 2018). Regions are formed from neighboring pixels that meet some similarity rule (e.g., color difference less than a threshold) and differ from pixels that do not belong to the region. One of the techniques most used by region-based algorithms is region growing. A pixel is chosen within the region to be obtained, and the similarity rule is applied to its neighbors. Those pixels that comply with the rule will be added to the growing region. On these newly added pixels, the similarity rule will be applied again to their neighbors iteratively. The algorithm will stop when the pixels neighboring the growing region do not meet the similarity criteria. To achieve the segmentation of the entire image, several seed pixels are chosen in the image randomly distributed or using some kind of rule such as those with extreme intensities or colors. A common technique based on this method is the *k*-means algorithm.

K-means (MacQueen 1967) is a clustering algorithm based on the region growing techniques that needs as input the number of regions (clusters), k in which the image is to be segmented. The algorithm selects the first random k pixels (seeds) in the image. In the first step, these single pixels are the centroids of the clusters. Then, those pixels accomplishing a function of minimum distance are aggregated to the clusters. The function of distance may correspond to the intensity of the pixels or any other characteristics on which the segmentation is to be based. The centroid is recalculated as the mean of the cluster, including the new samples added. This will generate a new pixel assignments, since other pixels will be now closer to the centroids. This process is repeated iteratively until all pixels belong to one of the clusters or there are no further changes in the centroids.

Other techniques, instead of using seeds for a region growing process, divide the image into arbitrary regions, so that if the region is very heterogeneous according to some type of rule, it will be divided, otherwise, it will be merged with the adjacent

regions holding the similarity rules. A common technique is to divide the image into four regions. If any of the regions is heterogeneous, it is divided into other four regions. The process is repeated until no changes occur.

2.4.3 Features of Objects of Interest

To identify an object from an image, colors are very important information for distinguishing various objects. In general, various objects have their own colors. Even for objects with similar colors to human eyes, their minute differences may be identified through image processing. For example, to distinguish immature green citrus fruit from surrounding green canopies, different amount of red and blue components can be identified using chromatic aberration map. In Zhao et al. (2016), chromatic aberration map was calculated using the following equations, and green fruit can be differentiated and background objects can be removed. Figure 2.7 below is an example of a chromatic aberration map.

$$\mathrm{ARB} = \mathrm{R} - \mathrm{Ratio} \times \mathrm{B}, \tag{2.6}$$

where ARB is an adaptive red and blue chromatic aberration map, R is red component, Ratio is a ratio of blue over red (= B/R), and B is blue component. In Fig. 2.7, green colors of the immature green oranges look different than those of surrounding green canopies, even though all objects in the image look green.

Shape features are also very useful in detecting objects. Since objects of interest generally have their unique shapes in the environment being imaged, these features can be used for object detection and localization in many agricultural and field robotic applications. For example, for orange or apple detection, roundness of those fruits cab be a very important feature to use. Some plant leaves are long and thin, which also can be used to differentiate them from round shaped weeds around them. Shape features include roundness, compactness, major axis, minor axis, centroid, perimeter, curvature, and elongation, among others. These features, alone or in combination with other features like color, are used for developing discriminant functions to identify objects of interests.

Texture is defined as repeated patterns of similar pixels in an area, and it also can be a powerful feature for distinguishing different objects. For example, in an image with strawberry plants in Fig. 2.8, the texture of strawberry fruit is very different than those of nonfruit parts. Of course, red color of strawberry fruit is a very unique feature; however, the texture of the fruit is also very unique for distinguishing fruit from other objects. This unique texture holds during immature stages, as shown in Fig. 2.8c, which could be a great feature to distinguish immature fruit.

Texture analysis can be divided into two categories: (i) statistical and (ii) structural methods (Jain 1989). Statistical methods include autocorrelation function, image transforms (e.g., coarseness, fineness), edge density, histogram features, and random texture models. Structural methods utilize invariant properties of a group of

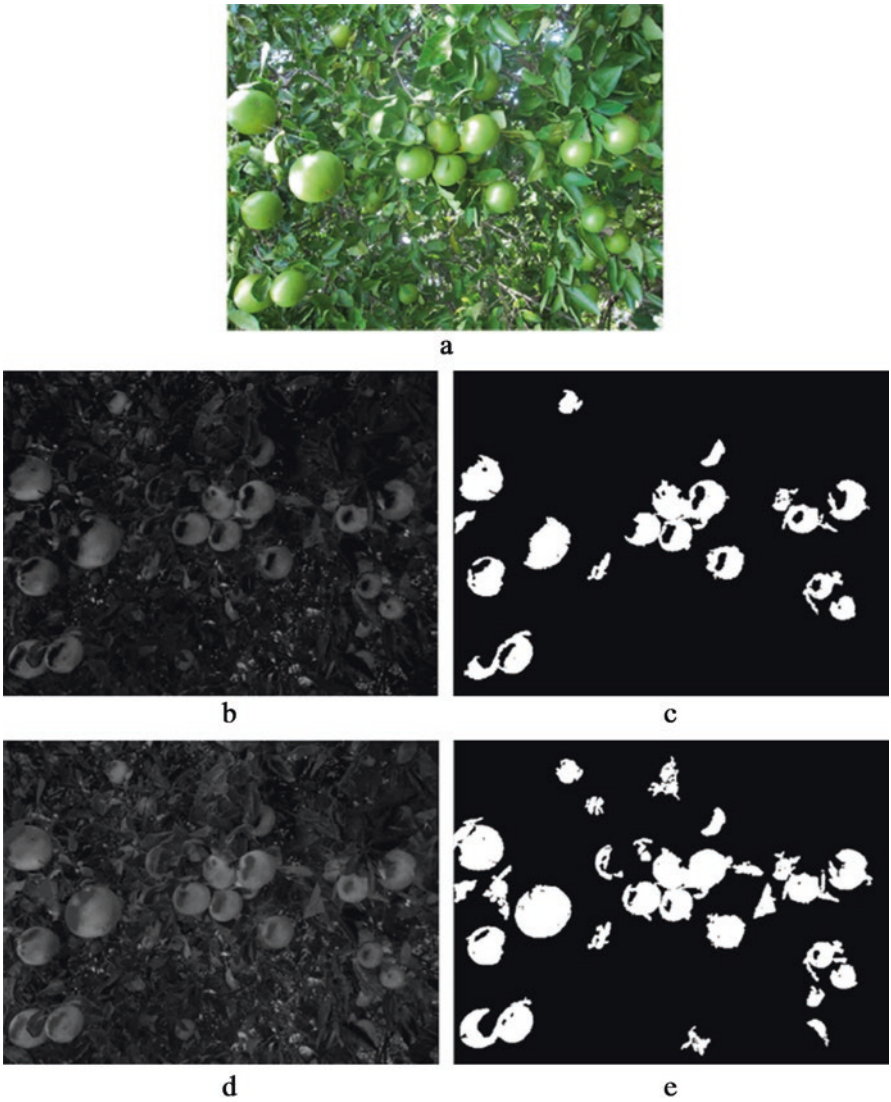


Fig. 2.7 An example chromatic aberration map: (**a**) original color image, (**b**) difference between red and blue components, (**c**) binary image after removing noise, (**d**) adaptive RB chromatic map, and (**e**) binary image of (**d**) after removing noise. (Adapted from Zhao et al. 2016)

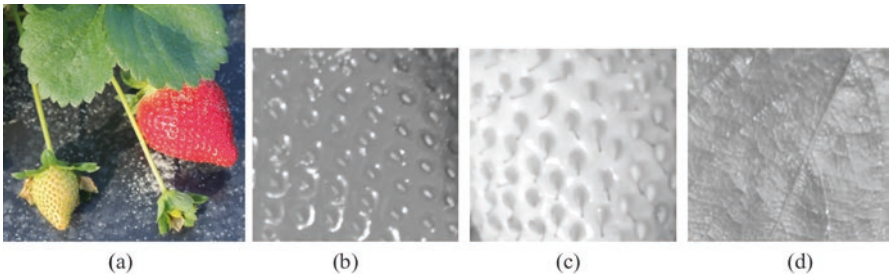


Fig. 2.8 Texture of strawberry fruit is very different than those of leaves and other objects: (**a**) strawberry plant, (**b**) texture of mature fruit, (**c**) texture of immature fruit, and (**d**) texture of leaf

Fig. 2.9 An example of circular Hough transform to identify immature green oranges. (Adapted from Sengupta and Lee 2014)

pixels, which is called a texel. Texels can be located closely or farther away, and their frequency of repetition may be used in defining textures with structural methods (Jain 1989).

2.4.4 Hough Transform

Hough transform was first developed by Paul V.C. Hough in 1959 (Hough 1959) to identify line segments in a bubble chamber photograph. Later it was generalized by Duda and Hart (1972), and eventually it became a very popular method for identifying circular objects, which was named as the circular Hough transform (CHT) afterwards. Using Eq. (2.7), CHT finds a circle or curve which are pixels satisfying a given condition of a center location (C_x, C_y) and radius (ρ) while the angle (θ) changes from 0° to 360°. More detailed information can be found in Kimme et al. (1975).

$$x = C_x + \rho cos\theta$$
$$y = C_y + \rho sin\theta \tag{2.7}$$

Since many objects in agricultural crops are in circular shapes, such as oranges and apples, this method has been used extensively for detecting objects in crop canopies for various robotic applications. Figure 2.9 below shows an example of Hough transform from an immature green orange image. In this example, circular objects can be seen clearly and easily identified after some postprocessing/cleaning techniques are applied. It is noted that the result of Hough transform would not generally be the final step for identifying various objects, such as fruit in tree canopies. More postprocessing steps would be necessary to achieve the desired final detection/identification results based on the specific application being considered.

2.5 Pattern Matching

Many times when objects of interest are similar, shape and pattern matching or template matching can be a good way to identify them in images. Pattern matching means to determine whether objects of interest are same (i.e., match) as the predetermined pattern or template by finding similarities between them. Some methods used in pattern matching are cross-correlation, sum of absolute differences, and fuzzy pattern matching, or some variations of these techniques.

One example technique for pattern matching would be the sum of absolute transformed difference (SATD), which is a commonly used method for motion estimation. SATD works by calculating differences of a template and pixels of interests in a gray scale image and taking a Hadamard transform of the difference (Kunz 1979). Zhao et al. (2016) implemented this method to identify immature green citrus fruit in color images along with a combination of other techniques, including adaptive red and blue chromatic aberration map, illumination enhancement, and image segmentation based on histogram. The study achieved a detection accuracy of 83% with 17% missed fruit and 11% false positives.

A fast normalized cross-correlation (FNCC) would be another pattern matching technique that can be applied to identify objects on interests in images. FNCC is a template matching method which calculates correlation between a template and objects of interests in images. As shown in Fig. 2.10, Li et al. (2016) applied this method to detect immature green citrus fruit from images and reported a detection accuracy of 84%. Different pattern matching methods may be used in various applications. However, it would be difficult to specify the best pattern matching method beforehand, as their success may vary depending on specific applications at hand.

Fig. 2.10 An example of an image of (**a**) immature green oranges and (**b**) FNCC result. (Adapted from Li et al. 2016)

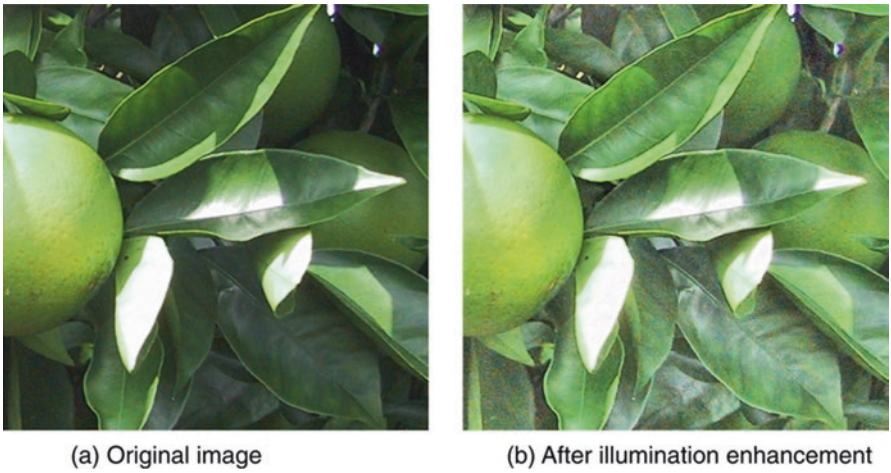


Fig. 2.11 An examples of illumination enhancement by a histogram equalization and logarithmic transformation. (Adapted from Kurtulmus et al. 2011)

2.6 Things to Consider

For outdoor imaging, *varying illumination* may be a big problem since it will affect the image quality tremendously. In agricultural and field applications, images are acquired usually in the natural environment without much control over the illumination sources, configurations, and conditions. Depending on the sunlight, some areas of the crop will be under direct sunlight, but some of them will be under the shade. In this situation, objects with similar colors might look very different in images due to the varying illumination. Figure 2.11a is an example of the effect of varying

Fig. 2.12 Example of illumination enhancement using CLAHE method (**a**) and (b) original images, and (**c**) and (**d**) enhanced images using CLAHE. (Adapted from Choi et al. 2016)

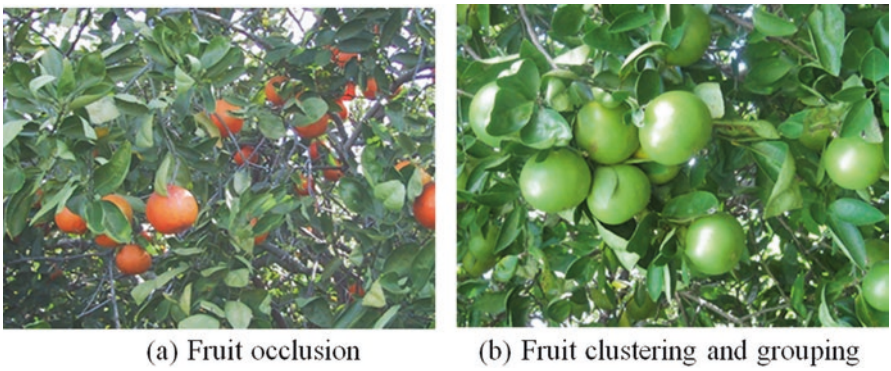


Fig. 2.13 Examples of (**a**) fruit occlusion and (**b**) clustering/grouping in an orange tree canopy

illumination, which shows immature oranges in bright and dark green colors. To eliminate or lessen this kind of problem, a histogram equalization method combined with a logarithmic transform can be used, as shown in Fig. 2.11b.

Contrast limited adaptive histogram equalization (CLAHE) is another method that can be used to adjust different brightness levels in an image. Choi et al. (2016) utilized CLAHE to identify and count the number of oranges prematurely dropped on the ground. Figure 2.12 shows an example of contrast enhancement.

Occlusion would be another big obstacle in outdoor imaging, particularly in agricultural fields such as fruit orchards. As shown in Fig. 2.13a, many objects of interest are hidden behind other objects and cannot be seen clearly. The human eye may be able to identify them; however, it would be difficult for a machine vision system

to do the same job. Perhaps acquiring multiple images at different angles could minimize occlusion.

Clustering or grouping of the same objects of interest would be another problem particularly prevalent in using a machine vision system in agriculture and field environments. When the same objects are touching each other and grouped together, it would be difficult to identify (and count) them individually. Oranges or blueberries are good examples facing such issues, as shown in Fig. 2.13b. Similar issues of clustering and occlusion can be faced in other applications, such as developing a machine vision system for robotic picking of fruit from tree canopies and robotic picking of rocks from agricultural fields.

Object movement due to wind during image acquisition would be another obstacle, especially for hyperspectral imaging, using a linescan camera. Since the tilting head will be used for acquiring hyperspectral images, it will take time to finish acquiring an image. If wind blows during image acquisition, objects will not align in the image, one line after another. Similar issues can exist in robotic fruit-picking application. A vision system can detect and locate a fruit at the beginning of harvesting cycle. As the fruits are picked, the changing load, wind, or other interferences can change the position of the fruit between imaging and picking, thus leading to failed picking. Repeated imaging and fruit detection approach can help minimize this issue, but at the cost of increased computational demand, which might impact the overall speed of the harvesting system.

2.7 Summary and Concluding Thoughts

This chapter describes important basic concepts of color imaging and basic image processing techniques. For developing successful robotic solutions for agricultural and field operations, it is very important to understand the basics of color spaces, different types of imaging sensors, image acquisition techniques, preprocessing steps, and some advanced image processing techniques. Agricultural and field applications usually require outdoor imaging, which likely means facing problems and challenges related to varying illumination, occlusion, and clustering of objects (e.g., fruits). Some strategies to minimize the issues caused by these challenges have also been described in this chapter. For any given task, developing efficient, robust, and accurate image processing algorithms is essential, which provides the foundation for developing successful robotic solutions to perform certain tasks in specific crops or other field conditions. It is noted that more discussion on various aspects of color imaging systems (including camera calibration and stereo-imaging) as it pertains to 3D sensing techniques will also be discussed in Chap. 3.

As technologies are advanced, task-specific machine vision systems may be developed in the near future that can automatically accomplish various tasks. Combined with artificial intelligence, many modular imaging systems may be developed for specific agricultural and field applications, such as nutrient or water stress detection or crop yield mapping. Such systems may solve difficult situations

of varying illumination or occlusion. Yet, fundamentals in color imaging and processing still remain important in such systems.

References

Blasco J, Cubero S, Gómez-Sanchis J, Mira P, Moltó E (2009) Development of a machine for the automatic sorting of pomegranate (*Punica granatum*) arils based on computer vision. J Food Eng 90(1):27–34

Bull DR (2014) Digital picture formats and representations. In: Bull DR (ed) Communicating pictures, Academic Press, Oxford, UK, pp 99–132

Catrysse PB (2015) Optics for digital imaging, in handbook of digital imaging. Volume 1: image capture and storage. Wiley, Hoboken

Cefola M, D'Antuono I, Pace B, Calabrese N, Carito A, Linsalata V, Cardinali A (2012) Biochemical relationships and browning index for assessing the storage suitability of artichoke genotypes. Food Res Int 48:397–403

Choi D, Lee WS, Ehsani R, Schueller JK, Roka FM (2016) Detection of dropped citrus fruit on the ground and evaluation of decay stages in varying illumination conditions. Comput Electron Agric 127:109–119

Cubero S, Lee W, Aleixos N, Albert F, Blasco J (2016) Automated systems based on machine vision for inspecting citrus fruits from the field to postharvest – a review. Food Bioprocess Technol 9:1623–1639

Cubero S, Marco-Noales E, Aleixos N, Barbé S, Blasco J (2020) RobHortic: a field robot to detect pests and diseases in horticultural crops by proximal sensing. Agriculture. Special Issue "Enhancing Surveillance and Detection of Invasive Harmful Plant Pathogens and Pests", 10, 276. https://doi.org/10.3390/agriculture10070276

Duda R, Hart P (1972) Use of the Hough transformation to detect lines and curves in pictures. Comm ACM 15:11–15

Ford A, Roberts A (1998) Colour space conversions. Technical report. University of Westminster, London. Retrieved from https://poynton.ca/PDFs/coloureq.pdf. Accessed Apr 2020

Fraser B, Murphy C, Bunting F (2004) Real world color management. Peachpit Press Publications, Berkeley

Goldstein E (2010) Sensation and perception, 8th edn. Cengage Learning, Belmont. ISBN: 978-0-495-60149-4

Gonzalez R, Woods R (2018) Digital image processing, 4th edn. Pearson, New York. ISBN: 9780133356724

Hastings G, Rubin A (2012) Colour spaces – a review of historic and modern colour models. S Afr Optometrist 71:133–143

Hough P (1959) Machine analysis of bubble chamber pictures. In: Proceedings of the international conference on high energy accelerators and instrumentation

Ibraheem N, Hasan M, Khan R, Mishra P (2012) Understanding color models: a review. ARPN J Sci Technol 2(3). Retrieved from https://haralick.org/DV/understanding_color_models.pdf. Accessed Apr 2020

Jain AK (1989) Fundamentals of digital image processing. Prentice Hall, Inc.

Jarquín-Enríquez L, Mercado-Silva E, Maldonado J, Lopez-Baltazar J (2013) Lycopene content and color index of tomatoes are affected by the greenhouse cover. Sci Hortic 155:43–48

Kahu SY, Raut RB, Bhurchandi KM (2019) Review and evaluation of color spaces for image/video compression. Color Res Appl 44:8–33

Kimme C, Ballard D, Sklansky J (1975) Finding circles by an array of accumulators. Short Commun Graph Image Process 18(2):120–122

Kunz H (1979) On the equivalence between one-dimensional discrete Walsh-Hadamard and multidimensional discrete Fourier transforms. IEEE Trans Comput C-28(3):267–268

Kurtulmus F, Lee WS, Vardar A (2011) Green citrus detection using eigenfruit, color and circular Gabor texture features under natural outdoor conditions. Comput Electron Agric 78(2):140–149

Li H, Lee W, Wang K (2016) Immature green citrus fruit detection and counting based on fast normalized cross correlation (FNCC) using natural outdoor colour images. Precis Agric 17(6):678–697

MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1: statistics, pp 281–297

Meyer G, Neto J (2008) Verification of color vegetation indices for automated crop imaging applications. Comput Electron Agric 63:282–293

Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Sys Man Cyber 9(1):62–66

Pathare P, Opara U, Al-Said F (2013) Colour measurement and analysis in fresh and processed foods: a review. Food Bioprocess Technol 6:36–60

Rey B, Aleixos N, Cubero S, Blasco J (2019) A field robot to detect olive trees infected by *Xylella fastidiosa* using proximal sensing. Remote Sens 11(3):221

Russ J.C, Neal F.B. (2017) The image processing handbook, 7th edn. CRC Press, Boca Raton. ISBN: 9781138747494

Sengupta S, Lee WS (2014) Identification and determination of the number of immature green citrus fruit in a canopy under different ambient light conditions. Biosyst Eng 117:51–61

Shapiro L, Stockman G (2001) Computer vision. Prentice Hall PTR, Upper Saddle River

Sharma G (2003) Digital color imaging handbook. CRC Press, Boca Raton

Xiong J, Lin R, Liu Z, He Z, Tang L, Yang Z, Zou X (2018) The recognition of litchi clusters and the calculation of picking point in a nocturnal natural environment. Biosyst Eng 166:44–57

XRITE (2018) A guide to understanding color. Retrieved from https://www.xrite.com/-/media/xrite/files/whitepaper_pdfs/l10-01_a_guide_to_understanding_color_communication/l10-001_understand_color_en.pdf. Accessed 1 Oct 2018

Yotter R, Wilson D (2003) A review of photodetectors for sensing light-emitting reporters in biological systems. IEEE Sensors J 3:288–303

Zhao C, Lee W, He D (2016) Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. Comput Electron Agric 124:243–253

Zhuang J, Luo S, Hou C, Tang Y, He Y, Xue X (2018) Detection of orchard citrus fruits using a monocular machine vision-based method for automatic fruit picking applications. Comput Electron Agric 152:64–73

Chapter 3
Sensors II: 3D Sensing Techniques and Systems

Manoj Karkee, Santosh Bhusal, and Qin Zhang

3.1 Introduction

In Chap. 2, we focused on understanding objects, recognizing them, and delineating or differentiating one type of object from another kind using color images without their depth information. Such 2D sensing and image processing techniques are essential for monitoring, evaluating, and/or understanding the objects and environment. A lot of real-world problems could be addressed quickly with 2D vision techniques. However, there are a lot of other applications where depth information of objects is essential. For example, a robotic application such as picking apples (Silwal et al. 2017; Silwal 2016), weeding vegetable crops (Chen et al. 2018; Chen 2019), and bundling red raspberry canes (Khanal 2018; Bhusal et al. 2018; Khanal et al. 2019) require both detection and localization (3D) of objects such as apples, weeds, and canes.

A robotic system used in agricultural and field environments generally consists of some or all of: (i) a vision system to detect and localize target objects and obstacles that may be on the way to get to the objects (e.g., fruit and branches in robotic apple harvesting); (ii) a manipulation and end-effector system for reaching and engaging with the target objects; (iii) a path planning and control system for robotic manipulation; (iv) pre- or postmanipulation object handling system (e.g., a conveyance system to bring harvested fruit onto a container/bin); (v) a vehicle/platform with navigation and guidance system for mobility; and (vi) a surround awareness system for operator safety. Various color, texture, geometric, spectral, and other parameters of the operating environment need to be measured during this process

M. Karkee (✉) · Q. Zhang
Washington State University, Prosser, WA, USA
e-mail: manoj.karkee@wsu.edu

S. Bhusal
Harvest Croo Robotics, Tampa, FL, USA

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_3

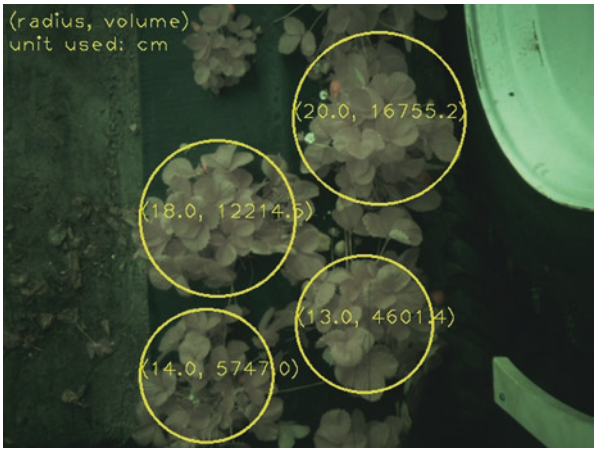


Fig. 3.1 Harvest CROO Robotics, Tampa FL utilizing 3D image technologies to estimate different canopy parameters of a strawberry plant. (Image courtesy of Bob Pitzer, CTO, Harvest CROO Robotics)

(e.g., plant/canopy size; Fig. 3.1). A 3D sensing/measurement system is essential and provides the backbone to implement all or most of these aspects of an agricultural and field robotic system.

A 3D sensing system includes a sensor(s) for capturing 3D information and a data/image processing technique to recreate a 3D environment (represented in a digital form) and locate objects of interest in the scene. Wide varieties of sensing techniques and sensor systems are available for 3D measurements, including color cameras/stereo vision techniques and time-of-flight (ToF) of light-based sensors such as Laser and 3D camera. These sensors are divided into *active* and *passive* systems depending on if separate illumination source is required to operate these sensors. *Passive* sensing systems consist of sensors that operate with the light energy that is available in the environment (does not need a dedicated lighting source). These sensors have a receiver that can record the electromagnetic energy reflected/emitted by the objects of interest in the given environment (e.g., stereo vision system based on color cameras). The light source, often, is the sunlight, mainly when we are using a vision system in the outdoor environment. However, increasingly, researchers and engineers are using artificial lights in the outdoor environments as well to both improve the uniformity of lighting conditions and to add the capability for nighttime operation (Silwal 2016; Wang et al. 2013). *Active sensors* (e.g., a Laser) emit an electromagnetic wave and generally receive it back after it gets reflected from the target. As discussed later, time or phase lag occurred between the emitted and received signals can be used to estimate the distance between the sensor and the objects.

Raw data collected with these sensors are processed in specific ways corresponding to the measurement technique used to create a 3D point cloud and to reconstruct (digitally create) 3D environment, which can be used both to identify objects and locate them in the 3D space. For example, there has been a study to identify apples using only 3D information (Soria et al. 2017). In addition, color, 3D, and other information/attributes of objects can be fused together to improve the accuracy of identifying target objects. For example, in various agricultural applications, 3D

information has been applied to remove unwanted background and keep only the areas of interest (e.g., remove the tree canopies and fruit in the rows behind the one being considered). This process has helped improve the accuracy of detecting and localizing objects of interest (e.g., apples in the trees adjacent to the camera). Finally, 3D information could be mapped onto color images and/or other information such as objects identified using a color vision system discussed in Chap. 2. As discussed before, the presentation of 3D information in the form of the reconstructed scene (Nielsen et al. 2011), 3D point cloud (Comba et al. 2018), or overlay with a 2D image become essential when any object manipulation is necessary using automated or robotic systems. In the following sections, major principles of distance measurement are introduced, followed by more detailed discussions on commonly used sensing systems/techniques and several case studies highlighting the importance of 3D vision systems in agricultural and field robotics.

3.2 3D Measurement Principles

There are many sensors or sensing systems that are available commercially[1] for 3D measurements. Increasingly large numbers of sensing systems have now shown capabilities for accurate measurements in outdoor and field conditions in addition to an indoor and structured environment. The capability of these sensing systems ranges from object presence/absence characterization (e.g., an ultrasonic system) to single point depth measurement (e.g., a Laser depth sensor) to create a 3D point cloud with RGB-D (Red, Green, Blue, and Depth) cameras. All of these sensing systems, however, operate with a few basic principles of 3D measurement. In this section, some of the major 3D measurement principles are introduced. In the following sections, individual sensing systems will be discussed in more detail.

3.2.1 3D from 2D Images

Among a few ways 3D information can be measured, generating 3D from 2D images (e.g., stereo vision, discussed in Sect. 3.3) is one of the most widely applied techniques (Fig. 3.2). In this technique, a number of images collected from the multiple perspectives of the object/environment will be mapped together using image correspondence (to be discussed in Sect. 3.3.4), which is a process to identify corresponding object points projected onto multiple 2D images. Features such as corner points of objects can be identified and used to establish the correspondence between multiple images, which can then be used to develop a trajectory of features

[1] **Disclaimer:** Commercial product mentioned in the chapter is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated.

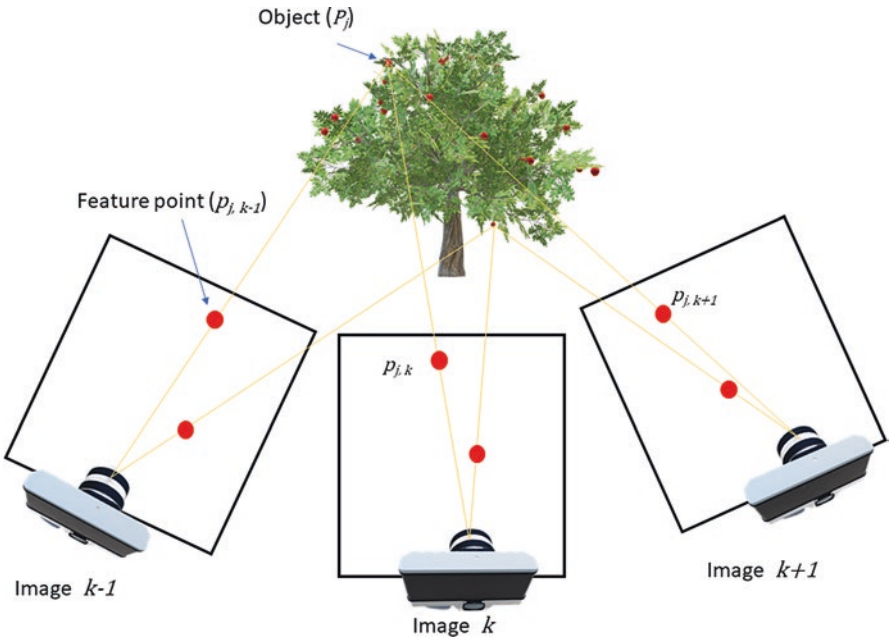


Fig. 3.2 A general framework for creating the 3D model using multiple 2D images

in the space leading to the 3D representation of the objects and scene. To be able to create a 3D trajectory of features, relative location, and orientation of the cameras while taking multiple images should be known.

One of the main advantages of this technique is that the 3D information can be generated for an entire scene at once using regular color cameras, which are comparatively more affordable. This technique also provides inherently co-registered 3D, color, and other information (e.g., infrared) which could provide a powerful tool for both object detection, classification, and localization. The 3D measurement resolution is much higher with this technique compared to other techniques discussed below, and the depth measurement range can be longer than some of the other techniques. However, the sensing structure with this technique can be relatively more complex and the computational time comparatively higher. Based on this technique, a large number of stereo vision systems have been commercialized for decades (e.g., Pointgray, FLIR Systems, USA). Recently commercialized RGB-D cameras such as ZED (Stereolabs Inc., San Francisco, USA) also operate with this principle and provide a much robust solution for outdoor applications at affordable costs. Sensing and image processing techniques for stereo vision-based 3D measurement are discussed in Sect. 3.3.

Structure from Motion Structure from motion is just another way 2D images are collected and used in creating 3D. Rather than using two or more cameras capturing multiple images, one camera is moved around the scene to scan multiple views of

objects. While doing so, the position and orientation of the camera at every instance of imaging would be known. Alternatively, there could be a stationary camera with objects moving around the camera with a known location and orientation. A sequence of images collected this way will then be matched using key image features, as discussed above. Due to the structural, operational, and processing complexities (long processing time), this technique has been less widely applied in the past in agricultural and field robotics. However, this technique has been well utilized in recent years in crop phenotyping applications as it offers a more affordable solution for this increasingly important area of research and development (e.g., Santos and De Oliveira 2012). More discussion on crop phenotyping techniques can be found in Chaps. 6 and 13.

3.2.2 3D with Time-of-Flight of Light

There are a large number of 3D measurement sensors/systems that operate on the principle of time-of-flight (ToF) of light. As the speed of light is known, the time elapsed for the light to travel between two points is the basis in this technique to measure the distance between two points. In any sensor operating with this principle, electromagnetic wave(s) at a certain wavelength(s) is generated and emitted by one unit of the system and is received by another unit to complete the process. The signal received is then compared with the original signal emitted by the source unit to estimate the time lag, which can then be converted into the distance between two points (Eq. 3.1).

$$D = c * \Delta t, \tag{3.1}$$

where D is the distance travelled by the signal, c is the speed of light, and Δt is the time elapsed.

For a lot of semiconductor devices, measuring the phase difference between emitted and received signal might be easier than estimating time elapsed. As an alternative to Eq. 3.1, distance calculation with estimated phase difference can be written as (Eq. 3.2):

$$D = c * \frac{\Delta\theta}{2\pi f}, \tag{3.2}$$

where $\Delta\theta$ is the phase shift between emitted and received signals and $f = 1/T$ is the frequency of the wave used.

The signals used in ToF sensors could be sinusoidal wave (Fig. 3.3a) or square wave (Fig. 3.3b). When a sinusoidal wave is used, the phase shift runs in a cycle between zero to 360°, and it is crucial to find out if there has been more than one full cycle of phase shift between the two signals. There are specific techniques used to solve the ambiguity with phase shifts, which will not be discussed in this chapter.

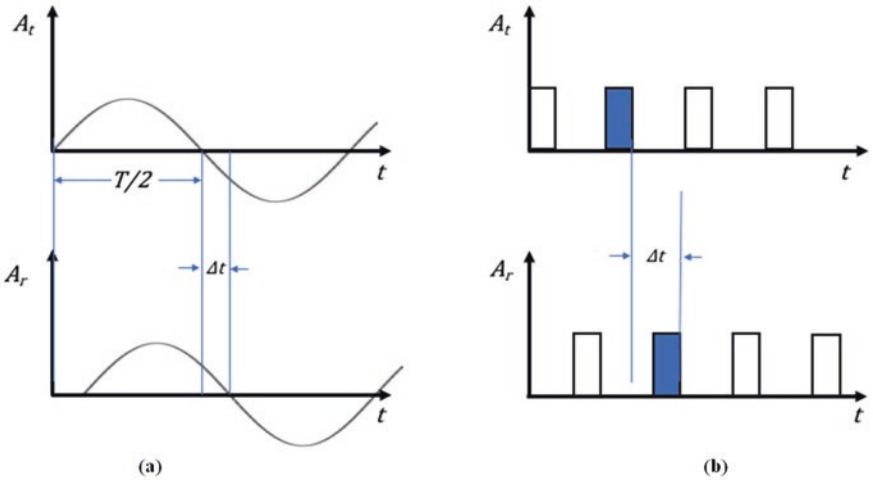


Fig. 3.3 Example waveforms used in measuring object distance using the principle of time-of-flight (ToF) of light; (**a**) sinosoidal wave; and (**b**) square wave

Without the solution for resolving phase ambiguity, the maximum distance (or the range) a ToF sensor can measure is given by (Eq. 3.3):

$$D_{\max} = \frac{c}{f} \tag{3.3}$$

If square waves are used, leading edges or trailing edges are detected to compare between what was emitted and what was received. As shown in Fig. 3.3b, it took Δt for the signal (trailing edge to trailing edge) to travel between the emitter and the receiver, which could be used in Eq. 3.1 to solve for the distance (range) between two points.

In developing geometry for ToF sensors, the source/emitting unit and receiving unit could be located together (e.g., in the case of a Laser sensor) or separated apart (e.g., in the case of a GPS). If the two units are located together, the signal emitted by the sensor travels to the object and the receiver records the signal reflected by the target. In this case, then, the distance to the object from the sensor would be half the total distance travelled by the light. Using Eq. 3.2, the distance d to objects from the sensor can then be given by the following equation:

$$d = \frac{D}{2} = c * \frac{\Delta\theta}{4\pi f} \tag{3.4}$$

ToF principle leads to one of the most accurate distance measurement systems for close range applications. Being an active sensing technique, sensors developed with this principle are also generally less influenced by other light sources and operate well in variable lighting conditions common in outdoor/field applications.

Sensors with this principle will also have inherent capability to operate in day and nighttime, which is not the case with the color imaging–based technique discussed above. One of the drawbacks is that this technique doesn't incorporate color information of the environment with the 3D information measured. The technique may be also more expensive to develop sensing system such as LIDAR that can be used to recreate 3D structure of the entire scene at the desired resolution. Some of the important 3D sensing/measurement technologies operating with this principle include Laser/LIDAR (light detection and ranging), global positioning system (GPS) or, more generically, global navigation satellite systems (GNSS), 3D cameras, and radio wave–based sensors. Some RGB-D sensors such as *Kinect V2 (Microsoft Inc., Redmond, WA)* also operate with this principle.

3.2.3 Structured Light

Structured lighting is the method of using active illumination of the scene with a varying light intensity pattern for 3D imaging (Valkenburg and McIvor 1998). With this technique, a known pattern (spatially varying) of light is generated by an illumination unit (Fig. 3.4). The intensity of each pixel on the structured light pattern can be obtained during calibration (Geng 2011). A sequence of these patterns are projected on to the environment similar to LCD video projectors (Salvi et al. 2004), which will be deformed by the objects in the environment differently based on their relative location and orientation. A camera is used to capture the scene illuminated by the structured light. For a smooth imaging plane, the captured structured light will be same as the projected illumination. However, if a 3D object is under illumination, the surface geometry of the object will distort the projected structured light. This distorted or deformed light pattern in these images will be used for triangulation to obtain the 3D surface geometry. As shown in Fig. 3.4, the geometric

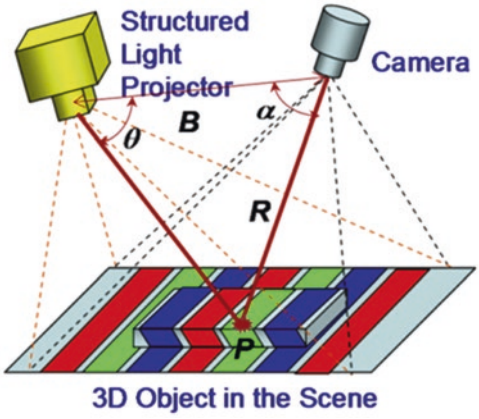


Fig. 3.4 3D imaging technique using structured light. (From Geng 2011)

relationship between an imaging sensor, a structured-light projector, and an object surface point can be expressed (Geng 2011) by the triangulation principle as:

$$R = B\frac{\sin\theta}{\sin(\theta+\alpha)} \tag{3.5}$$

Sensors operating with this principle are generally fast and accurate but are more appropriate for indoor condition as variable outdoor/field lighting condition can interfere with the patterned light used by the system. Since a projector is used to illuminate the entire scene, power consumption by these cameras is high, often requiring a dedicated power source. The technique is also more suitable for close range depth measurement as the accuracy decreases with increasing distance to objects and can operate in both day and nighttime. A few commercially available sensors operating with this principle include Kinect V1 (Microsoft Inc., Redmond, WA) and Intel RealSense SR300 (Intel Corporation, Santa Clara, CA).

3.3 Stereo-Vision System

3.3.1 Introduction

Let's take a moment and give a look at the image in Fig. 3.5a. If one has practiced single-image stereo viewing in the past, it would not be difficult to see the 3D environment represented by this picture (technically a pair of pictures, which will be discussed later). There are two diamond-shaped objects (dices) and concentric circular hills, all in 3D, in the scene. One of the dices is in the center of the smallest circle and another one towards the lower-left corner of the environment. To be able to see the 3D environment, we need to focus behind the physical surface (paper or computer screen) of this image so that our left eye can see a slightly displaced version of the image compared to what the right eye sees.

Let's also look at another image pair presented in Fig. 3.5b. These images are created for what is called parallel stereo viewing, in which the left eye will be looking at the left image and the right would be looking at the right image (another technique would be cross-viewing with a flipped sequence of images). In this case of parallel viewing, eyes are focused behind the surface of these images to bring the two images together, which makes the brain assume that these two images are created by two eyes in a natural environment. Such a technique creates a sensation of a real 3D environment (through processing by the brain) very similar to how our eyes work in an actual 3D environment. This process is called the stereo vision technique, which is one of the most commonly used 3D sensing techniques in agricultural and field robotics.

Similar to stereo-viewing with human eyes as discussed above, stereo vision systems consist of two same or similar cameras placed right next to each other

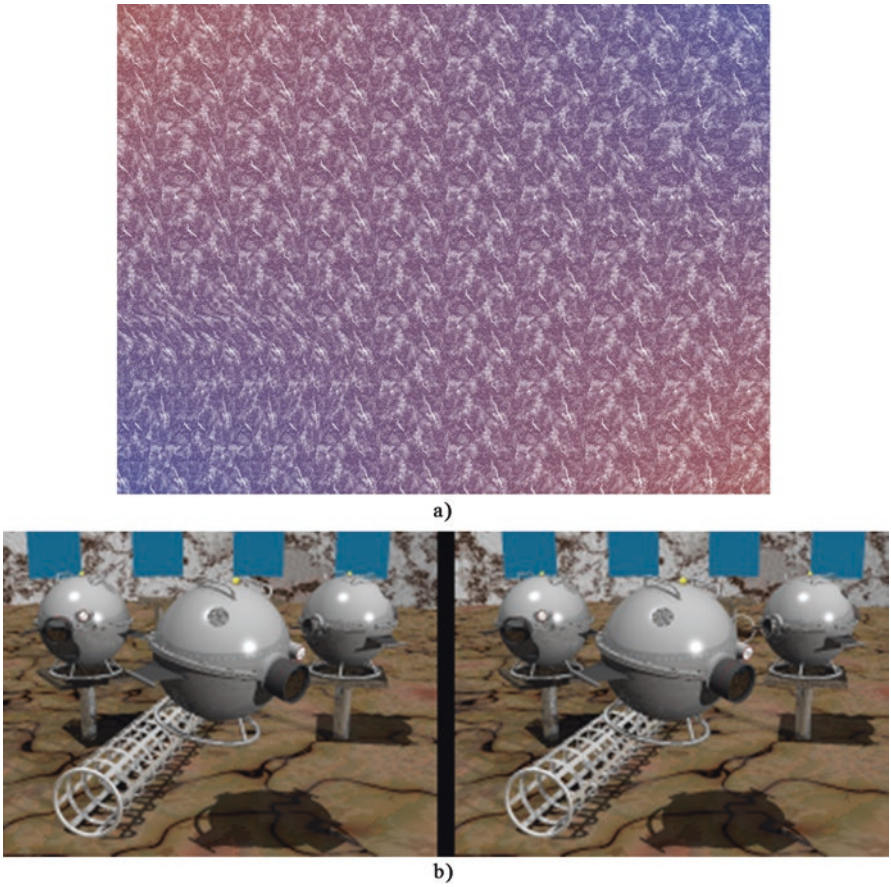


Fig. 3.5 Example stereo vision images; (**a**) single image stereo; and (**b**) parallel viewing stereo pair. (From: http://een.se/niklas/sis/color/alia.jpg)

(separation distance is called *baseline*), which are used to collect two different images (called left and right images) of the environment (e.g., BumbleBee, FLIR Systems; Fig. 3.6; combination of any two cameras of the BumbleBee sensor creates a stereo system). Alternatively, a single sensor/camera could be used to collect image-pair over time with slightly displaced cameral locations. Depending on a range of depth measurement, the baseline, B could be varied. To improve the accuracy in measuring the distance to objects far away, the baseline needs to be wider. The drawback of such a system would be its inability to measure the distance to the objects close to the cameras, similar to how our eyes fail to estimate the depth of the objects that are very close to the eyes. In other words, further apart the sensors, longer the close range of the 3D measurement becomes. Therefore, the baseline is one of the most important parameters to be optimized to achieve the desired measurement range for specific applications.

Fig. 3.6 BumbleBee XB3 stereo vision camera from FLIR Systems used in various research projects at Washington State University (e.g., Bhusal et al. 2018; Hohimer et al. 2019; Khanal et al. 2019)

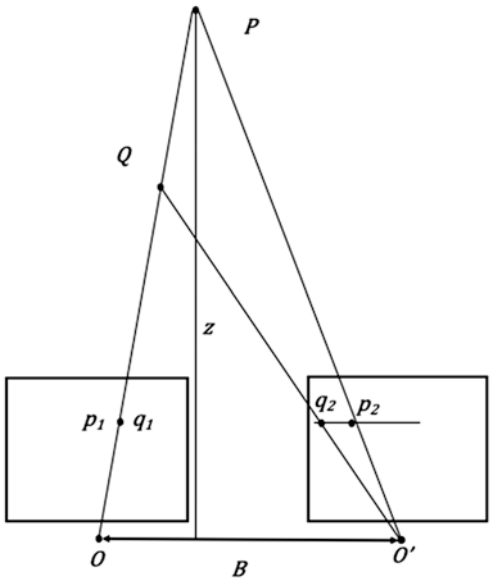


Fig. 3.7 Stereo vision system for estimating disparity between corresponding projections of a point P onto left and right imaging planes

3.3.2 Depth Estimation Using Stereo-Vision Camera

As shown in Fig. 3.7, the *imaging planes* of two cameras are co-planner, their *optical axes* are parallel, and they are separated by a *baseline*, *B*. The cameras are separated such that the majority of the field-of-views (within the desired range of measurement, discussed above) of two cameras overlap. In such a system, if there is an object at a point, *P* in the common field-of-view, it would be projected at point p_1 on the left-side image and p_2 in the right-side image. The distance (in pixels) to p_1 from the left edge of the images is higher in the left image than in the right image. This difference in the corresponding image positions of the same object/pixel location is called *Stereo Parallex* or *Disparity*. Let's assume there is another point, *Q* in the scene, which is closer to the cameras than *P*. For the simplicity of discussion, it has been located in the same line connecting *P* and the focal point of the left camera. That means *Q* would be projected at the same location as *P* on the left image.

However, q_2, projection of Q on the right image, is different than p_2, which is closer from the left edge of the image. This geometric representation shows that the closer the objects are to the sensors, the disparity becomes higher. In general, then, the distance to the object is inversely proportional to the disparity. This relationship provides the basic principle of how depth is measured using a stereo vision camera system.

In Fig. 3.7, f is the focal length (same for both cameras), P is the physical location of the object of interest in the scene with x and x' being the distance to the projection of P on left and right images, as discussed above. The goal of the stereo vision system is to find the distance/depth from the cameras to the object, z. Using the similarity triangles, disparity can be calculated as:

$$\text{Disparity}, d = x - x' = \frac{Bf}{z} \tag{3.6}$$

As the baseline, B is known and so is the focal length f, Eq. 3.6 can be solved for z, if we know corresponding disparity d.

3.3.3 Camera Calibration

Precise camera calibration is required for accurate 3D interpretation of the images, reconstruction of the world models, as well as robot interaction with the world (hand-eye coordination). As discussed above in Sect. 3.3.1, calculating depth with the stereo vision system using Eq. 3.6 requires known focal lengths of both the cameras used in the stereo vision system as well as the disparity between corresponding pixels/objects in the left and right images. Estimating disparity is based on stereo-matching (image correspondence, Sect. 3.3.4), which requires an accurate estimation of a set of stereo camera properties, including both intrinsic and extrinsic parameters. The intrinsic parameters include focal length along both x and y directions (f_x and f_y), principal point or the image center (c_x, c_y), and the skew γ between the two axes. On the other hand, the extrinsic parameters consisting of rotation and translation parameters describe the position and orientation of the cameras in the world. These parameters are often unknown and need to be computed externally through a process called camera calibration.

The pinhole camera model is a geometric model that describes how the points in space, say $\boldsymbol{X} = \begin{bmatrix} u & v & w & 1 \end{bmatrix}^T$, are projected onto the image plane, $\boldsymbol{X}_{\text{img}} = \begin{bmatrix} x & w & 1 \end{bmatrix}^T$. According to this model, $\boldsymbol{X}$ and $\boldsymbol{X}_{\text{img}}$ are related (Szeliski 2010) by:

$$\boldsymbol{X}_{\text{img}} = \boldsymbol{A}\begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \end{bmatrix}\boldsymbol{X}, \tag{3.7}$$

where A (Eq. 3.8) is a 3 × 3 matrix containing the intrinsic parameters of the camera, $\boldsymbol{R}$ is a 3 × 3 rotation matrix, and $\boldsymbol{t}$ is a 3 × 1 translation vector.

$$\boldsymbol{A} = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3.8}$$

As mentioned before, $\boldsymbol{R}$ and $\boldsymbol{t}$ are commonly referred to as the extrinsic or lens distortion parameters of the camera. $\boldsymbol{R}$ is an orthonormal matrix with $\boldsymbol{RR}^{\mathrm{T}} = \boldsymbol{I}$ and $|\boldsymbol{R}| = \boldsymbol{1}$. Completely parameterizing the intrinsic and extrinsic parameters is not a straightforward process, and interested readers are advised to refer to other materials (e.g., Hartley and Zisserman 2003; Szeliski 2010) to understand the mathematical formulation behind them.

Pinhole camera model leads to simpler camera architecture and low-cost systems where straight lines in the scene will be a straight line in the image. However, modern camera systems are designed with different lenses that result in significant image distortions as discussed before. In general, radial and tangential distortions become noticeable due to lens optics. The presence of radial distortion introduces the "fish-eye" effect in the image. As a result, the images are observed curved inward (Barrel distortion) or outward (Pincushion distortion) or a combination of both (Mustache distortion) (Szeliski 2010). The tangential distortion, on the other hand, occurs due to imperfection in the camera lens and the imaging plane and leads to visual curvature in the straight lines (Szeliski 2010).

Since a stereo vision system consists of a pair of cameras (referred to as left and right camera), intrinsic parameters of both cameras need to be known beforehand. Let A_1 and $\boldsymbol{A_2}$ be the intrinsic parameters of the stereo camera system or left and right cameras. In terms of extrinsic parameters, the stereo camera calibration process will compute a single set of rotation and translation matrix representing stereo pair. This information will be used to compute the point/image correspondences (Sect. 3.3.4) between the left and right images. If a 3D point in the world coordinate is projected at $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ on the left and right image, respectively (Fig. 3.7), then the correspondence problem can be defined as:

$$\boldsymbol{p_1}^T \boldsymbol{F} \boldsymbol{p_2} = 0 \tag{3.9}$$

Now that we have defined all necessary unknown parameters ***(F, A_1, A_2, R, and t)*** for the stereo vision system, the next step would be to compute these parameters. For this chapter, we present a 2D camera calibration method that uses a series of images to compute the intrinsic parameters and the lens distortion parameters. We will skip the linear algebra on how to compute these parameters and focus more on the software or GUI implementation using MATLAB and OpenCV utilities using checkerboard images (Fig. 3.8) captured at different orientations with respect to the stereo camera. Another popular approach for calibrating camera using circular points is described in Meng and Hu (2003). Readers can also go through reference books such as Hartley and Zisserman (2003), Szeliski (2010), and Zhang (2000) for more a detailed discussion on the calibration process, including the mathematical formulation.

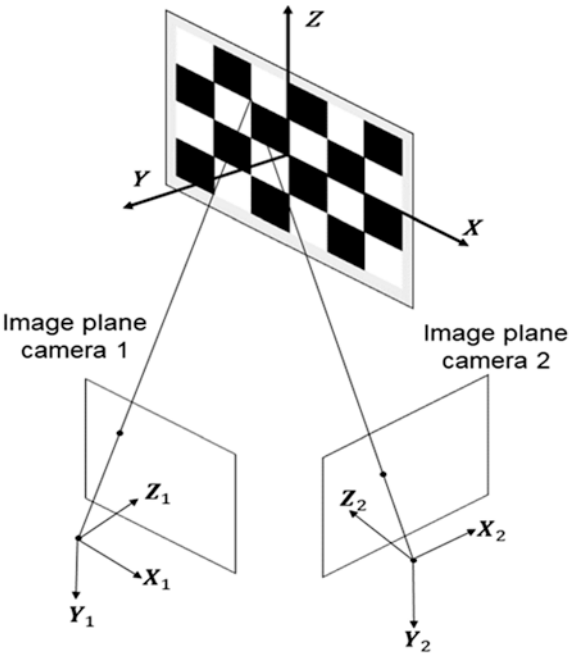


Fig. 3.8 A camera and checkerboard configuration used for camera calibration

A rectangular (rather than a square) checkerboard is desired (especially when using MATLAB) as it allows the camera calibrator to determine the orientation of the checkerboard patterns. The corner of the world coordinate system is set to the corner of the checkerboard. Before beginning the imaging process, it is important to note the length of each square unit of the checkerboard and the rectangular dimension of the checkerboard (number of checkerboard corners along the x and y direction). The checkerboard surface is prepared to be as flat as possible to minimize the impact of surface imperfection on the calibration process. Also, any camera settings that change the camera focal length must not be used during checkerboard imaging as the change in focal length of the camera tend to change the size of the checkerboard in the image. In general, all camera settings are kept consistent throughout the imaging process to minimize any unwanted variations between images. While capturing images, it is recommended to place the calibration checkerboard at a distance close to the intended depth of objects the 3D vision system is used for. To avoid the impact of compression, a lossless image compression format is preferred. During the calibration imaging, the checkerboard is placed such that the board appears towards the edges of the images, which is essential to account for the lens distortion discussed earlier. In general, 20–30 image pairs are captured for calibration with varying checkerboard orientations, making sure the checkerboard is fully visible in both camera field-of-views.

Once the stereo image pairs are ready, MATLAB's Stereo Camera Calibrator App (Fig. 3.9) can be used to calibrate the images. The Stereo Camera Calibrator App utilizes the information provided by the user (checkerboard dimension and

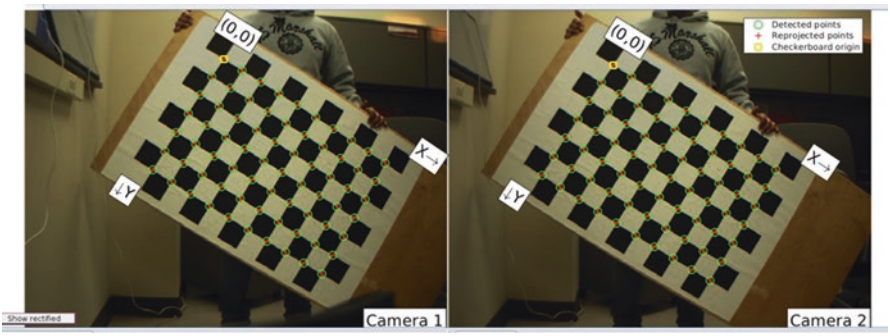


Fig. 3.9 Checkerboard images used for camera calibration in MATLAB. (Images from Bhusal et al. 2018)

length of the square) to create a model of checkerboard at Z = 0 plane in each image representing the world coordinate. These coordinates are later mapped with the detected checkerboard corners within the App to establish the relationship between the 2D detected corner points to the world coordinates, which then can be used to calculate the intrinsic and extrinsic parameters. Before exporting the intrinsic and extrinsic parameters, it is important to evaluate the calibration results from the App. There might be cases when the App fails to detect corners or detect different corners for origin. With the App, one can examine reprojection errors and camera extrinsic parameters as well as view and refine the calibration images. Once the calibration process is complete, the App outputs both the intrinsic as well as extrinsic parameters of the stereo vision system along with the distortion parameters. The Brown–Conrady model (Brown 1965; Conrady 1919) is used to correct radial distortion and for tangential distortion and generally higher-order coefficients are not considered. The App outputs three radial distortion coefficients and two tangential distortion coefficients. Once all the intrinsic and extrinsic parameters are computed, the stereo pair can be used to estimate depth or recreate 3D environment.

3.3.4 Image Correspondence

One of the most important steps in estimating depth with the stereo vision system is matching the pair of images so that corresponding pixel locations can be found between two images. This process is called image correspondence or stereo-matching. The matched images will then be used to generate a disparity map, which, as mentioned before, represents the separation between corresponding object/pixel locations in two images. The accuracy of this map is directly related to the accuracy in the depth estimation, and therefore, how accurately two images can be matched becomes vital for the overall performance a stereo vision system. The pair of images collected by two cameras of a stereo vision system need to have a huge amount of

overlap so that corresponding or matching pixels can be located between two images with a greater ease.

Image correspondence or stereo image matching is performed using various statistical and nonstatistical techniques. One such statistical method is called cross-correlation (Eq. 3.10). This is one of the oldest techniques used in stereo-matching. As can be seen from the equation, cross-correlation is a process of convoluting one image with a part of another image of the stereo-pair. In other words, a small section (user defined) of the first image is compared against a small section of the second image, pixel by pixel, which is repeated by moving the small section of the first image over the entire second image. Through this process, a particular section/instance in the second image that generates the highest correlation or the best matching between two sections is found. This process is then repeated for each section in the first image so that a complete matching between two images can be created.

$$G(x,y) = F(x,y) \otimes H(x,y); G(x,y) = \sum_{i=-w/2}^{w/2} \sum_{j=-h/2}^{h/2} F(x+i, y+j) H(i,j) \quad (3.10)$$

Feature-Based Matching It is another approach used in stereo vision systems. In this approach, rather than trying to match entire images, specific features of objects such as sharp corners, lines, and center of objects that are prevalent in the stereo-pair are identified and matched. In this approach, specific objects and/or features of interest are first segmented out (see Chaps. 2 and 13 for segmentation techniques) in each of the two images. For example, if we are using a vision system for robotic apple harvesting, apples would be the objects of interest. Once the apples are segmented out using color images, individual apples (based on their center/location, size, and shape) could be matched in two stereo images. If the features/objects identified can be uniquely defined (e.g., using color, shape, size, and texture features), stereo-matching can computationally be more efficient and the matching results could be more accurate with this technique compared to the same with the cross-correlation-based approach. It is important, however, that the spatial relationship between features/objects be matched as well to minimize false matching between objects. In addition, feature/object-based matching would provide disparity map and then distance of only specific locations within the field-of-view. For example, in the case of matching with apples as objects of interest, only areas covered by apples could be matched well and could lead to depth estimation with higher accuracy. If 3D measurement is necessary for other regions in the environment, interpolation-based technique or other matching methods may have to be used to fill the gaps.

Another popular and most recently, widely used approach for image correspondence is *semi-global matching* (SGM) (Hirschmuller 2008). For rectified image pair, the cost of matching a base image (let's say left image) pixel p with all the pixels on right image on Epiplolar line (to be discussed below, Sect. 3.3.4) is computed using some similarity measures such as Mutual Information (MI) (Viola and Wells III 1997). Thus, for each base image pixel and its potential correspondences in the right image, matching results in a 3D cost structure. Cost calculation can

result in ambiguous match and results in wrong disparity estimate. This method assumes that the observed disparity surfaces are quiet smooth so that any disparity shift or irregularities can be penalized. The final matching cost can be aggregated from all directions in the neighborhood, and a mach pixel is chosen that closely matches with the cost of neighboring pixels. The number of directions or the neighborhood considered affects the computation time of this algorithm. In general, computation from 16 directions ensures good quality of disparity estimation; however, a lower number of directions can be used to achieve faster computation.

3.3.5 Epipolar Geometry

As discussed before, accuracy of image correspondence is crucial for the overall accuracy of 3D measurement with stereo vision systems. To minimize the complexity and improve the accuracy of correspondence, a technique called *Epipolar geometry* is used (Shapiro and Stockman 2001). Using a specific Epipolar geometry in designing the stereo vision system, search space for corresponding pixels could theoretically be reduced down to a line. Basically, if the relative position and orientation of two cameras with respect to each other and their internal parameters, including focal length, are known, then a matching axis called Epipolar axis between the two images can be found, as shown in Fig. 3.10. As shown in the figure, two images were acquired by two cameras with their centers C_1 and C_2. The location P in the field-of-view was imaged at p_1 in the left image and p_2 in the right image. When two camera centers C_1 and C_2 are connected (which is the baseline), the line intersects the images at E_1 and E_2. Based on this geometry, it is found that any point in the line P_1E_1 in left image would have a matching point located along P_2E_2 in right image. This technique, therefore, minimized the search space down to a line from the entire image. Stereo vision system uses a specific example of Epipolar geometry created by two cameras with same internal parameters and orientation separated by a short distance (baseline) along their imaging planes (orthogonal to their optical axis; Fig. 3.7). With such a geometry, the search space for finding a corresponding pixel would be a horizontal line at the same pixel location in the vertical direction. The Epiplolar geometry is utilized by various image correspondence algorithms, such as the SGM technique discussed before.

3.3.6 Tools for Stereo-Vision-Based Distance Measurement

MATLAB and OpenCV (open source computer vision library) are some of the most used tools by engineers and scientists working in agricultural and field robotics area for 2D and 3D image processing. As discussed before, the first step for depth

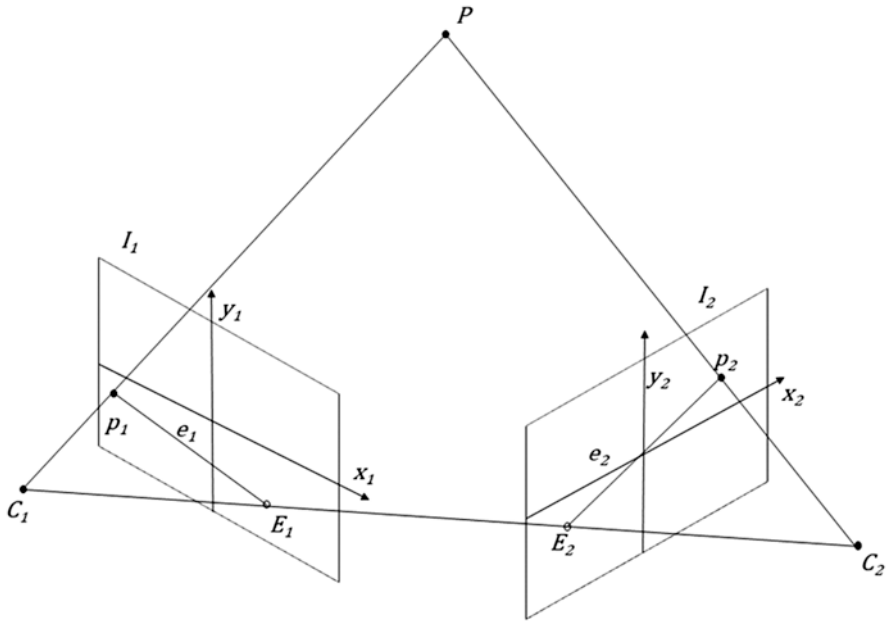


Fig. 3.10 Epipolar constraints. (Adopted from Shapiro and Stockman 2001); the projection of point *P* in the left image is at P_1, which would be found along epiline E_2P_2, in the right image

estimation with a stereo vision system is to minimize the radial and tangential distortions in the images captured by the stereo camera(s). MATLAB provides "*undistortImage*()" and OpenCV provides a similar function "*undistort*()" to improve images by minimizing distortions or to compensate for issues related to lens imperfection. Next, images can be rectified, which is a process to make the images Epipolar (Sect. 3.3.4). The function *stereoRectify*() in OpenCV and *rectifyStereoImages*() in MATLAB can be used for stereo images rectification. After the images are rectified, image correspondence is performed, as discussed in Sect. 3.3.4. Both MATLAB and OpenCV provide functions for image/pixel correspondence and to compute disparity between the two rectified images. Readers are encouraged to explore more on *StereoSGBM*() function and *disparityMap*() functions in OpenCV and MATLAB, respectively, which utilizes semi-global matching for image correspondence. Once disparity maps are generated, object depth can be computed easily using Eq. 3.6 (with known focal length and baseline) that can be easily implemented in MATLAB or any other tools).

3.4 Other 3D Measurement Systems

3.4.1 Visual Servoing

Particularly in manipulative robotics (where robots use some sort of manipulator and end-effector to miniplate objects), approaching the target object can be achieved by repeated acquisition and use of 2D images of the target object and its surrounding. First, the general target locations/objects are detected in 2D images using image processing techniques discussed in Chaps. 2 and/or 13. Once the targets are located in 2D images and approach sequence is determined, manipulator movement towards the target object is guided based on the instantaneous object location in the 2D image sequence as a feedback to the manipulator control system (Ringdahl et al. 2019). This technique of approaching target objects/locations using repeated use of 2D images collected using an end-effector-based imaging system is called visual servoing. For example, in a fruit picking robot utilizing a visual servoing technique, approaching a fruit would involve detecting the fruit and its position (often in 2D image space) on a regular basis and actuating the robotic manipulate joints in such a way the target fruit always remains at the desired image coordinates.

Visual servoing is a more robust technique for localizing target objects for manipulation than other 3D measurement techniques operating as a global sensing system (camera systems not attached to an end-effector). As the object position is dynamically updated while approaching the object, the technique can ensure reaching the object even in a noisy environment. However, the system can be slower as it requires the acquisition and processing of images during the approach. Dynamic adjustment of manipulator joint positions also reduces the efficiency at which the robotic manipulator can be moved to the target. In a global camera system–based manipulation without a visual servoing technique, however, objects might be missed if the environment is noisy, 3D measurement system has errors and/or uncertainties, and/or unintended movement in objects occur between imaging and the intended instance of manipulation (e.g., wind gusts causing an apple to move before the end-effector reaches the fruit). More discussion on visual servoing can be found in Chap. 9.

3.4.2 Laser and LIDAR

Laser and LIDAR (light detection and ranging), operated with the principle of ToF, are one of the most accurate techniques for 3D measurements. These two techniques are basically the same in terms of how they function, and the terms are often used interchangeably. In this chapter, we define Laser to represent the basic unit that uses a single beam of light to estimate distance to a single point in the environment. LIDAR, on the other hand, would include a scanning mechanism so that the device could be rotated around to gather depth to the objects around the sensor covering a

specific line and/or area of interest in the environment, not just a single point. As discussed in Sect. 3.2.2, Laser and LIDAR use either continuous ranging (sinusoidal waves; Fig. 3.3a) or pulse ranging (square waves; Fig. 3.3b) to transmit and receive the light back for distance calculation using Eq. 3.4. As discussed before, with this technique, either the time elapsed or the phase shift between the emitted and the reflected lights are estimated, which is then used to calculate the distance travelled by the light using known speed of light. These devices provide highly accurate range measurement in the close distance but are not suitable for long range distance estimation. They are also generally less sensitive to variable lighting conditions common in the field environment and have the capability for nighttime operation. The technology is relatively slow and expensive to build high-resolution 3D environment.

3.4.3 3D Camera

3D cameras also operate using the principle of ToF. However, these sensors have a grid of Laser emitters and sensors that receive all of them back. These cameras, therefore, generate 3D point cloud of the space (within a field-of-view) at once. Contrary to Laser and LIDAR, 3D cameras use thousands or even millions of emitters operating in parallel to create the 3D maps without the need for using a scanning mechanism. These cameras, therefore, are faster in terms of creating 3D structure of a given field-of-view; however, they generally have relatively low pixel resolution, are more costly, and are not well suited to handle shiny objects. Being the active sensors, these cameras also have an ability to function both in day and nighttime. Some examples (Fig. 3.11) include Camcube 3.0 (PMD Technologies, Siegen, Germany) and Azure Kinect (Microsoft Inc., Redmond, WA). These example sensors have resolution around 0.1 to 1.0 megapixels. These cameras have more commonly been used by research communities around the world in the past. Recently, more cost-effective RGB-D cameras utilizing ToF principle (or other techniques such as structured light and stereo vision) have been utilized more

(a) (b)

Fig. 3.11 Example 3D Cameras; (**a**) CamCube 3 from PMD Technologies (Siegen, Germany); and (**b**) Azure Kinect from Microsoft Inc. (Redmond, WA)

commonly in research and development of agricultural robotic solutions compared to 3D cameras.

3.4.4 Global Navigation Satellite Systems (GNSS)

GNSS, including GPS and Galileo systems, also operates using ToF principle. But the distance measurement mechanism is slightly different. For example, our cell phones can estimate their location in the space using GPS technology. However, these devices do not actively emit light and receive it back to estimate their location. In this 3D measurement system, generally, we have an emitting system (satellites) and a receiving unit (e.g., a cell phone). When a receiver receives the signal, it can estimate the time taken by the signal to travel from the satellite to the receiver, which will be used to estimate the distance between the satellite and the receiver. As the satellites have known position, a trilateration technique (Fig. 3.12) is then used to estimate the specific 3D location of the receiver using signals from at least four visible satellites at any given time. Four satellites need to be visible, because the trilateration is being used to solve for four unknown variables, x, y, z and time. In practice, five or more satellites are used (when available/visible) to improve the accuracy of localization.

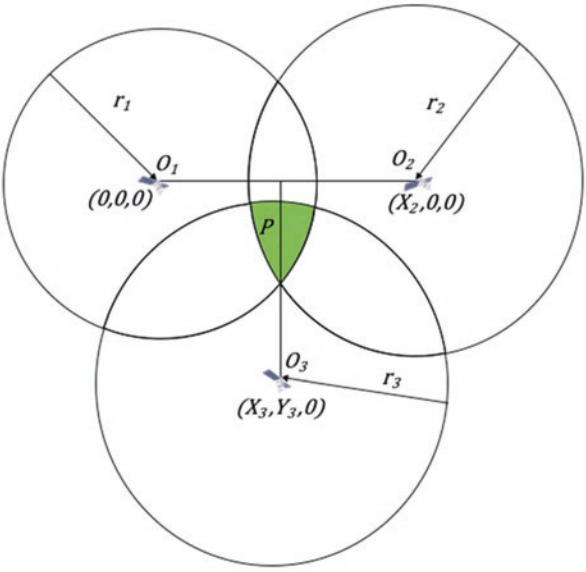


Fig. 3.12 Distance measurement using the GNSS trilateration. Three circles represent the cross-sections of spheres created around three individual satellites with the radius being the respective distances between the GNSS receiver and the satellites. The cross-section of three spheres provides the potential receiver location, *P*

Trilateration for positioning with the GNSS system works as follows: with an estimated distance from a receiver to one GNSS satellite, r_1, the receiver position could be constrained down to the surface of the sphere, with the center being the location of the specific satellite and the radius being r_1. The circles in the diagram (Fig. 3.12) are just the cross-sections of the spheres with a possible receiver location. Next, if the distance from the receiver to another satellite is estimated to be r_2, the receiver could be anywhere on the surface of the sphere drawn the same way. Now, with two spherical surfaces of possibilities, the interacting line between two spheres would define the narrowed down area of the potential location of the receiver. When the distance to a third satellite is estimated, now the intersection between three spheres provides two potential locations for the receiver. Two resolves the ambiguity between two locations, known position of earth can be used as one of the two potential locations of the receiver would be impossibly far away from the earth's surface.

Assuming the location of the first satellite be (0, 0, 0), the second be (X_2, 0, 0), and the third be (X_3, Y_3, 0) and using the algebraic representation of a sphere, trilateration can be represented as follows (Eqs. 3.11, 3.12, and 3.13). In this formulation, the receiver position is defined by $P(x,y,z)$.

$$r_1^2 = x^2 + y^2 + z^2 \tag{3.11}$$

$$r_2^2 = \left(x - X_2\right)^2 + y^2 + z^2 \tag{3.12}$$

$$r_3^2 = \left(x - X_3\right)^2 + \left(y - Y_3\right)^2 + z^2 \tag{3.13}$$

Eqs. 3.11, 3.12, and 3.13 can then be solved for x,y, and z (receiver position, P in 3D) as given by Eqs. 3.14, 3.15, and 3.16.

$$x = \frac{r_1^2 - r_2^2 + {X_2}^2}{2X_2} \tag{3.14}$$

$$y = \frac{r_1^2 - r_3^2 + {X_2}^2 + {X_3}^2 - 2X_3 x + {Y_3}^2}{2Y_3} \tag{3.15}$$

$$z = \mp\sqrt{r_1^2 - x^2 - y^2} \tag{3.16}$$

As mentioned before, however, in practice, more than three satellites are used to estimate time as a new variable and also to improve the accuracy of estimations. Frequently, in the open space, there can be ten or more satellites visible (e.g., in GPS), allowing us to improve positioning accuracy. However, even with such a technique, a regular GPS receiver provides a position with an accuracy of about 4 to 5 m (van Diggelen and Enge 2015). This level of accuracy is sufficient for various applications, such as driving around a city. However, when we are using an auto-steering system in cornfields or vineyards, much higher accuracy would be necessary for achieving the desired field operations without damaging crops.

Differential and Real-Time Kinematic GPS To improve the positioning accuracy of roaming GPS receivers, Differential GPS (D-GPS) and Real-Time Kinematic GPS (RTK-GPS) techniques are used. In D-GPS, a base station with the precisely known position will be used (Fig. 3.13) in addition to roaming GPS receivers. Base stations could be installed in the field or could even be other low-orbit satellites. The location of the base station is estimated using satellite signals as if it were a regular GPS receiver. Because the location of the base station is known accurately, the difference between the actual and estimated location can be calculated as the positioning error at the base station. The promise with this technique is that the atmospheric and other attenuations/errors in the satellite signals will be similar for two receivers that are close to each other. Therefore, the positioning error estimated at the base station can be used to correct/improve the positioning accuracy of other receivers that operate in the close vicinity of the base station. With this technique, GPS positions are estimated with receivers that roam around the base station. The estimated roaming receiver position will then be corrected as a postprocessing (offline) step or in real-time using the correction signal recorded/transmitted by the base station. In a real-time D-GPS, the roaming receivers are capable of receiving both the satellite signals and the correction signals from the base station and apply the correction to the estimated position in real time. The positioning accuracy of the D-GPS system can be less than a meter.

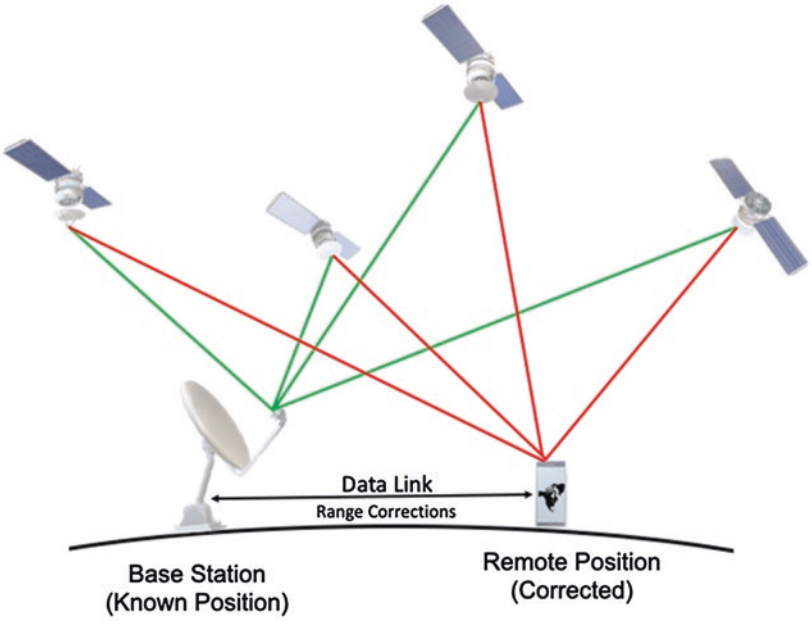


Fig. 3.13 A basic framework for Differential GPS and Real-time Kinematic GPS. The base station (with precisely known position) estimates its position using satellite signals and computes a correction comparing the known and estimated position, which is then sent to a remote receiver for correcting its estimated position

RTK-GPS uses the same principle of operation as D-GPS. However, the correction signals are sent to the roaming receivers using the carrier phase signaling method. The carrier phase information received by two receivers (base station and roaming receiver) can be used to avoid the error in satellite and receiver clocks, which helps further improve the positioning accuracy of the receiver (because that of the base station is already known; Langley 1998). With this technique, roaming receivers could achieve a positioning accuracy of up to 1 to 2 cm (e.g., Lin 2004).

3.4.5 Interferometric Synthetic Aperture RADAR (InSAR)

InSAR (*Interferometric Synthetic Aperture RADAR*) is another technique for 3D estimation, mostly used by satellite systems (called *Satellite Interferometry*). This technique uses radio waves that are generated by two sensors separated by some distance. The configuration could be developed by two sensors in the same satellite or two different satellites flying over the same area slightly offset in space and time. The radio wave generated by the sensors travels to the objects and returns back to both the sensors. The phase difference in the wave received by the two sensors is used to estimate the distance to the object. Because radio waves have a wider wavelength, phase shift analysis could be performed more robustly. However, to find out the distance to objects longer than what could be estimated by one full phase of the signal, a technique called *phase unwrapping* needs to be applied.

The depth estimation process with interferometry, briefly, involves four specific steps. First, images collected by two sensors or collected at two different instances need to be *co-registered*. This step requires matching pixels from one image to another, which is similar to stereo-matching techniques used in the stereo vision systems discussed earlier (Sect. 3.3.4). After images are matched, a phase map is generated, which is also called *interferogram* generation. As the phase map includes cyclic information between 0 and wavelength of the radio wave used, phase unwrapping is performed on the interferogram to create absolute values of the total phase shift. Finally, phase unwrapped maps are converted into depth or elevation maps. Because InSAR operates using radio waves, it has certain advantages, including better penetration of the signal leading to uninterrupted operation in cloudy days as well, which would not be possible with visible signal-based techniques such as stereo vision or stereoscopy. More details on this technique can be found in Karkee (2005). Based on this technique, the US government has generated digital elevation models or altitude information covering almost the entire globe using a mission called Shuttle Radar Topographic Mission with spatial interferometry (same shuttle carrying two sensors separate by some distance). More information on this dataset and mission can be found at https://www2.jpl.nasa.gov/srtm/.

3.4.6 Ultrasonic and Infrared Techniques

There are a few other 3D sensing techniques that have been used in precision and automated agricultural systems for approximate estimation of crop canopy location and size. One of those techniques is the use of ultrasonic or infrared signals to detect the presence or absence of canopy and estimate various canopy parameters, including canopy depth, volume, and foliage density. These sensors also operate in the principle of time-of-flight (ToF) of a specific type of wave being used. An ultrasonic sensor generates a sound wave in a frequency generally inaudible to humans. The signal, after being reflected by the target, is recorded by a sensor to estimate the time difference and, consequently, estimate the distance. Canopy parameters estimated with these sensors can be used to perform various field operations more accurately. One such example operation would be to turn various sections of a chemical sprayer ON or OFF depending on the estimated canopy density and volume in a target region of a tree canopy as demonstrated by Jeon et al. (2011) and Maghsoudi et al. (2015). Sparse representation of 3D structures of tree canopies can also be achieved by a motion tracker that works via the relative displacement of an electromagnetic field. Sinoquet et al. (1997) used this technique to model walnut trees and Vougioukas et al. (2016) to model pear trees for understanding fruit accessibility and harvestability using an automated harvester. The sensor can be used to record three-dimensional positions in space at branch junctions, for example, which can be used to represent the tree shape. The techniques discussed in this sub-section generally provide only coarse resolution data. However, sensors operating with these principles can be simpler and low-cost with much simpler data structure and faster processing speed beneficial to meet various needs of automated or robotic systems in agriculture.

3.5 Case Studies

In agricultural and field robotics, a wide range of 3D measurements systems have been applied covering a great swath of applications, including guidance and situation/surround awareness (e.g., Malavazi et al. 2018); monitoring crop growth and other crop phenomics applications (e.g., Guo et al. 2018); robotic operations in orchards (e.g., Gebbers et al. 2013) such as harvesting (Silwal et al. 2017), training (Zeng et al. 2019; Majeed et al. 2020), pruning (Karkee and Adhikari 2015; Botterill et al. 2017), and thinning (Nielsen et al. 2011); robotics operations in vegetable crop production, including pollination (Yuan et al. 2016), harvesting (Klein et al. 2019), and weed control (Raja et al. 2020; Chen 2019); and robotic operations in vertical and controlled environment farming (e.g., greenhouse and vertical farming); to name a few. There could be varied needs of 3D measurement systems and approaches depending on the type of applications. For applications that need information on only the presence or absence of canopy without regard to identifying individual

canopy objects or parts, nonimaging sensors such as ultrasonic and Laser could provide desired 3D information at a faster speed. As discussed above, these sensors can provide a quick assessment of plant canopy characteristics such as canopy depth, canopy size, canopy shape, and foliage density (e.g., Jeon et al. 2011; Maghsoudi et al. 2015; Rosell Polo et al. 2009; Liu and Zhu 2016). In applications where accurate 3D positioning of a point or a line in the environment is needed, a Laser system could be an ideal option (Jiménez et al. 2000). For example, in identifying trunks for targeted chemical applications in vineyards (Yaxiong et al. 2017) and to estimate the canopy volume in an orchard (Underwood et al. 2016), a Laser sensor installed on a moving vehicle at a certain height might provide sufficient data. A similar example where a plane of Laser data would be of great importance is the guidance of robotic machines through orchards. A Laser scanner could provide object locations on a plane, which could be used in the robotic system to identify the center of the two rows of trees and to detect obstacles in front of the machine (Ye et al. 2017).

On the other hand, 3D location information of the entire environment of interest or surrounding area might be necessary to perform various other field activates, including understanding, operating, and traveling through an unstructured environment. For example, a 3D vision system with the capability to reconstruct entire tree canopies would also be necessary for applications requiring the identification and localization of specific canopy objects (e.g., flower, fruit, leaves, branches, trunks) or specific parts (e.g., a cluster of flowers or fruit, an area of foliage with a particular disease). Commonly used sensors for estimating the 3D locations of objects in such situations include stereo vision camera (Gebbers et al. 2013; Yuan et al. 2016), ToF-based 3D cameras (Silwal et al. 2017), RGB-D sensors (Xiao et al. 2017), and systems to create structure from motion (Dey et al. 2012). In the following subsections, a few robotic applications (case studies) of various types of 3D measurement techniques in agriculture have been discussed.

3.5.1 Crop-Load Estimation in Orchards

Accurate crop-load estimation is essential for efficient management of pre- and postharvest operations in various fruit crops, including apples. This information will help arrange the desired amount of harvesting labor and equipment and also estimate needs for storage and marketing. Crop-load estimation in any fruit crop, such as apples, requires detection and counting of fruit as well as determining the size of the detected fruit. Various object detection approaches (convectional as well as deep-learning-based techniques) are discussed in Chaps. 2 and 13. In this case study, a brief discussion on methods for fruit size estimation in the orchard environment is presented.

In its basic form, distance to a segmented object would be necessary to convert the pixel area occupied by an object into the physical size of the object. Though there have been a lot of studies in detecting fruits such as apples, only limited

studies can be found in utilizing 3D vision systems for fruit sizing in the orchard environment. Gongal et al. (2018) presented one such study using a ToF 3D camera-based system to estimate apple size in tree canopies. An over-the-row imaging platform with artificial lighting was developed to create a uniform lighting environment (Fig. 3.14). The study used the major axis (longest axis) of the apples as the size of the fruit, which was estimated based on (i) 3D coordinates of pixels within segmented apples and (ii) 2D size of individual pixels within apple surfaces. In the first method, 3D coordinates estimated with the 3D camera were used to calculate Euclidean distance between pixels. The maximum distance between two pixels within an apple surface was considered to be the major axis of the respective apples. In the second method, the physical size of each pixel along the major axis of the apples was estimated using a calibration model, which was then summed to estimate the major axis length. The calibration model was developed based on the distance to pixels from the camera and the pixel coordinates. The study found that the 3D coordinate-based method achieved only about 69% accuracy in estimating fruit size in the orchard environment, whereas the accuracy with the pixel-based method was 85%. Other studies in this area include Wang et al. (2013), which scanned apple trees from both sides in a row and detected and registered apples to generate yield estimation.

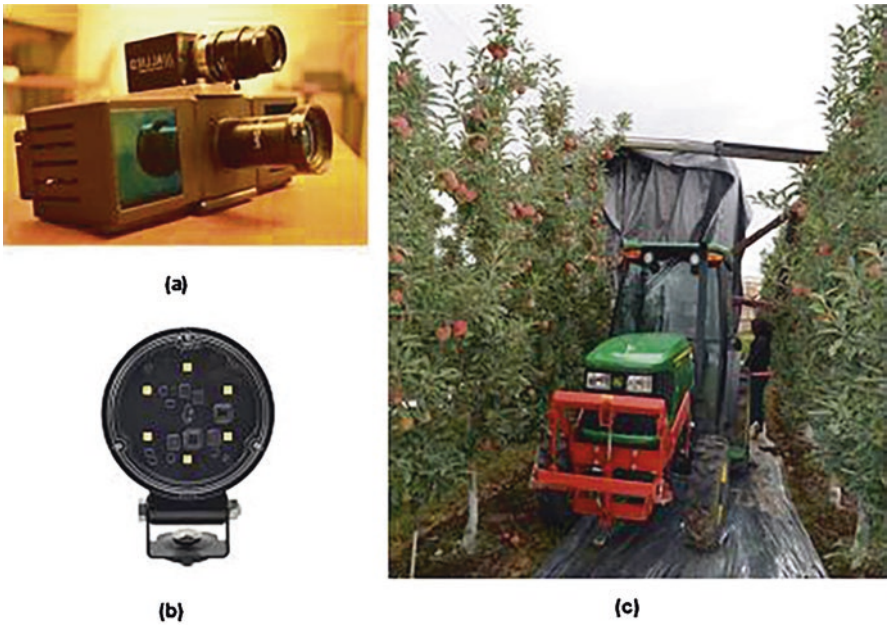


Fig. 3.14 Cameras (**a**), artificial lighting (**b**), and an over-the-row image acquisition system (c) used for apple sizing in tree canopies. (From Gongal et al. 2018)

3.5.2 *Robotic Fruit Harvesting*

Robotic fruit harvesting has been investigated for many decades. However, no commercial success has been achieved so far because of a few major constraints, including lack of the desired level of speed, accuracy, robustness, and high cost of the technology. Fueled by declining the labor force and increasing labor costs and encouraged by the recent advancement in sensing and image processing techniques, including deep learning, and increasingly cost-effective parallel computing platforms, robotic fruit picking has been an area of active research and development over the last decade. Both private companies and university research teams around the world are actively perusing fruit picking research and development for various types of crops, including apples (Hohimer et al. 2019; Silwal et al. 2017), citrus (Lin et al. 2019), kiwifruit (Williams et al. 2019), sweet pepper (Arad et al. 2020), and strawberries (Xiong et al. 2020).

In the recent efforts on robotic fruit picking, various types of 3D measurement systems have been used in detecting and localizing fruit. Sukkar et al. (2019) used an RGB-D sensor (SR 300, Intel Inc., Hillsboro, OR) installed on a robotic arm, which was then moved around a target tree canopy to detect apples using a 3D model created using the point cloud data. Fruit detection was based on features such as shape and size. This technique used images acquired from multiple directions, which improved the robustness of fruit detection and localization. Over the last few years, a transdisciplinary team of researchers from FFRobotics (Haifa, Israel) and Washington State University (WSU) have also been developing a full-scale robotic harvesting machine (Fig. 3.15) utilizing a similar sensor (RealSense D435). The machine used 12 Cartesian robotic arms (linear movement in and out of canopies) supported by 12 RealSense RGB-D cameras, which were selected for their low cost and ease of use while also providing the desired level of accuracy and robustness in the field environment. The research focused on estimating the 3D location of not

Fig. 3.15 End-to-end system developed for robotic fruit harvesting, conveying, and bin filling; (**a**) Intel RealSense 435 Cameras used as the color and depth sensors; and (**b**) the complete harvest system

only apples but also of various canopy objects, including trunks, branches, and trellis wires, which would be essential to create a collision-free path of robotic-end effector to travel to the desired fruit, grab it and detach from the tree.

In the integrated robotic machine, RealSense cameras were installed at the base of individual robotic arms (Fig. 3.15a). The machine was then evaluated in commercial apple orchards in Washington (2019 harvest season) and Israel (2018 harvest season) to harvest entire tree height and transport fruit to bins (Fig. 3.15b, https://youtu.be/YUdqTov9GeU). During the field test, the image acquisition process began by scanning the canopy directly in front of the initial multiarms robot position. As some apples were blocked by other apples, leaves, branches, trunks, and trellis wire and were therefore difficult to be accessed and picked by the robotic hand, a 3D image processing technique was used to locate all potentially obstructing objects (Fig. 3.16). This technique was able to detect and locate apples that were not obstructed by other canopy objects, which could be completely visible and accessible, and may freely be picked by robotic hands. Similar efforts are underway with other teams in fruit picking, including apple harvesting by Abundant Robotics (Hayward, CA), and strawberry harvesting by Harvest Croo (Tampa, FL).

3.5.3 Robotic Fruit Tree Pruning

There have been several studies in creating the 3D structure of fruit trees and identifying pruning branches using various kinds of 3D vision systems. Some of the frequently used systems for 3D reconstruction of fruit trees include ToF-based 3D cameras and RGB-D sensors like Microsoft Kinect 2 (which also operates with ToF). Karkee et al. (2014) and Karkee and Adhikari (2015) developed a skeletonization-based technique to create apple tree structures using 3D point cloud data acquired with a ToF 3D camera. They used a pan-and-tilt system to increase the field of view of the camera system so that entire tree canopies could be imaged and reconstructed. The topological structure of trunks and branches created with this

Fig. 3.16 (**a**) Apples marked in the image are identified and located to be safe to harvest in the given view; (**b**) Trellis wire and trunk detection and localization to avoid end-effector and trellis wire collision

technique was then used to develop and implement tree pruning rules utilizing branch spacing and branch length. Elfiky et al. (2015), Chattopadhyay et al. (2016), and Akbar et al. (2016b) used a ToF-based Kinect camera to perform a similar task of creating apple tree structures. Reconstruction of 3D structures of apple trees utilized KinectFusion API (Izadi et al. 2011) for surface reconstruction, and then skeletonization and feature extraction (Elfiky et al. 2015). The work then led to estimating branch diameter and other canopy features that could be useful for designing and implementing automated pruning process for apple trees (Chattopadhyay et al. 2016; Akbar et al. 2016a). It is noted that Microsoft has recently discontinued Kinect sensors, but there are other similar cameras such as ZED camera showing promise to continue using low-cost RGB-D sensing for agricultural and field robotics.

LIDAR sensors (Medeiros et al. 2017) and multiple 2D image-based 3D reconstruction techniques (Tabb 2013) have also been successfully used for estimating fruit tree shape for pruning. Medeiros et al. (2017) used one-dimensional LIDAR to acquire large depth maps of fruit trees, which was then used to estimate branch diameter, branch length, and other geometric features of tree canopies. Tabb (2013) and Tabb and Medeiros (2017) used multiple color cameras to acquire apple tree images, which was then used to reconstruct a 3D structure using a technique called structure from a silhouette. One common drawback of 3D machine vision systems used in fruit tree reconstruction and automated pruning has been the data acquisition and processing time. For example, Medeiros et al. (2017) and Tabb and Medeiros (2017) reported a data acquisition and processing time of more than 10 min per tree using a high-performance computer, which is limiting for real-time applications.

To further advance the robotic pruning technologies, a team of researchers from WSU, Carnegie Mellon University, and Oregon State University is currently working on novel sensing and data processing techniques. A specialized 3D measurement system was developed, which includes two global stereo-camera systems and a small, eye-in-hand stereo-camera that moves with the robotic manipulator. Synchronized lighting was used to suppress external light in the environment and to create a uniform illumination for stereo-imaging, which substantially minimized the noisy, unwanted objects in the images (Fig. 3.17). The global camera system includes two camera orientations and a sliding mechanism to vary imaging position so that a dense 3D point cloud could be developed for a given section of the tree canopy. The local camera system was used to improve the accuracy of approaching the pruning location. The camera system has been used in both laboratory and field conditions to acquire 3D information of apple and cherry trees and creating 3D point clouds (Figure 3.17e).

Faster R-CNN proposed by Ren et al. (2015) was used to identity branching points from color images after the 3D point cloud generation. These branching points were strong visual cues that were used to detect branch occlusion that would be necessary to segregate individual branches as a whole. The link between the detected branching points was associated using a skeleton image generated by a generative adversarial network (GAN). The output of GAN was a binary image with

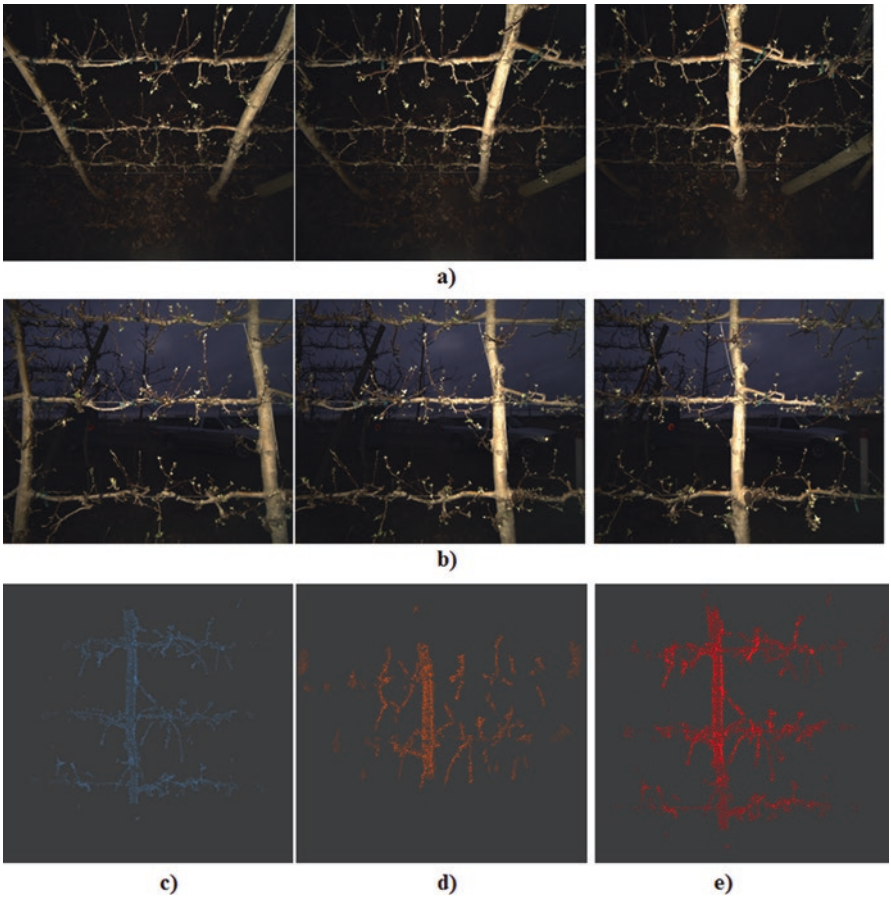


Fig. 3.17 Image processing sequence; (**a**) color images collected with a custom-built stereo vision camera from three different locations; (**b**) color images collected similarly with another stereo vision camera placed under the first one; (**c**) 3D point cloud created with the first stereo-camera, (**d**) 3D point cloud created with the second stereo-camera; and (**e**) combined 3D point cloud. (Image courtesy of Dr. Abhisesh Silwal, Carnegie Mellon University)

an array of connected binary pixels that traced the midsection of branches in the color images. A multichannel GAN was used to generate skeleton image for branches and the main trunk. Once the skeleton was identified, the curvature of the limbs and trunk was warped using the depth information obtained in the previous steps to reconstruct the 3D models. This information was used to estimate the length and size (diameter) of trunks and branches. The 3D skeleton and geometric parameters obtained were then used in identifying pruning branches in apple, cherry, and other fruit trees.

An integrated robotic system was also developed by the team (Fig. 3.18), which has been tested in the laboratory environment (You et al. 2020) to prune out cherry

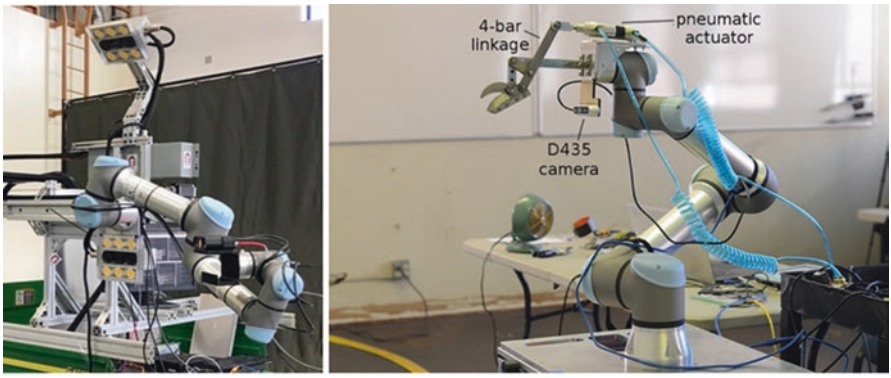


Fig. 3.18 Pruning robot setup; (**a**) a UR5e robot arm with specially designed two global stereo vision cameras and one eye-in-hand camera at the end of the arm; (**b**) an Intel RealSense D435 camera replacing the eye-in-hand stereo camera and a custom pneumatically actuated pruning end-effector

trees trained to an architecture called Upright Fruiting Offshoot (UFO). To simplify the architecture and to reduce the overall cost, a RealSense D435 was also investigated (Fig. 3.18b) as an alternative to the specialized local camera mentioned above. The study (You et al. 2020) showed that the integrated system with the RealSense D435 camera performed the path planning and a sequence of branch pruning operations successfully. These efforts are showing promising results for improved accuracy and computational efficiency for 3D vision system–based robotic pruning. Nevertheless, robotic pruning remains one of the most difficult tasks for automating tree fruit production, and further research and development are crucial to advance these technologies so that the accuracy, robustness, and cost of the technology could be achieved to the desired level for practical adoption.

3.5.4 Automated Red Raspberry Bundling

Red raspberry is a specialty crop grown commercially in some parts of the world, including the northwest region of Washington State. Similar to many other fruit crops, red-raspberries also require various canopy management operations, including cane pruning, bundling, and tying. To minimize labor use and associated cost, automated or robotic operations are in need of these canopy management activities (Bolda et al. 2012). Primarily because of low volume production and unique needs around the world, there has been limited research and development in automating canopy management activities in red raspberries. Identifying this research gap and unique needs of this industry in Washington and around the world, Bhusal et al. (2018) and Khanal et al. (2019) developed a novel, automated cane bundling and

tying solution using a stereo vision system for cane localization (Fig. 3.19). The vision system used a BumbleBee XB3 stereo vision camera (FLIR Systems, Wilsonville, OR) to acquire stereo-pair images of red raspberry canopies during the dormant season. The stereo vision system was mounted under the bundling and taping/tying end-effector and calibrated, as discussed in Sect. 3.3.2.

The system was tested in the field by driving the automated system in the center between crop rows. Stereo-images were collected by stopping the platform at the center of individual plants/canopies, and the images were then used to perform stereo vision computation, including image correspondence and disparity map generation (Sect. 3.3.4). One of the images (left image of the stereo pair) was also used to detect canes before 3D information was overlaid on to the detected canes. In addition, depth information was used to remove unwanted canes and other information from the background (anything outside the range of target distance to canes of interest), which helped improve the accuracy of estimating depth and height of the bunch of canes being bundled and tied together. When the color information (obtained as a segmentation mask from semantic segmentation) of detected canes was overlaid on distance-filtered depth information, binary blobs containing 3D information of target canes were created, which was used to find the average depth to the target canes for bundling. Bundling depth and height was provided to a microcontroller to actuate the mechanisms for bundling with the L-shaped end-effector and tying using a taping mechanism (Fig. 3.19). The automated cane bundling and taping mechanism achieved an overall success rate of 90% when tested in a red-raspberry plot located in a research farm at Washington State University. The manual tying and automated tying were compared in terms of crop yield. The mean fruit yield between machine taped and manually tied raspberry plants showed no significant difference. The study showed a potential for the development of an automated red raspberry

Fig. 3.19 Robotic cane bundling machine for red raspberries based on a stereo vision system (**a**) and raspberry canes detected and separated using 2D and 3D image-processing techniques (**b**). (From Bhusal et al. 2018)

cane bundling and taping mechanism. However, more work is necessary to increase speed, and address concerns related to using plastic tapes in tying canes.

3.6 Summary and Concluding Thoughts

Why 3D Sensing

The world is inherently a 3D space, and therefore, a robotic machine, to be robust, must operate using an accurately reconstructed 3D environment. Particularly for robots operating in uncertain and variable field conditions (agricultural and other fields), precise sensing, representation, and understanding of the 3D environment become crucial. For example, an apple-picking robot would need to have a sufficiently detailed and accurate description of the tree canopy environment, including the location and spatial relationship between various canopy objects such as fruit, branches, trunks, and trellis wires. Such 3D representation would be necessary to plan collision-free and efficient paths and then to implement the desired trajectory to approach, grab, pick, and place the fruit on to a conveyor or a container. Similar needs would be there for countless other robotic operations in the field environment, such as moving strawberry boxes out of fields to loading zones (Seyyedhasani et al. 2020) and flying aerial robots in between trees to apply specific chemicals to desired canopy parts. A large number of sensing systems are available and have been used in agricultural and field robotics applications. These sensors operate using one of three principles: (i) 3D from 2D images, (ii) time-of-flight of light, and (iii) structured light. The sensors range from single point depth estimation (e.g., a Laser sensor operating with ToF) to an imaging system for full environment 3D development (e.g., stereo vision system working with the principle of 3D from 2D images). Depending on specific applications at hand, single point depth estimation such as ultrasound and Laser techniques to imaging systems such as stereo vision and 3D camera systems have been used. As the 3D technologies are becoming increasingly cheaper, more reliable, and easier to operate, they have been implemented alongside the 2D vision system almost everywhere. However, the choice of the required type of technologies largely depends on the functional requirement of the work, space required, cost, and performance.

Similar to color and spectral sensing systems (Chaps. 2 and 4), 3D sensing techniques also suffer from various issues in agricultural and field environment, including variable lighting, uncertainty of field environment, inherent biological variability of canopies, occlusion of objects of interests, and variation in object shape, size, color, and other parameters. Laser sensors are generally less influenced by these challenges and are considered the best in terms of depth measurement accuracy in close range. But if the full environment (of interest) needs to be created at a higher resolution using a Laser scanning or LIDAR system, it could be slow and expensive (Zhang et al. 2019). Stereo vision system could provide a faster (in terms of image acquisition) and more cost-effective solution in such situations, but are more prone to the variability and uncertainty in lighting conditions and biological system

variabilities. In recent years, however, these systems are showing to be increasingly more robust and accurate for agricultural and field environments as well due to recent advancements in sensing hardware and data/image processing techniques, including deep learning. In addition, synchronized use of artificial lighting to suppress the interference created by sunlight and adding the capabilities for nighttime operation has been and will continue to be a powerful tool in improving the robustness of 3D vision system for field applications (Gongal et al. 2018; Nuske et al. 2011).

Integrated 2D/3D Sensing

In recent years, 3D information has also been used extensively in understanding the spatial variability of color, spectral, and other characteristics of objects (e.g., plant canopies). In many of the studies listed earlier in this chapter, two or more sensors were used for object detection and localization, which requires a precise calibration and registration to project location information estimated by a 3D sensor onto objects detected using a 2D image. To overcome this issue, in recent years, integrated color-3D cameras (also called RGB-D camera, e.g., RealSense and ZED cameras) and spectral-3D sensors like cameras being produced by MantaPoole Technologies (Reston, VA) are being developed and utilized. These cameras provide co-registered color/multispectral and 3D information and thus offer reduced postprocessing time and improved 2D-3D registration accuracy. For example, Xiao et al. (2017) used the Microsoft Kinect sensor to detect foliage using color information and to estimate leaf area density and average distance to canopy surfaces using 3D data. Newer, consumer-grade cameras with 2D, spectral, and 3D information are providing exceptional capabilities and a higher level of robustness at an affordable cost (a few hundred dollars) and thus are gaining broader applicability in agricultural and field robotics.

These newer sensors also provide an opportunity to conveniently implement depth filtering. The co-registered 3D information has been widely used to remove unwanted background in the field environment. For example, when a robot is working on a specific row of apple trees, part of canopies captured by the color images from the adjacent rows are nothing but the noises. With co-registered 3D information, such background noise can easily be removed from the images before further processing them using simple depth filters (removing image pixels that are closer and/or further away than the depth range a robotic system operates in). Postprocessing the depth filtered images has often shown improved accuracy in object detection and localization while also reducing the computational time as the noise (or noisy information) has been reduced in the images. For example, Zhang et al. (2018) used 3D information collected by a Kinect sensor with color images to train a faster R-CNN, which showed better accuracy in classifying branches, trunks, and background compared to the same without 3D information. As mentioned before, it is expected that the decreasing cost and increasing capability of consumer-grade, RGB-D, and spectral-3D sensors/cameras will lead to the widespread use of these cameras/sensors in the future. This advancement, therefore, is expected to help increase the viability of various type of robotic/automated systems for achieving desired speed,

accuracy, and robustness and to reduce the cost of the technology so that it becomes more viable for practical adoption.

Need for Higher Computational Performance
Agricultural robots work as an integrated system including vision system (including 3D sensing), robot control, field task management, guidance, inspection and quality control, and conveying and packaging system as depicted in example systems developed by FFRobotics (Hafia, Israel), Blue River Technology (California, USA) and Harvest CROO Robotics (Florida, USA). Multiple robotic subsystems need to be used in an integrated and collaborative manner for efficient field operation. For example, the robotic apple picking machine developed by FFRobotics utilized 12 robot manipulator (arm) subsystems with separate cameras for each manipulator. All the tasks performed need to be optimized and synchronized for the smooth operation of the integrated robotic system. As these mobile robotic platforms will be moving in the field from plant to plant and row to row, the desired image processing, including 3D processing techniques, needs to include sufficiently higher framerate (number of images processed per second) to provide the desired continuity and object tracking capabilities. In addition, for field operation, the sensing system used are placed in close proximity of the plant to view and understand minute details of the plants/canopies and/or due to the design constraint (such as row spacing, robotic arm length design, and so on). The proximity of the sensor to plants results in the need for acquiring a large number of images to cover the required operation area. Thus, the designer of the 3D agricultural vision system should pay special attention to the computational speed so that desired system performance can be achieved in real-time operation of the integrated robot system.

As seen from the different case studies discussed in the chapter, it is difficult to isolate the 2D and 3D image processing in agricultural and field robots. The choice of deep learning–based vision system has shown to be very promising in object detections and segmentation; however, some of these algorithms might hurt the operational speed of integrated agricultural robots. On top of that, some of the 3D image-processing algorithms such as SGM (discussed earlier in Sect. 3.3.4) requiring larger block size for matching, postintegration of 2D and 3D data from modern consumer-grade camera systems might require additional computational time. The computational performance of many vision algorithms, including 3D processing, can be increased by utilizing parallel programming software architecture, using multiple embedded computer systems such as Nvidia Jetson TX2. Other hardware architectures such as high-performance graphical processing units (GPU) can also provide promising platforms, which will, however, increase the cost of the integrated system. Some of the recent consumer-grade cameras such as ZED (Stereolabs Inc., San Francisco, CA) can be built with CUDA support, a parallel computing platform developed by Nvidia for faster 3D computation. Designers of the agricultural vision system (including 3D Vision) should be able to find the right balance between the required accuracy, speed, cost, power, and the physical space without affecting the performance of the integrated robot.

Additional Practical Considerations

In agricultural and field robotics where robots are in motion, there might be other artifacts such as motion blurriness, which can substantially reduce the accuracy of 3D mapping. In such a scenario, cameras/sensors that can operate at a higher frame rate and provide the flexibility of controlling exposure time, and color balance will be preferable. Also, trigger control for applications involving selective imaging (e.g., vision-based inspection system that can take images when fruit is present in the conveyor belt), and connection type and speed are also essential factors to consider. These choices directly affect the image data transmission rate on a multiimaging system connected to a single computer in an integrated robotic system.

As discussed before, the stereo vision system, especially consumer-grade hardware systems, recently has been more widely used in both researches as well as industrial development due to ease of use (precalibration), low cost, high resolution, low power consumption, smaller size, and flexibility in connection. These consumer-grade cameras (e.g., RealSense and ZED) also come with their own software development kit (SDK) and can be easily integrated with major programming languages such as C++ and Python (along with supporting wrappers for MATLAB and OpenCV libraries) for customizing the applications to meet the needs of specific research and development projects. Such SDKs will be the real asset in the future research and development of agricultural and field robots utilizing 3D vision systems.

References

Akbar S, Chattopadhyay S, Elfiky N, Kak A (2016a) A novel benchmark RGBD dataset for dormant apple trees and its application to automatic pruning. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 81–88

Akbar S, Elfiky N, Kak A (2016b) A novel framework for modeling dormant apple trees using single depth image for robotic pruning application. In: 2016 IEEE international conference on robotics and automation (ICRA), pp 5136–5142

Arad B, Balendonck J, Barth R, Ben-Shahar O, Edan Y, Hellström T, Hemming J, Kurtser P, Ringdahl O, Tielen T, van Tuijl B (2020) Development of a sweet pepper harvesting robot. J Field Robot 37:1027–1039

Bhusal S, Khanal K, Karkee M, Zhang Q (2018) Cane detection and localization for automated cane management in Red Raspberry Plant. Robotic Vision and Action in Agriculture, International Conference of Robotics and Automation

Bolda M, Tourte L, Klonsky K, Moura R (2012) Sample costs to produce fresh market raspberries: primocane bearing. University of California Cooperative Extension, Davis, CA. Retrieved from https://coststudyfiles.ucdavis.edu/uploads/cs_public/9e/28/9e286208-f8d8-4a6c-bbcc-d6d04d0f61d3/raspberrycc2012.pdf, Accesed on 15 May 2020

Botterill T, Paulin S, Green R, Williams S, Lin J, Saxton V, Mills S, Chen X, Corbett-Davies S (2017) A robot system for pruning grape vines. J Field Robot 34(6):1100–1122

Brown D (1965) Decentering distortion and the definitive calibration of metric cameras. Annual Convention of the American Society of Photogrammetry, Washington, DC, p 29

Chattopadhyay S, Akbar SA, Elfiky NM, Medeiros H, Kak A (2016) Measuring and modeling apple trees using time-of-flight data for automation of dormant pruning applications. In: 2016 IEEE Winter conference on applications of computer vision (WACV), 1–9

Chen L (2019) A weeding robot auto-levelling system for typical vegetable fields in pacific northwest region. PhD dissertation, Washington State University

Chen L, Karkee M, He L, Wei Y, Zhang Q (2018) Evaluation of a leveling system for a weeding robot under field condition. IFAC-PapersOnLine 51(17):368–373

Comba L, Biglia A, Aimonino DR, Gay P (2018) Unsupervised detection of vineyards by 3D point-cloud UAV photogrammetry for precision agriculture. Comput Electron Agric 155:84–95

Conrady A (1919) Lens-systems, decentered. Mon Not R Astron Soc 79:384–390

Dey D, Mummert L, Sukthankar R (2012) Classification of plant structures from uncalibrated image sequences. In: 2012 IEEE workshop on the applications of computer vision (WACV), pp 329–336

Elfiky N, Akbar S, Sun J, Park J, Kak A (2015) Automation of dormant pruning in specialty crop production: an adaptive framework for automatic reconstruction and modeling of apple trees. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 65–73

Gebbers R, Pflanz M, Betz A, Hille B, Mattner J, Rachow-Autrum T, Özyurtlu M, Schischmanow A, Scheele M, Schrenk J, Schrenk L (2013) OptiThin–implementation of precision horticulture by tree-specific mechanical thinning. In: *Massendatenmanagement in Der Agrar-Und Ernährungswirtschaft–Erhebung–Verarbeitung–Nutzung*. Ges. für Informatik, Bonn

Geng J (2011) Structured-light 3D surface imaging: a tutorial. Adv Opt Photon 3(2):128–160

Gongal A, Karkee M, Amatya S (2018) Apple fruit size estimation using a 3D machine vision system. Inf Process Agri 5(4):498–503

Guo Q, Wu F, Pang S, Zhao X, Chen L, Liu J, Xue B, Xu G, Li L, Jing H, Chu C (2018) Crop 3D—a LiDAR based platform for 3D high-throughput crop phenotyping. Sci China Life Sci 61(3):328–339

Hartley R, Zisserman A (2003) Multiple view geometry in computer vision. Cambridge university press, Cambridge

Hirschmuller H (2008) Stereo processing by semiglobal matching and mutual information. IEEE Trans Pattern Anal Mach Intell 30(2):328–341

Hohimer C, Wang H, Bhusal S, Miller J, Mo C, Karkee M (2019) Design and field evaluation of a robotic apple harvesting system with a 3D-printed soft-robotic end-effector. Trans ASABE 62(2):405–414

Izadi S, Kim D, Hilliges O, Molyneaux D, Newcombe R, Kohli P, Shotton J, Hodges S, Freeman D, Davison A, Fitzgibbon A (2011) KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, pp 559–568

Jeon H, Zhu H, Derksen R, Ozkan E, Krause C (2011) Evaluation of ultrasonic sensor for variable-rate spray applications. Comput Electron Agric 75(1):213–221

Jiménez A, Ceres R, Pons J (2000) A vision system based on a laser range-finder applied to robotic fruit harvesting. Mach Vis Appl 11(6):321–329

Karkee M (2005) Fusion of stereo-optical and interferrometric SAR DEMs. Master thesis, Asian Institute of Technology, 2005

Karkee M, Adhikari B (2015) A method for three-dimensional reconstruction of apple trees for automated pruning. Trans ASABE 58(3):565–574

Karkee M, Adhikari B, Amatya S, Zhang Q (2014) Identification of pruning branches in tall spindle apple trees for automated pruning. Comput Electron Agric 103:127–135

Khanal K (2018) Cane Management in Red Raspberry Crops: a proof of concept towards automation. MS Thesis, Washington State University

Khanal K, Bhusal S, Karkee M, Scharf P, Zhang Q (2019) Design of improved and semi-automated Red Raspberry Cane Bundling and taping machine based on field evaluation. Trans ASABE 62(3):821–829

Klein F, Wilmot A, De Tejada V, Rodriguez B, Requena I, Busch S, Rondepierre A, Auzeeri T, Sauerwald T, Andrews W, Rihan H (2019) Proof-of-concept modular robot platform for cauliflower harvesting. In: 12th European conference on precision agriculture, ECPA 2019

Langley RB (1998) Rtk gps. GPS World 9(9):70–76
Lin L (2004) Application of GPS RTK and Total Station system on dynamic monitoring land use. National Science Council, Taiwan, pp 92–2415
Lin G, Tang Y, Zou X, Li J, Xiong J (2019) In-field citrus detection and localisation based on RGB-D image analysis. Biosyst Eng 186:34–44
Liu H, Zhu H (2016) Evaluation of a laser scanning sensor in detection of complex-shaped targets for variable-rate sprayer development. Trans ASABE 59(5):1181–1192
Maghsoudi H, Minaei S, Ghobadian B, Masoudi H (2015) Ultrasonic sensing of pistachio canopy for low-volume precision spraying. Comput Electron Agric 112:149–160
Majeed Y, Zhang J, Zhang X, Fu L, Karkee M, Zhang Q, Whiting MD (2020) Deep learning based segmentation for automated training of apple trees on trellis wires. Comput Electron Agric 170:105277
Malavazi F, Guyonneau R, Fasquel J, Lagrange S, Mercier F (2018) LiDAR-only based navigation algorithm for an autonomous agricultural robot. Comput Electron Agric 154:71–79
Medeiros H, Kim D, Sun J, Seshadri H, Akbar S, Elfiky N, Park J (2017) Modeling dormant fruit trees for agricultural automation. J Field Robot 34(7):1203–1224
Meng X, Hu Z (2003) A new easy camera calibration technique based on circular points. Pattern Recogn 36(5):1155–1164
Nielsen M, Slaughter D, Gliever C (2011) Vision-based 3D peach tree reconstruction for automated blossom thinning. IEEE Trans Ind Inf 8(1):188–196
Nuske S, Achar S, Bates T, Narasimhan S, Singh S (2011) Yield estimation in vineyards by visual grape detection. In: 2011 IEEE/RSJ international conference on intelligent robots and systems, pp 2352–2358
Raja R, Nguyen T, Slaughter D, Fennimore S (2020) Real-time weed-crop classification and localisation technique for robotic weed control in lettuce. Biosyst Eng 192:257–274
Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. Adv Neural Inf Proces Syst:91–99
Ringdahl O, Kurtser P, Edan Y (2019) Evaluation of approach strategies for harvesting robots: case study of sweet pepper harvesting. J Intell Robot Syst 95(1):149–164
Rosell Polo J, Sanz Cortiella R, Llorens Calveras J, Arnó Satorra J, Ribes Dasi M, Masip Vilalta J, Camp F, Gràcia F, Solanelles Batlle F, Pallejà Cabrè T, Val L (2009) A tractor-mounted scanning LIDAR for the non-destructive measurement of vegetative volume and surface area of tree-row plantations: a comparison with conventional destructive measurements. Biosyst Eng 102(2):128–134
Salvi J, Pages J, Batlle J (2004) Pattern codification strategies in structured light systems. Pattern Recogn 37(4):827–849
Santos T, De Oliveira A (2012) Image-based 3D digitizing for plant architecture analysis and phenotyping. In: Conference on graphics, patterns and images. Retrieved from http://www.alice.cnptia.embrapa.br/alice/handle/doc/948383
Seyyedhasani H, Peng C, Jang W, Vougioukas SG (2020) Collaboration of human pickers and crop-transporting robots during harvesting–Part I: model and simulator development. Comput Electron Agric 172:105324
Shapiro L, Stockman G (2001) Computer vision. Prentice Hall, Englewood
Silwal A (2016) Machine vision system for robotic apple harvesting in fruiting wall orchards. Washington State University, Pullman
Silwal A, Davidson JR, Karkee M, Mo C, Zhang Q, Lewis K (2017) Design, integration, and field evaluation of a robotic apple harvester. J Field Robot 34(6):1140–1159
Sinoquet H, Rivet P, Godin C (1997) Assessment of the three-dimensional architecture of walnut trees using digitising. Silva Fennica 31:265–273
Soria P, Sukkar F, Martens W, Arrue B, Fitch R (2017) Multi-view probabilistic segmentation of pome fruit with a low-cost RGB-D camera. Iberian Robotics Conference, pp 320–331
Sukkar F, Best G, Yoo C, Fitch R (2019) Multi-robot region-of-interest reconstruction with Dec-MCTS. In: 2019 International conference on robotics and automation (ICRA), pp 9101–9107

Szeliski R (2010) Computer vision: algorithms and applications. Springer, London
Tabb A (2013) Shape from silhouette probability maps: reconstruction of thin objects in the presence of silhouette extraction and calibration error. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 161–168
Tabb A, Medeiros H (2017) A robotic vision system to measure tree traits. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 6005–6012. IEEE
Underwood J, Hung C, Whelan B, Sukkarieh S (2016) Mapping almond orchard canopy volume, flowers, fruit and yield using lidar and vision sensors. Comput Electron Agric 130:83–96
Valkenburg R, McIvor A (1998) Accurate 3D measurement using a structured light system. Image Vis Comput 16(2):99–110
van Diggelen F, Enge P (2015) The worlds first gps mooc and worldwide laboratory using smartphones. In: Proceedings of the 28th international technical meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2015), pp 361–369
Viola P, Wells WM III (1997) Alignment by maximization of mutual information. Int J Comput Vis 24(2):137–154
Vougioukas S, Arikapudi R, Munic J (2016) A study of fruit reachability in orchard trees by linear-only motion. IFAC-PapersOnLine 49(16):277–280
Wang Q, Nuske S, Bergerman M, Singh S (2013) Automated crop yield estimation for apple orchards. Exp Rob: 745–758. Springer
Williams H, Jones M, Nejati M, Seabright M, Bell J, Penhall N, Barnett J, Duke M, Scarfe A, Ahn H, Lim J (2019) Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. Biosyst Eng 181:140–156
Xiao K, Ma Y, Gao G (2017) An intelligent precision orchard pesticide spray technique based on the depth-of-field extraction algorithm. Comput Electron Agric 133:30–36
Xiong Y, Ge Y, Grimstad L, From PJ (2020) An autonomous strawberry-harvesting robot: design, development, integration, and field evaluation. J Field Robot 37(2):202–224
Yaxiong W, Shasha X, Wenbin L, Feng K, Yongjun Z (2017) Identification and location of grapevine sucker based on information fusion of 2D laser scanner and machine vision. Int J Agri Biol Eng 10(2):84–93
Ye Y, Wang Z, Jones D, He L, Taylor ME, Hollinger GA, Zhang Q (2017) Bin-dog: a robotic platform for bin management in orchards. Robotics 6(2):12
You A, Sukkar F, Fitch R, Karkee M, Davidson JR (2020) An efficient planning and control framework for pruning fruit trees. In: 2020 International Conference on Robotics and Automation (ICRA)
Yuan T, Zhang S, Sheng X, Wang D, Gong Y, Li W (2016) An autonomous pollination robot for hormone treatment of tomato flower in greenhouse. In: 2016 3rd international conference on systems and informatics (ICSAI), pp 108–113
Zeng L, Feng J, He L (2019) Semantic segmentation of sparse 3D point cloud based on geometrical features for Trellis structured Apple Orchard. 2019 ASABE Annual International Meeting
Zhang Z (2000) A flexible new technique for camera calibration. IEEE Trans Pattern Anal Mach Intell 22(11):1330–1334
Zhang J, He L, Karkee M, Zhang Q, Zhang X, Gao Z (2018) Branch detection for apple trees trained in fruiting wall architecture using depth features and regions-convolutional neural network (R-CNN). Comput Electron Agric 155:386–393
Zhang Q, Karkee M, Tabb A (2019) The use of agricultural robots in orchard management. ArXiv Preprint ArXiv 1907:13114

Chapter 4
Sensors III: Spectral Sensing and Data Analysis

Rajeev Sinha, Lav R. Khot, Zongmei Gao, and Abhilash K. Chandel

4.1 Introduction

Spectral sensing of plants refers to the domain where an optical sensing device measures the response, typically reflectance in arbitrary units, representative of the health status of the sampled plant segment (Carter and Knapp 2001; Sankaran et al. 2011; Thomas et al. 2018; Sinha et al. 2019). Spectral sensing as a system involves an optical sensing device that is either connected to onboard or standalone computing unit and has spectral data acquisition firmware/software as well as storage unit. A sophisticated system may include the pertinent data analytics algorithms running in the background for real-time diagnostic and result visualization.

Depending on the configuration, spectral sensing system can have a few (up to 15) to hundreds of optical bands characterizing the plant response in either spectral or in both spectral and spatial dimensions. Spectroscopy signals emitted by the objects can be sensed in visible (350–700 nm), near-infrared (NIR) (700–1400 nm), and in short-wave infrared (1400–2500 nm) regions (Fig. 4.1). Such response then can be analyzed using chemometric methods to relate it to plant health status. The goal of this chapter is to introduce some of the basic optical sensing principles, typical optical system (device) configurations, and associated spectral data analytics. Through pertinent case studies, this chapter also summarizes the application domain of the optical spectral sensing method toward better understanding of the plant response to biotic and abiotic stressors.

Disclaimer: Commercial product mentioned in the chapter is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated.

R. Sinha · L. R. Khot (✉) · Z. Gao · A. K. Chandel
Washington State University, Prosser, WA, USA
e-mail: lav.khot@wsu.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_4

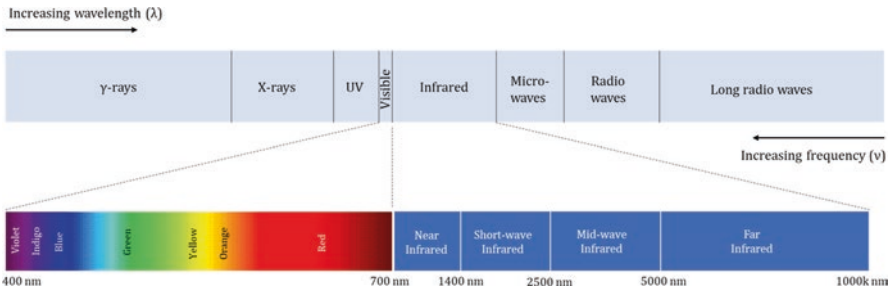


Fig. 4.1 Schematics showing the electromagnetic spectrum and wavelengths pertinent to visible and infrared regions of the spectrum (not drawn to scale)

4.2 Spectroradiometers

4.2.1 Working Principle

Spectroradiometers are calibrated spectral sensing devices having an inbuilt optical measurement unit that evaluates the interaction between the biological matter and the electromagnetic spectrum (EMS). The basic components of a spectroradiometer are (a) a light source producing the spectrum, (b) an optical unit to collimate and disperse the spectrum, and (c) a detector measuring the dispersed spectrum (Fig. 4.2).

An optical unit consisting of specific lenses is used to converge the incoming light or photon flux from the targeted field of view to a narrow slit. This is termed as fore-optic configuration (Fig. 4.3a). In this configuration, the photon flux from a target surface is directly received by the spectroradiometer. The photon flux can also be guided from the target surface toward the spectroradiometer using a fiber optics cable or a "light pipe" (Fig. 4.3b). A fiber optics cable may be analogous to a garden hose guiding water from one location to the other. It works on the principle of total internal reflection and guides light through twists and turns to the spectroradiometer or any other optical device. It has two parts, namely, a core and a cladding. Both the core (with a higher refractive index) and the cladding (with a lower refractive index) are made of glass. The refractive indices and the angle of light incidence are selected in a way to achieve total internal reflection of the incident light inside the fiber optics cable.

The slit width (μm) regulates the amount of light and thus governs the resolution and throughput of a spectroradiometer. Reduction in the slit width below a certain limit will not change the unit resolution. Hence, once the resolution criterion is met for a spectroradiometer, also a factor of pixel, picture element, and width of a detector, the slit width may be increased only for augmenting the throughput characteristics.

As the light passes through the slit, it diverges owing to the natural tendency. This divergent beam is then collimated or parallelized by a concave mirror, sized to

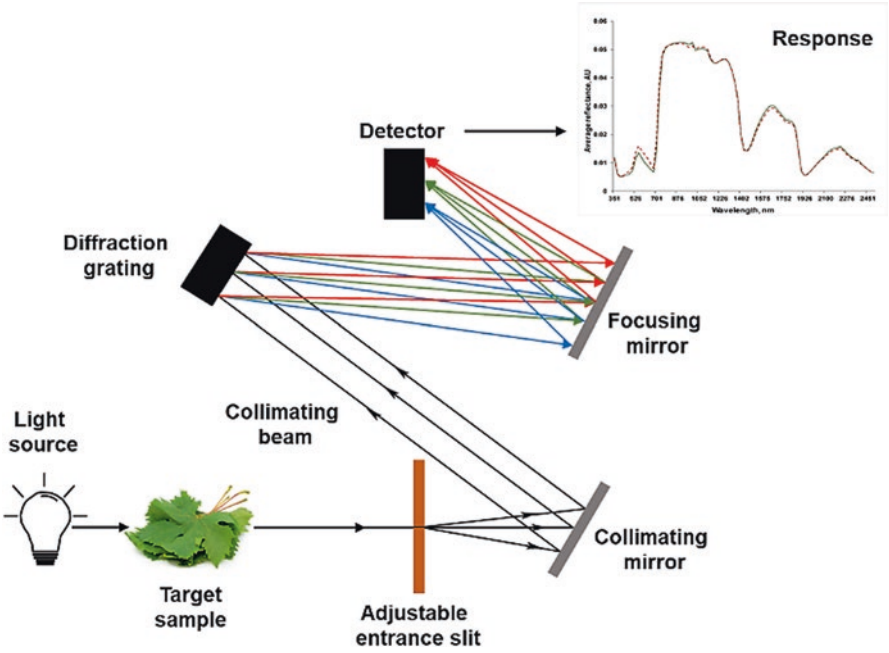


Fig. 4.2 Typical components assembly depicting working principle of a spectroradiometer

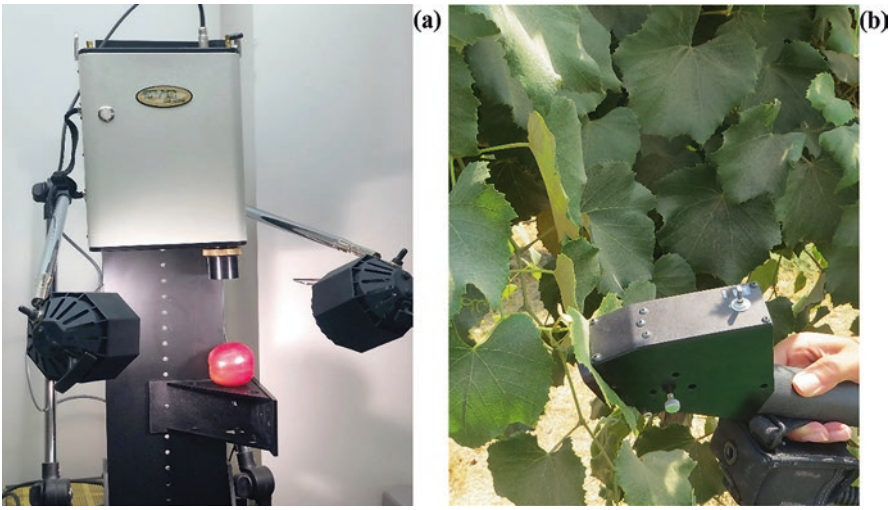


Fig. 4.3 Illustrations of a commercial spectroradiometer (SVC HR-1024i, Spectra Vista Corporation, Poughkeepsie, NY, USA) in (**a**) fore and (**b**) fiber optic configuration for spectral data acquisition

encompass incoming light, before being directed toward a diffraction grating (Fig. 4.2). The latter is an optical instrument to disperse the collimated light into multiple divergent beams which are nothing but the component wavelengths or colors or energies. The wavelength range, and to some extent the spectral resolution of a spectroradiometer, is determined by a diffraction grating. Ruled and holographic grating are the two basic types of grating. Ruled gratings are formed by physically creating parallel and equally spaced grooves over a reflective surface. Holographic gratings are generated over a photosensitive material by exposing it to laser beams (e.g., ultraviolet and argon) which are intersecting, coherent, and monochromatic in nature to create sinusoidal interference fringes over the photosensitive material. Using laser beams for creating fringes or grooves in holographic grating has certain advantages over the ruled grating. Intersecting beams in the former generates a highly uniform surface compared to the latter, providing a highly uniform spectral response. A ruled grating does account for a higher amount of stray light due to imperfect surface and causes ghosting effect. Overall, the holographic gratings are preferred over the ruled gratings where signal to noise ratio (SNR) is critical.

Grating dispersed beam then gets reflected by another optical setup consisting of a concave mirror. This mirror also separates the wavelengths that then reach to the detector(s). A detector is an electronic device highly sensitive to photons and capable of generating electric charge upon the incident of the separated wavelengths from the mirror (Fig. 4.2). Semiconductor materials like silicon (Si) or indium gallium arsenic (InGaAs) alloy are the most popular detector elements. The InGaAs alloy detectors have higher bandgap energy and can have higher upper detection limit compared to Si-based detectors. The detector-generated charge then gets digitized, filtered, and read/logged as a time series spectrum using associated instrumentation.

In earlier spectroradiometers, a single detector element was typical with rotated grating to measure the light intensity as a function of the incident wavelength. Moving parts add complexity to these designs and device power requirements. The modern spectroradiometers thus use charged couple device (CCD) arrays and linear detectors, making them more compact and power efficient.

The detection limit of detectors can be improved by cooling the array through inbuilt cooler (Wiecek 2005). Si-based detectors can be configured as CCDs, black-thinned (BT) CCDs, and photodiode arrays (PDAs). Overall, BT-CCDs are highly sensitive and can overcome the light scattering issue typical in CCDs. Thinning of the substrate in BT CCDs causes reduced absorption and increased efficiency of the detector. PDAs use linearly arranged complementary metal oxide semiconductor (CMOS). Though CCDs are excellent where low-noise data is required, they tend to consume more power. As CMOS technology uses inbuilt amplifiers and readout transistors at each detector, it can achieve rapid digitization with significantly less power compared to CCDs.

4.2.2 Spectroradiometer Types

Spectroradiometers can either be multispectral or hyperspectral depending upon the number of specific spectral bands within which the reflected energy is measured. A multispectral radiometer measures reflected energy within a few specific bands (<15 bands) with somewhat wider spectral bandwidth (per band). Hyperspectral radiometer measures reflected energy for hundreds of bands with narrower spectral bandwidth. Thus, the latter can sense subtle variations in the reflected energy and be more specific in detecting crop stressor specific signatures.

Multispectral spectroradiometers can be handy and practical for field-level sensing due to small size, portability, and comparatively lower unit cost. For example, GreenSeeker® (NTech Industries Inc., Ukiah, CA, USA) is an active canopy multispectral sensing device that captures plant-reflected light in red (660 ± 12 nm) and near-infrared (NIR) (770 ± 12 nm) bands. These two spectral bands can be used to estimate the normalized difference vegetation index (NDVI) (Kim et al. 2012). The reflected energy from the crops can be used to calculate different crop health–related indices like normalized difference red edge (NDRE) and red edge chlorophyll index (CI_{RE}), among others. There are several of such active canopy sensors (ACS) which measure the reflected energy in couple or few spectral bands. A CC–470 (Holland Scientific Inc., Lincoln, NE, USA) is one such ACS that can be configured by a user with a choice of six spectral bands (blue [450 nm]; green [550 nm]; red [650 nm, 670 nm]; red edge [730 nm]; and NIR [>760 nm]). However, only three of the six bands could be used at a time. The company has another ACS, CC–430 model, which measures the reflected energy in three specific bands (red [670 nm]; red edge [730 nm]; and NIR [>760 nm]) (Cao et al. 2018).

Commercial spectroradiometers such as "Weed-Seeker®" and "DetectSpray®" have monochromatic diodes for red and NIR bands and can detect green vegetation on a bare soil. Green plants reflect a minimum of red light; however, they reflect most in NIR range. A threshold can be set for plant identification based on the reflected energy from the target in these two bands. Besides commercially available units, there have been a few attempts to develop multispectral radiometers as a specific plant health diagnosis tool (Feyaerts et al. 1998; Larbi et al. 2013).

Similar to multispectral units, several hyperspectral spectroradiometers are available commercially with different spectral sensing ranges. For example, the HR–512i (Spectra Vista Corporation, Poughkeepsie, NY, USA; Fig. 4.3) is a field portable unit that uses a linear array of Si detectors and has a spectral range of 350–1050 nm. The spectral resolution of this unit is less than 3.2 nm with a nominal bandwidth less than or equal to 1.5 nm. This unit comes with a 4° field of view (FOV) with an optional fiber optics with a 25° FOV with automatic/selectable feature for spectrum averaging. The measurements can be acquired and viewed using a 32-bit internal central processing unit (CPU), and the device can communicate with up to 16 external sensors using a secondary Bluetooth™ radios. Fig. 4.4 shows typical spectral responses pertinent to green vegetation under healthy and stressed conditions (Sinha et al. 2017a).

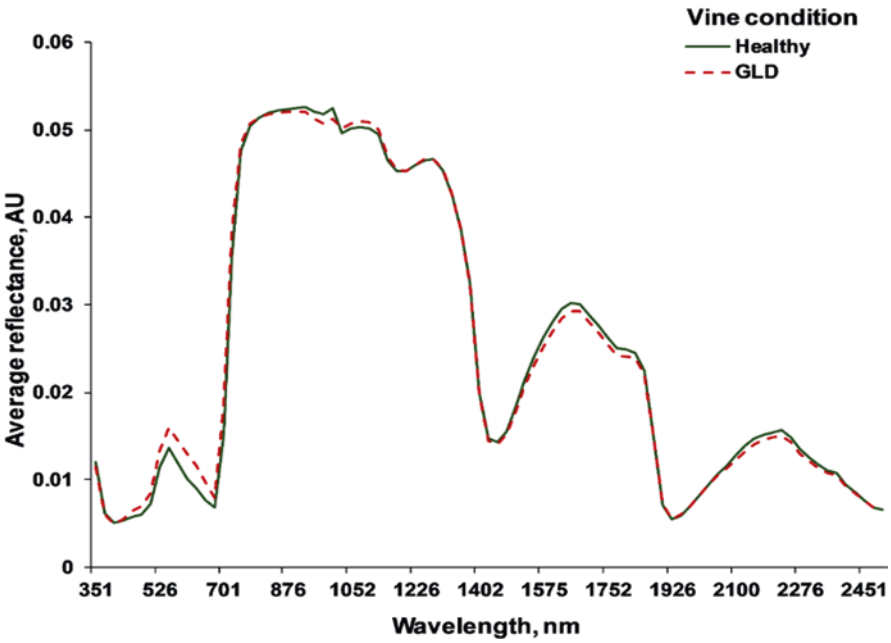


Fig. 4.4 Representative spectral responses from healthy and grapevine leafroll disease (GLD) infected vine leaves

4.2.3 Spectral Data Analysis

4.2.3.1 Preprocessing

Preprocessing steps involve data normalization, smoothing, scatter correction (Rinnan et al. 2009), derivatives, and continuum removals (Dotto et al. 2018). Preprocessing can help improve classification and multivariate as well as exploratory analysis. Preprocessing can be divided into (1) scatter correction methods and (2) spectral derivatives. While the scatter correction methods are mostly used to reduce the scatter effects between samples, the spectral derivatives have the capability to remove baseline effects and any offset differences between the data. The most common way of selecting a preprocessing method is to try different methods and select the optimal one. This strategy is often influenced by a prior expertise with the equipment, sample, and different preprocessing methods (Gerretzen et al. 2015).

Multiplicative scatter correction (MSC), inverse MSC, continuum removal, normalization by range, and standard normal variates (SNV) are typical examples of scatter corrections (Rinnan et al. 2009). Typical examples of spectral derivative–based preprocessing include Savitzky-Golay (SG) polynomial derivative and Norris-Williams (NW) derivatives (Peng et al. 2014). Spectral derivatives have the capability to remove additive as well as multiplicative effects from the reference spectra. This is because it does smoothen the spectra and then estimate derivative(s).

MSC is one of the most widely used methods and has been successfully applied to preprocess spectral data from meat (Geladi et al. 1985), dairy (Chen et al. 2002), grains (Wang et al. 2004), and soil samples (Kooistra et al. 2001). Overall, the MSC is performed by correcting the recorded spectrum by a reference spectrum. The average spectrum of the calibration set is often used as the reference spectrum. Windig et al. (2008) proposed a loopy MSC which iteratively searches for the reference spectrum within the dataset being corrected by the MSC. The SNV transformation centers each spectrum in the dataset, followed by scaling by its own standard deviation, and is linearly related to the MSC corrections (Dhanoa et al. 1994). Normalization is another method for scatter correction with "Euclidian" as one of the widely used methods (Sankaran and Ehsani 2011).

Smoothing of spectral data is the first step in both the SG- and NW-based preprocessing methods. It helps in SNR reduction. This is followed by the estimation of first (removing the baseline) and second derivatives (removing both baseline and linear trends). The SG method uses moving average to preserve the intrinsic properties of the spectral data, such as relative maxima, minima, and width. In this method, a small window of spectral wavelengths is fitted by a polynomial equation. Once the parameters of the fitted model are known, they are used to calculate the derivative. Thus, selection of window width and degree of the polynomial fit are critical steps in the SG-based spectral data preprocessing.

4.2.3.2 Spectral Feature Extraction

Data dimensionality is a critical factor that needs to be carefully considered during hyperspectral spectroradiometers use in crop sensing. An efficient high-dimensional dataset handling approach has been to search for a subset of wavebands that contain salient spectral information (Hira and Gillies 2015) about the plant stressor(s) symptoms. This process is often termed as feature selection (Balabin and Smirnov 2011) after removal of nonpertinent, noisy, and redundant features.

Stepwise multiple linear regression (SMLR) is one such feature selection method. It is a sequential approach of feature selection with forward selection and backward elimination (Balabin and Smirnov 2011). In the forward selection, the features are selected iteratively based on their correlation with the response variable. This selection is followed by the determination of model regression coefficient and significance testing (mostly using a t–test) at a critical value. The selected feature is retained if the regression coefficient is significant, followed by the selection of another variable. The added variable is selected based on its correlation with the model residuals obtained with the first variable. The two models are then tested for significance based on their regression coefficients, and the nonsignificant features are removed in a process called backward elimination. These two steps are repeated until there is no significant improvement in the model that can be achieved by adding or removing more features. While keeping too many variables in the model may lead to overfitting, different types of cross-validation methods (e.g., leave-one-out cross-validation) can be used to reduce this risk.

Besides SMLR, other multivariate feature selection approaches are principal component analysis (PCA), principal component regression (PCR), and partial least square regression (PLSR) (Balabin and Smirnov 2011). These methods overcome issues like collinearity, band overlaps, and interactions which are frequently found in spectroscopic data. PCA is a dimensionality reduction technique which transforms a high-dimensional dataset into a smaller set of uncorrelated variables known as principal components (PC) that explain most of the variance in the feature dataset (Sinha et al. 2017b; Sinha et al. 2018). In PCR, major PC scores of the features are retained and the remaining scores are discarded. The lower ranked PC scores often account for linear dependencies among the features. The removal of such PCs adequately addresses the issue of multicollinearity in the dataset (Lee et al. 2015). While SMLR aims at finding a single factor correlating the feature dataset as well as the response variable, PCR searches for the factors capturing maximum variance in the dataset prior to regression analysis. PLSR captures the variance and correlates the data by maximizing the covariance in the feature dataset. Thus, PLSR is deemed superior to SMLR and PCR in terms of predictive power (Xiaobo et al. 2010).

Algorithms like interval partial least square (iPLS), backward iPLS, and forward iPLS regressions can also be used for feature selection. In iPLS, the feature dataset is divided into sections which do not overlap. Each of the sections is separately modeled using PLSR to find the suitable variable range (Nørgaard et al. 2000). Backward iPLS is similar to iPLS in the interval selection with added determination of a sequence model leaving one interval at a time. For example, if '*n*' intervals are selected, each PLS model is calculated with (*n*-1) intervals excluding one interval at a time. Forward iPLS is an inverse form of backward iPLS. It builds PLSR model with an interval that has the best performance in terms of root mean square error of cross-validation (RMSECV). In addition to the interval selection algorithms where the selected intervals are fixed, moving window methods can also be used for feature selection. In such algorithms (e.g., moving window PLSR), a window of spectral features is built and moved through the entire spectrum. At each of the window locations, the RMSECV is calculated for a PLSR or SMLR model, and information about a critical spectral region is obtained. In some algorithms, the size of the spectral window is not fixed, and hence, they can be called as changeable size moving window PLSRs. A combination of moving window and changeable size moving window, known as searching combination moving window, can also be implemented within PLSR.

Successive projection algorithm (SPA) is another feature selection method that reduces the collinearity issues associated with SMLR (Araújo et al. 2001). SPA projects feature dataset to a vector space and allows selection of a feature set with low collinearity. Feature with the highest projection value in the orthogonal space of the prior selected features is selected in a forward selection process (Balabin and Smirnov 2011). The uninformative variable elimination (UVE) is another approach of variable selection (Centner and Massart 1996). In this technique, some random variables are added to the feature dataset. The new added features that contribute less in the model development, compared to primary selected random variables, are eliminated from the model. A combination of SPA and UVE (UVE-SPA method) is

also feasible with SPA selecting a set of features and UVE removing less informative features. Such approach is reported to have a greater precision of prediction.

In addition to the above-discussed algorithms, there exist a number of advanced algorithms for spectral feature selection. They include simulated annealing (SA), artificial neural networks (ANN), and genetic algorithm (GA). The SA is a search-based probabilistic optimization technique which approximates a global optimal solution in a large space. It starts with an initial solution which is continuously modified based on a control parameter. The algorithm searches for a global optimal solution by iterating through a number of local optimal solutions (Swierenga et al. 1998). ANNs are machine learning models inspired by neural networks in human brain. Back propagation (BP) is one of the most commonly used ANNs. BP ANN consists of different layers of neurons, namely, at input, hidden, and output layers. For feature selection, the causal index of the trained ANN model is analyzed to estimate the influence of each input on each ANN output (Balabin and Smirnov 2011). Among multiple approaches, a combination of PCA and conjugate gradient algorithm (GA) can be employed to determine the number of neurons in the hidden layer that represents the data dimensionality. The contribution of inputs to the total model variance governs retention of the significant features. GA is a heuristic optimization method for features selection based on the theory of probability. This technique searches for all possible subsets of features that would have maximum model prediction accuracy. GA combined with PLS (GA-PLS) has been reported as a better feature selection approach as the combination offers a number of feature sets which can provide an optimum or near-optimum solution (Mehmood et al. 2012).

4.2.3.3 Spectral Data Classification/Prediction Models

Several types of models can be used to classify different categories or classes in the feature dataset and subsequently predict the class which relates to plant stressor(s). Classification is defined as the process of identification of the category of a new sample based on a training set of samples with known categories (Tang et al. 2018). Some of the most common classification/prediction models for Vis-NIR hyperspectral dataset are logistic regression (LR), linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), and Naïve Bayes (NB). LR determines the probability of a sample to be associated with a specific class or category. However, for well distinct classes and small datasets, the estimation of regression parameters is unstable. The LDA and QDA methods, respectively, model the response variable with a linear and nonlinear combination of predictor variables and can be used for multiclass classification (Sinha et al. 2017c; Sinha et al. 2018). Data dimensionality, sample size and spectral feature set, is critical for any of the classifiers, including that of LDA and QDA. The NB classifier assumes independence among the features prior to assigning a probability value to a specific class (Srivastava et al. 2007). Soft independent modeling of class analogies (SIMCA) is also a well-known classification model used for classifying Vis-NIR spectral

datasets. SIMCA is an extension to PCA. However, it is reported as a weak classifier compared to LDA and QDA techniques (Rácz et al. 2018).

Some of the advanced classification algorithms used with spectral datasets are k-nearest neighbor (kNN), artificial neural network (ANN), decision trees (DT), random forest (RF), support vector machines (SVM), and ensemble learning (EL). The kNN can be used for pattern recognition where the class of a sample is ascertained by the class of its nearest '*k*' neighbors (Guo et al. 2018). ANN are supervised machine learning technique. The traditional ANNs use RB function, perceptron, and back propagation algorithms, among others. DTs are extremely fast classification models devised in a tree-like shape. Each node in the tree performs a pairwise comparison on a selected feature with the branches coming out of the nodes that symbolize the outcome. A leaf node signifies the final classification or decision-making. RFs are a collection of several DTs and can overcome the issue of overfitting associated with DTs. In RF, outcomes of various DTs are merged for final prediction or classification. Recently, SVMs have been used extensively for classification/prediction of spectral datasets. SVMs create a linear hyperplane to classify the observations. In addition, significant improvements in the performance of SVM classifier can be obtained using kernels which map the feature vectors into a high dimension (Mishra et al. 2012).

4.2.4 Application Case Studies

Table 4.1 reports some of the applications of spectral sensing in plant stressors detection. Pertinent feature selection and classification/prediction models implemented are also summarized therein. Subsections below also detail a few case studies where visible near-infrared spectroscopy has been employed in plant stress detection.

4.2.4.1 Case Study 1. Biotic and Abiotic Stress Detection in Grapevines

Spectral reflectance technique has been successfully employed for detecting biotic and abiotic stresses in grapevines. Naidu et al. (2009) assessed the potential of Vis-NIR spectroscopy for the detection of grapevine leafroll-associated virus-3 (GLRAV–3)-related grapevine leafroll disease (GLD) symptoms in red-berried cultivars (*cv.* Cabernet Sauvignon and Merlot). Leaf spectra were collected from detached healthy and infected vines using a portable spectroradiometer following a destructive sampling approach. Significant differences ($p < 0.05$) between the healthy and GLD-infected leaf samples were observed in the visible (550 nm), infrared (900 nm), and mid-infrared regions (1600 and 2200 nm).

In a continuing study, Sinha et al. (2019) employed a nondestructive leaf sampling strategy for GLD detection. For two crop seasons (2016 and 2017), in-field spectral data was collected from healthy, symptomatic, and asymptomatic leaf

Table 4.1 Commonly used classification/prediction models with Vis-NIR/NIR spectral datasets in agricultural applications

Model	Application	Spectral range, nm	Spectral features	References
LR	Citrus greening detection	350–2500	PCs, >95% variance	Mishra et al. (2012)
	Maize yield prediction	350–2500	Full spectrum	Ferreiro-González et al. (2017)
	Blood spot detection in eggs	200–1100	576.94, 600.31 nm	Chen et al. (2015)
LDA	Tea-type classification	190–800	PCs, first 7	Dankowska and Kowalewski (2019)
	Wine grape quality assessment	350–2500	Full spectrum	Costa et al. (2019)
	Adulteration in processed meat	400–1000, 900–1700	PCs, >99% variance	Rady and Adedeji (2018)
QDA	Citrus greening detection	350–2500	PCs, >99.9% variance	Sankaran et al. (2011)
	Grapevine leafroll disease detection	350–2500	626, 701, 726, 901, 976, 1001, 1027, 1052, 1101 nm	Sinha et al. (2019)
	Bitter pit detection in apples	800–2500	971.2, 978, 986.1, 987.3, 995.4, 1131.5, 1135.3, 1139.1, 1142.8 nm	Kafle et al. (2016)
NB	Grapevine leafroll disease detection	350–2500	626, 701, 726, 901, 976, 1001, 1027, 1052, 1101 nm	Sinha et al. (2019)
	Unfertilized duck eggs detection	330–1030	500–940 nm	Dong et al. (2019)
	Adulteration in processed meat	400–1000, 900–1700	PCs, >99% variance	Rady and Adedeji (2018)
SIMCA	Citrus greening detection	350–2500	PCs, >99.9% variance	Sankaran et al. (2011)
	Nutrient estimation in oil palm leaf	350–2500	354, 732, 2129, 2292 nm for N 352, 356, 568 nm for P, 356 nm for K	Khorramnia et al. (2014)
	Weed species discrimination	920–2500	1078, 1435, 1490, 1615 nm	Shirzadifar et al. (2018)
kNN	Citrus greening detection	350–2500	PCs, >99.9% variance	Sankaran et al. (2011)
	Adulteration in processed meat	400–1000, 900–1700	PCs, >99% variance	Rady and Adedeji (2018)
	Blood spot detection in eggs	200–1100	576.94, 600.31 nm	Chen et al. (2015)

(continued)

Table 4.1 (continued)

Model	Application	Spectral range, nm	Spectral features	References
ANN	Soil lead and zinc content estimation	350–2500	460, 900, 1400, 1900, 2200 nm for PB 600, 900, 1100, 1400, 1900, 2200 nm for Zn	Khosravi et al. (2018)
	Nutrient estimation in oil palm leaf	350–2500	354, 732, 2129, 2292 nm for N, 352, 356, 568 nm for P, 356 nm for K	Khorramnia et al. (2014)
DT	Discrimination of apple varieties	325–1075	500–700, 720–960 nm	He et al. (2007)
	Classification of orange growing locations	1000–2500	Full spectrum	Dan et al. (2015)
	pH-based classification of meat quality grades	400–2500	400, 448, 482, 540, 568, 600, 602, 604, 606, 608, 622, 626, 654, 666, 684, 698, 968, 1376, 1710, 1874, 2476, 2494 nm	Jr et al. (2018)
SVM	Bitter pit detection in apples	800–2500	971.2, 978, 986.1, 987.3, 995.4, 1131.5, 1135.3, 1139.1, 1142.8 nm	Kafle et al. (2016)
	Nutrient estimation in oil palm leaf	350–2500	354, 732, 2129, 2292 nm for N, 352, 356, 568 nm for P, 356 nm for K	Khorramnia et al. (2014)
	Adulteration in processed meat	400–1000, 900–1700	PCs, >99% variance	Rady and Adedeji (2018)
RF	Soil nitrogen and carbon assessment	305–2200	Full spectrum	Nawar and Mouazen (2017)
	Soil organic carbon estimation	350–2500	Full spectrum	Vašát et al. (2017)
	Soil quality parameters	400–2500	Full spectrum	de Santa et al. (2018)
PLSR	Wine grape quality assessment	350–2500	Full spectrum	Costa et al. (2019)
	Bitter pit development prediction in apples	935–2500	PLS components, >95% variance	Jarolmasjed et al. (2017)
	Adulteration in processed meat	400–1000, 900–1700	PCs, >99% variance	Rady and Adedeji (2018)
PCR	Egg yolk cholesterol quantification	190–2500	Full spectrum	Puertas and Vázquez (2019)
	Winter wheat biomass estimation	350–2500	5 vegetation indices	Yue et al. (2018)
	Soil properties assessment	1300–2500	20 PCs, smallest RMSECV	Chang et al. (2001)

LR Logistic regression, *LDA* Linear discriminant analysis, *QDA* Quadratic discriminant analysis, *NB* Naïve Bayes, *SIMCA* Soft independent modeling by class analogy, *kNN* k–Nearest neighbors, *ANN* Artificial neural networks, *DT* Decision tree, *SVM* Support vector machines, *RF* Random forest, *PLSR* Partial least square regression, *PCR* Principal component regression

samples using a field portable spectroradiometer. PLSR- and SMLR-based feature selection followed by QDA and NB classifiers were employed in the study. The reflectance between the healthy and GLD-infected leaves (both symptomatic and asymptomatic) were found to be significantly different in visible (351, 377, 501, 526, 626 and 676 nm) and NIR (701, 726, 826, 901, 951, 976, 1001, 1027, 1052 and 1101 nm) regions. Some of the feature wavelengths (1001, 1027 and 1052 nm) were repetitive across crop seasons as well as temporally within a specific crop season. The PLS score plots for healthy-symptomatic and healthy-asymptomatic spectral datasets exhibited inherent clustering for the samples (Fig. 4.5). A study reported reliable asymptomatic samples classification (Fig. 4.5a) similar to the symptomatic samples (Fig. 4.5b). The QDA had superior performance with overall classification accuracies in the ranges of 75–99% for both healthy-symptomatic and healthy-asymptomatic datasets.

Al-Saddik and Cointault (2019) evaluated Vis-NIR spectroscopy for obtaining optimal spectral wavelengths for the detection of "*Flavescence dorée*" in French vineyards. The reflectance of healthy leaves was observed to be higher in the visible (500–700 nm) and lower in the NIR (800–1300 nm) and infrared regions (>1300 nm) compared to the infected leaves. The preprocessing of spectral data yielded a classification accuracy >97% using SVM and LDA classifiers.

Beghi et al. (2017) utilized Vis-NIR spectroscopy for water stress monitoring in grapevines (*cv.* Cabernet Sauvignon). The study reported high R^2 values for the prediction of total water content (TWC) and leaf water potential (ψ) (0.67 and 0.72, respectively) using spectral response in the ranges of 400–1000 nm. However, better prediction of TWC ($R^2 = 0.91$) was observed with spectra from 1000–2000 nm. In a similar study, de Bei et al. (2010) measured leaf water potential in three different grapevine cultivars (*cv.* Cabernet Sauvignon, Chardonnay and Shiraz) using a spectroradiometer (300–1100 nm). While acceptable prediction of ψ was observed for all the cultivars, the best prediction was reported for Shiraz cultivar ($R^2 = 0.92$, standard error in cross-validation = 0.09 MPa).

4.2.4.2 Case Study 2. Citrus Disease Detection

Vis-NIR spectroscopy has been successfully employed to detect various diseases in citrus production. Citrus greening (or Huanglongbing [HLB]), caused by a phloem-limiting bacteria, *Candidatus Liberibacter asiaticus* (Sankaran and Ehsani 2011), has been one of the devastating citrus diseases. In a study, Sankaran et al. (2011)) utilized Vis-NIR spectroscopy to classify healthy and HLB-infected citrus leaves under field conditions. Suitable feature extraction techniques (SDA and SMLR) were employed for feature wavelength selection, followed by QDA- and SIMCA-based classification. The study reported that spectral bands from visible, NIR, and infrared regions (537, 612, 688, 713, 763, 998, 1066, 1120, 1148, 1296, 1472, 1597, 2121, 2172, 2348, 2493 nm) had the potential to discriminate between healthy and HLB-infected leaves. Overall classification accuracies were in the ranges of 74% to

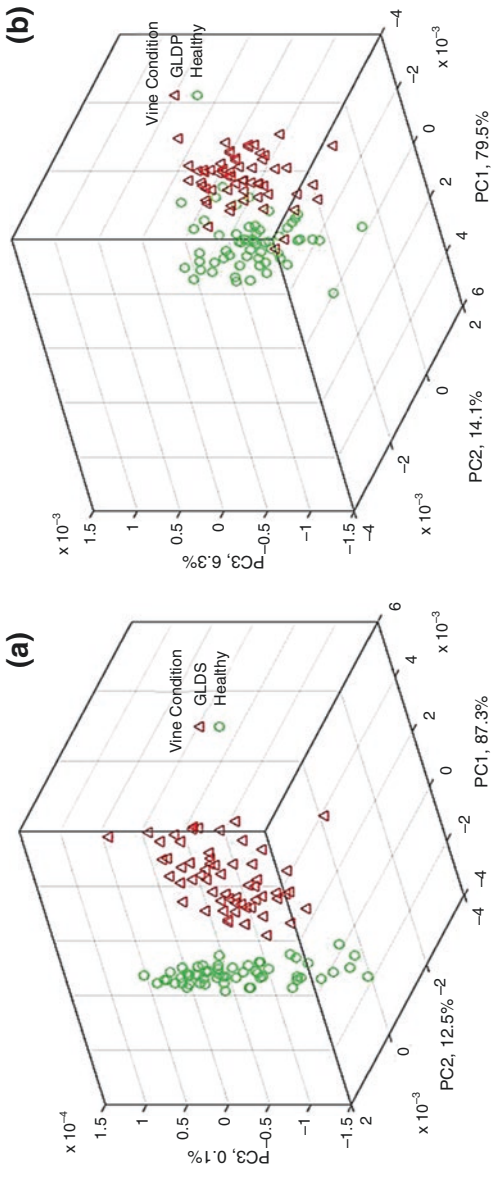


Fig. 4.5 Leaf spectral response–based principal components score plots for healthy and GLD-infected leaves with (**a**) symptomatic (GLDS) and (**b**) asymptomatic (GLDP) samples

91% with QDA performing better compared to SIMCA. In a similar study, Mishra et al. (2012) evaluated Vis-NIR spectroscopy with three different classification techniques (kNN, LR and SVM). Similar to prior studies, the reflectance of HLB-infected leaves was found higher in the visible region compared to healthy leaves. This was possibly due to chlorosis in the infected leaves causing less absorption of visible light. Lorente et al. (2015) also used Vis-NIR spectral measurements to automatically detect citrus fruit decay due to fungus *Penicillium digitatum* in mandarins.

4.2.4.3 Case Study 3. Apple Bitter Pit Disorder Detection

Apple bitter pit is a physiological disorder that causes significant economic losses to growers. Bitter pit development occurs before harvest and intensifies during postharvest storage. Optical sensing techniques have shown the potential in detecting this disorder. For example, Kafle et al. (2016) studied the robustness of spectral features for the nondestructive detection of bitter pit in fresh market apples (*cv.* Honeycrisp). The SMLR and RF algorithms were used for feature selection. A study reported 971, 978, 986, 987, 995, 1132, 1135, 1139, and 1143 nm as common bands associated with the disorder. Using the selected features, QDA and SVM classifiers yielded high bitter pit apple classification accuracies (78–87%). In a continuing study, Jarolmasjed et al. (2017) employed NIR spectroscopy for prediction of bitter pit development in Honeycrisp, Granny Smith, and Golden Delicious varieties. Spectral data acquired about 63 days after harvest resulted in 100% classification accuracy of healthy and bitter pit samples of Honeycrisp variety, which typically had the highest visual symptom on the fruit surface. Similar approach has been validated with hyperspectral imaging for bitter pit detection in storage apples (Jarolmasjed et al. 2018). Beyond above research studies, commercial systems that use such optical sensing approach do exist in the market. For example, Ellips Technologies (https://ellips.com/technology/) has a range of fruit-sorting system based on optical detection of internal qualities of the produce, including that of apples.

4.3 Spectral Imaging

Spectral imaging combines spectroscopy and imaging principles to simultaneously provide both spectral and spatial information of a target surface (e.g., plant canopy). Multispectral and hyperspectral are the common spectral imaging techniques that enable capturing spectral responses of a complete surface in comparison to an averaged spectrum of a limited FOV pertinent to spectrometer or spectroradiometer. Multispectral imaging refers to imaging spatial signatures within multiple but limited (or selected) number of broad wavebands (3–15) while hyperspectral imaging (HSI) captures spectra typically within hundreds of narrow wavebands (Wu and Sun 2013). Following sections elaborate on these techniques and pertinent applications in production agriculture with selective example case studies.

4.3.1 Multispectral Imaging

Multispectral imaging outputs surface reflectance in the Vis-NIR spectrum (typically 450–1100 nm) that can be related to crop stressor(s). It has gained immense adaptability in the production agriculture for monitoring soil and vegetation traits such as physiology, nutrient status, water demands, pest/disease infestation, and yield potential forecasting (Ranjan et al. 2019; Vargas et al. 2019; Guo et al. 2020; Munyati et al. 2020; Rahman et al. 2020). Table 4.2 lists some of the commercially available multispectral imaging sensors. Typically, each sensor would have few spectral bands with broad bandwidth and be configured to sense spectral reflectance at red, green, blue, and red edge and near-infrared (< 900 nm) bands, among others.

4.3.1.1 Imaging Platforms

Multispectral imaging sensors could be integrated on various satellites (Peter et al. 2020; Sicre et al. 2020), aerial platforms (Bagheri 2020; Su et al. 2020), and ground (Fig. 4.6, Sankaran et al. 2013; Svensgaard et al. 2014; Ranjan et al. 2019) platforms. Platform governs the spatial resolution, that is, regional to field to plant/leaf-level sensing and may also be a major factor in determining temporal resolution. Satellite-based imaging could be conducted from high or low earth orbits to provide details

Table 4.2 Specifications of some of the commercially available multispectral imaging sensors

Make	Model	Spectral bands [peak ± bandwidth] (nm)	Spatial resolution[a] (mm/pixel)	FOV (H° × V°)
Micasense	RedEdge	Blue (475 ± 10), green (560 ± 10), red (668 ± 5), red edge (717 ± 5) and NIR (840 ± 20)	66.60	47.9° × 36.9°
Parrot	Parrot Sequoia	RGB, green (550 ± 20), red (660 ± 20), red edge (735 ± 5) and NIR (790 ± 20)	91.67	61.9° × 48.5°
Sentera	Quad sensor	Blue (446 ± 26), green (548 ± 22), red (650 ± 32), red edge (720 ± 19.5) and NIR (839 ± 10)	22.5	50° × 39°
Sentera	Double 4 K	Blue (446 ± 26), green (548 ± 22), red (650 ± 32), red edge (720 ± 19.5) and NIR (839 ± 10)	22.5	60° (HFOV)
MAIA	M2	2 bands from visible-NIR spectrum (395–950), customizable	40	36° × 27°
Tetracam	MCA4	Blue (450 ± 5), green (530 ± 10), red (685 ± 5), red edge (740 ± 5), NIR (780 ± 5) and RGB	54.1	38.26° × 30.97°

[a]At 100 m above ground level (AGL). Disclaimer: mention of a commercial product is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated

Fig. 4.6 Multispectral imaging with (**a**) ground and (**b**) airborne platforms

from regional to field scale at geospatial resolutions ranging from low (about 3 m/pixel) to very low (30 m/pixel). Satellite-based imagers could be limited majorly due to cloud covers, atmosphere dynamics (Sun et al. 2013; Stettz et al. 2019) within high altitudes, and low revisit frequency (2–16 days). Such limitations could be eliminated with low-altitude airborne and ground-based imaging systems that provide on-demand data with minimum atmospheric interference (Dash et al. 2018). Airborne imaging systems can aid in acquiring spectral data at high spatial resolution (mm–cm/pixel) suitable to understand crop trait details at plant scale. Ground-based imaging provides data at a very high resolution (mm/pixel) to assess detailed crop characterizations from plant to leaf scales. It may however be subjected to some inconsistencies of spatial resolution, geometric and radiometric calibration, sensor's perspective, and reflectance variation due to ambient light changes between imagery samples. Ground-based imaging can also be laborious and does not provide flexibility to get high temporal resolution that is feasible with aerial approach.

4.3.1.2 Multispectral Imagery Processing

When compared to HSI, aerial multispectral imaging offers ease of data handling and may require less computational resources per unit coverage area, that is, acreage. However, accurate surface reflectance retrieval could be challenged by factors including limited number of spectral wavebands as well as imaging requirements under clear or overcast sky and near the solar noon. Noise in ground and airborne imagery due to varying light conditions and aforementioned factors could be corrected with radiometric calibrations using calibrated reflectance panels (CRP) and simultaneous solar irradiance measurements by skyward-facing sunshine sensors. Also, customized correction functions have been developed for satellite-specific multispectral imaging to minimize noises due to cloud covers and atmospheric parameters (Tasumi et al. 2007).

Meaningful information from multispectral imagery could be extracted by a series of preprocessing and processing steps. Raw imagery often contains pixelated digital numbers corresponding to the imaged surface within the sensor's FOV. These numbers are initially converted to pixel radiances using a radiometric calibration model (e.g., Eq. 4.1, source: Micasense, Inc., Seattle, WA, USA). This model compensates for sensor's black level, sensitivity, gain, exposure, and vignette effects. The resultant pixel radiance could then be converted to a surface reflectance matrix using CRP radiance and reflectance scale factors (Eq. 4.2) pertinent to each imaging waveband. For the airborne imagery, additional steps of stitching, georectification, and orthorectification need to be performed prior to conversion from digital numbers to pixel radiance. All such steps could be performed in aerial photogrammetry and mapping software platforms (e.g., Pix4DMapper from Pix4D Inc., Agisoft from Agisoft LLC, Drone2Map from Esri) that provide reflectance orthomosaics as final outputs.

$$L = V\left(x,y\right) \times \frac{a_1}{g} \times \frac{p - p_{BL}}{t_e + a_2 \times y - a_3 \times t_e \times y} \tag{4.1}$$

$$F_i = \frac{\rho_i}{avg\left(L_i\right)} \tag{4.2}$$

Where p is the normalized raw pixel value, p_{BL} is the normalized black level value, a_1, a_2, a_3 are the radiometric calibration coefficients, $V(x, y)$ is the vignette polynomial function for pixel (x, y), t_e is the image exposure time, g is the sensor gain, x, y are the pixel column and row numbers, respectively, and L is the spectral radiance in $Wm^{-2}sr^{-1}nm^{-1}$. F_i is the reflectance calibration factor for band i, ρ_i is the average reflectance of the CRP for the i^{th} band (provided by the manufacturer), $avg(L_i)$ is the average value of the radiance for the pixels inside the panel for band i.

Numerous multispectral image processing techniques have been developed and explored for crop monitoring, such as texture analysis, vegetation index (VI), and supervised classifications (Navulur 2006). Outputs from these processes can be used to extract regions of interest, zonal statistics, and prescription maps. All such processing could be performed through custom analysis algorithms and user interfaces or scripts.

4.3.1.3 Multispectral Imaging Applications

Multispectral imagery–derived VI features (Table 4.3) have been widely used in assessing a range of biotic and abiotic stressors that include nutrient deficiencies, heat and water stress, disease, and pest infestations in different field and orchard/tree crops (Table 4.4). For example, predictive modelling of chlorophyll content was performed for open tree canopy orchards (Berni et al. 2009) using transformed chlorophyll absorption in reflectance index (TCARI) normalized by the optimized soil-adjusted vegetation index (OSAVI). A high correlation (0.94) was obtained

Table 4.3 Some of the widely used vegetation indices in production agriculture

Index	Equation	Reference
Normalized Difference Vegetation Index (NDVI)	$NDVI = \frac{R_{NIR} - R_{Red}}{R_{NIR} + R_{Red}}$	Rouse et al. (1973)
Green Normalized Difference Vegetation Index (GNDVI)	$GNDVI = \frac{R_{NIR} - R_{Green}}{R_{NIR} + R_{Green}}$	Gitelson and Merzlyak (1998)
Normalized Difference RedEdge Index (NDRE)	$NDRE = \frac{R_{NIR} - R_{Red-Edge}}{R_{NIR} + R_{Red-Edge}}$	Leprieur et al. (1996)
Optimized Soil-Adjusted Vegetation Index (OSAVI)	$OSAVI = \frac{1.16 \times \left(R_{NIR} - R_{Red}\right)}{R_{NIR} + R_{Red} + 0.16}$	Rondeaux et al. (1996)
Green-Red Vegetation Index (GRVI)	$GRVI = \frac{R_{Green} - R_{Red}}{R_{Green} + R_{Red}}$	Motohka et al. (2010)
Ratio Vegetation Index (RVI)	$RVI = \frac{R_{NIR}}{R_{Red}}$	Quan et al. (2011)

between the field-measured chlorophyll content and the predictive model. Similarly, a photochemical reflectance index (PRI) was used to assess crop water stress in olive orchards, and it exhibited high correlation (0.92) with the field-measured tree xylem water potential. In another study, normalized difference vegetation index (NDVI) was used to detect sheath blight disease in a paddy field showing a high correlation (0.79) with the ground-assessed disease severity (Zhang et al. 2018). Citrus greening disease symptoms were also assessed using multispectral imagery–derived VI features and statistical support vector machine algorithm. High classification accuracy, specificity, and sensitivity above 80% were observed for the symptomatic tree leaves (Sankaran et al. 2013).

4.3.2 Hyperspectral Imaging

Hyperspectral imaging is one of the most advanced spectral sensing methods (Sankaran et al. 2015) because it can obtain both spectral and imaging information, covering hundreds of wavelengths and obtaining a wider range of internal and external information of the object (Mendoza et al. 2018). With HSI, there is an image for each wavelength, and one pixel of each image has a wavelength covering the whole spectral range. The most sensitive wavelengths can then be identified to detect crop stress or to build the low cost miniaturized optical sensing module(s). Collecting spectral information for each pixel in the image of a scene, HSI generates a

Table 4.4 Multispectral imaging applications in crop stress monitoring

Platform	Sensor (model, manufacturer)	Vegetation index	Application	Reference
UV-based mast	6-band (MCA-5, Tetracam, Inc., CA, USA)	SIPI, VOG, mNDVI, mSR, RE-NDVI, SR, NDVI	Citrus greening	Sankaran et al. (2013)
UV-based mast	5-band (Red edge 3, MicaSense, Inc., WA, USA)	Broadband vegetation indices	Pinto bean crop stress and yield	Ranjan et al. (2019)
Small UAS	6-band (MCA-6, Tetracam, Inc., CA, USA)	NDVI, TCARI, OSAVI	Leaf area and chlorophyll	Berni et al. (2009)
Small UAS	5-band (Red edge M, MicaSense, Inc., WA, USA)	NDVI, RVI and OSAVI	Wheat yellow rust	Su et al. (2018)
Small UAS	5-band (Sequoia, parrot, Paris, France)	NDVI	Post fire vegetation survey	Fernández-Guisuraga et al. (2018)
Small UAS	5-band (Red edge 3, MicaSense, Inc., WA, USA)	NDVI	Rice sheath blight	Zhang et al. (2018)
Small UAS	5-band (Red edge 3, MicaSense, Inc., WA, USA)	NDVI, GRVI	Citrus greening	Javan et al. (2019)
Small UAS	Double 4 K (Sentera LLC, MN, USA)	NDVI, NDRE	Pea biomass phenotyping	Vargas et al. (2019)
Small UAS	5-band (Red edge M, MicaSense, Inc., WA, USA)	NDVI	Cotton plant phenotyping	Xu et al. (2019)
Low orbital satellite	13-band (sentinel-2A L1C, Paris, France)	NDVI	Cotton root rot	Song et al. (2017)
High orbital satellite	4-band (HJ-CCD sensor, China)	NDVI, SAVI, GNDVI, SIPI, OSAVI, PSRI	Wheat grain protein	Tan et al. (2020)

UV Utility Vehicle, *SR* Simple Ratio, *SIPI* Structure Insensitive Pigment Index, *VOG* Vogelmann Red Edge Index, *NDVI* Normalized Difference Vegetation Index, *mNDVI* Modified NDVI, *RE-NDVI* Red Edge Normalized Difference Vegetation Index, NDRE Normalized Difference Red Edge Index, *mSR* Modified Red Edge Simple Ratio, *GRVI* Green Red Vegetation Index, *RVI* Relative Vigor Index, *OSAVI* Soil Adjusted Vegetation Index, *PSRI* Plant Senescence Reflectance Index, *TCARI* Transformed Chlorophyll Absorption Reflectance Index. Disclaimer: mention of a commercial product is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated.

three-dimensional (3D) dataset of spatial and spectral information, known as hypercube. This 3D dataset comprises two spatial dimensions (x rows × y columns) and one spectral dimension (λ wavelengths), as illustrated in Fig. 4.7.

Three common techniques of acquiring the 3D (x, y, λ) HSI datasets include spatial, spectral, and spatial–spectral scanning methods (Fig. 4.8). The spatial scanning method uses either point scanning or line scanning, the spectral scanning

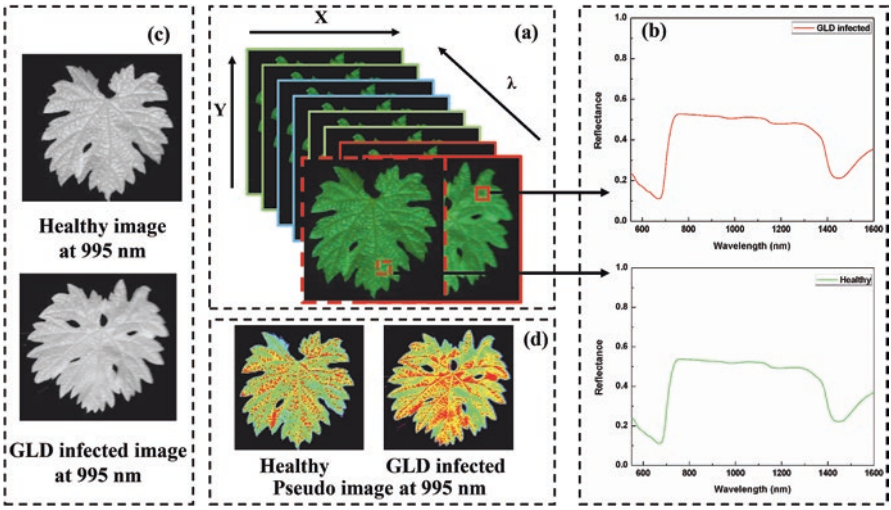


Fig. 4.7 (**a**) Hyperspectral image data cube, (**b**) point sampled spectra of grapevine leafroll disease infected and healthy leaf, (**c**) related grayscale images, and (**d**) pseudo color images

method uses the area scanning approach, and the spatial–spectral scanning method uses the snapshot imaging approach (Sellar and Boreman 2005). Selecting the suitable hyperspectral technique depends on application and imaging requirements.

In the point or whiskbroom method of spatial scanning, a single point is scanned along two spatial dimensions by moving either the sample or the sensor (Fig. 4.8). The point scanning method is used to acquire a spectrum for each pixel in the scene. The line scanning or push broom method can be considered as an extension of the point scanning method. In this approach, the sensor is configured to simultaneously acquire a slit of spatial information as well as spectral information corresponding to each spatial point in the slit. A special 2D image (y, λ) with one spatial dimension and one spectral dimension is taken at a time. A complete hypercube is then generated by aggregating slit scans in the direction of motion (x).

The spectral scanning method commonly employs the area scanning method, also known as sequential method. This method acquires a single band 2D grayscale image (x, y) with full spatial information at once (Fig. 4.9). The hypercube containing a stack of single band images is built up as the scanning is performed in the spectral domain through a number of wavelengths. No relative movement between the sample and the detector is required during such scanning approach. The spatial–spectral or snapshot scanning method is intended to record both spatial and spectral information of the FOV with one exposure. No scanning in either spatial or spectral domain is needed for obtaining a 3D image cube (Grusche 2014).

The spatial scanning does not provide live display of spectral images, which is calculated from the scanned spectra of the corresponding area. Spectral scanning thus has advantage in that it displays live spectral images that may be essential for adjusting optical settings related to FOV and focus within FOV. As of today, the

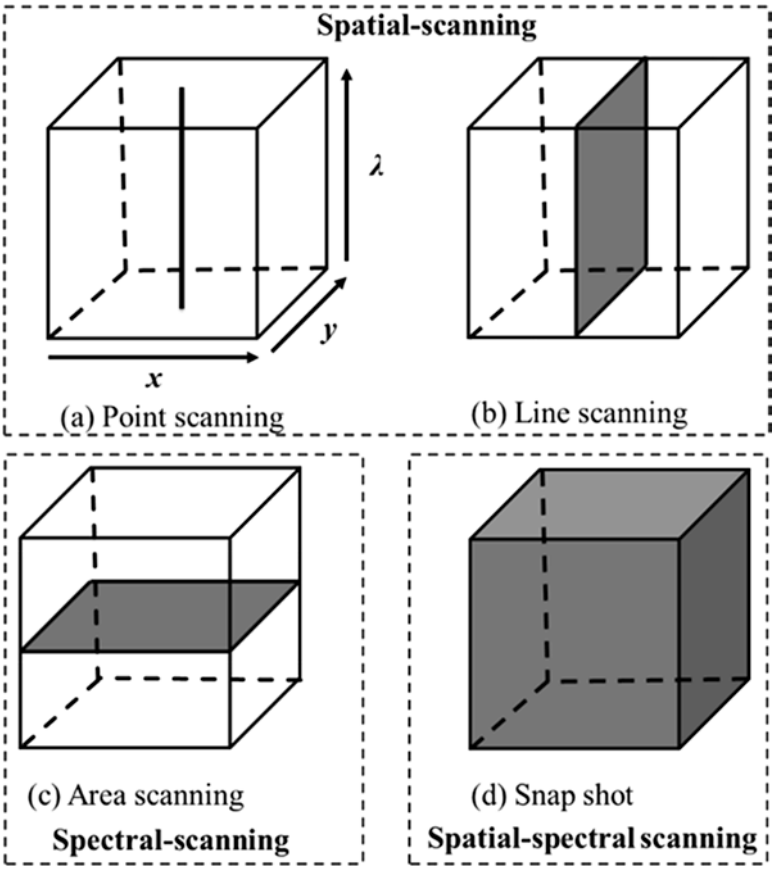


Fig. 4.8 Hypercube acquisition modes: spatial (**a**) point scanning and (**b**) line scanning, (**c**) spectral area scanning, and (**d**) spatial–spectral snapshot scanning

available spatial–spectral-based HSI imaging sensors provide low spatial or spectral resolutions. On the positive side, as this approach has all the information required to build a hypercube using captured data, it eliminates motion artifacts typical to other acquisition methods. This allows image capture at video frame rates, making this technology suitable for high-speed remote sensing applications (Kester et al. 2011).

4.3.2.1 Data Analysis Methods

Data Preprocessing The purpose of hyperspectral imagery preprocessing is to eliminate unintended distortions. Preprocessing can also enhance salient image features. The field of HSI data processing is well developed, and to keep this chapter scope focused, we are not covering those steps (Baxes 1994; Nixon and Aguado 2012). In terms of image preprocessing, four basic methods including brightness

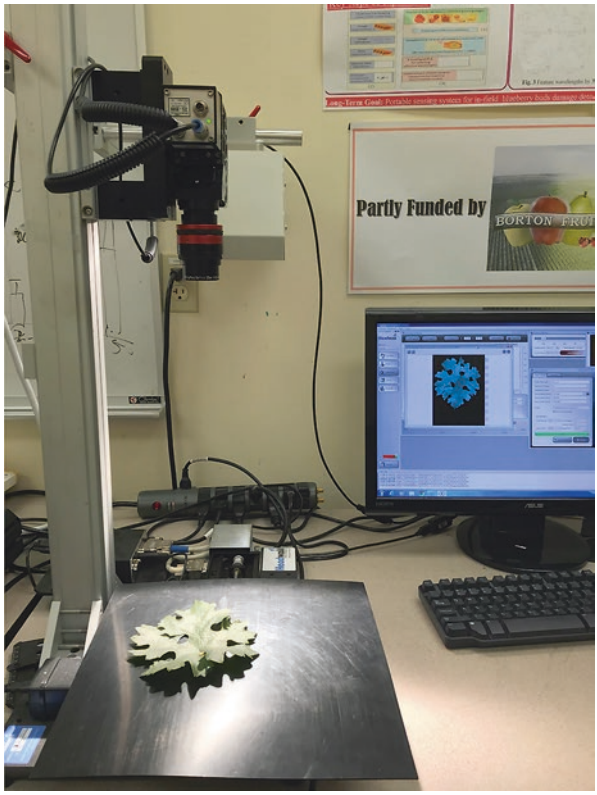


Fig. 4.9 Hyperspectral imaging sensor acquiring spectral response for grapevine leafroll disease detection

transformations, geometric transformations, local neighborhood preprocessing, and image restoration can be performed on the pertinent spectral data.

There exist different types of random spectra noises including the environment, instruments, and the biological sample itself. To remove such noise, many spectral data preprocessing methods have been utilized by the researchers. Simple multivariate chemometric methods include PCA and PLSR analysis. Furthermore, baseline correction (BC), derivative, Fourier transform (FT), mean centering (MC), multiplicative scatter correction (MSC), normalization, orthogonal signal correction (OSC), smoothing, standard normal variate (SNV), wavelet transform (WT), or combinations of those can be used to preprocess the hyperspectral imagery data.

Feature Selection Spectral images could provide physical attributes of the objects, such as plant canopy vigor, leaf color, leaf texture, size, and shape information (Mahlein 2016). Color and texture features are important for identifying the characteristic difference between different objects, that is, healthy and symptomatic plants. Normally used color features are RGB (Red, Green, Blue), YCBCR (Luminance, Chrominance), and HSV (Hue, Saturation, Value) spaces. Contrast, homogeneity, dissimilarity, energy, and entropy features often describe texture-level changes.

In a recent review, Zou et al. (2010) have grouped the spectral imaging feature selection methods into manual, regression, and sophisticated methods. In the manual method, typically an experienced researcher scans the data for the peak absorbance of the known chemical components, such as the C···H (aliphatic), C···H (aromatic), C···O (carboxyl), O···H (hydroxyl), and N···H (amine and amide). Regression methods could be single and multiple linear regression (MLR). Stepwise, forward selection, and backward elimination–based MLR methods are often used by the scientists (Balabin and Smirnov 2011; Liu et al. 2015) while analyzing HSI data. Lastly, as discussed in the prior section, sophisticated methods such as SPA, UVE, Gas, and iPLS can be used for salient wavelengths/feature selection (Cheng et al. 2016; Jiao et al. 2016; Li et al. 2017). These methods have been discussed in detail in Sect. 4.2.3 of this chapter. Several new methods such as band ratio, competitive adaptive reweighted sampling (CARS), receiver operating characteristic (ROC) analysis, branch and bound (BB) algorithm, minimum redundancy-maximum relevance (MRMR), normalized difference spectral indices (NDSI), and ratio spectral indices (RSI) have also been explored (Liu et al. 2014; Wang et al. 2017; Chen et al. 2018) during HSI data analysis.

Classification/Prediction Models In HSI data analysis, the multivariate regression techniques aim at establishing a relationship between a desired physical, chemical, or biological attribute of an object and its observed spectral and spatial response. Multivariate classification can be done using either linear or nonlinear regression analysis. The conventional linear regression methods include hierarchical linear models, linear least squares, ridge regression, Bayesian linear regression, and principal component regression (Yan & Su, 2009). Similarly, the conventional nonlinear regression methods include nonlinear least squares, Gauss–Newton algorithm, and gradient descent algorithm (Bates et al., 1988). Classification/prediction models are necessary to assess spectral features robustness in detecting plant stress. For example, such a model's performance can be evaluated by correlation coefficient (r), standard error of calibration (SEC), standard error of prediction (SEP), and root mean square error estimated by cross-validation (RMSECV). A good model should have a lower value of SEC, SEP, and RMSECV as well as a higher value of r and a small difference between SEC and SEP (Cen and He 2007).

4.3.2.2 Hyperspectral Imaging Applications

Table 4.5 summarizes HSI applications domain in plant stressors detection with pertinent sensor, associated feature section and classification/prediction model details. Overall, most applications have used HSI in laboratory settings. To date, it is challenging to employ the hyperspectral sensor in the field conditions due to sensor sensitivity to the varying light and environment conditions. For example, the wind will cause the plant leaves motion and could result in the blurred images acquired using push broom–type HSI imaging system. Current HSI system development efforts thus are focused on developing snapshot-type sensors which would have needed spatial and spectral resolution for use in crop stressors mapping.

Table 4.5 Plant diseases detection using hyperspectral imaging with common feature selection and classification methods

Sensor (model, manufacture)	Feature selection	Classification/ prediction	Crop type	Disease type	References
HSI system–based on LCTF	ROC curve	ELM	Citrus fruit	Decay	Verlag et al. (2013)
Short-wave infrared (headwall photonics Inc., MA, USA)	PLS-DA	PLS-DA, LS-SVM	Watermelon seeds	Cucumber green mottle mosaic virus	Lee et al. (2016)
Pika XC (Resonon Inc., MT, USA)	GA	SVM	Soybean	Charcoal rot disease	Nagasubramanian et al. (2018)
ImSpectorV10 (spectral imaging ltd., Finland)	Features ranking	kNN	Tomato	Gray mold	Xie et al. (2017)
ImSpectorV10 (spectral imaging ltd., Finland)	SPA	BPNN, ELM, LS-SVM	Tobacco	Tobacco mosaic virus	Zhu et al. (2016)
ImSpectorV10 (spectral imaging ltd., Finland)	PCA	PLS-DA, SVM, RBFNN, ELM	Oilseed rape	Sclerotinia sclerotiorum	Kong et al. (2018)
FieldSpec UV/ VNIR (ASD Inc. Colorado, USA)	NDSI	LRA	Peanut	Leaf spots disease	Chen et al. (2019)
Pika XC (Resonon Inc., MT, USA)	SPA	QDA, KNN and SVM	Oilseed rape	Waterlogging stress	Xia et al. (2019)
ImSpectorV10 (spectral imaging ltd., Finland)	PCA, SPA	BPNN	Wheat	Wheat stripe rust	Yao et al. (2019)

Note: *LR* Logistic regression; *PLS* Partial least squares, *PLS–DA* Partial least squares–discriminant analysis, *LS–SVM* Least–Squares Support Vector Machine, *GA* Genetic Algorithm, *SVM* Support vector machine, *KNN* k–Nearest Neighbors, *ELM* Extreme Learning Machine, *BPNN* back propagation neural network, *LRA* linear regression analysis, *LCTF* liquid crystal tunable filters, *NDSI* Normalized difference spectral index

4.4 Summary and Concluding Thoughts

The spectroscopy and spectral imaging technologies offer great promise for enhanced characterization of plant stressors. However, as discussed in the above sections, many hurdles need to be resolved for practical field use of these techniques. Spectrometers on the market are being customized for use case scenarios, and additional features that integrate real-time analytics, wireless network

connectivity, and on-board/cloud computing ability would make them more appealing to end users. In case of hyperspectral spectroradiometers and imaging sensors, real-time sensing remains a challenge due to lack of on-board "big data" processing options with streamlined workflow in a chosen software suite. Those sensors are also bulky to be integrated with low altitude aerial sensing platforms, limiting their wider use. Thus, additional research is needed to have miniaturized modules with autocalibration and fast-capturing high radiometric resolution detectors. Some standardization efforts are also needed to capture pertinent metadata, both for multi- and hyperspectral domain, that is compatible and accepted universally in a range of data analytics tools. As a wide range of explorations are on-going globally, it would also be beneficial if there exists a common database of spectral libraries for given crop and stressor type. This would ultimately avoid repetition of efforts to some extent and/or validation of a given technique in a similar type of stress detection in a geographically distinct region.

References

Al–Saddik H, Cointault J (2019) Assessment of the optimal spectral bands for designing a sensor for vineyard disease detection: the case of '*Flavescence dorée*'. Precis Agric 20(2):398–422

Araújo M, Saldanha T, Galvão R, Yoneyama T, Chame H, Visani V (2001) The successive projections algorithm for variable selection in spectroscopic multicomponent analysis. Chemom Intell Lab Syst 57:65–73

Bagheri N (2020) Application of aerial remote sensing technology for detection of fire blight infected pear trees. Comput Electron Agric 168:105147

Balabin R, Smirnov S (2011) Variable selection in near–infrared spectroscopy: benchmarking of feature selection methods on biodiesel data. Anal Chim Acta 692(1–2):63–72

Bates DM, Watts DG (1988) Nonlinear regression analysis and its applications (Vol. 2). New York: Wiley

Baxes GA (1994) Digital image processing: principles and applications (pp. I-XVIII). New York: Wiley

Beghi R, Giovenzana V, Guidetti R (2017) Better water use efficiency in vineyard by using visible and near infrared spectroscopy for grapevine water status monitoring. Chem Eng Trans 58:691–696

Berni J, Zarco–Tejada P, Suarez L, Fereres E (2009) Thermal and narrowband multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle. IEEE Trans Geosci Remote Sens 47(3):722–738

Cao Q, Miao Y, Shen J, Yuan F, Cheng S, Cui Z (2018) Evaluating two crop circle active canopy sensors for in–season diagnosis of winter wheat nitrogen status. Agronomy 8:1–17

Carter G, Knapp A (2001) Leaf optical properties in higher plants: linking spectral characteristics to stress and chlorophyll concentration. Am J Bot 88(4):677–684

Cen H, He Y (2007) Theory and application of near infrared reflectance spectroscopy in determination of food quality. Trends Food Sci Technol, 18(2):72–83

Centner V, Massart D (1996) Elimination of uninformative variables for multivariate calibration. Anal Chem 68:3851–3858

Chang C, Laird D, Mausbach M, Hurburgh C (2001) Near–infrared reflectance spectroscopy–principal components regression analyses of soil properties. Soil Sci Soc Am J 65:480–490

Chen J, Iyo C, Terada F, Kawano S (2002) Effect of multiplicative scatter correction on wavelength selection for near infrared calibration to determine fat content in raw milk. Near Infrared Spectrosc 10:301–307

Chen M, Zhang L, Xu H (2015) On–line detection of blood spot introduced into brown–shell eggs using visible absorbance spectroscopy. Biosyst Eng 131:95–101

Chen T, Zeng R, Guo W, Hou X, Lan Y, Zhang L (2018) Detection of stress in cotton (Gossypium hirsutum L.) caused by aphids using leaf level hyperspectral measurements. *Sensors 18*(9):2798

Chen T, Zhang J, Chen Y, Wan S, Zhang L (2019) Detection of peanut leaf spots disease using canopy hyperspectral reflectance. Comput Electron Agric 156:677–683

Cheng J, Sun D, Pu H (2016) Combining the genetic algorithm and successive projection algorithm for the selection of feature wavelengths to evaluate exudative characteristics in frozen–thawed fish muscle. Food Chem 197:855–863

Costa D, Mesa N, Freire M, Ramos R, Mederos B (2019) Development of predictive models for quality and maturation stage attributes of wine grapes using vis–nir reflectance spectroscopy. Postharvest Biol Technol 150:166–178

de Bei R, Cozzolino D, Sullivan W, Cynkar W, Fuentes S, Dambergs R, Pech J, Tyerman S (2010) Non–destructive measurement of grapevine water potential using near infrared spectroscopy. Aust J Grape Wine Res 17(1):62–71

de Santa F, de Souza A, Poppi R (2018) Visible and near infrared spectroscopy coupled to random forest to quantify some soil quality parameters. Spectrochim Acta A Mol Biomol Spectrosc 191:454–462

Dan S, Yang S, Tian F, Den L (2015) Classification of orange growing locations based on the near–infrared spectroscopy using data mining. Intell Autom Soft Computing 22(2):229–236

Dankowska A, Kowalewski W (2019) Tea types classification with data fusion of UV–Vis, synchronous fluorescence and NIR spectroscopies and chemometric analysis. Spectrochim Acta A Mol Biomol Spectrosc 211:195–202

Dash J, Pearse G, Watt M (2018) UAV multispectral imagery can complement satellite data for monitoring forest health. Remote Sens 10(8):1216

Dhanoa M, Lister S, Sanderson R, Barnes R (1994) The link between multiplicative scatter correction (MSC) and standard Normal variate (SNV) transformation of NIR spectra. J Near Infrared Spectrosc 2(1):43–47

Dong J, Dong X, Li Y, Peng Y, Chao K, Gao C, Tang X (2019) Identification of unfertilized duck eggs before hatching using visible/near infrared transmittance spectroscopy. Comput Electron Agric 157:471–478

Dotto A, Dalmolin R, ten Caten A, Grunwald S (2018) A systematic study on the application of scatter-corrective and spectralderivative preprocessing for multivariate prediction of soil organic carbon by Vis-NIR spectra. Geoderma 314:262–274

Fernández-Guisuraga J, Sanz-Ablanedo E, Suárez-Seoane S, Calvo L (2018) Using unmanned aerial vehicles in postfire vegetation survey campaigns through large and heterogeneous areas: opportunities and challenges. Sensors 18(2):586

Ferreiro-González M, Barbero G, Álvarez J, Ruiz A, Palma M, Ayuso J (2017) Authentication of virgin olive oil by a novel curve resolution approach combined with visible spectroscopy. Food Chem 220(1):331–336

Feyaerts F, Pollet P, Gool L, Wambacq P (1998) Sensor for weed detection based on spectral measurements. In: Proceedings of the 4th international conference on precision agriculture, pp 1537–1548

Geladi P, McDougel D, Martens H (1985) Linearization and scatter–correction for near infrared reflectance spectra of meat. Appl Spectrosc 39:491–500

Gerretzen J, Szymańska E, Jansen J, Bart J, van Manen H, van den Heuvel E, Lutgarde MC, Buydens L (2015) Simple and effective way for data preprocessing selection based on design of experiments. Anal Chem 87:12096–12103

Gitelson A, Merzlyak M (1998) Remote sensing of chlorophyll concentration in higher plant leaves. Adv Space Res 22:689–692

Grusche S (2014) Basic slit spectroscope reveals three–dimensional scenes through diagonal slices of hyperspectral cubes. Appl Opt 53(20):4594–4603
Guo Y, Cao H, Han S, Sun Y (2018) Spectral–spatial hyperspectral image classification with k–nearest neighbor and guided filter. IEEE Access 18:18582–18591
Guo L, Fu P, Shi T, Chen Y, Zhang H, Meng R, Wang S (2020) Mapping field–scale soil organic carbon with unmanned aircraft system–acquired time series multispectral images. Soil Tillage Res 196:104477
He Y, Li X, Shao Y (2007) Fast discrimination of apple varieties using Vis/NIR spectroscopy. Int J Food Prop 10(1):9–18
Hira Z, Gillies D (2015) A review of feature selection and feature extraction methods applied on microarray data. Adv Bioinforma 198363:1–13
Jarolmasjed S, Espinoza C, Sankaran S (2017) Near infrared spectroscopy to predict bitter pit development in different varieties of apples. J Food Meas Charact 11(3):987–993
Jarolmasjed S, Khot L, Sankaran S (2018) Hyperspectral imaging and spectrometry–derived spectral features for bitter pit detection in storage apples. Sensors 18:1561
Javan F, Samadzadegan F, Pourazar S, Fazeli H (2019) UAV-based multispectral imagery for fast Citrus greening detection. J Plant Dis Prot 126:307–318
Jiao L, Bing S, Zhang X, Wang Y, Li H (2016) Application of fluorescence spectroscopy combined with interval partial least squares to the determination of enantiomeric composition of tryptophan. Chemom Intell Lab Syst 156:181–187
Jr S, Barbon A, Mantovani R, Barbin D (2018) Machine learning applied to near-infrared spectra for chicken meat classification. J Spectrosc 2018:8949741
Kafle G, Khot L, Jarolmasjed S, Yongsheng S, Lewis K (2016) Robustness of near infrared spectroscopy based spectral features for non–destructive bitter pit detection in honeycrisp apples. Postharvest Biol Technol 120:188–192
Kester RT, Bedard N, Tkaczyk TS (2011, May) Image mapping spectrometry: a novel hyperspectral platform for rapid snapshot imaging. In: Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XVII (Vol. 8048, p. 80480J). International Society for Optics and Photonics
Khorramnia K, Khot L, Shariff A, Ehsani R, Mansor S, Rahim A (2014) Oil palm leaf nutrient estimation by optical sensing techniques. Trans ASABE 57(4):1267–1277
Khosravi V, Ardejani F, Yousefi S, Aryafar A (2018) Monitoring soil lead and zinc contents via combination of spectroscopy with extreme learning machine and other data mining methods. Geoderma 318:29–41
Kim Y, Glenn D, Park J, Ngugi H, Lehman B (2012) Characteristics of active spectral sensor for plant sensing. Trans ASABE 55(1):293–301
Kong W, Zhang C, Huang W, Liu F, He Y (2018) Application of hyperspectral imaging to detect *Sclerotinia sclerotiorum* on oilseed rape stems. Sensors 18(2):123
Kooistra L, Wehrens R, Leuven R, Buydens L (2001) Possibilities of visible–near–infrared spectroscopy for the assessment of soil contamination in river floodplains. Anal Chim Acta 446:97–105
Larbi P, Ehsani R, Salyani M, Maja J, Mishra A, Neto J (2013) Multispectral–based leaf detection system for spot sprayer application to control citrus psyllids. Biosyst Eng 116(4):509–517
Lee H, Park Y, Lee S (2015) Principal component regression by principal component selection. Commun Stat Appl Methods 22(2):173–180
Lee H, Kim M, Lim H, Park E, Lee W, Cho B (2016) Detection of cucumber green mottle mosaic virus-infected watermelon seeds using a near–infrared (NIR) hyperspectral imaging system: application to seeds of the "Sambok Honey" cultivar. Biosyst Eng 148:138–147
Leprieur C, Kerr Y, Pichon J (1996) Critical assessment of vegetation indices from AVHRR in a semi-arid environment. Int J Remote Sens 17(13):2459–2463
Li C, Zhao T, Li C, Mei L, Yu E, Dong Y, Chen J, Zhu S (2017) Determination of gossypol content in cottonseeds by near infrared spectroscopy based on Monte Carlo uninformative variable elimination and nonlinear calibration methods. Food Chem 221:990–996

Liu D, Sun DW, Zeng XA (2014) Recent advances in wavelength selection techniques for hyperspectral image processing in the food industry. Food Bioprocess Technol, 7(2):307–323
Liu K, Chen X, Li L, Chen H, Ruan X, Liu W (2015) A consensus successive projections algorithm – multiple linear regression method for analyzing near infrared spectra. Anal Chim Acta 858:16–23
Lorente D, Escandell-Montero P, Cubero S, Gómez-Sanchis J, Blasco J (2015) Visible–NIR reflectance spectroscopy and manifold learning methods applied to the detection of fungal infections on citrus fruit. J Food Eng 163:17–24
Mahlein A (2016) Plant disease detection by imaging sensors – parallels and specific demands for precision agriculture and plant phenotyping. Plant Dis 100(2):241–251
Mehmood T, Liland K, Snipen L, Sæbø S (2012) A review of variable selection methods in partial least squares regression. Chemom Intell Lab Syst 118:62–69
Mendoza F, Cichy K, Sprague C, Goffnett A, Lu R, Kelly J (2018) Prediction of canned black bean texture (Phaseolus vulgaris L.) from intact dry seeds using visible/near infrared spectroscopy and hyperspectral imaging data. J Sci Food Agric 98(1):283–290
Mishra A, Karimi D, Ehsani R, Lee W (2012) Identification of citrus greening (HLB) using a VIS-NIR spectroscopy technique. Trans ASABE 55(2):711–720
Motohka T, Nasahara K, Oguma H, Tsuchida S (2010) Applicability of green–red vegetation index for remote sensing of vegetation phenology. Remote Sens 2(10):2369–2387
Munyati C, Balzter H, Economon E (2020) Correlating Sentinel–2 MSI–derived vegetation indices with in–situ reflectance and tissue macronutrients in savannah grass. Int J Remote Sens 41(10):3820–3844
Nagasubramanian K, Jones S, Sarkar S, Singh AK, Singh A, Ganapathysubramanian B (2018) Hyperspectral band selection using genetic algorithm and support vector machines for early identification of charcoal rot disease in soybean stems. Plant Methods 14(1):86
Naidu R, Perry E, Pierce F, Mekuria T (2009) The potential of spectral reflectance technique for the detection of Grapevine leafroll–associated virus−3 in two red–berried wine grape cultivars. Comput Electron Agric 66:38–45
Navulur K (2006) Multispectral image analysis using the object-oriented paradigm. CRC press, Boca Raton
Nawar S, Mouazen A (2017) Comparison between random forests, artificial neural networks and gradient boosted machines methods of on-line Vis-NIR spectroscopy measurements of soil total nitrogen and total carbon. Sensors 17(10):2428
Nixon M, Aguado AS (2012) Feature extraction and image processing for computer vision. Academic Press
Nørgaard L, Saudland A, Wagner J, Nielsen J, Munck L, Englesen S (2000) Interval partial least-squares regression (iPLS): a comparative chemometric study with an example from near–infrared spectroscopy. Appl Spectrosc 54(3):413–419
Peng X, Shi T, Song A, Chen Y, Gao W (2014) Estimating soil organic carbon using VIS/NIR spectroscopy with SVMR and SPA methods. Remote Sens 6:2699–2717
Peter B, Messina J, Carroll J, Zhi J, Chimonyo V, Lin S, Snapp S (2020) Multi-spatial resolution satellite and sUAS imagery for precision agriculture on smallholder farms in Malawi. Photogramm Eng Remote Sens 86(2):107–119
Puertas G, Vázquez M (2019) Cholesterol determination in egg yolk by UV-VIS-NIR spectroscopy. Food Control 100:262–268
Quan Z, Xianfeng Z, Miao J (2011) Eco-environment variable estimation from remote sensed data and eco-environment assessment: models and system. Acta Bot Sin 47:1073–1080
Rácz A, Gere A, Bajusz D, Héberger K (2018) Is soft independent modeling of class analogies a reasonable choice for supervised pattern recognition? RSC Adv 8:10–21
Rady A, Adedeji A (2018) Assessing different processed meats for adulterants using visible–near–infrared spectroscopy. Meat Sci 136:59–67

Rahman G, Sohag H, Chowdhury R, Wahid K, Dinh A, Arcand M, Vail S (2020) SoilCam: a fully automated minirhizotron using multispectral imaging for root activity monitoring. Sensors 20(3):787

Ranjan R, Chandel A, Khot L, Bahlol H, Zhou J, Boydston R, Miklas P (2019) Irrigated pinto bean crop stress and yield assessment using ground based low altitude remote sensing technology. Inf Process Agri 6(4):502–514

Rinnan A, van den Berg F, Engelsen S (2009) Review of the most common pre–processing techniques for near–infrared spectra. Trends Anal Chem 28(10):1201–1222

Rondeaux G, Steven M, Baret F (1996) Optimization of soil–adjusted vegetation indices. Remote Sens Environ 55:95–107

Rouse J, Haas R, Schell J, Deering D (1973) Monitoring vegetation systems in the Great Plains with ERTS. In: Proceedings of third ERTS symposium, pp 309–317

Sankaran S, Ehsani R (2011) Visible–near infrared spectroscopy based citrus greening detection: evaluation of spectral feature extraction techniques. Crop Prot 30:1508–1513

Sankaran S, Mishra A, Maja J, Ehsani R (2011) Visible–near infrared spectroscopy for detection of Huanglongbing in citrus orchards. Comput Electron Agric 77:127–134

Sankaran S, Maja J, Buchanon S, Ehsani R (2013) Huanglongbing (Citrus greening) detection using visible, near infrared and thermal imaging techniques. Sensors 13:2117–2130

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

Sellar R, Boreman G (2005) Classification of imaging spectrometers for remote sensing applications. Opt Eng 44(1):013602

Shirzadifar A, Bajwa S, Mireei S, Howatt K, Nowatzki J (2018) Weed species discrimination based on SIMCA analysis of plant canopy spectral data. Biosyst Eng 171:143–154

Sicre C, Fieuzal R, Baup F (2020) Contribution of multispectral (optical and radar) satellite images to the classification of agricultural surfaces. Int J Appl Earth Obs Geoinf 84:101972

Sinha R, Gao Z, Rathanayake A, Khot L, Naidu R (2017a) Visible–near infrared spectroscopy based grapevine leafroll–associated virus–3 detection from undetached leaves under field condition. ASABE St. Joseph, Mich. ASABE paper no. 1700499

Sinha R, Khot L, Schroeder B (2017b) FAIMS based sensing of Burkholderia cepacia caused sour skin in onions under bulk storage condition. J Food Meas Charact *11*:1578–1585

Sinha R, Khot L, Schroeder B, Si Y (2017c) Rapid and non–destructive detection of *Pectobacterium carotovorum* causing soft rot in stored potatoes through volatile biomarkers sensing. Crop Prot 93:122–131

Sinha R, Khot L, Schroeder B, Sankaran S (2018) FAIMS based volatile fingerprinting for real-time postharvest storage infections detection in stored potatoes and onions. Postharvest Biol Technol 135:83–92

Sinha R, Khot L, Rathnayake A, Gao Z, Naidu R (2019) Visible–near infrared spectroradiometry–based detection of grapevine leafroll–associated virus 3 in a red–fruited wine grape cultivar. Comput Electron Agric 162:165–173

Song X, Yang C, Wu M, Zhao C, Yang G, Hoffmann WC, Huang W (2017) Evaluation of sentinel–2A satellite imagery for mapping cotton root rot. Remote Sens 9(9):906

Srivastava S, Gupta M, Frigyik B (2007) Bayesian quadratic discriminant analysis. J Mach Learn Res 8:1277–1305

Stettz S, Zaitchik B, Ademe D, Musie S, Simane B (2019) Estimating variability in downwelling surface shortwave radiation in a tropical highland environment. PLoS One 14(2):e0211220

Su J, Liu C, Coombes M, Hub X, Wang C, Xuc X, Li Q, Guo L, Chen W (2018) Wheat yellow rust monitoring by learning from multispectral UAV aerial imagery. Comput Electron Agric 155:157–166

Su J, Yi D, Liu C, Mi Z, Su B, Hu X, Xu X, Guo L, Chen W (2020) Aerial visual perception in smart farming: field study of wheat yellow rust monitoring. IEEE Trans Ind Inf 17:2242–2249

Sun Z, Gebremichael M, Wang Q, Wang J, Sammis T, Nickless A (2013) Evaluation of clear–sky incoming radiation estimating equations typically used in remote sensing evapotranspiration algorithms. Remote Sens 5(10):4735–4752

Svensgaard J, Roitsch T, Christensen S (2014) Development of a mobile multispectral imaging platform for precise field phenotyping. Agronomy 4(3):322–336

Swierenga H, de Groot P, de Weijer A, Derksen M, Buydens L (1998) Improvement of PLS model transferability by robust wavelength selection. Chemom Intell Lab Syst 41:237–248

Tan C, Zhou X, Zhang P, Wang Z, Wang D, Guo W, Yun F (2020) Predicting grain protein content of field-grown winter wheat with satellite images and partial least square algorithm. PLoS One 15(3):e0228500

Tang J, Alelyani S, Liu H (2018) Feature selection for classification: a review. In: Data classification: algorithms and applications, 37. Retrieved from https://pdfs.semanticscholar.org/310e/a531640728702fce6c743c1dd680a23d2ef4.pdf. Accessed April 2019

Tasumi M, Allen R, Trezza R (2007) Estimation of at surface reflectance and albedo from satellite for routine, operational calculation of land surface energy balance. J Hydrol Eng 13(2):51–63

Thomas S, Kuska M, Bohnenkamp D, Brugger A, Alisaac E, Wahabzada M, Behmann J, Mahlein A (2018) Benefits of hyperspectral imaging for plant disease detection and plant protection: a technical perspective. J Plant Dis Prot 125:5–20

Vargas J, Zhang C, Smitchger J, McGee R, Sankaran S (2019) Phenotyping of plant biomass and performance traits using remote sensing techniques in pea (*Pisum sativum*, L.). Sensors 19(9):2031

Vašát R, Kodešová R, Borůvka L (2017) Ensemble predictive model for more accurate soil organic carbon spectroscopic estimation. Comput Geosci 104:75–83

Verlag S, Lorente D, Aleixos N, Gómez-Sanchis J, Cubero S, Blasco J (2013) Selection of optimal wavelength features for decay detection in citrus fruit using the ROC curve and neural networks. ROC Curve and Neural Networks. Food Bioprocess Technol 6(2):530–541

Wang W, Pailwal J, Jayas D (2004) Determination of moisture content of ground wheat using near-infrared spectroscopy. ASAE/CSAE Meeting presentation, paper no. MB04–200

Wang L, Chang CI, Lee LC, Wang Y, Xue B, Song M, li S (2017) Band subset selection for anomaly detection in hyperspectral imagery. IEEE Transactions on Geoscience and Remote Sensing, 55(9):4887–4898

Wiecek B (2005) Cooling and shielding systems for infrared detectors. In: Proceedings of the 2005 IEEE. Engineering in medicine and biology. 27th annual conference

Windig W, Shaver J, Bro R (2008) Loopy MSC: a simple way to improve multiplicative scatter correction. Appl Spectrosc 62(10):1153–1159

Wu D, Sun D (2013) Advanced applications of hyperspectral imaging technology for food quality and safety analysis and assessment: a review- Part I: fundamentals. Innovative Food Sci Emerg Technol 19:1–14

Xia J, Cao H, Yang Y, Zhang W, Wan Q, Xu L, DaoKuo G, WenYu Z, YaQi K, Huang B (2019) Detection of waterlogging stress based on hyperspectral images of oilseed rape leaves (Brassica napus L.). Comput Electron Agric 159:59–68

Xiaobo Z, Jiewen Z, Povey M, Holmes M, Hanpin M (2010) Variables selection methods in near-infrared spectroscopy. Anal Chim Acta 667(1–2):14–32

Xie C, Yang C, He Y (2017) Hyperspectral imaging for classification of healthy and gray mold diseased tomato leaves with different infection severities. Comput Electron Agric 135:154–162

Xu R, Li C, Paterson AH (2019) Multispectral imaging and unmanned aerial systems for cotton plant phenotyping. PLoS One 14(2):e0205083

Yan X, Su X (2009) Linear regression analysis: theory and computing. World Scientific

Yao Z, Lei Y, He D, Yao Z, Lei Y, He D (2019) Early visual detection of wheat stripe rust using visible/near–infrared hyperspectral imaging. Sensors 19(4):952

Yue J, Feng H, Yang G, Li Z (2018) A comparison of regression techniques for estimation of above–ground winter wheat biomass using near–surface spectroscopy. Remote Sens 10:66

Zhang D, Zhou X, Zhang J, Lan Y, Xu C, Liang D (2018) Detection of rice sheath blight using an unmanned aerial system with high–resolution color and multispectral imaging. PLoS One 13(5):e0187470

Zhu H, Cen H, Zhang C, He Y (2016) Early detection and classification of tobacco leaves inoculated with tobacco mosaic virus based on hyperspectral imaging technique. In: 2016 ASABE International Meeting, 1

Zou X, Zhao J, Malcolm J, Mel H, Mao H (2010) Variables selection methods in near-infrared spectroscopy. Analytica chimica acta, 667(1–2):14–32

Chapter 5
Crop Scouting and Surrounding Awareness for Specialty Crops

Francisco Rovira-Más and Verónica Saiz-Rubio

5.1 Introduction to Crop Scouting

At the beginning of the nineteenth century, farmers managed limited pieces of land from which one family could survive. Decisions were made according to the knowledge accumulated after spending long days in small fields. The size of the fields allowed for a very precise management of the land. With the advent of mechanization, the land that every farmer could manage grew exponentially, and decisions had to be made through sampling, which often led to unrealistic generalizations. With the introduction of precision agriculture, massive data acquisition provides a way to systematically scouting crops as a tool for enhancing decision-making at management level. The successful application of precision agriculture principles to wine production, for example, leads to an optimized management of the vineyard, especially in everything related to grape harvesting. It is possible to make great wine by dumb luck or recipe, but not consistently; only by measuring key parameters can you make the best possible wine year after year (Cox 1999), using the variability inside the vineyard in your favor. However, methodical crop scouting to support key decisions on objective and precise field data is currently a dream for many producers due to the following main reasons:

- *Monitoring cost*: it is very expensive to pay operators for acquiring field data. Due to these high costs, it can only be done once per year at most, which impedes updating information and assesssing the evolution of crops along the entire growing season. The nitrogen content of the leaves, for example, continuously varies as fertilizer or water is applied.

F. Rovira-Más (✉) · V. Saiz-Rubio
Universitat Politècnica de València, Valencia, Spain
e-mail: frovira@upv.es

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_5

- *Low sampling rate*: since it is not feasible to sample data every meter, measurements are realized once every 400 m^2 (20 m × 20 m) for advanced management of high-value specialty crops, and usually at a much less resolution for the rest, if data is sampled at all. Hence, data sampling rate is so low that conclusions are often biased. For example, the weekly assessment of grape ripening status in 6 weeks prior to harvesting is either biased (low-sampled) or unaffordable.
- *Weight of current handheld devices*: data sampling for hours with a handheld device that may easily weigh several kg becomes a torture for the operators, who in addition have typically to walk under the sun in the summer.
- *Costs of service providers*: although service providers offer data maps, airborne information tends to be of low resolution, and if several measurements are needed to check the evolution of the plants along the season, say every week or 2 weeks, the cost per season may become dissuasive.

The use of noninvasive monitoring devices for agriculture started with remote sensing, where optical sensors to perceive the vegetative activity of plants provided images that by recording the light reflected from the canopy, typically in the red and near-infrared spectra, allowed the calculation of certain vegetative indices such as the NDVI (normalized difference vegetative index). The acquisition of data from satellites, airplanes, or unmanned aerial vehicles presents a number of drawbacks that have discouraged their use in many practical applications in commercial orchards. Among them, it can be mentioned the need of a clear atmosphere for image taking, the low resolution when the aircrafts fly far away from the field, the unresolved legal problems in many countries caused by UAVs that require special licenses, the need for paying every image – something growers are not happy to do – and in general the lack of temporal flexibility to get vital information from the field that end users would like to have access to as soon and as many times as fields need. Aerial images have been convenient to assess vegetation differences – e.g., due to water stress or chlorophyll activity – over large areas of cropland, but result quite inappropriate for the precision needed in modern specialty crops, especially when the canopy is stretched along vertical structures. Due to these disadvantages, the use of proximal sensing has been growing in the last decade and is becoming the de facto management system of the future, through which precision agriculture may be successfully implemented in orchards.

There exist several parameters that have awaken interest among specialty crop producers, such as the nutritional content in the leaves or nitrogen uptake, water status and stress, maturity rate for fruits, disease and pest attack spatial distribution, and the estimation of yield at different periods of time before harvest. Yield estimation, in particular, has been claimed by growers as key piece of information for the optimal management of orchards, groves, and vineyards, as it can significantly fluctuate among seasons and also spatially within the same field and season. Although there are no commercial solutions yet, several approaches have produced promising results. Nuske et al. (2011) applied machine vision techniques to estimate yield in vineyards. The system uses a sideways-facing color camera of resolution 3264 × 2448 (pixels) with halogen lamp lighting to detect and count the number of

berries per vine at 0.8 Hz, based on shape and texture, with a satisfactory detection rate, even for green grapes over green leaves. To reduce the presence of false positives (mistaken berries that were mostly isolated), as grape berries naturally grow in clusters, berries that do not have at least five other berries within their immediate neighborhood were eliminated. An average error of 10 % and a correlation coefficient of determination of 0.74 were reported when predicting the actual harvest weight with this experimental setting. As important as with vineyards is crop yield estimation for apple orchard management, mainly due to the fact that current manual sampling-based yield estimation is time-consuming, labor-intensive, and quite inaccurate. Optimizing fruit thinning and harvesting efforts as well as packing house strategies can reduce operation costs. To face this challenge, Wang et al. (2013) deployed a computer vision system that uses a stereoscopic camera with 28 cm baseline between two lenses of 11 mm focal length and artificial lighting (ring flashes) for yield estimation (number of apples) during the night. The system works well with both red (Red Delicious) and green (Granny Smith) apples in a tall spindle planting structure in such a way that apples are approximately 2 m from the stereo rig. The implemented algorithm uses the color space HSV (hue-saturation-value) to discriminate the apples from the background, even green apples from foliage, given the robustness of this model to decouple the light intensity from the color information in the image. For the red apple rows that received conventional fruit thinning, the computer was very close to ground truth, with errors around 3 %. However, apples were undercounted by 41 % for the trees without fruit thinning. Similarly, undercounts for the green apples, which have thicker foliage and more occlusions, reached 30 %. In order to account for occlusions, a finer calibration was added to the estimation model, lowering errors below 2 %. At the end, the orchard manager received a 2D yield map indicating the spatial distribution of red apples for the upcoming harvest.

It is no random coincidence the fact that both yield prediction cases exposed in Nuske et al. (2011) and Wang et al. (2013) together with the use case example for water stress monitoring of Sect. X require supporting structures that enhance canopy and fruit exposure. When tractors coexisted with animals to farm orchards, it soon became apparent that old orchards had been outlined for horses, and therefore, many tractors did not fit between rows, encountering serious problems to maneuver at the end of the rows in narrow headlands. As tractors eventually replaced animals, orchards were delineated with wider row spacing and larger headlands. This is an excellent example of how farming practices changed to better adapt to technology; it made no sense to design tractors with the size of horses to fit well in the traditional orchards. Several decades have passed and we are again facing a similar dilemma. Orchard automation, scouting, and site-specific approaches are not fully efficient with the way many fields are set today. Fruit exposure highly depends on the way trees are shaped. A critical barrier to yield prediction, and robotic harvesting of fruits in trees, has been the high percentage of fruits occluded by branches and leaves, which puts fruits out of reach of end-effectors commanded by computerized vision systems. However, new production structures are being incorporated for high-value specialty crops, some of them already implemented to facilitate

mechanized operations. Such is the case of vineyards for wine production. The traditional goblet or head-trained vines are steadily being replaced by more modern vertical shoot position (VSP) trellis, where most of the bunches are openly exposed in either side of the vine. The percentage of fruit exposure may be further enhanced by field operations such as canopy defoliation. Figure 5.1 illustrates the differences between these two production systems for wine grapes; Fig. 5.1a shows a traditional goblet vine, whereas Fig. 5.1b represents the trellis system used for VSP architectures. As evidenced in the images, VSP is far more convenient for automation and data acquisition from a moving vehicle.

Although wine-producing vineyards were pioneers in the adoption of vertical and flat supporting structures for specialty crops, other crops have been gradually introducing trellises to convert spherical canopies into vegetation walls. This kind of high-exposure architectures is favorable to automation because they are simple, narrow, accessible, and productive (SNAP). Figure 5.2a depicts the case for apples, Fig. 5.2b provides a production alternative for cherries, Fig. 5.2c shows a vertical structure for almonds, and Fig. 5.2d brings the innovative high-density plantations proposed for olive trees. Unlike orchards, row crops and short plants are easier to automate and monitor, as agricultural equipment is not tightly constrained inside rigid structures and GPS signal is less prone to be blocked.

Given that proximal sensing requires keeping a limited distance between the measuring sensor and the target, typically between 1 and 3 meters, a ground vehicle is necessary to carry the data acquisition equipment and scan the field in a reasonable time while assuring a sampling rate with statistical significance. This excludes manual sampling and therefore favors automation and the introduction of intelligent off-road equipment in the field. One possibility is represented by conventional equipment – tractors, harvesters, and sprayers – endowed with the necessary suite of sensors and computers to automatically save crop data. Another option is to deploy a scouting robot to autonomously register large amounts of data, as described in detail in Sect. 5.2. This alternative has been gaining momentum over the last years, as a consequence of the rapid expansion of robotics and big data applications.

Fig. 5.1 Vineyard architectures: (**a**) head-trained vines or goblet (France); (**b**) vertical shoot position or VSP (Portugal)

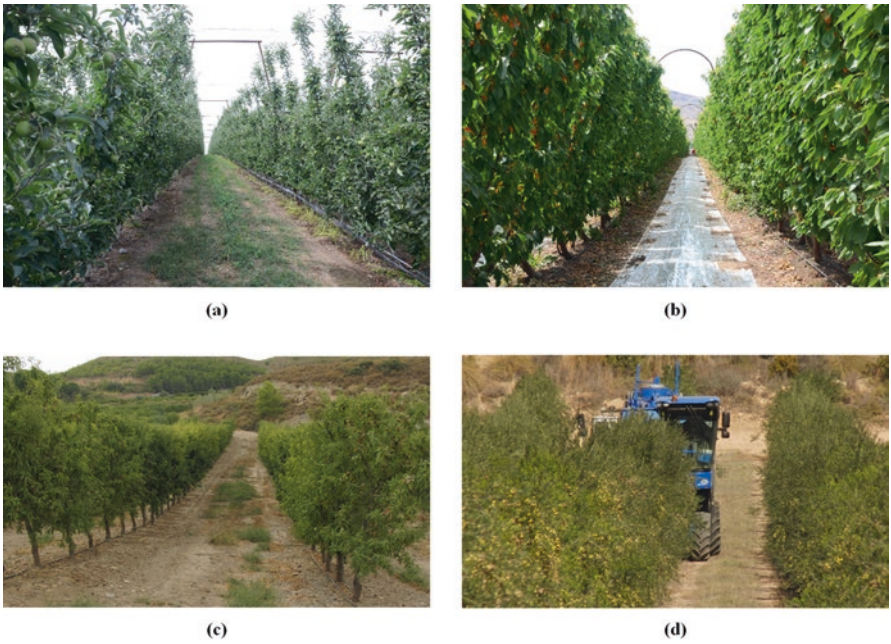


Fig. 5.2 Vertical supporting systems for specialty crops: (**a**) apples (USA); (**b**) cherries (USA); (**c**) almonds (Spain); (**d**) olive trees being harvested with a modified grape harvester. (Spain, courtesy of prof. Antonio Torregrosa)

Application	2016	2017	2018	2025
Field Farming	238.6	281.6	332.5	1,077.8
Dairy Management	146.8	176.3	211.7	766.3
Indoor Farming	79.5	92.7	108.0	318.3
Horticulture	42.8	50.1	58.7	178.9
Others	104.0	120.4	139.5	390.7
Total	611.8	721.0	850.4	2,732.0

Fig. 5.3 US agricultural robots market by application (Verified Market Intelligence 2018)

The amount of data retrieved from commercial orchards is not massive and pervasive, generally speaking, and as a result, we are not into big data yet, but having access to a high-rate data-collecting machine will likely lead the way to it. A great number of financial publications coincide in forecasting a rapid growth of the service robots market, with a significant role of agricultural applications for the next two decades. Figure 5.3 shows the expected growth of the US agricultural robots market by application for 2015 (USD million) (Verified Market Intelligence 2018). Although milking robots and dairy management will still dominate the deployment

of robotic solutions in the next decade, the introduction of digital farming through advanced management of soil and crops with precision agriculture techniques, in combination with a stronger presence of automation and intelligent systems in field farming, will result in the emergence of a growing group of first adopters who will manage their land at the same technological level reached by other industrial sectors. This transformation for the twenty-first-century farmer from laborer to digital age manager may be instrumental to attract young generations to agricultural production, counterweighting the risks of leaving such a high responsibility as feeding the world to an ever-aging population of farmers.

5.2 Scouting Robots: Architecture and Design

There are two practical approaches to endow agricultural machinery with intelligence and automation: (a) modify conventional equipment by introducing digital technology and automatic controls and (b) design a new vehicle that integrates a mechatronics design from scratch. At the dawn of digital technology, the former type was mostly represented by large vehicles powered by diesel engines, while the latter consisted of small- or medium-sized platforms run by electrical drives and traditional lead batteries. At present, however, solutions are not so clearly divided, and large equipment manufacturers are already offering big tractors with a good dose of electrification, whereas innovative farm robots may incorporate small combustion engines to fulfill high-demand field tasks, such as blast spraying or mowing, in platforms propelled by lithium batteries, which have increased efficiency significantly thanks to research conducted by the automotive industry. The convenient size of a farm vehicle endowed with certain degree of automation is still an unresolved question. Productivity and liability are antagonist concepts that need to be balanced until finding a reasonable and cost-effective trade-off. Large machines such as combine harvesters are very efficient because they can harvest big amounts of product per day, but at the same time, they are hard to automate because of their size, weight, and power; if a navigation or control problem arises, it is very difficult to stop a machine of such magnitude. Small robots, or moderate-sized autonomous assistance platforms, on the contrary, entail lighter risks when operating in the field, but, unfortunately, are limited in their capacity to produce work because of small operating widths and low power autonomy. Better batteries and safer architectures will probably lead to larger off-road vehicles propelled by electricity on one hand, and farm machinery more modest in size but with higher levels of automation on the other.

When conventional agricultural equipment is modified to adopt digital farming techniques, it retains a highly effective mechanical design inherited from many years of continuous evolution. However, when a new farm robot is developed, there is a risk of overlooking its mechanical design in favor of software, sensors, or electronics. A vehicle built to operate in off-road conditions must include a strong chassis, a responsive suspension, and an effective steering mechanism; without them, its

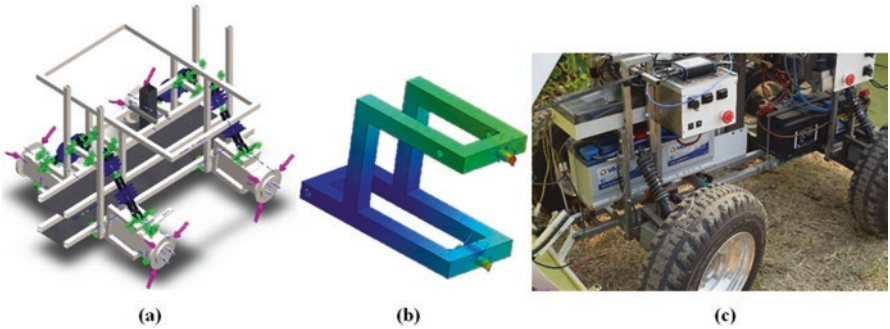


Fig. 5.4 Mechanical design in agricultural robots: (**a**) 3D model for robot chassis (courtesy of Andrés Cuenca); (**b**) high stress joints in motor carriers; (**c**) independent suspension in actual robot

behavior will likely become inconsistent and unpredictable with time. Figure 5.4a depicts the computer-based design of the entire chassis for a vineyard robot, Fig. 5.4b provides a close-up evaluation of highly stressed and highly strained joints in its chassis (wheel motor carriers), and Fig. 5.4c depicts the four-wheel independent suspension constructed for the designed robot. We should evaluate the dynamic performance of agricultural robots as systematically as any conventional off-road vehicle, taking into consideration such crucial parameters as slippage, weight transfer, and traction. The size and type of the wheels, or the potential need of ballast, may improve field performance significantly.

The dynamic performance of a farming robot, which mostly moves on off-road terrains, is crucial for the safety of the vehicle and the quality of the data collected on the fly. Its dimensions and weight distribution over the wheel axles are important design parameters, but the critical point for reaching a stable behavior is the optimal design of suspension and steering. The prototype of Fig. 5.4, for instance, was equipped with four custom-made coil springs with an elastic constant of 15,675 $N \cdot m^{-1}$, with the purpose of improving the results of a previous prototype featuring commercial suspension springs with an elastic constant of 27,210 $N \cdot m^{-1}$. In order to compare the dynamic performance of both prototypes (Saiz-Rubio & Rovira-Más 2016), a three-axis accelerometer was mounted on the front right wheel of each prototype to record vertical accelerations at a frequency of 16 Hz. The robots were set to overcome two obstacles at approximately 1 $km \cdot h^{-1}$: first, a round curb of approximately 4 cm in height and 8 cm wide causing a sudden drop and, second, an office footrest forming a wedge of 45 cm in length and 16 cm in height (slope angle of 21°). Figure 5.5 shows the outcome of this comparison. As expected, the softer suspension of the second prototype (Fig. 5.4c) absorbs better the vertical accelerations provoked by the obstacles, especially with the curb.

The study of the most favorable suspension system for each particular robotic platform needs to be complemented by its corresponding design of the steering mechanism. When robots are endowed with autonomous navigation, this stage in the design process becomes fundamental. An advantageous steering system must provide crisp and accurate corrections when the robots advances in straight

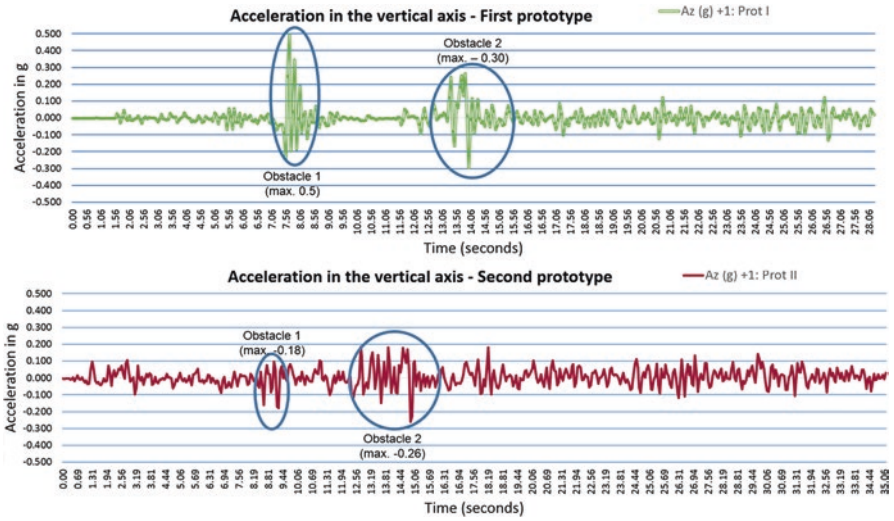


Fig. 5.5 Suspension test for two prototypes of agricultural robots

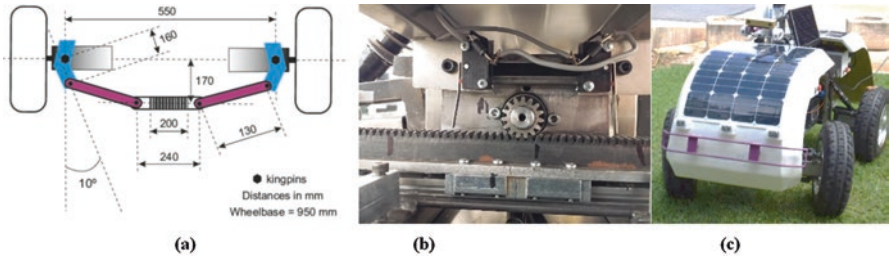


Fig. 5.6 Steering design in scouting robots: (**a**) tie-rod assembly; (**b**) pinion-rack actuation; (**c**) caster angle

guidance between orchard or vineyard rows, but at the same time must allow sharp turns at the headlands to change rows with the minimum slippage. Figure 5.6 gives an overview of the key design components that form the steering mechanisms of scouting robots. In this particular case, as the robot is four-wheel-drive two-wheel-steer, the electric motors powering the front wheels must turn with them as the robot steers, which complicates the outline of the steering linkages. The model of Fig. 5.6a shows an example of a tie rod assembly that facilitates wheel turns with electric motors attached to each wheel. The actual rotation of the wheels is accomplished by displacing a sliding bar actuated by the pinion-rack coupling of Fig. 5.6b. In general, a *significant change in the suspension system implies a reformulation of the steering linkage design*. The steering system for the prototype of Fig. 5.4 was also improved by introducing slight caster and camber angles in the front wheels. A caster angle induces an inclination angle at maximum turns that facilitates the negotiation of sharp turns in a similar way automobiles and tractors steer (Fig. 5.6c). The

light positive camber angle helps the robot self-center, which is key for automatic steering along the rows.

The benefits of optimizing the steering system of a scouting robot must necessarily translate to the actual field, where the modified robot will have to outperform the navigation aptitudes shown by the robot before the improvements. A comparison similar to that of Fig. 5.5, but focused on automated guidance, is plotted in Fig. 5.7. The robot of Figs. 5.4 and 5.6c was automatically guided in Spain, whereas a previous prototype with a simpler steering mechanism was tested in France. Both vineyards had a row spacing of 2 m, and the forward speed oscillated between 1 km·h^{-1} and 1.8 km·h^{-1}. The plot graphs the Ackerman angle registered by the robot's central computer and clearly reveals a superior behavior for the enhanced steering system (blue line), which behaved more stable than the old prototype (red line), that additionally showed some asymmetry in the actuation of the wheels. Experience taught that slight deviations in the construction of the tie rod assembly had significant consequences in the navigation performance of autonomous vehicles. But the opposite was true too; a committed *design* of the *suspension* and *steering* systems *as a whole* resulted in gratifying outcomes.

The deployment of automated systems, even in the case of low-level automation, always involves solving important technical difficulties. Out of all of them, the biggest barrier to commercialization has typically been reliability and safety. There is a long way between a working prototype and a product ready for the market. As a result, it takes years for a solution to hit the market and become accessible to average end users. A safe solution must be reliable in both hardware and software. The environmental conditions found in agriculture – low temperatures in the winter, high humidity, strong sun radiation, application of chemicals, and potential knocks – are usually a threat to delicate off-the-shelf electronics, so a first step in making scouting machines stronger is by protecting sensors, electronic cards, and wires

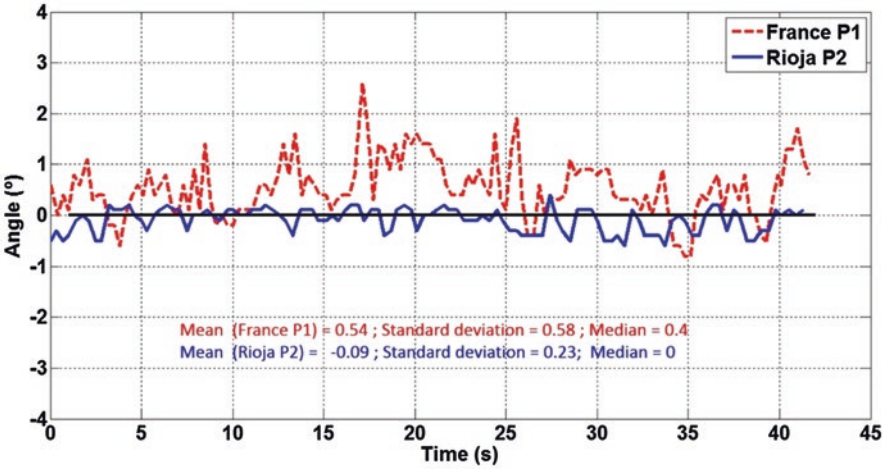


Fig. 5.7 Steering performance comparison between two robotic prototypes automatically guided

from weather aggressions, in addition to using components with an IP rate of 65 or above. Once hardware is protected and secure, communication among sensors, actuators, and computers or processors has to be granted. Furthermore, this communication must be permanent and take place at the necessary frequency to assure a stable and predictable performance. Even though each sensor requires a particular interface, we can enunciate some general rules. Serial data are usually transmitted through universal serial ports (USB) or RS-232. When both protocols are available, the latter is preferable for agricultural applications as it is stronger and can be tightly screwed, providing a sturdy link between sensors and computers. The fact that computers permanently number serial ports is a guarantee of stability before device confusion. Typical sensors that use RS-232 communication are GPS receivers and lidar rangefinders. The increasing popularity of USB ports for video streaming has led to their dominance in imaging sensors over the traditional I2C and firewire (IEEE 1394). However, caution is necessary when using USB-connected devices; first, the connector itself has been designed for desktop computers and, as a result, is weak and easy to unhook under the vibration of off-road equipment; and second, the power supplied through USB ports is limited to 5 V up to 5 A, which many times requires the addition of extra power for power-avid devices.

The transmission between sensors and computers – or processors – must be set at a conservative sample rate, in such a way that if a few measurements are missed, information is still available in a timely manner. However, the control of actuators, especially in navigation and safeguarding, has to be as reliable as possible. For such cases, the commands sent by the processing units must arrive integral and at the right pace. This necessity immediately discards any kind of wireless communication between the central computer and the control of motors and electrohydraulic valves. The most accepted protocol in industry for serial data transmission is the controller area network (CAN) bus, developed by Bosch in the 1980s. It allows controllers, sensors, and actuators be connected on a common serial bus that behaves like a "data highway." Based upon the CAN bus, the manufacturers of agricultural equipment are developing the ISO bus, a protocol to connect farm implements and tractors in a general, safe, and efficient way, with only one control console for all the implements regardless of the brand. This standard is still in progress but many manufacturers already offer it.

There has been a considerable gap between technology development and user adoption with regard to digital farming, and a significant contribution to it comes from the lack of harmony between solutions offered and user needs. Some solutions tend to be very complex and not in tune with current demands made by growers. The average user is not an information technology (IT) expert with an engineering degree; rather, we expect to find practical workers in search of consistent solutions. Intelligent systems must be straightforward and intuitive, easy to use, and easy to explain. In this context, only indispensable sensors and functions should be used to keep complexity at reasonable levels. Adding extra sensors, even if they are low cost, will increase the failure rate of any system. To ease usability, human-machine interaction must be clear and easy, with well-defined graphic user interfaces (GUI) and ergonomic joysticks for manual operations. For autonomous operations, a

well-designed network of emergency stops is mandatory to stop the vehicle in case of unexpected behavior. Figure 5.8a shows the graphic user interface of the robot utilized in the use case of Sect. 5, and Fig. 5.8b illustrates its manual control with an ergonomic joystick. Overall, the skills required to operate these scouting vehicles should be similar to using a cell phone and not much more advanced.

5.3 Surrounding Awareness for Scouting and Data Collection

The principal task of any scouting machine is the perception of its surroundings, and that will only be possible with a combination of sensors holding the capacity of measuring physical properties of what is enclosed in the machine's vicinity. Not only the sensors must have the capacity of perceiving these physical properties in a quantitative or numerical format, but they also must facilitate their recording in a data-logging device for their further processing. Data may be processed in real time for immediate actuation, but the majority of scouting applications require onboard data storage. The development of sensors is in continuous growth, and we will see new devices hitting the market in the following years, but the implementation of some sensors over the last decade has resulted particularly attractive and effective for the complex scenario of agricultural environments, where open off-road terrains that require working outdoors pose challenging situations to the advanced manager willing to apply the principles of precision agriculture and digital farming. This section reviews some of the most popular sensors onboard intelligent vehicles for crop scouting applications, providing a general overview of their workings to show their potential, but without getting into technical details.

Fig. 5.8 User-centered design in agricultural robots: (**a**) intuitive GUI; (**b**) manual control with ergonomic joystick

5.3.1 Visible Light Machine Vision

Machine vision is the computer version of the farmer's sight; the eyes are represented by cameras and the brain by computers. The output of vision systems are digital images. A monocular camera yields digital images of a scene at a given rate (frames per second or fps), and a stereoscopic camera produces a depth image as a result of comparing two images taken simultaneously by two equal cameras mounted on a rigid rig. In this section, we will deal with cameras that perceive the environment in the visible range, which means that the cameras register the same scene seen by humans, filtering out other electromagnetic sources such as ultraviolet or infrared. A digital image consists of little squares called pixels (picture elements), each of which carries information on its level of intensity. If the image is in black and white (technically called monochrome image), the intensity level is in reality a gray level between a minimum value (0) and a maximum value (i_{max}). The number of gray levels depends on the number of bits (binary digits) in which the image has been coded. Most of the images used in agriculture are 8 bits, which means that the image can distinguish 256 gray levels (2^8) and the maximum value is 255 representing pure white. In practical terms, our eyes cannot distinguish so many levels, and 8 bits are many times more than enough. When digital images reproduce a scene in color, pixels carry information of intensity levels for three channels red, green, and blue, leading to RGB images, whose processing is more intricate than monochrome images and therefore falls outside the scope of this chapter.

Monocular cameras constitute the basic component of typical vision systems and retrieve powerful and useful information from plants, trees, and other objects of interest in agricultural fields. In addition to selecting the camera manufacturer and image type (monochrome or color), users must also choose important technical parameters such as the lens focal distance, the size of the sensor, and optical filters when there is a need to block determined spectral ranges (colors). Although the detailed description of these parameters would take too long for an overview like this, the focal distance (f) is related to the scope of scene that fits into the image, and as a result, the smaller value of *f*, the wider angle of view covered by the camera, and therefore a bigger portion of the targeted scene will be captured in the resulting image. Once images are taken, we have completed the first step of the vision process – image acquisition – but the images still have to be properly interpreted to be meaningful, the image processing stage, which involves the delicate step or extracting the useful information from the image, something that must be efficiently carried out by computer algorithms tailored for each given application. Figure 5.9a shows a monocular monochrome camera used to capture zenithal scenes from a vineyard, and Fig. 5.9b reproduces the results of a color-based segmentation algorithm to extract oranges from a citrus grove, application that uses the HSV color space in similar fashion to the apple detection algorithm of Wang et al. (2013).

Even though digital images reproduce scenes with great detail, this representation is flat, that is, in two dimensions, whereas the real scene is actually in three dimensions. What dimension is left then? The depth, or distance between the camera and the scene. In the image shown in Fig. 5.9b, we can locate a given orange

Fig. 5.9 Monocular vision for scouting: (**a**) digital camera; (**b**) color image processed to detect oranges

with precision in the horizontal and vertical axis, but we cannot know how far it is. This information would be essential, for example, if we had to program a robotic arm to retrieve the orange, as it could be out of reach. Stereo cameras allow the acquisition of two (or more) images in a certain relative position to which the principles of stereoscopy can be applied. These principles mimic how human vision works, as the images captured by our eyes in the retinas are slightly offset – as are our eyes – and this offset, known as disparity, is what allows the brain estimate the depth. In agriculture, stereovision is used for assisting in autonomous navigation (Rovira-Más et al. 2015) and for generating three-dimensional (3D) maps of crops (Rovira-Más et al. 2008), which allows the estimation of plant growth and canopy vigor. Figure 5.10a shows a compact stereo camera mounted on a rice harvester, and Figs. 5.10b, c and d, depict a commercial vineyard monitored with a stereo camera mounted at the front of a tractor, its depth image (c) where darker pixels indicate farther distances, and its 3D virtual representation (frontal view d) that depicts the canopy at both sides of the vehicle and the row spacing ahead showing that is free for navigation. Notice that stereo matching is a pixel-by-pixel operation that requires texture to identify the same objects in both images of the stereo pair. Therefore, image areas without enough texture (dark shadows under the canopy or uniformly lit patches of soil) produce (white) pixels with no information in the disparity image (c), eventually creating voids in the 3D representation of the scene. These voids are easily identifiable within the vineyard canopy of Fig. 5.10d.

5.3.2 Biosensing for Crop Monitoring

With the development of non-invasive crop sensing tools that can operate on the fly, i. e., while a vehicle is moving in the field, the availability of crop information has grown significantly. The accessibility to massive field data is key to apply the principles of precision agriculture, a modern approach to grant economical and

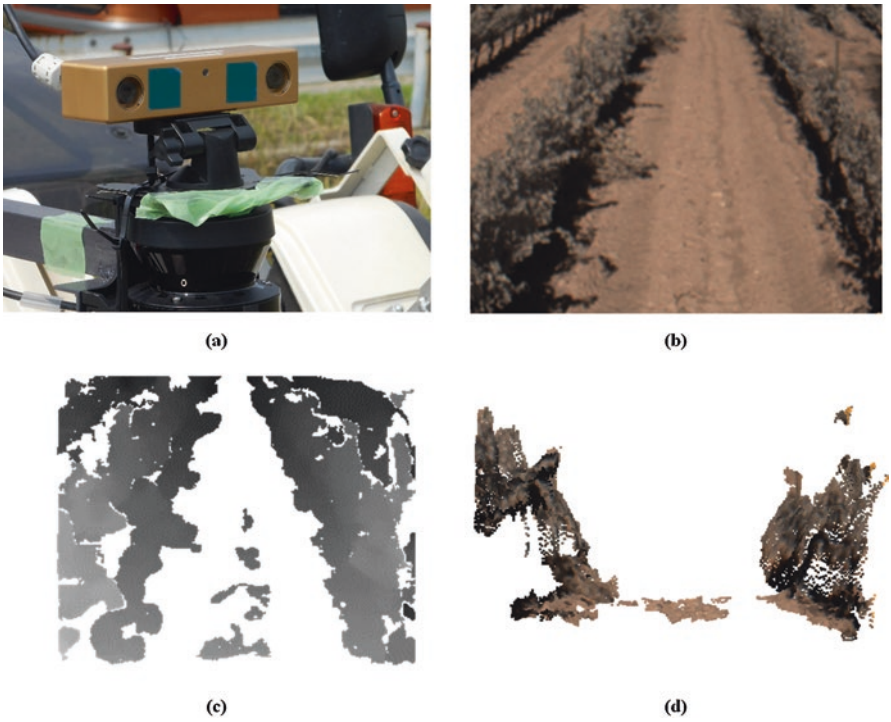


Fig. 5.10 Stereo vision: (**a**) binocular camera; (**b**) real scene; (**c**) depth or disparity image; (**d**) 3D frontal representation

environmental sustainability in farms. Machine vision with cameras sensitive to non-visible spectral bands, infrared radiometers, and other optical devices can produce amounts of data that growers can fit into decision support models to make better choices. Among the crop parameters that have risen more interest in agriculture, it is worth mentioning the canopy volume, the temperature of the leaves measured with infrared radiometers (Sect. 5.5) or, alternatively, with thermographic cameras yielding the CWSI (crop water stress index), and plant vigor assessed through the calculation of the NDVI (normalized difference vegetation index) or the NBI (nitrogen balance index). The use of vegetative indices, such as NDVI, NBI, and CWSI, provides objective and quantitative methods to evaluate the status of crops at any given moment of the productive cycle and, therefore, allows farmers reach a deeper understanding of how crops evolve with time. The fact that many intelligent farm vehicles include a GPS receiver facilitates the acquisition of crop information in map format, which is convenient to compare the spatial variability of key parameters and the generation of historical data series. At present, there exist several off-the-shelf devices to calculate the NDVI in real time and apply a variable rate of fertilizer according to the nutritional status of plants. The systematic record of plant physical properties at high resolution is known as plant phenotyping and is

becoming instrumental to understand the relationship between the genetic makeup of a plant and its complex traits such as growth or water stress resistance. Figure 5.11a shows a multispectral camera that allows the calculation of NDVI and NBI, and Fig. 5.11b depicts an infrared radiometer that can estimate the canopy temperature of specialty crops.

5.3.3 Nonvisual Range Perception

Ultrasonic rangefinders are devices that use sound to measure distances, and for that reason, they are also known as sonar sensors. The principle that lays behind sonars is the fact that the speed of sound is known (343 $m \cdot s^{-1}$ at 20 °C), and measuring the time that the wave needs to hit an obstacle and return – the echo – allows the estimation of that distance. The speed of sound in the air depends on the ambient temperature, but this relationship is known and easy to apply. However, the use of sonar sensors in agricultural fields is subjected to some difficulties that can be detrimental for the accuracy and consistency of measurements. The sound wave sent by the sensor has to hit an object and return to the sensor receiver that must capture it to measure the time used by the wave in its round trip. The physical properties of the target objects, especially their reflective nature, are essential to obtain reliable results. Sound waves do not behave as linear beams; instead, they propagate in irregular cones that expand in coverage with distance. As a result, when the target object is uneven (as canopies) and the cone is not narrow, the measurements may become too noisy. An important design feature to consider is the distance between adjacent

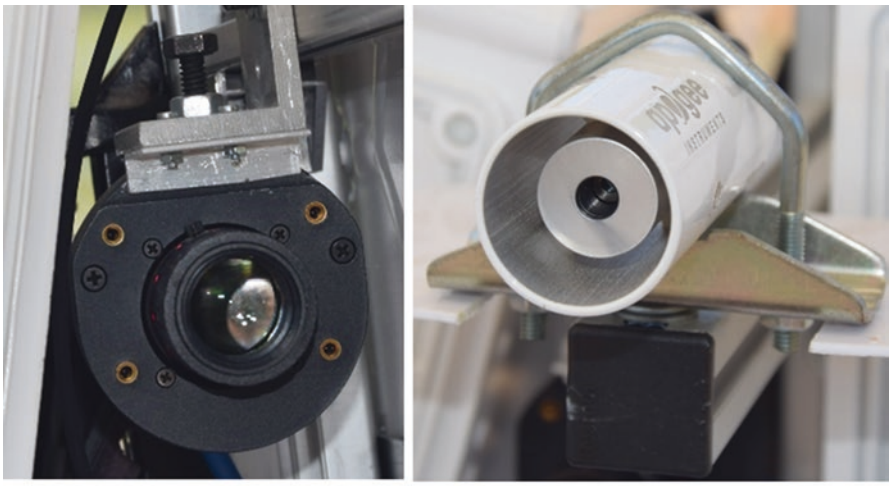


Fig. 5.11 Sensors for crop monitoring: (**a**) multispectral camera; (**b**) infrared radiometer

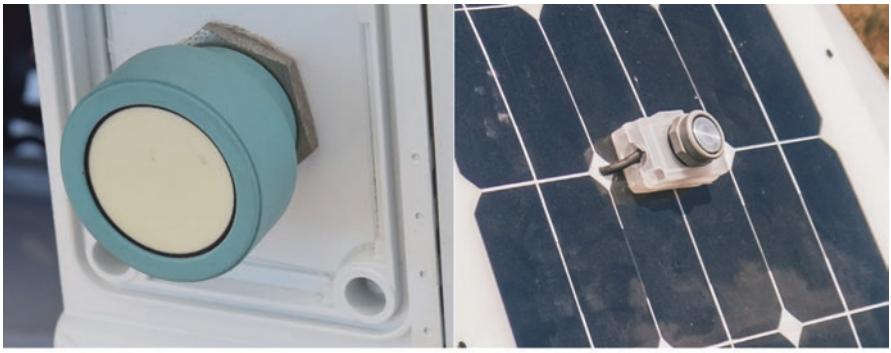


Fig. 5.12 Ultrasonic sensors for estimating distances in agricultural vehicles

ultrasonic sensors, as echo interference is a source of unstable behavior. Overall, sonar rangefinders are helpful sensors to estimate short distances, say up to 2 m, cost-efficiently, and when accuracy and reliability are not essential, for instance, when detecting the distance of a vigor-estimating sensor to the canopy for fine-tuning measurements. Figure 5.12 shows a robust ultrasonic sensor to estimate the lateral distance of a robot to the canopy of grape vines (a) and a lower cost sonar to issue warnings when the robot moves backward and objects interfere in its trajectory (b).

Lidar stands for *light detection and ranging* and is an optical device that provides distances to objects. Although different light sources can be used to estimate ranges, most of the lidars use laser pulses because the beam density and coherency make them very accurate. However, laser beams are very narrow and one static emitter cannot cover the amplitude of zones typically needed. As a result, off-the-shelf rangefinders come in two working modes: (1) a rotating head with an emitter that sweeps an area that can reach 270° and (2) a stationary head with several emitters equally spaced to cover an area, usually smaller than that spanned by rotating heads. Lidars are not affected by sunlight unless it hits the emitter directly and work excellently at night when most vision systems provide poor perceptive capabilities. *Machine vision and lidar are technologies that complement well each other* to provide local perception to intelligent vehicles. Figure 5.13 shows a perception unit comprising a stereovision camera, two sonar sensors on the sides for lateral perception, and a frontal lidar rangefinder with a motionless head consisting of 11 emitters that cover an angle of 88°, with beams angularly spaced by 8.8°.

5.4 Crop Monitoring and Mapping

Regardless of the level of intelligence embedded in a farm machine, field work has to be carried out in the most efficient way. In fact, automation and control is introduced to enhance performance of machinery and therefore increase efficiency,

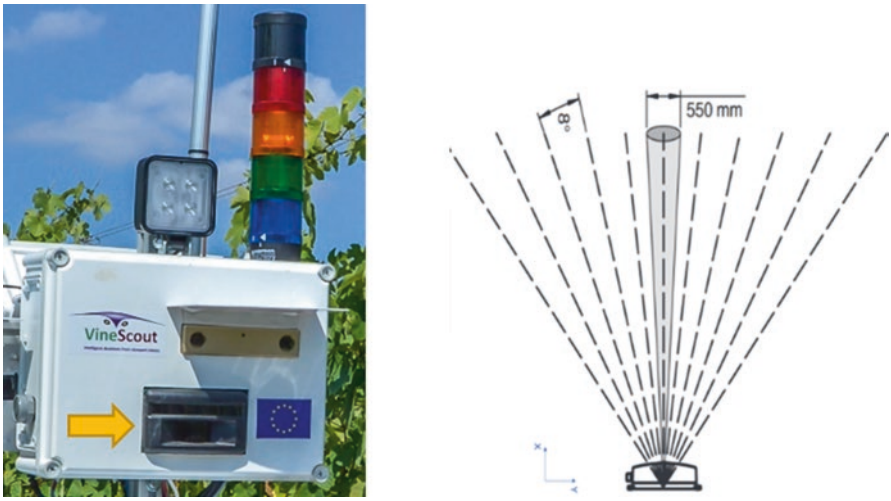


Fig. 5.13 Lidar rangefinder (black component) with 11 beams covering a horizontal field of view of 88°

safety, and comfort for the operators. In a general formulation of crop scouting, a vehicle operating in the field can be ground-based or airborne. In either case, it will have to cover an area in a given time, and this is actually a relevant parameter when comparing different options to do a given task, i.e., the number of ha·h^{-1} that the machine can cover. Let us put the example of a sprayer equipped with a multispectral camera to monitor a vineyard NDVI with a row spacing of 2 m. The sprayer is supposed to pass along every row, with an average speed of 5 km·h^{-1}. The time invested in headland turns for this example has been estimated in 20 % of the total operation. At that pace, the sprayer advances 1.4 m every second, and considering the regular row spacing of 2 m, the area covered is 2.8 m^2·s^{-1}, or 10,080 m^2 per hour, which can be approximated to 1 ha·h^{-1}. Taking into account that a working day is equivalent to 8 h, and 20 % of the time is lost in headland maneuvers, the final efficiency will be approximately 6.4 ha·day^{-1}. This is the working efficiency we need to compare among alternative solutions.

At present, there is a raising interest in electrical drives and actuators. The automotive industry has pushed the development of high-performance batteries for hybrid or totally electric vehicles. Electric actuators facilitate vehicle control and provide a clean alternative to fossil fuels. This technological move is obviously affecting agricultural equipment, and major manufacturers are already developing innovative vehicles with an ever-growing presence of electrical components. However, performing demanding tasks in off-road conditions requires a considerable amount of energy, and it is critical to estimate the autonomy of electrically powered vehicles. The efficiency of the example mentioned above will hardly be accomplished if the sprayer cannot run for at least 8 h per day. The pervasive advent of unmanned aerial vehicles for data acquisition is greatly dependent on battery performance and payload. The use of renewable sources of energy, while

recommendable, must always be compatible with the actual needs of power and workload encountered in the field to satisfy a minimum degree of efficiency.

Crops can be monitored by simple visual inspection or just by reading a sensor output in a display, but the large amount of data logged by most of current sensors, even if running at low frequencies, makes this approach inefficient. Some applications require a real-time processing of sensor data, for example, variable rate fertilizing with an NDVI-based nitrogen assessment, but the majority of monitoring applications require saving data for their later processing. The storage of data in numerical tables and oversized matrices is suitable for engineering work, but field managers and crop growers find it more intuitive when field data is conveyed in a map format. Therefore, the preferred method to deal with crop data is through two-dimensional maps. The fact that GPS receivers are ubiquitous and easily accessible favors the coupling of crop information with global positioning. However, the standard geodetic coordinates coded in NMEA (National Marine Electronics Association) strings that all GNSS receivers get are not convenient for the precision and detail needed in proximal sensing applications, where the position of individual trees is often necessary. In addition, working in degrees, minutes, and seconds is not intuitive for the daily planning of field managers, who would rather use meters instead of degrees. The local tangent plane (LTP or NED) offers several advantages that make it the ideal coordinate frame for proximal sensing monitoring. It uses Cartesian coordinates *north* and *east*, which are by themselves intuitive for growers to orientate in the field, and also features a local origin chosen by the user in every field. This allows both, the use of Euclidean geometry for the calculation of distances and areas, and having convenient magnitudes for the coordinates, which will normally be expressed in meters. The flat coordinates NED (North-East-Down) provide global references for scouting vehicles and therefore for each crop measurement carried out from them by simply translating each sensor location to the GPS antenna's location.

Choosing an advantageous coordinate system for representing the spatial distribution of field data is a necessary step, but the way crop information is displayed in the user map is crucial. Accurate maps of difficult interpretation are useless for the average field manager. Three features may ruin the usability of monitoring maps: (1) the measurement scale has so many intervals or classes that practical actuation is not easy to figure out; (2) measurements are not coinciding in space, and therefore multiple variables cannot be correlated; and (3) the parameter displayed in the map has no practical meaning or application for the daily operations of the farm, either because variability is too high or the parameter itself is more scientific than operational. The first two drawbacks are counterweighted by discretizing both the space and the measurements. A grid approach that maintains global references in LTP coordinates offers good guarantees for map usability (Rovira-Más 2012) and facilitates the comparison of data through the years. This approach lets users decide the cell size – map resolution – and the number of classes that are manageable for each specific situation, typically represented in a friendly color scale. The temperature maps of Fig. 5.19 are examples of grid representations in the LTP frame. Notice that a scouting robot recording a temperature measurement every second will gather

several data points for each cell of 4 m^2 area, and without computing the average measurement of each cell, comparison and correlation between zones is not practical.

The third drawback mentioned above requires a different sort of solution. A grid map that directly represents raw data acquired in the field will naturally present a lot of variability, mainly with cells of reduced area. It may be the case of having a crop map with the right spatial resolution and with a reasonable interval for the studied parameter, but still yielding a large dispersion among cell values. For such cases, and if no real-time decisions are to be made, geospatial techniques for data smoothing and clustering become very practical to make crop maps operative. When field data is scarce, interpolation has been the typical technique used, but many times it has led to artificial mapping that has little to do with field reality. When massive data is available, however, smoothing and clustering are the most appropriate ways to treat data. In any case, geostatistics in precision agriculture must avoid the common plague of overfitting and overtransforming data (Schueller 2010). The ideal situation will be when a crop map has a few treatment zones for a parameter directly applicable with the equipment available in the farm. To reduce the number of treatment zones, cells that fall within a "close" vicinity must be classified as equal *dose*. The problem is how to determine the radius – known as the *range* – around any given cell above which influence from other cells is negligible, following the accepted principle of Tobler stating that everything is related to everything else, but nearby objects are more related than distant objects (Tobler 1970). Clustering techniques are usually optimal when data follow a Gaussian distribution. The estimation of the maximum distance of influence can be found through the plotting of semivariograms, which consists of computing statistical variances (Y axis) for growing distances (X axis) until the variance remains close to a steady state called the *sill*. The geometrical locus of the variances is then fit to an equation whose intersection with the sill establishes the maximum range, the distance considered for clustering and smoothing data. Figure 5.14a shows the distribution of nitrogen (0 is poor; 1 is maximum) in leaves of a vineyard where 1062 measurements were made noninvasively (Saiz-Rubio et al. 2017). Note the closeness of the actual distribution of nitrogen values with a perfect Gaussian distribution. Figure 5.14b plots its corresponding semivariogram where the best fit was exponential and the maximum range found was 30 m.

Once the range has been set, a raw map plotting data directly retrieved from crop sensors can be further processed to indicate site-specific treatment zones. A multitude of algorithms may be applied, taking into account the range as the radius of influence. A simple way of smoothing data would be by estimating the median in the surrounding cells considering two concentric rings, that is, by finding the median of each cell and its surrounding 24 cells (5 × 5 window) (Saiz-Rubio et al. 2017). Figure 5.15a shows a grid map of nitrogen in leaves directly plotted with sensor data, and Fig. 5.15b plots the result of applying the 25-cell smoothing filter based on the median. The filtered map facilitates decision-making based on zoning and allows a rational use of variable-rate equipment.

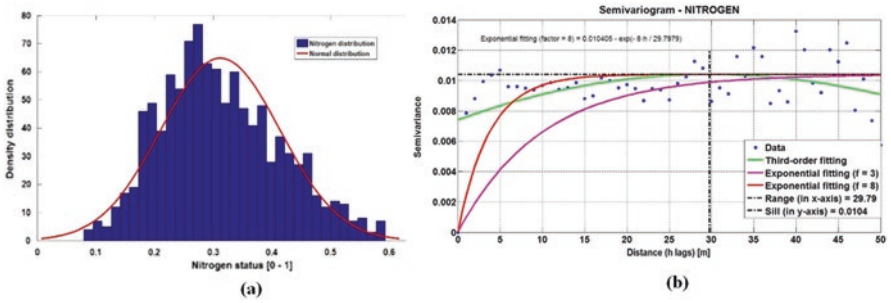


Fig. 5.14 Normal distribution of foliar nitrogen (0–1) in a vineyard (**a**) and semivariogram (**b**)

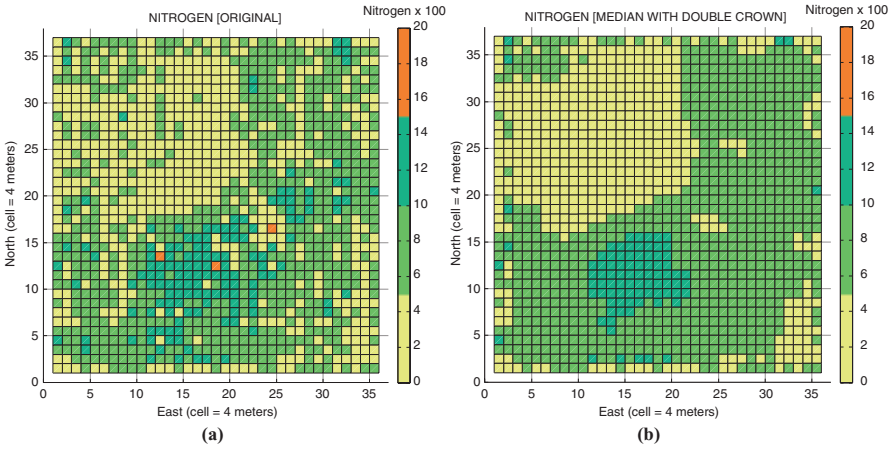


Fig. 5.15 Original grid-based nitrogen map (**a**) and results after applying smoothing techniques (**b**) (Data courtesy of Mª Paz Diago)

5.5 Use Case: Vineyard Scouting with Ground Robots for Water Status Assessment

The Alto Douro Wine Region is a UNESCO World Heritage site centered on the Douro (or Duero) river, which is sheltered by mountains from the coastal influence of the Atlantic Ocean. This region is home of the most important wines elaborated in Portugal and includes both table wine – typically known as Douro wines – and the world-famous Porto wine. Producing wine in the Alto Douro is hard due to steep terrain and high temperatures, and even though wine makers are used to these conditions, and grape vines, most of which indigenous local varieties, are acclimated to the region, the steadily rising temperatures of the last years have posed serious concerns among growers, who need to keep the yield and reputation of their products before an uncertain availability of water for irrigation. An excess of water stress in

Fig. 5.16 Water stress in grape vines: (**a**) scanty canopy; (**b**) lime treatment to decrease leaf temperature

vines may lead to important economic losses and long-standing damage to renowned Porto brands. Figure 5.16a shows the effects of water stress (1 August 2018) in the canopy of *Touriga Nacional* vines, where plant coverage is too poor. A traditional method in the region to palliate the effect of extreme sun radiation has been to spray the vine canopy with lime to lower leaf temperature. Figure 5.16b depicts the deposition of lime on the leaves.

The natural response of a stressed vine is to limit transpiration by closing the leaf stomata, with the purpose of preserving plant water as much as possible when soil water is not available. The predominant method so far to assess the degree of water stress in a plant tissue has been by estimating the water potential of leaves with a Scholander pressure chamber. These pressure bombs are field portable and easy to operate. However, while the measurements yielded are quite reliable, the actual operation is time-consuming and physically demanding. It requires transporting a nitrogen tank along the field to reach the sampled points of measurement. Single leaves must be introduced into the chamber, and for every test, an equilibrium must be reached: as the pressure increases, at some point the liquid contents of the sample will be forced out of the xylem and will be visible at the cut end of the stem or petiole. In order to see the little drop coming out of the stem, a magnifying lens is typically used. In addition to the time needed for a single sample, measurements must be carried out either at midday or before dawn, which are not the best time for being in the field. All these inconveniencies have resulted in very few growers, even in large wineries, assessing water stress with the proper instrumentation, and therefore, there is an urgent claim to develop a new methodology using non-invasive technologies that allow getting more data, more often and less cumbersomely. Figure 5.17 shows the use of the Scholander pressure chamber in a commercial vineyard both at midday (center, right) and before dawn (left).

The effects of closing stomata result in an increase of leaf temperature; therefore, by tracking the thermal dynamics of vine canopies, we can define objective indicators of how stressed plants are. Unfortunately, plant biology involves complex, not fully understood physiological phenomena, and additional parameters must be

Fig. 5.17 Sampling and measurement of plant water stress in a vineyard with the Scholander pressure chamber (Courtesy of Symington Family Estates)

(a) (b)

Fig. 5.18 Robot VineScout to assess vine water stress: (**a**) day mapping; (**b**) night mapping

taken into account to enrich plant monitoring. For example, the sun-exposed side of the rows will evolve differently from the shady side, and the vegetative vigor of each single plant will result in different resistance to stress. The EU-funded research project VineScout (2017-2020) aimed at understanding and estimating water status in vineyards by the massive measurement of canopy temperature and vine vigor. To do so, the robot of Fig. 5.18 was equipped with an infrared radiometer to measure temperature and a multispectral camera (Fig. 5.11a) to track vegetation indices. The onboard GPS receiver assigned a submeter precise location for each point sensed in the field.

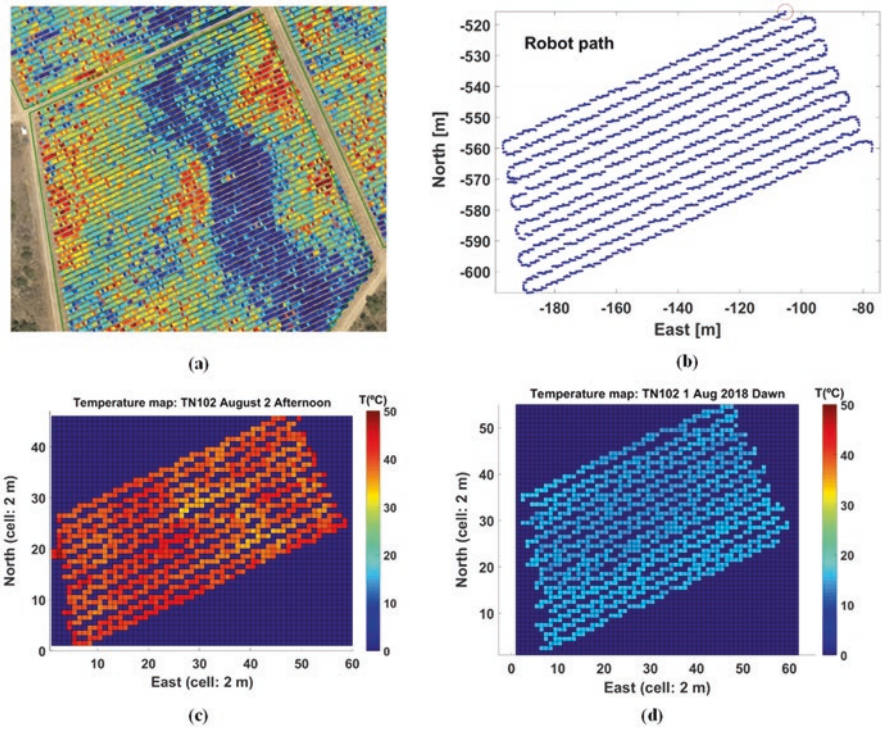


Fig. 5.19 Crop monitoring maps: (**a**) aerial NDVI map of monitored field (Courtesy of Symington F. E.); (**b**) robot trajectory for building the grid map of afternoon temperature; (**c**) grid map of afternoon temperature; (**d**) grid map of dawn temperature

The output of a field monitoring session comes in the format of a map, where a text file stores the parameters measured in the field and their corresponding GPS location. One of the maps generated by the robot of Fig. 5.18 to assess water stress is the canopy temperature, which was measured with an infrared radiometer (Apogee Instruments, Logan, UT, USA) placed in the right side of the robot (Fig. 5.11b) facing the canopy central section at an approximate separation of 50 cm. This map was displayed in real time in the robot's console (Fig. 5.8a) and also saved in a text file for its further processing. Figure 5.19 shows various canopy temperature maps recorded in the Portuguese vineyard of Fig. 5.16 in the summer, where the winery Symington Family Estates established an experimental irrigation scheduling for research. Figure 5.19a is an aerial image of the studied field that highlights differences in vine vigor caused by uneven soil and terrain. Figure 5.19b plots the trajectory of the robot during one of the mapping missions conducted in the vineyard in 2018, when some of the highest temperatures of the summer were registered in the afternoon. The robot only mapped one side of the canopy and scanned every two rows. The temperature map in grid format recorded over the afternoon is shown in Fig. 5.19c. In contrast, Fig. 5.19d represents the temperature map for the same field

the day before at dawn. The map of Fig. 5.19c covers 12 rows and sums up a total of 16,529 points (including the headland turns where no valid measurements were made) acquired in a time frame of 67 min. After removing the headland section where the temperature was not registered, the total number of points was 15,469. The map of Fig. 5.19d comprises 15 rows and 23,012 data points, which were reduced to 18,945 after removing the headlands.

According to Fuchs (1990), radiation, air temperature, humidity, and wind speed modify leaf temperature and may mask indications of water stress. Furthermore, the position, inclination, and orientation of leaves within the canopy also produce considerable variation of leaf temperature. The intricacies behind the knowledge of how much stress a vine is actually supporting in a given period of time lead to complex models and a multiplicity of influencing parameters. However, the majority of field data used so far has been taken remotely and occasionally. The opportunities brought by the automatic monitoring of crops at submeter distances, producing massive amounts of data at the grower request, open a new paradigm for understanding crop growth and fruit production. The temperature maps of Fig. 5.19 were taken in August and involve 34,414 measurements grabbed from dawn to evening. When comparing these figures with manual sampling, the resolution of robot-generated maps reaches significantly higher orders of magnitude. Maps 5.19c and d involve an area of 4800 m^2 ($\approx$ 0.5 ha), considering an approximate row length of 100 m, a row spacing of 2 m, and a mapping frequency of one pass every two rows. The resolution achieved is 3.2 measurements per m^2. The same rationale can be applied to the map of Fig. 5.19d, yielding a similar sampling resolution. The data represented in the maps are the straight measurements carried out by the infrared radiometer, and no filtering has been introduced yet. However, some patterns and preliminary conclusions may be extracted from these results. The distribution of temperatures graphed in Fig. 5.19c reveals that many locations in the 15 rows sampled reached temperatures above 40 °C, with an uneven distribution of temperatures that resembles the S shape of Fig. 5.19a (dark blue in Fig. 5.19a). The map recorded at dawn, from 5:27 am to 6:55 am, produced a more homogeneous result, and more data is required for a deeper understanding of thermal dynamics along the day. Fortunately, the specific measurements recorded by the robot at every location are available for the user, and can be easily plotted for the entire battery of tests. Figure 5.20 shows the precise temperatures acquired for the maps of Fig. 5.19c and d.

Every row in the field of Fig. 5.19 has an approximate length of 100 m and presents a basin-like lower section toward the center of the row, where plants are more vigorous. The extremes of the rows (headlands) are higher and plant canopies significantly thinner. The robot velocity oscillated between 1.2 $km \cdot h^{-1}$ and 1.7 $km \cdot h^{-1}$ according to the motion being upslope or downslope. Overall, each run took between 4 and 5 min. Figure 5.20 confirms that there is much higher dispersion for the temperatures measured during the afternoon, but also reveals a pattern in the chilly temperatures at dawn; at the central part of each row, where vines have more foliage and the terrain profile forms a mild basin, registered temperatures were typically several degrees lower. As time progressed over the morning test, the average

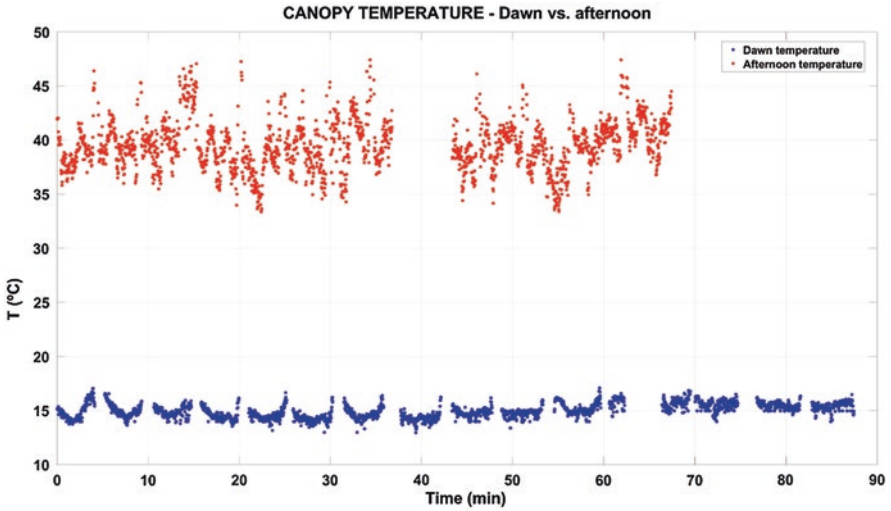


Fig. 5.20 Specific temperatures acquired by the robot in the maps of Fig. 5.19c and d

temperature slowly increased, and after 1 h, the differences between headlands and row centers practically disappeared. The thermal behavior of the vineyard over time and location was transferred to the vineyard manager, who will add this information to other field data for a better decision-making in the management of this plot.

5.6 Summary and Concluding Thoughts

Feeding an ever-growing population is becoming a big challenge for the agricultural engineers of the future. Fortunately, disruptive technologies tend to appear to face up such challenging scenarios, and digital farming has brought powerful tools to do so, such as precision agriculture and robotics. Making the right managerial decisions is key for the sustainable production of specialty crops, and to make good decisions, reliable data is essential. This chapter has explained the basic conditions for crop scouting and the principles for designing monitoring vehicles, paying especial attention to the workings of fundamental sensors, and how to display their information in legible maps. However, although results are positive and technology is more accessible every day, systematic field monitoring is still scarce. Many high-interest crop properties such as maturity and yield have no commercial device to monitor noninvasively from moving vehicles. In some cases, measuring sensors are available, but their practical implementation or the data that they generate are too cumbersome for regular applications at user level. Nevertheless, digital farming has come to stay and these difficulties will eventually be overcome with time. Agriculture is not a source of big data yet, but it is a question of time.

References

Cox J (1999) From vines to wines: the complete guide to growing grapes and making your own wine. Storey Publishing: North Adams, MA, USA

Fuchs M (1990) Infrared measurement of canopy temperature and detection of plant water stress. Theor Appl Climatol 42(4):253–261

Nuske S, Achar S, Bates T, Narasimhan S, Singh S (2011, September) Yield estimation in vineyards by visual grape detection. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 2352–2358. IEEE

Rovira-Más F (2012) Global-referenced navigation grids for off-road vehicles and environments. Robot Auton Syst 60(2):278–287

Rovira-Más F, Zhang Q, Reid JF (2008) Stereo vision three-dimensional terrain maps for precision agriculture. Comput Electron Agric 60(2):131–143

Rovira-Más F, Millot C, Sáiz-Rubio V (2015) Navigation strategies for a vineyard robot. In: 2015 ASABE annual international meeting, p 1. American Society of Agricultural and Biological Engineers

Saiz-Rubio V, Rovira-Mas F (2016) Preliminary approach for real-time mapping of vineyards from an autonomous ground robot. In: 2016 ASABE annual international meeting, p 1. American Society of Agricultural and Biological Engineers

Saiz-Rubio V, Rovira-Más F, Millot C (2017) Performance improvement of a vineyard robot through its mechanical design. In: 2017 ASABE Annual international meeting, p 1. American Society of Agricultural and Biological Engineers

Schueller JK (2010) Geostatistics and precision agriculture: a way forward. In: Geostatistical applications for precision agriculture. Springer, Dordrecht, pp 305–312

Tobler WR (1970) A computer movie simulating urban growth in the Detroit region. Econ Geogr 46(sup1):234–240

Verified Market Intelligence (VMI). (2018). Global agricultural robots: market size, status and forecast to 2025. Boonton, NJ, EEUU.

Wang Q, Nuske S, Bergerman M, Singh S (2013) Automated crop yield estimation for apple orchards. In: Experimental robotics. Springer, Heidelberg, pp 745–758

Chapter 6
Crop Sensing and Its Application in Precision Agriculture and Crop Phenotyping

Geng Bai and Yufeng Ge

6.1 Introduction

The need to enhance agricultural productivity and resilience in the face of increasing world population and changing climate has called enormous interests from both academics and industry. The median values of global population prediction are 8.1 billion, 9.6 billion, and 10.9 billion in the years 2025, 2050, and 2100, respectively (UN 2013). With a population of 7.8 billion in 2020, continuously increasing the crop yield is always a practical but challenging way to feed the increasing population. The consumption of food and other crop-related production per person per year will also keep increasing with the progress of the local economic development, especially in developing countries where the consumption rate is substantially lower than that in developed ones (Alexandratos and Bruinsma 2012). The negative influence of climate change on crop production systems has drawn substantial concern (Olesen et al. 2011).

Crop production is the largest sector of fresh water usage, which uses around 70% of total fresh water withdrawn (Morison et al. 2008). In addition, excessive chemical application in crop production can lead to substantial environmental and economic cost (Pimentel and Burgess 2014). Thus, breeding new crop cultivars with better adaptation to the climate and developing transformative technologies for sustainable field management are two promising ways to boost the crop yield while minimizing the negative impact of crop production on the environment.

Disclaimer: Mention of a commercial product is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated.

G. Bai · Y. Ge (✉)
University of Nebraska Lincoln, Lincoln, NE, USA
e-mail: yge2@unl.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_6

Sensing of morphological, physiological, and biochemical properties of crops plays an increasingly important role in agriculture. In the research domain, plant sensors enable us to gather a large volume of information across multiple scales on crop performance to elucidate relationships between plant genetics, environmental stresses, and management practices (also known as G × E × M). In production settings, sensors generate valuable information used by producers for decisions in seeding, fertilization, irrigation, and chemical applications. Satellite remote sensing and grain yield monitors are examples of early crop sensing. More recently, crop sensing research and applications have seen a trend of rapid expansion, which is likely due to the confluence of the following two factors. The wide availability of low-cost sensors, coupled with a substantial increase in computational power, has created flexible and cost-effective solutions in agriculture to generate, curate, model, and share very large-scale sensor data.

In the main body of this chapter, we review the major spectroscopic, imaging, and ranging sensors; how they are used to measure plants and crops in different environments; and their advantages and limitations. Applications of crop sensing in precision agriculture and plant phenotyping are also introduced. Finally, we end the chapter with a perspective discussion on the open challenges in crop sensing/phenotyping and future directions.

6.2 Crop Sensing Systems and Strategies

6.2.1 Spectroscopy

Reflectance or transmission spectra of crop leaves or canopies in the visible (400–700 nm, VIS), near-infrared (700–1100 nm, NIR), and shortwave infrared (1100–2500 nm, SWIR) regions of the electromagnetic (EM) spectrum (collectively known as VIS-NIR-SWIR) are often used to measure physiological and chemical properties of plants. Portable spectrometers such as those from Analytical Spectral Devices and Ocean Optics are often used. For the leaf-level measurement, a leaf clip with an artificial light source (such as a tungsten-halogen lamp) is commonly used; whereas for the canopy-level measurement, fiber-optical cables are pointed toward the canopy to measure reflected solar energy. Compared to hyperspectral imagers, spectrometers often have significantly lower cost with higher spectral resolution, faster measurement speed, and smaller and lightweight sensors.

Incident radiation, spectral properties of plants, environmental parameters, and the instrumental parameters are the main factors determining the spectral signal of field plants. In most cases of crop sensing research in the field using spectroscopy, the measurement of spectral reflectance is adopted instead of collecting absorbance and transmittance because it is more practical to set up spectrometers above the canopy for data collection. To understand the dominant factors which influence the spectral properties of field plants, it is better to start from studying the spectral

properties of an "ideal" green leaf sample in the lab, whose spectral properties do not change with time, under a stable light source and a controlled environment. In this way, we exclude other extraneous factors except for the spectral properties of the plant leaf.

Figure 6.1 shows an example of spectral reflectance of a maize leaf in the VIS-NIR-SWIR region. Figure 4.3 (Chap. 4) also presents a similar spectral signature for a specific application of detecting grapevine leafroll disease (GLD). In the VIS spectrum, a small bump in the green band (centered around 550 nm) exists because chlorophylls in the leaf cells absorb more strongly in the blue and red regions. Another important feature is the significantly lower reflectance in the VIS region compared to that in the NIR region. In NIR, a significantly higher reflectance is collected by the spectroscopy due to the internal leaf structure and the scattering of light by mesophyll tissues. Photosynthetic pigments, including chlorophylls, carotenoids, xanthophylls, and others, significantly affect the spectral reflectance properties, while chlorophylls a and b contribute the most among them in a healthy green leaf. In the SWIR region, the leaf reflectance curve shows several deep valleys due to the absorption of light by water in the leaf. In addition, many organic components (such as proteins and structural carbohydrates in the cell wall) in the leaf also contribute to the absorption of light energy in the NIR and SWIR regions.

Different vegetation indices (VIs) have been developed from wavelength-wise spectral reflectance to monitor terrestrial vegetation properties using satellite

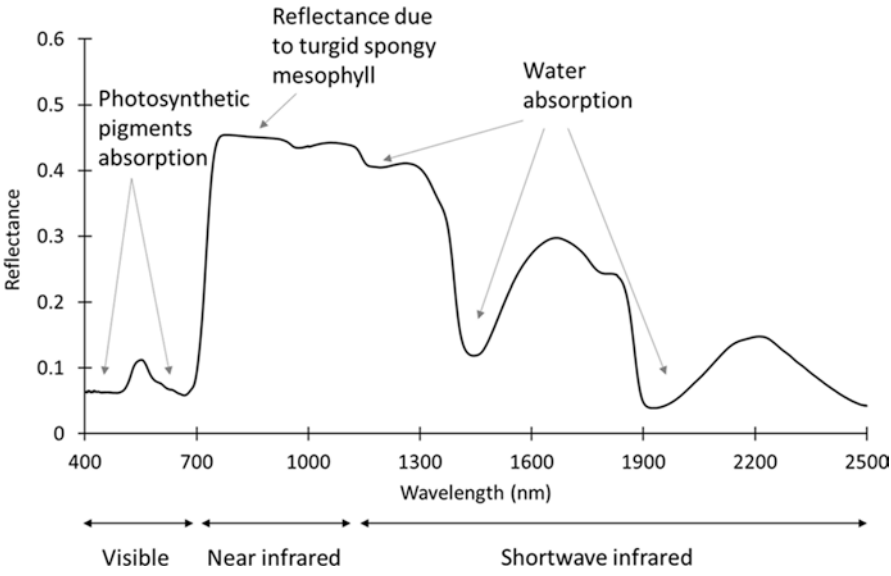


Fig. 6.1 A sample of spectral reflectance of a maize leaf ranging from visible, near-infrared, and shortwave-infrared spectrum (400–2500 nm) measured by an ASD spectrometer. The absorption bands of photosynthetic pigments and water molecular are also generally marked

platforms at seasonal or longer temporal scale. A large number of VIs exist for different objectives, but only a few are introduced as follow. Normalized difference vegetation index (NDVI) utilizes the difference between the low reflectance at the red band and the high reflectance at the NIR band of plant canopy by creating a normalized index ranging from −1 to 1 (Tucker, 1978). Based on NDVI, soil-adjusted vegetation index (SAVI) has been developed to minimize the soil brightness influence (Huete 1988) while enhanced vegetation index (EVI) is more influenced by canopy biophysical parameters and less affected by the atmosphere (Huete et al. 2002). To overcome the saturation issue of NDVI when measuring dense crop canopies, the spectral reflectance at the red-edge region has been used (Mutanga and Skidmore 2004).

A narrow band VI, photochemical reflectance index (PRI) has been brought forward to estimate the diurnal dynamics of the photosynthetic efficiency using the spectral reflectance at 531 nm with the help of a reference wavelength at 570 nm (Gamon et al. 1992). Measuring solar-induced chlorophyll fluorescence (SIF) has been considered as a promising approach to directly monitor photosynthesis activity in recent years (Frankenberg and Berry 2018). SIF is emitted by the plant chlorophylls between the wavelength of 650 nm and 800 nm as one of the energy dissipation pathways during the light reaction of photosynthesis (Meroni et al. 2009). However, no standard pipeline of SIF data collection and retrieval has been established. NIR_v, which is the product of NDVI and the terrestrial reflectance at NIR spectrum, is a new VI with a close relationship with SIF and has the potential to estimate global gross primary production (GPP) accurately (Badgley et al. 2017).

Many other factors need to be considered when studying the reflectance properties of plant canopy in the field. These factors include the difference in canopy structures, variation in incident solar irradiance, plant physiological activities, observation angle of the spectroscopic sensors, soil reflectance, wind disturbance, and so on. One limitation of the spectroscopic sensors (or spectrometers) is that it is a point-measurement instrument which cannot provide pixel-by-pixel information like hyperspectral imaging systems. Thus, the spectral noise from background objects is difficult to exclude.

Spectrometer specifications (e.g., wavelength range, spectral resolution, dark current level, signal to noise ratio, detector-cooling technology, and wavelength shift due to the body temperature of the spectrometer) and calibration protocols (radiometric and spectral calibration) are the main factors that could significantly affect the quality of the raw data. Spectral calibration is necessary to be carried out periodically to make sure the measured data is correctly assigned to each wavelength. Radiometric calibration converts the raw data set from digital count to the energy unit. Reference targets with known reflectance values can be used for reflectance calibration.

6.2.2 RGB Imaging

RGB cameras are the most common imaging sensors for plant measurements. They capture plant images in the visible part of the EM spectrum in three broad bands (Red, Green, and Blue) commonly by CCD detectors. More discussion on the various types of sensors and image processing techniques used for RGB imaging can be found in Chap. 2. A plethora of consumer-grade digital RGB cameras are found commercially that are also low-cost, flexible, and easy to operate. RGB images are generally easier to interpret because they appear similar to how and what human eyes see. In the field, depending on the type of crops being imaged, cameras are mounted on a sensor platform facing generally downward to capture nadir view images of crop canopy (e.g., imaging row crops like corns) or sideways to capture the side view of the canopies (e.g., scanning perennial crops such as fruit trees). Sideways imaging can also be used in the greenhouse applications (Ge et al. 2016). In such cases, single potted plants are generally imaged with cameras mounted on the side and side view images from multiple camera angles can be derived. RGB images are useful to quantify the morphological traits of crops, such as vegetation cover, height, width, biomass, and growth dynamics. This is achieved by segmenting the vegetation (or green) pixels from the image and quantitatively relating the pixel information to the actual traits. Figure 6.2 shows the image examples of different crops in the field and greenhouse conditions.

In the field, canopy height information can be derived from overlapped RGB images taken by mobile imaging systems using Structure from Motion (Díaz-Varela et al. 2015). With multiple-angle RGB images, 3D reconstruction of the plant or

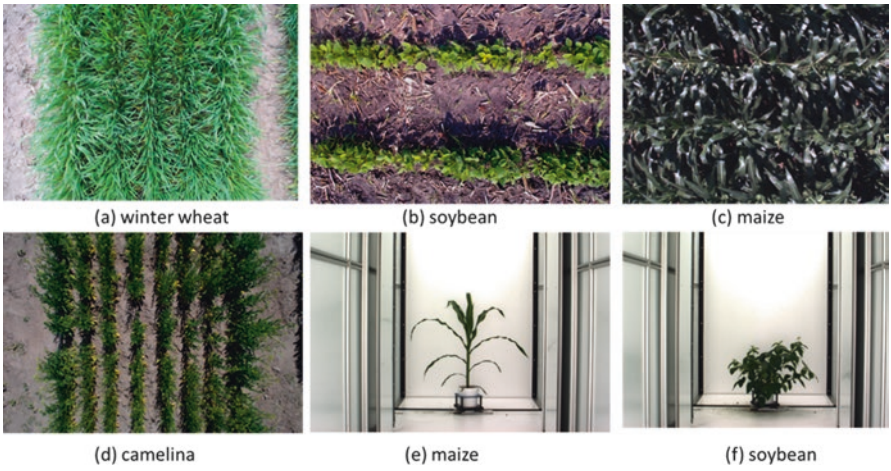


Fig. 6.2 Raw RGB images collected in the field (**a–d**: winter wheat, soybean, maize, and camelina) and greenhouse (e–f: maize and soybean) conditions. Various phenotypic parameters related to the plant/canopy structure and color distribution can be extracted from the raw images through image processing

canopy can be realized to retrieve important traits, including leaf number, leaf angle distribution, leaf area, and so on. Figure 6.3 shows an initial attempt of 3D reconstruction results of a maize plant in the laboratory environment and a sorghum plot in the field condition using RGB images taken from multiple angles. In addition, RGB images can help the plant segmentation work for other imaging systems with lower spatial resolution if they are sharing large area of the camera field of views (FOVs).

Robust algorithms for image processing is usually challenging under varying illumination conditions in a field with high weed density. This challenge is usually much less influential on the crop segmentation in the greenhouse environment where uniform lighting condition might be created for the imaging system. Color calibration might be needed if images are taken using multiple cameras, and color information is critical for quantitative analysis of the scene. A flexible imaging platform with Pan-Tilt capability can generate raw data set for 3D reconstruction of the top canopy. Noise introduced by unwanted crop movement during the image acquisition process due to wind could be minimized by using an image array to capture images simultaneously.

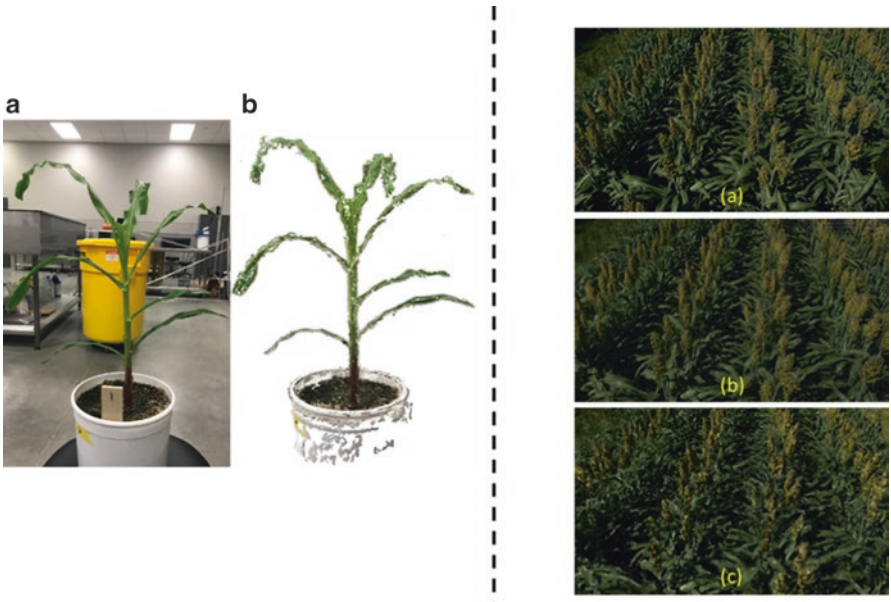


Fig. 6.3 Preliminary result of 3D reconstruction of plant and canopy using multiangle RGB images. Left: individual maize crop under indoor illumination environment with (**a**) an RGB image to show the target maize crop and (**b**) point clouds generated from multiangle imaging *(Courtesy of Suresh Thapa)*. Right: sorghum plot in field condition with (**a**) an RGB image of the target sorghum plot, (**b**) generated point clouds, and (**c**) 3D meshes for further process

6.2.3 *Multispectral Imaging*

As discussed in Chap. 4, multispectral cameras usually cover multiple spectral bands in the VIS and NIR spectral range to capture the distinct difference of the canopy reflectance at visible and near-infrared bands with the sharp increase feature at the red edge spectrum. Reliable result of crop segmentation from soil background could be achieved by leveraging NDVI image generated from the multispectral imager. Other imagers can further utilize the segmentation results through image registration. For example, it can help crop segmentation of the thermal infrared image to retrieval the canopy temperature and soil temperature. Figure 6.4 shows an image-processing example to use multispectral camera for plant segmentation for calculation of different phenotypic parameters (Bai et al. 2019). In addition, specific bands can be used for the estimation of nitrogen deficiency and other important traits based on various VIs (Berni et al. 2009; Svensgaard et al. 2014; Zaman-Allah et al. 2015).

A good strategy or algorithm to avoid under or over exposure of the images at each spectral band is critical to the quality of the raw images. Radiometric calibration is normally necessary if the images from different collection dates will be quantitatively compared, although variable light condition during a single data collection

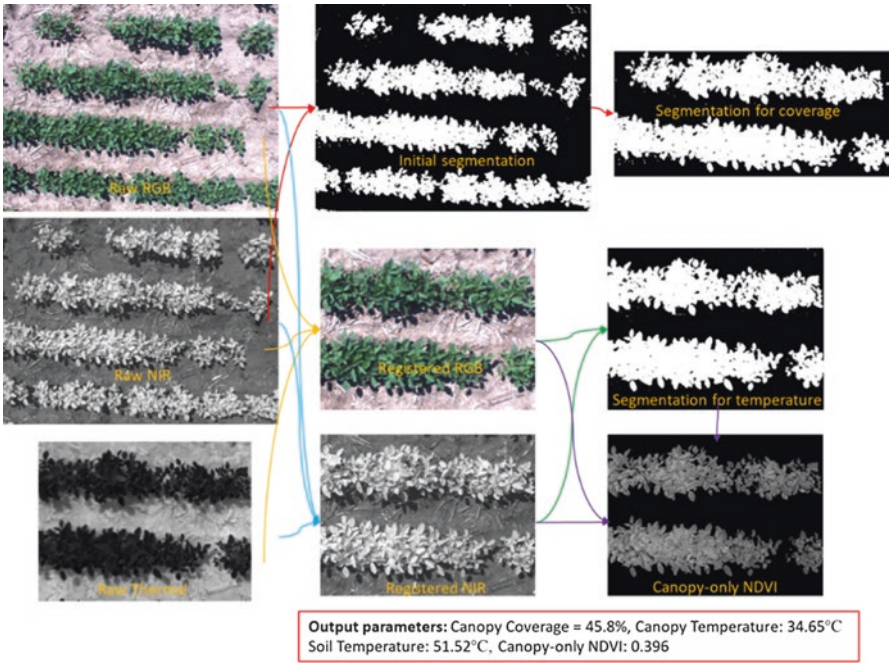


Fig. 6.4 An example of plant segmentation using a multispectral imaging system. The segmentation result also helps the canopy segmentation in thermal infrared image for the calculation of canopy temperature. The arrows indicate the workflows from raw images to the results

is still challenging. Comparing to RGB imaging, the spatial resolution of multispectral imagers (cameras) are usually coarser, and a good balance between the spatial resolution and the coverage area needs to be considered if the parameters of the sensing platform can be modified.

6.2.4 Thermal Infrared Imaging

Any object with a temperature above −273.15 °C emits electromagnetic radiation, and the flux density of the radiation energy is a function of the object's surface temperature and its emissivity. Thus, thermal infrared imagers can capture surface temperature information in the FOVs by measuring the infrared radiation at specific wavelength range emitted by the target. When crops are water stressed, leaf stomata tend to be closed to prevent excess water loss and plant dehydration. This process leads to a reduced heat dissipation through transpiration, which gives rise to higher leaf temperature. Thermal infrared imaging therefore is a useful nondestructive tool to estimate plant water stress through accurately measuring the canopy temperature. Plant water stress can be estimated by calculating the crop water stress index (CWSI) based on canopy temperatures and other meteorological parameters (Jackson et al. 1981). Evapotranspiration estimation of the canopy can also be carried out by measuring the temperature of the plant and canopy along with other environmental parameters (Norman et al. 1995; Li et al. 2005).

Canopy temperature in the field is highly variable with time due to the variation of the incident radiation, wind disturbance, air temperature, physiological activities of the plants, and other parameters. Thus, temperature measurement with high temporal and spatial resolutions could provide more information for the study. Figure 6.5 shows a series of thermal infrared images of a soybean plot captured under different times on a sunny day in July 2018 (Bai et al. 2019). Soil temperature experienced much more increase and decrease in the whole afternoon than that of the canopy. In addition, measurement accuracy of different cameras differs significantly depending on its performance as well as calibration protocols. How to capture the canopy

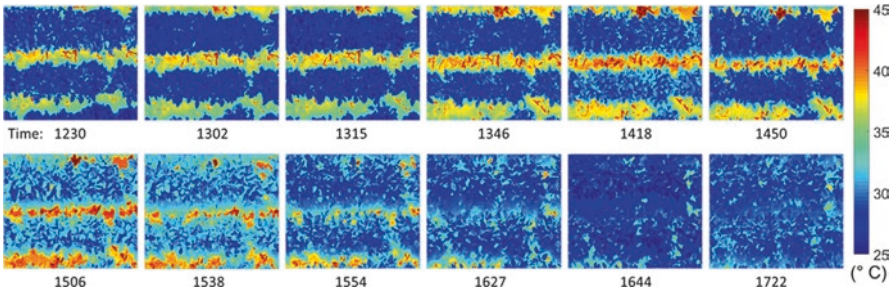
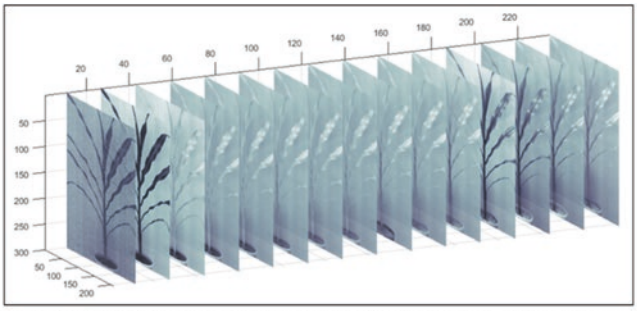


Fig. 6.5 Thermal image at different times from noon to late afternoon. Canopy and soil temperature ranges from 25 to 45 °C and the local solar noon was around 1330

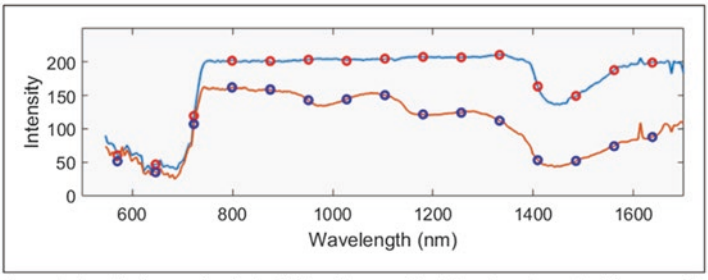
temperature of the target field quickly and accurately with enough spatial resolution is challenging. It is also a challenge to capture useful temperature data when using conveyer belt to transport potted plants to the imaging chamber in the greenhouse because of the temperature difference of the greenhouse and the imaging chamber.

6.2.5 *Hyperspectral Imaging*

Functioning of hyperspectral imaging systems has been described in Chap. 4 (Sect. 4.3.2). Briefly, this imaging system is capable of providing the largest amounts of spectral reflectance information of the individual plant or canopy. Figure 6.6 shows an example of a 3D image cube (called a hypercube) collected for a maize plant in an imaging chamber by a hyperspectral camera (Pandey et al. 2017). Regardless of the operation mechanism, the product of hyperspectral imaging systems could be considered as a hypercube, which not only includes spectral reflectance at each wavelength like a spectrometer but also delivers 2D images for each wavelength band in the FOV. Thus, each pixel array in the image has the spectral information in the whole wavelength range, while pixels at a certain wavelength consist of a 2D image at that wavelength. Either stacked 2D images at individual wavelength bands



(a) Image stack illustration of one image cube from a hyperspectral imager



(b) Reflectance intensity from a leaf pixel (blue line, red dots) and a stem pixel (orange line, blue dot)

Fig. 6.6 Hyperspectral image example. Figure (**a**) illustrates the stacked images at all wavelength bands, while figure (**b**) shows the reflectance intensity from a stem and leaf pixel from the hypercube

or the reflectance spectrum of individual image pixels from all wavelength bands could be used for data processing (Mahlein et al. 2012). With appropriate statistical methods, hyperspectral images are promising to estimate the spatial distribution of leaf properties, including water content, macro- and micronutrient content, and other important chemical concentrations of the plant, along with the ability to exclude the background using specific image-processing algorithms (Li et al. 2014).

Similar to spectrometer, careful calibration is needed to convert the pixel values into radiance or reflectance. The environmental illumination needs to be uniform and constant for the application of hyperspectral imaging in comparing different plant canopies (e.g., comparison between different pot plants in a greenhouse). In addition, the target plant needs to be kept completely stationary during the imaging/scanning period, which might take several minutes. For field application of the hyperspectral imaging system, the effect of variation of the illumination condition on the data, the large weight of the sensor, the stability of the sensor platform, wind disturbance, and the calibration protocols are the main factors influencing the data quality. However, a single hyperspectral camera could potentially replace the RGB, VNIR, and other imaging systems and provide more information as a single data set in the near future.

6.2.6 LiDAR

Canopy structure parameters (like canopy height) are important traits and affect its spectral reflectance signal. Plant canopy is a 3D leaf–stem structure which reflects, absorbs, scatters, and transmits incident light. Except the leaves at the top of canopy that reflect light directly back to the air, most of the light penetrates into the canopy architecture and is scattered in the canopy multiple times. Thus, different canopy structures can result in a difference in the total light absorption by the canopy.

Unlike other sensors mentioned above, light detection and ranging (LiDAR) is an active sensor which sends out laser pulses to quickly measure the distance between the laser source and the target based on the travel time of the pulse. The raw distance of a point target is usually based on polar coordinates and needs to be converted into Cartesian coordinates based on the physical set up of the LiDAR on the sensing platform. By continuously changing the laser generation angle slightly, the distance datum of different target locations is obtained. After the coordinate conversion and removing the unwanted scanning area, structural parameters of the crop canopy can be calculated from the processed point cloud with appropriate algorithms.

LiDAR has been widely used in vegetation sensing with aerial and ground systems for canopy height measurement, estimation of leaf area index, and other parameters related to canopy structure (Sun et al. 2018; Yuan et al. 2018; Jin et al. 2018; Zhao and Popescu 2009). 3D reconstruction work using LiDAR has also been investigated in order to retrieve more structure parameters. Usually, a moving platform for the LiDAR or a rotation platform for the plant itself is necessary to generate a high-quality 3D point cloud for retrieving structure traits if the LiDAR itself can only scan a single line of the target. 3D LiDAR can generate a 3D point cloud

without movement by rotating the laser housing at a certain degree. Figure 6.7 shows two examples of the LiDAR data collected in the field and greenhouse by different sensing platforms (Bai et al. 2019; Thapa et al. 2018). Obtaining point clouds with high density and good accuracy is a critical step to retrieving quantitative parameters.

Specific algorithms are needed if researchers need to reconstruct the plant structure and estimate various attributes such as leaf area and leaf angle distribution (Thapa et al. 2018). A very accurate measurement using LiDAR requires the absolutely stable/stationary instrument and target, which is not always possible for crop sensing in the field due to the plant movements caused by wind. Accurate and fast data logging of the platform status (roll, yaw, rotation), time stamp, and geographic tag with the LiDAR measurement can improve the LiDAR measurement accuracy when operating LiDAR on mobile sensing platforms. A good multireturn capacity is needed if more canopy structure information inside the canopy is needed.

6.3 Applications of Crop Sensing Systems

6.3.1 Variable Rate Fertilization and Irrigation

Over application of various crop inputs such as fertilizer and irrigation water often leads to low utilization efficiency of resources and negative economic and environmental impact (Carpenter et al. 1998). Applied fertilizer that cannot be taken up by

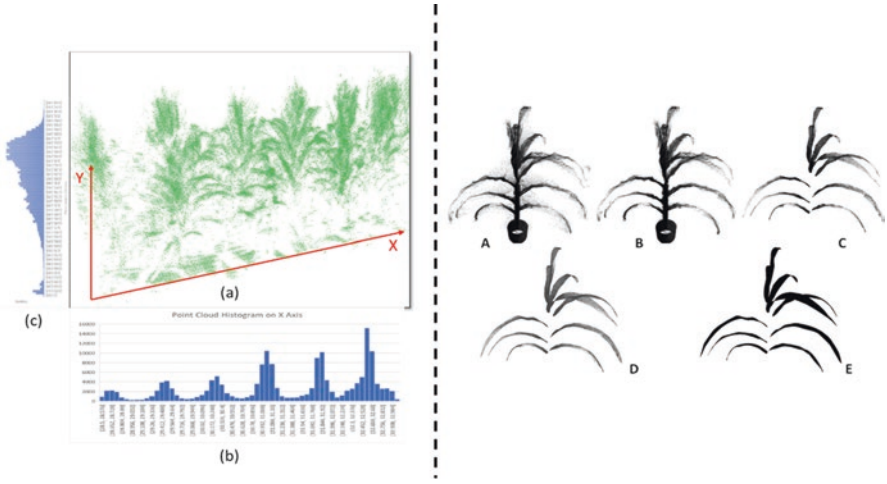


Fig. 6.7 Two examples of LiDAR point clouds of maize plants under different scenarios. Left: point cloud collected from 6 maize plants with a row spacing of 30 inches in the field condition with (**a**) LiDAR point clouds, (**b**) a histogram to show the count of the data points along X axis, and (**c**) a histogram to illustrate the count of the data points along Y axis. Right: different stages of the data processing for a maize plant under indoor environment started from (**a**) raw point cloud to (**e**) a 3D rendering of the leaf surface model

the plants goes into underground water by deep percolation, which leads to excessive nitrogen concentration in the water resources. Over-irrigation results in more deep percolation and surface runoff, which brings more nitrogen into the surface and underground waterbodies. Thus, site-specific application of irrigation water and chemicals based on how much the crop actually needs has been studied for decades (McCann et al. 1997). Data collection and processing, application map generation, and application are the main pipeline of variable rate fertilization (VRF) and irrigation (VRI).

Developing a reliable sensing and scheduling system which could bring more net profit for the users remains the major challenge for a faster adoption of VRI technology (O'Shaughnessy et al. 2019). Water saving in crop irrigation could make significant contribution to the conservation of freshwater resources because agriculture is the largest component of freshwater consumption (Elliott et al. 2014). Economic benefits could also be achieved by saving pumping cost and water bill at certain regions (Lo et al. 2016). With over 30 years' research and development, off-the-shelf VRI application systems are available, but the generation of application maps are still evolving along with the advancement of the sensing technologies. Soil sensing and crop sensing are the two sensing categories for the decision-making of VRI applications. Soil sensing has provided valuable information for precision agriculture since the mid1980s from intensive soil sampling to real-time proximal sensing (Mulla 2013). Off-the-shelf soil mapping system for soil electrical conductivity sensing, which is related to soil texture and moisture content, is also available. With the advancing of remote sensing technologies using satellite platforms, scientists started developing application tools for soil and crop sensing from the early 1990s (Bhatti et al. 1991). Historically, the low temporal (1 week or more) and spatial resolution (30 m or more) of the data from satellite images were not enough for most of the VRI applications. However, new commercial satellites with biweekly, submeter resolution are already operating and have the potential to be used in precision agriculture.

Surface irrigation, sprinkler irrigation, and drip irrigation are the three main irrigation methods in production, while the currently available VRI packages are for the large-scale moving sprinkler systems, including center pivot and lateral-moving systems. High-frequency, cloud-free imagery is critical in the initial data collection phase if satellite platforms are used (Barker et al. 2018). Canopy temperature data collected from infrared thermometers mounted on the center pivots are used to detect crop water stress and trigger the VRI application (Andrade et al. 2017).

Compared to VRI, VRF has attracted more attention because it has more potential to be economically sound if less fertilizer is applied. Significant lower application of nitrogen and irrigation water by VRF could be economically beneficial to producers and minimize the negative impact to the local environment (Diacono et al. 2013) while maintaining or increasing crop yield and quality. One of the biggest challenges of the data collection for VRF is to find a reliable parameter for estimating leaf nutrient concentration using crop sensing technologies. Tractor-mounted NDVI sensors have been used for real-time VRF after researchers found a strong relationship between the crop spectral reflectance and the crop nitrogen

uptake (Stone et al. 1996). A big challenge is that these real-time VRF systems need users to set and sense "reference plots" strategically located at different locations of the field to calibrate the sensor readings (Kitchen et al. 2010). Unmanned aerial systems (UAS) provide another promising tool for soil and crop sensing based on the remote sensing technologies due to its significantly higher resolutions and operation flexibility (Ferguson and Rundquist 2018).

6.3.2 Site-Specific Mechanical Weeding

Site-specific mechanical weeding systems for in-row and intra-/between-row weed has the potential to diminish herbicide usage in the future and completely avoid its negative environmental effect (Slaughter et al. 2008). A real-time weed sensing technology is a critical component to guide the mechanical weeding components. Optical sensors on different sensing platforms have been widely studied for weed sensing (Thorp and Tian 2004; Brown and Noble 2005; Peteinatos et al. 2014; Bajwa et al. 2015).

Because the mechanical weeding component needs the precise location information of the weed plant for accurate physical weeding, an on-board weed sensing system on the weeding machine becomes more promising than a separate sensing platform. In addition, the weed detection algorithm needs to react fast enough to trigger the weeding components, which means time-consuming algorithms are not preferred even with better accuracy. Nonimaging sensors, like spectrometer, can indicate a potential area with high weed stress based on abnormally high NDVI values. In addition to the spectral parameters, imaging systems can also leverage the morphological differences between crops and weeds (Perez et al., 2000; Søgaard 2005; Ishak et al. 2009; Burgos-Artizzu et al. 2011). Using RGB imager alone, good performance of plant segmentation algorithms under changing light condition is still challenging (Hamuda et al. 2016). An alternative approach is to create a controlled, artificial illumination environment in the field which naturally eliminates the issues of the changing light condition (Lottes et al. 2016). However, this solution also introduces more limitations due to the shading structure and the illumination chamber. Fluorescence and hyperspectral imagers are also being used for weed detection (Behmann et al., 2015; Underwood et al. 2017).

6.3.3 High-Throughput Plant Phenotyping

With the rapid increasing global population, meeting the needs for food, fiber, and fuel with the limited arable land and under the changing climate remains a major challenge. Breeding new crop cultivars with better tolerance for biotic and abiotic stresses is one of the most important solutions for a continuous yield improvement. Collection of genetic and phenotypic data (crop height, lodging, flowering date,

growth speed, stress-tolerance ability, to name a few) are carried out by breeders for selecting the top performers among the candidate genotypes. Breeders are still using labor-intensive methods to collect phenotypic data by portable tools and by observations from experienced researchers (Furbank and Tester 2011; Araus and Cairns 2014). In recent years, engineers have been working with breeders to develop high-throughput phenotyping systems at different scales, which could potentially replace the conventional ways of phenotyping with much higher efficiency.

Lab, greenhouse, and field-based plant phenotyping facilities have been developed to carry out phenotypic data collection for breeding research. The lab-scale facility is usually designed for *Arabidopsis thaliana*, which is a model plant for plant science research (Granier et al. 2006). Automatic high throughput greenhouses have been developed in different fashions. They are generating big phenotypic data set from integrated sensors, including RGB camera, multispectral imaging, fluorescence imaging, thermal IR imaging, and hyperspectral imaging (Humplík et al. 2015; Ge et al. 2016). Figure 6.8 shows the concentration prediction of six macronutrients of maize and soybean leaves using the hyperspectral image data with the partial least square regression (PLSR) method (Pandey et al. 2017).

Off-the-shelf phenotyping systems at lab and greenhouse scales have been on the market, while different field systems are being developed for specific research objectives with their own merit and limitations. Large-scale ground facilities enable accuracy and highly repeatable measurements with high payload capacity for data fusion (Virlet et al. 2017; Kirchgessner et al. 2017; Bai et al. 2019). Figure 6.9 shows two measurement campaigns using a large-scale phenotyping facility at leaf level with matched environmental parameters. These facilities could also be used

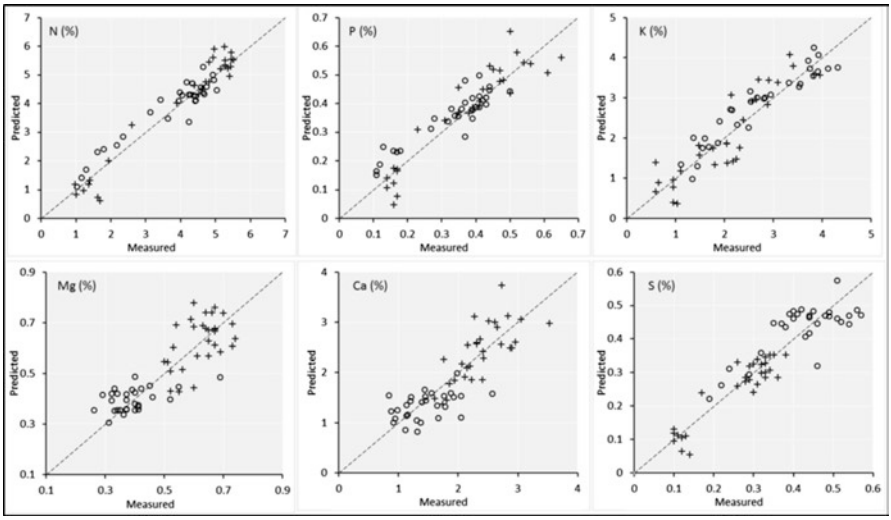


Fig. 6.8 Concentration prediction of six macronutrients in corn and soybean leaves from hyperspectral image cube using the PLSR method. Maize plants are denoted by circles, and soybean plants are indicated by crosses. (Pandey et al. 2017)

for the selection of sensor packages for mobile platforms with limited payload capacity by testing different combinations of sensors. However, these systems are not currently affordable for most of the independent research groups due to the cost. In addition, this type of system cannot be transported to another experiment location.

UAS could cover a large area of the field with limited payload and can be easily transported to different locations (Sankaran et al. 2015; Shi et al. 2016). However, limited flight time and payload, accident possibilities, and platform stability could be the main concerns to use UAS for plant phenotyping. Ground platforms have more payload than UAS and provide a more stable platform for crop sensing (Deery et al. 2014). It also has the potential to carry out contact sensing by integrating robotic arms into the systems. Under-canopy navigation, field accessibility, soil compaction, and crop damage are the main limitations for the ground platforms.

6.4 Summary and Concluding Thoughts

In this chapter, several widely used sensing systems for crop sensing study have been introduced with example data set collected using proximal sensing platforms. Three different applications in precision agriculture based on crop sensing has also been discussed, including variable rate application technology, site-specific mechanical weeding, and high throughput plant phenotyping.

Thanks to the rapid development and enhancement of the modern sensing technologies, lots of affordable sensors are now available to carry out crop sensing research with acceptable speed and accuracy. With the further evolution of the instruments, more and more sensors with low price, low energy consumption, fast and accurate measurement, and lightweight will be applied. It is certain that the plant traits that can be currently measured will be measured at much higher

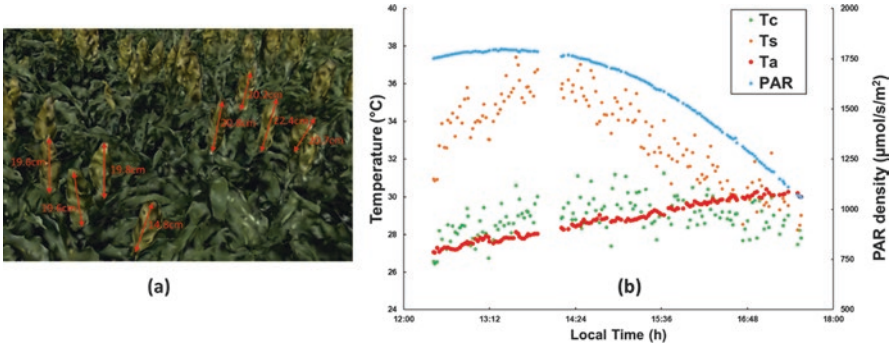


Fig. 6.9 Preliminary data set using a cable-suspended field phenotyping facility at the University of Nebraska-Lincoln. Figure (**a**) illustrates the measured length of panicle length of sorghum plants. Figure (**b**) shows the relationships between the temperature of canopy and soil and the environmental parameters

temporal and spatial resolutions, while the plant traits which cannot be measured nowadays will be accurately observed by new instruments. A close collaboration between scientists and engineers in crop sensing research would continuously advance the crop sensing technologies.

As mentioned above, the continuous research on VRF, VRI, site-specific mechanical weeding, and other solutions for precise field management will provide much more sustainable solutions for crop producers, which is economically and environmentally friendly. In the near future, these technologies are expected to be seamlessly integrated into the agricultural activities by improved connectivity in the field (Ojha et al. 2015; Vasisht et al. 2017). In the long terms, different robotic platforms and humanoid robots driven by powerful artificially intelligent (AI) technologies might provide more environmental-friendly and low-cost solutions for precision agriculture (Nelson et al. 2019) and crop phenotyping.

When it comes to high-throughput plant phenotyping, one of the main objectives of developing such a technology is to leverage new technologies to revolutionize the current phenotyping work in plant breeding by significantly improving the efficiency and accuracy of phenotyping. In addition, new breeding protocols with significantly short breeding cycle might be possible by combining advanced genotyping, high-throughput phenotyping, and advanced statistical techniques.

References

Alexandratos N, Bruinsma J (2012) World agriculture towards 2030/2050: the 2012 revision

Andrade MA, O'Shaughnessy SA, Evett SR (2017) ARSPivot, a sensor based decision support tool for the integrated irrigation management of VRI center pivot systems. In: Proceeding of the 28th annual central plains irrigation conference

Araus JL, Cairns JE (2014) Field high-throughput phenotyping: the new crop breeding frontier. Trends Plant Sci 19(1):52–61

Badgley G, Field CB, Berry JA (2017) Canopy near-infrared reflectance and terrestrial photosynthesis. Sci Adv *3*(3):e1602244

Bai G, Ge Y, Scoby D, Leavitt B, Stoerger V, Kirchgessner N et al (2019) NU-Spidercam: a large-scale, cable-driven, integrated sensing and robotic system for advanced phenotyping, remote sensing, and agronomic research. Comput Electron Agric 160:71–81

Bajwa AA, Mahajan G, Chauhan BS (2015) Nonconventional weed management strategies for modern agriculture. Weed Sci 63(4):723–747

Barker JB, Heeren DM, Neale CM, Rudnick DR (2018) Evaluation of variable rate irrigation using a remote-sensing-based model. Agric Water Manag 203:63–74

Behmann J, Mahlein AK, Rumpf T, Römer C, Plümer L (2015) A review of advanced machine learning methods for the detection of biotic stress in precision crop protection. Precis Agric 16(3):239–260

Berni JA, Zarco-Tejada PJ, Suárez L, Fereres E (2009) Thermal and narrowband multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle. IEEE Trans Geosci Remote Sens 47(3):722–738

Bhatti AU, Mulla DJ, Frazier BE (1991) Estimation of soil properties and wheat yields on complex eroded hills using geostatistics and thematic mapper images. Remote Sens Environ 37(3):181–191

Brown RB, Noble SD (2005) Site-specific weed management: sensing requirements—what do we need to see? Weed Sci 53(2):252–258

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

Carpenter SR, Caraco NF, Correll DL, Howarth RW, Sharpley AN, Smith VH (1998) Nonpoint pollution of surface waters with phosphorus and nitrogen. Ecol Appl 8(3):559–568

Deery D, Jimenez-Berni J, Jones H, Sirault X, Furbank R (2014) Proximal remote sensing buggies and potential applications for field-based phenotyping. Agronomy 4(3):349–379

Diacono M, Rubino P, Montemurro F (2013) Precision nitrogen management of wheat. A review. Agron Sustain Dev 33(1):219–241

Díaz-Varela RA, De la Rosa R, León L, Zarco-Tejada PJ (2015) High-resolution airborne UAV imagery to assess olive tree crown parameters using 3D photo reconstruction: application in breeding trials. Remote Sens 7(4):4213–4232

Elliott J, Deryng D, Müller C, Frieler K, Konzmann M, Gerten D et al (2014) Constraints and potentials of future irrigation water availability on agricultural production under climate change. Proc Natl Acad Sci 111(9):3239–3244

Ferguson R, Rundquist D (2018) Remote sensing for site-specific crop management. Precision Agriculture Basics, (precisionagbasics), 103–118

Frankenberg C, Berry J (2018) 2018_solar induced Chlorophyll fluorescence:_Origins, relation to Photosynthesis and retrieval. Available at http://www2.geog.ucl.ac.uk/~mdisney/teaching/GEOGG141/papers/sif.pdf, accesson on May 15, 2020

Furbank RT, Tester M (2011) Phenomics–technologies to relieve the phenotyping bottleneck. Trends Plant Sci 16(12):635–644

Gamon JA, Penuelas J, Field CB (1992) A narrow-waveband spectral index that tracks diurnal changes in photosynthetic efficiency. Remote Sens Environ 41(1):35–44

Ge Y, Bai G, Stoerger V, Schnable JC (2016) Temporal dynamics of maize plant growth, water use, and leaf water content using automated high throughput RGB and hyperspectral imaging. Comput Electron Agric 127:625–632

Granier C, Aguirrezabal L, Chenu K, Cookson SJ, Dauzat M, Hamard P et al (2006) PHENOPSIS, an automated platform for reproducible phenotyping of plant responses to soil water deficit in Arabidopsis thaliana permitted the identification of an accession with low sensitivity to soil water deficit. New Phytol 169(3):623–635

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

Huete A (1988) Huete, AR a soil-adjusted vegetation index (SAVI). Remote Sens Environ 25:295–309

Huete A, Didan K, Miura T, Rodriguez EP, Gao X, Ferreira LG (2002) Overview of the radiometric and biophysical performance of the MODIS vegetation indices. Remote Sens Environ 83(1–2):195–213

Humplík JF, Lazár D, Husičková A, Spíchal L (2015) Automated phenotyping of plant shoots using imaging methods for analysis of plant stress responses–a review. Plant Methods 11(1):29

Ishak AJ, Hussain A, Mustafa MM (2009) Weed image classification using Gabor wavelet and gradient field distribution. Comput Electron Agric 66(1):53–61

Jackson RD, Idso SB, Reginato RJ, Pinter PJ Jr (1981) Canopy temperature as a crop water stress indicator. Water Resour Res 17(4):1133–1138

Jin S, Su Y, Wu F, Pang S, Gao S, Hu T et al (2018) Stem–leaf segmentation and phenotypic trait extraction of individual maize using terrestrial LiDAR data. IEEE Trans Geosci Remote Sens 57(3):1336–1346

Kirchgessner N, Liebisch F, Yu K, Pfeifer J, Friedli M, Hund A, Walter A (2017) The ETH field phenotyping platform FIP: a cable-suspended multi-sensor system. Funct Plant Biol 44(1):154–168

Kitchen NR, Sudduth KA, Drummond ST, Scharf PC, Palm HL, Roberts DF, Vories ED (2010) Ground-based canopy reflectance sensing for variable-rate nitrogen corn fertilization. Agron J 102(1):71–84

Li F, Kustas WP, Prueger JH, Neale CM, Jackson TJ (2005) Utility of remote sensing–based two-source energy balance model under low-and high-vegetation cover conditions. J Hydrometeorol 6(6):878–891

Li L, Zhang Q, Huang D (2014) A review of imaging techniques for plant phenotyping. Sensors (Basel) 14:20078–20111

Lo TH, Heeren DM, Martin DL, Mateos L, Luck JD, Eisenhauer DE (2016) Pumpage reduction by using variable-rate irrigation to mine undepleted soil water. Trans ASABE 59(5):1285–1298

Lottes P, Hoeferlin M, Sander S, Müter M, Schulze P, Stachniss LC (2016) An effective classification system for separating sugar beets and weeds for precision farming applications. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp 5157–5163. IEEE

Mahlein AK, Steiner U, Hillnhütter C, Dehne HW, Oerke EC (2012) Hyperspectral imaging for small-scale analysis of symptoms caused by different sugar beet diseases. Plant Methods 8(1):3

McCann IR, King BA, Stark JC (1997) Variable rate water and chemical application for continuous-move sprinkler irrigation systems. Appl Eng Agric 13(5):609–615

Meroni M, Rossini M, Guanter L, Alonso L, Rascher U, Colombo R, Moreno J (2009) Remote sensing of solar-induced chlorophyll fluorescence: review of methods and applications. Remote Sens Environ 113(10):2037–2051

Morison JIL, Baker NR, Mullineaux PM, Davies WJ (2008) Improving water use in crop production. Philos Trans R Soc B Biol Sci 363(1491):639–658

Mulla DJ (2013) Twenty five years of remote sensing in precision agriculture: key advances and remaining knowledge gaps. Biosyst Eng 114(4):358–371

Mutanga O, Skidmore AK (2004) Narrow band vegetation indices overcome the saturation problem in biomass estimation. Int J Remote Sens 25(19):3999–4014

Nelson G, Saunders A, Playter R (2019) The PETMAN and Atlas robots at Boston dynamics. In: Humanoid robotics: A reference, pp 169–186

Norman JM, Kustas WP, Humes KS (1995) Source approach for estimating soil and vegetation energy fluxes in observations of directional radiometric surface temperature. Agric For Meteorol 77(3–4):263–293

O'Shaughnessy SA, Evett SR, Colaizzi PD, Andrade MA, Marek TH, Heeren DM et al (2019) Identifying advantages and disadvantages of variable rate irrigation-an updated review. Appl Eng Agric 35(6):837

Ojha T, Misra S, Raghuwanshi NS (2015) Wireless sensor networks for agriculture: the state-of-the-art in practice and future challenges. Comput Electron Agric 118:66–84

Olesen JE, Trnka M, Kersebaum KC, Skjelvåg AO, Seguin B, Peltonen-Sainio P et al (2011) Impacts and adaptation of European crop production systems to climate change. Eur J Agron 34(2):96–112

Pandey P, Ge Y, Stoerger V, Schnable JC (2017) High throughput in vivo analysis of plant leaf chemical properties using hyperspectral imaging. Front Plant Sci 8:1348

Perez AJ, Lopez F, Benlloch JV, Christensen S (2000) Colour and shape analysis techniques for weed detection in cereal fields. Comput Electron Agric 25(3):197–212

Peteinatos GG, Weis M, Andújar D, Rueda Ayala V, Gerhards R (2014) Potential use of ground-based sensor technologies for weed detection. Pest Manag Sci 70(2):190–199

Pimentel D, Burgess M (2014) Environmental and economic costs of the application of pesticides primarily in the United States. In: Integrated pest management. Springer, Dordrecht, pp 47–71

Sankaran S, Khot LR, Espinoza CZ, Jarolmasjed S, Sathuvalli VR, Vandemark GJ et al (2015) Low-altitude, high-resolution aerial imaging systems for row and field crop phenotyping: a review. Eur J Agron 70:112–123

Shi Y, Thomasson JA, Murray SC, Pugh NA, Rooney WL, Shafian S et al (2016) Unmanned aerial vehicles for high-throughput phenotyping and agronomic research. PLoS One 11(7)

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

Søgaard HT (2005) Weed classification by active shape models. Biosyst Eng 91(3):271–281

Stone ML, Solie JB, Raun WR, Whitney RW, Taylor SL, Ringer JD (1996) Use of spectral radiance for correcting in-season fertilizer nitrogen deficiencies in winter wheat. Trans ASAE 39(5):1623–1631

Sun S, Li C, Paterson AH, Jiang Y, Xu R, Robertson JS et al (2018) In-field high throughput phenotyping and cotton plant growth analysis using LiDAR. Front Plant Sci 9:16

Svensgaard J, Roitsch T, Christensen S (2014) Development of a mobile multispectral imaging platform for precise field phenotyping. Agronomy 4(3):322–336

Thapa S, Zhu F, Walia H, Yu H, Ge Y (2018) A novel LiDAR-based instrument for high-throughput, 3D measurement of morphological traits in maize and sorghum. Sensors 18(4):1187

Thorp KR, Tian LF (2004) A review on remote sensing of weeds in agriculture. Precis Agric 5(5):477–508

Tucker CJ (1978) Red and photographic infrared linear combinations for monitoring vegetation

Underwood J, Wendel A, Schofield B, McMurray L, Kimber R (2017) Efficient in-field plant phenomics for row-crops with an autonomous ground vehicle. J Field Robot 34(6):1061–1083

United Nations (2013) World population prospects: the 2012 revision, highlights and advance tables. New York United Nations Department of Economic & Social Affairs

Vasisht D, Kapetanovic Z, Won J, Jin X, Chandra R, Sinha S, ... Stratman S (2017) Farmbeats: an IOT platform for data-driven agriculture. In: 14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17), pp 515–529

Virlet N, Sabermanesh K, Sadeghi-Tehran P, Hawkesford MJ (2017) Field Scanalyzer: an automated robotic field phenotyping platform for detailed crop monitoring. Funct Plant Biol 44(1):143–153

Yuan W, Li J, Bhatta M, Shi Y, Baenziger PS, Ge Y (2018) Wheat height estimation using LiDAR in comparison to ultrasonic sensor and UAS. Sensors 18(11):3731

Zaman-Allah M, Vergara O, Araus JL, Tarekegne A, Magorokosho C, Zarco-Tejada PJ et al (2015) Unmanned aerial platform-based multi-spectral imaging for field phenotyping of maize. Plant Methods 11(1):35

Zhao K, Popescu S (2009) Lidar-based mapping of leaf area index and its use for validating GLOBCARBON satellite LAI product in a temperate forest of the southern USA. Remote Sens Environ 113(8):1628–1645

Part II
Mechanisms, Dynamics and Control

Chapter 7
Robotic Manipulation and Optimization for Agricultural and Field Applications

Changki Mo, Joseph Davidson, and Cameron Hohimer

7.1 Introduction

The design of robot kinematics and its optimization for agricultural and field applications should be based on a specific task. Robots with kinematics suitable for a specific task may perform better than a universal robot expected to perform several tasks. The tasks needed for precision agriculture including planting, pruning, thinning, harvesting, weeding, spraying, and post-harvesting will require specific robotic manipulation. Those are also largely dependent on crop types and working environment – outdoor fields or greenhouses (Van Henten et al. 2009; Bechar and Vigneault 2016; Bloch et al. 2015, 2018).

An essential step in the design of a robotic manipulator for a specific task is the creation of a kinematic framework flexible enough to accommodate the crop environment. However, the unstructured environment of agricultural fields and orchards pose unique engineering challenges compared to other possible applications such as automation in manufacturing plants. Some of these challenges include variable outdoor conditions; complex plant structures; inconsistency in product pose, shape, size, and color; and delicate products. The most critical source of variation affecting

Disclaimer: Mention of a commercial product is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated.

C. Mo (✉)
Washington State University Tri-Cities, Richland, WA, USA
e-mail: changki.mo@wsu.edu

J. Davidson
Oregon State University, Corvallis, OR, USA

C. Hohimer
Harvard University, Cambridge, MA, USA

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_7

the design of the robotic manipulator for agricultural and field applications is the highly irregular and unstructured environment (Bac et al. 2014; Davidson and Mo 2015; Silwal et al. 2017). In addition, a target object (e.g., specific crop) can be surrounded by densely spaced obstacles (e.g., branches and leaves, support wires, and crop clusters in a fruit orchard). The manipulator and end-effector need to avoid these obstacles to successfully approach a target object (e.g., a fruit) and prevent damage to the other objects in the environment (e.g., tree branches or nearby fruit) (Van Henten et al. 2010; Bac et al. 2016).

Noting that successful planning and implementation of the manipulator motion determines the feasibility of a robotic system for a specific task in a predefined workspace, it would be useful to understand the influence of design parameters affecting motion planning success.

7.2 Design Considerations/Constraints

Traditionally, robotic manipulators for agricultural and field applications are designed for specific manipulative tasks by looking mainly to target crop type, working environment, workspace volume, payload capacity, and target performance metrics that may include manipulability and velocity performances. Although it is quite reasonable to consider these aspects as fundamental criteria for manipulator design, in general they can give contradictory results in the design phase. Therefore, a formulation as multi-objective optimization problem can be convenient in order to consider them simultaneously (Ceccarelli et al. 2005).

As for manipulability and velocity performance, design requirements and operational feasibility should be investigated for any possible singularity conditions. In fact, it is desirable to ensure a given workspace volume within which the manipulator can be movable, controllable, and far enough from singularities. The singularity analysis for robotic manipulators can be performed by means of the Jacobian matrices. In general, the condition for identifying singular configurations can be represented by surfaces in the n dimensional joint space, and they can be identified by finding the configurations at which the determinant of the Jacobian matrix $\mathbf{J}$ is zero. The concept of singularity has been extensively studied, and several classification methods have been defined. In the agricultural and field robotic systems, serial manipulators typically are used and singularity occurs when a manipulator reaches internal or external boundaries of its workspace and the output link loses one or more degrees of freedom (DOFs).

7.2.1 Target Crop Type

Design criteria for autonomous robotic manipulation depend on the working environment including crop types. For example, some high-value crops for which automated harvesting has been a pressing need to reduce production costs are grown in the greenhouse environment, such as tomatoes, cucumbers, and sweet peppers. The main design criteria for these crops will be the ability of the robot to reach the crops within a predefined greenhouse volume, its size because of the limited workspace and the possibility of collisions with plants and greenhouse structure, its maximum payload, and the speed and ability of the machine to operate in adverse climate conditions with high temperature and humidity levels (Van Henten et al. 2009). On the other hand, fresh market apples and citrus are commonly grown in outdoor orchards. Currently, the working environment for modern apple orchards, for example, is a well-maintained row of trees that are trained, pruned, and thinned to have planar canopies. The trees are shaped like a vertical wall or a tilted wall supported by a V-trellis. This is a simplified environment compared to traditional highly unstructured and biologically driven orchards. It is relatively obstacle-free with improved fruit accessibility. This architecture enables better apple detection and reduces the complexity of planning and control algorithms. Nevertheless, it is worthwhile to note that robotic systems will operate in tight working environments either in outdoor fields or greenhouses. To reach a crop in an obstacle-dense crop environment in either working environment, robotic manipulation requires a collision-free motion of the manipulator and end-effector.

The optimal design parameters considered and motion planning in robotic harvesting high-value crops in a dense obstacle environment will be different from harvesting apples from a fruiting wall environment. For the former, the effect of the angle of approach of the end-effector to the fruit on suitable motion planning avoiding risky paths may need to be investigated. In order to find a collision-free goal configuration and path, the parameters specifying the crop environment (e.g., fruit location and stem spacing), the robot (end-effector dimensions and robot position), and the planning algorithm can be considered. Bac et al. (2016) indicated that reducing end-effector dimensions and widening stem spacing improved goal configuration success. Furthermore, the fruit location at the stem is the strongest influencing parameter and therefore provides an incentive to train or breed plants that develop more fruit at the front side of the plant stem. For the latter, it would be ideal to be able to determine an optimal distance from the vehicle to the fruiting wall at each robotic harvesting stop to reach the maximum number of apples.

7.2.2 Workspace Considerations

One of the objectives of the manipulator kinematic design for a specific task is to cover a predefined workspace. The workspace of a manipulator can be described as the total volume swept out by the end-effector as the manipulator executes all possible motions. The workspace is constrained by the geometry of the manipulator as well as mechanical constraints on the joints (Spong et al. 2006). It is one of the most important kinematic properties required for robotic manipulation, which impacts significantly on manipulator design and location in a worksite. Although this workspace definition is commonly referred to as the reachable workspace in robotic literature, the workspace can be categorized into a reachable workspace and a dexterous workspace.

The reachable workspace is the entire set of points reachable by the manipulator, whereas the dexterous workspace of a manipulator is the volume of space which can be reached by the manipulator with arbitrary orientation. The dexterous workspace is useful in the context of task planning since it allows the orientation of the end-effector to be ignored when positioning objects in the dexterous workspace (Murray et al. 1994). Ceccarelli et al. (2005) expanded the definition of the workspace even further as position workspace refers to reachable points by a reference point on the manipulator, and orientation workspace describes the angles that can be swept by reference axes on the manipulator. In addition, the task space can be defined by all feasible positions and orientations of the manipulator's end-effector. The task space can be conveniently represented in Cartesian coordinates with Euler angles.

The workspace can be computed with the forward kinematics equations or graphically constructed while considering the mechanical joint limits and the geometry of the robot. Because the forward kinematics function is continuous, the volume can be defined by an enclosing surface (Baur 2015). In particular, an operation of the manipulator at the workspace limit should in general be avoided, because the freedom to move the end-effector in any direction declines. It is worthwhile to note that the overlap of the arm's workspace and the crop space is the key to increase the reachability instead of only increasing the volume of arm's workspace (Wang et al. 2018).

7.2.3 Target Performance Metrics

The primary metric for robot performance evaluation is task success followed by speed in robotic literature (Righetti et al. 2014). To evaluate the robotic manipulation, a suitable quantitative performance criterion needs to be defined. For task success, manipulability has been commonly used as a criterion for manipulator design evaluation and optimization. Almost all manipulability measures found in the literature involve the "Jacobian" of the manipulator $\mathbf{J}$. Typically the determinant of Jacobian is used for a measure of manipulability. The design specifications

mentioned can be formalized by two quantitative performance criteria, one evaluating the length of a collision-free path to the target and the other evaluating the manipulability.

Alternatively, optimality criteria for manipulator design can be identified in performance evaluations regarding positioning and orientation capability and velocity response. Positioning and orientation capability can be evaluated by computing position and orientation workspaces that give the reachable regions by the manipulator as function of the mobility range of the manipulator joints. The velocity response can be evaluated by looking at the velocity mapping that can be described by the Jacobian of the manipulator. The Jacobian is also useful to identify singular configurations (singularities) of a manipulator at which DOFs are lost that should be avoided in a controlled movement.

As for the speed requirement, the cycle time of a single robotic operation should be optimized to be economically feasible. This time will include the time for the vision system (e.g., fruit detection, ripeness assessment, and 3D localization of the crops, in a robotic picking example) and the time for robotic manipulation (motion planning, the motion of the end-effector to the crop, gripping, and cutting (usually of the crops in greenhouse, whereas grasping and detaching of those in outdoor orchard) and the return motion for the storage). To satisfy this performance requirement, the motion trajectory of the manipulator should be as short as possible (Van Henten et al. 2009).

Bac et al. (2016) used performance measures in terms of success rate, planning time, and path quality in manipulator motion planning for sweet pepper harvesting in an obstacle-dense greenhouse environment. The success rates comprised goal configuration success and path planning success. The goal configuration success measure represents the percentage of fruits for which a collision-free goal configuration was accessed. A path was considered collision-free if it was collision-free before and after path smoothing. The planning time refers to the time elapsed to plan a path. Path quality was quantified by the joint angles index of curvature (JAIC). JAIC, measured in the configuration space, is based on the city block distance metric capturing the overall sum of the joint angle paths, i.e., JAIC measures how much the joints rotate. JAIC was normalized by the sum of joint angle paths for the "line-of-sight" path (the shortest path from start point to goal point); thus, JAIC ranges from 1 (the shortest path) to 0. The normalization facilitated path quality comparisons among groups of crops.

7.3 Orientation: What Is It Good For?

For a robot manipulator, the number of joints determines the number of DOF. A rigid object in three-dimensional free space has six DOFs: three for position and three for orientation. Therefore, a manipulator should typically possess at least six independent DOFs. With fewer than six DOFs, the robot cannot reach every point in its workspace with arbitrary orientation. Certain applications such as reaching

around or behind obstacles may require more than six DOFs. A manipulator having more than six DOFs is referred to as a kinematically redundant manipulator (Spong et al. 2006).

The manipulator should have maximum dexterity once the end-effector has arrived at the target position. Dexterity is a measure quantifying the ability of a manipulator to move and rotate the end-effector in all directions. The dexterity significantly matters at the endpoint of the motion path for a specific task such as the manipulator for a crop harvesting in greenhouse needs a large amount of dexterity at the endpoint to perform the picking task.

7.3.1 What Types of Tasks Are Required?

The robot design for precision agriculture, in particular, must be based on a well-defined task model to provide a good fit between a robot and its task because different tasks would require different robots to perform optimally in each task. Each precision agriculture task including planting, pruning, thinning, harvesting, weeding, spraying, and post-harvesting will require specific robotic manipulation with a task-based optimized robot. The optimization cost function can include time, energy, weight, and/or cost and depends on the customer demands (Bloch et al. 2015).

The main objective of our case study (Sect. 7.6) for the robotic apple harvester in an orchard was to optimize the robot manipulation to be able to reach all (or most) apples in each machine vision frame (Wang et al. 2018). For a precision agriculture task (e.g., for selectively picking apples in an orchard), the optimal robot must be able to reach all (or a certain majority) of the apples, i.e., interact with the task environment. This interaction with the environment is the optimization constraint: if it is not fulfilled, the robot is not suitable for the task (Bloch et al. 2015).

7.3.2 Required Manipulability

Manipulability can be used to determine the optimal configurations in which to perform certain tasks. In some cases, it is desirable to perform a task in the configuration for which the end-effector has the maximum manipulability. Manipulability can also be used to aid in the design of manipulators (Spong et al. 2006). For efficient Cartesian motion, the manipulator must be kept away from singular configurations, which can be evaluated using the manipulability measure.

The concept of manipulability of a manipulator was introduced by Yoshikawa (1985). The manipulability is defined as the square root of the determinant of the product of the manipulator Jacobian by its transpose:

$$\mu = \sqrt{\det\left(\mathbf{J}\mathbf{J}^{\mathrm{T}}\right)} \tag{7.1}$$

The manipulability μ is equal to the absolute value of the determinant of the Jacobian **J** in case of square Jacobian (Tenev and Stoyanov 2000).

Baur et al. (2012) performed the manipulability measure for the first manipulator prototype to validate the proposed manipulator kinematic design for the selective harvesting of sweet pepper in greenhouse. The manipulability measure H_2 normalized on the minimum value $H_{2,min}$ was calculated for discrete points on the paths for the weighting factors as depicted in Fig. 7.1. At a height of 0.8 m, the robot manipulability is lowest. At this height, the manipulator must fold itself up since the first prismatic joint has reached its mechanical limit. With respect to a high manipulator, an operation in this region should be avoided. The manipulability along γ is minimalat $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. Here the manipulator must take a stretched configuration since it isapproaching the limits of the workspace, and therefore the manipulability is reduced.

In addition, visual servoing, in which a camera is mounted on the end-effector, is commonly used for final alignment of the end-effector with the target crop in case of inaccurate positioning information from the main camera system of the robot. In this case, considerable dexterity is needed to be able to make the required corrections to the position and orientation of the end-effector. More details on visual servoing can be found in Chap. 9.

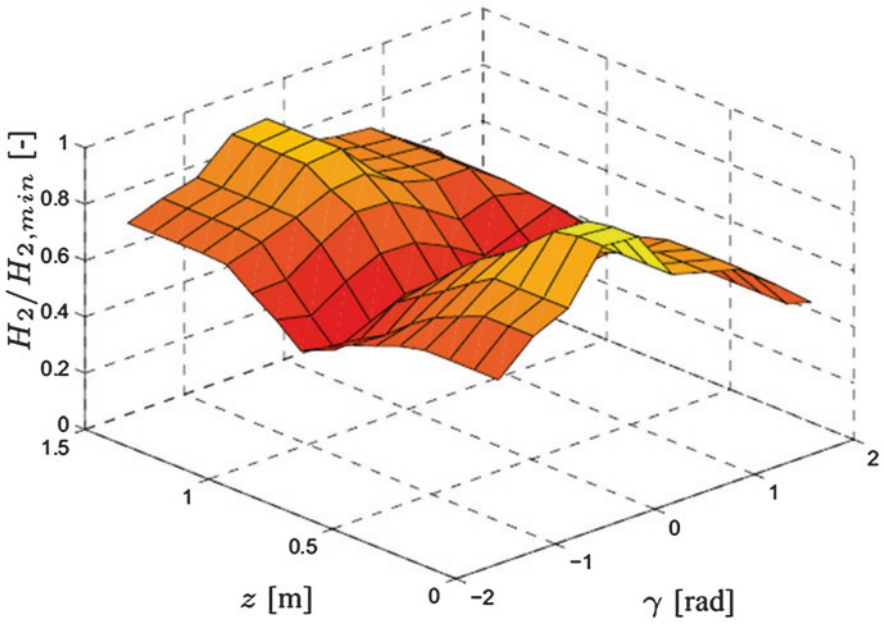


Fig. 7.1 Normalized manipulability on discrete points along the paths. (From Baur et al. 2012)

7.3.3 Speed vs Robustness

Speed or cycle time is one of the key performance metrics to evaluate robotic systems. Manipulators used for agricultural and field robotics (e.g., fruit harvesting) typically consist of two to seven DOFs. An advantage of using many DOFs is an improved target reachability in cluttered environments. However, the disadvantage is that common motion planning algorithms have difficulty in finding a collision-free path within a reasonable time. Furthermore, additional DOFs decrease speed and reliability and increase the cost of the manipulator. Visual servoing or force feedback can improve robustness in terms of position errors, but it may reduce the cycle time.

For the robotic apple harvester, our case study, the manipulator's planned trajectory is executed using a simple, open-loop look-and-move approach. To reduce design complexity and enhance speed of harvesting, both of which are design criteria, the end-effector has no sensors (visual servoing or force feedback is not used) and utilizes open-loop, feedforward control. As such, an important initial design step was to develop an environmental model for the actuation torque required to produce the desired link normal forces (Davidson and Mo 2015). However, field evaluation indicated that force feedback would improve success rate of robotic harvesting. Prismatic-only motion also can be slower than the combination of prismatic and rotation by revolute joints to reach a certain target point.

7.4 Robotic Manipulation

In design of a robotic manipulator, the first thing to be considered is the kinematic structure of the system. Typically, it is possible to design very specialized and optimized kinematics for a well-defined task, although modular manipulators can be designed to reconfigure the kinematics for the needs of the different applications (Baur et al. 2012; Baur 2015; Bloch et al. 2018).

The optimality criterion for robotic manipulation is to fulfill the given task within a predefined workspace and desired cycle time with a limited collision interacting suitably with the task environment.

7.4.1 Kinematics and Dynamics

7.4.1.1 Kinematics

The kinematics equations describe the motion of the robot without consideration of the forces and torques producing the motion.

Forward Kinematics

A manipulator is a kinematic chain of rigid bodies, or links, connected by discrete joints. Here we consider only open serial chains and exclude closed chains (i.e., chains that form a loop) as well as continuum structures with more complex motion. One end of the chain is constrained to a base, and the robot's end-effector is fixed at the other end of the chain. Joints can be either revolute or prismatic, and joint articulation defines each degree of freedom and constitutes a joint variable. The objective of forward kinematics is to determine the position and orientation of the end-effector given the values for the joint variables of the robot.

There are two general approaches to the forward kinematics problem. The first approach, which is covered in detail by Siciliano et al. (2010), incorporates the kinematic relationship between consecutive links in a recursive procedure to define the overall manipulator kinematics. Consider an open chain, serial link manipulator with n links where reference frames have been assigned to each link. Then, the forward kinematics describing the position and orientation of the end-effector frame n with the base frame 0 can be written as

$$\boldsymbol{T}_n^0(\boldsymbol{q}) = \boldsymbol{A}_1^0(q_1)A_2^1(q_2)\ldots\boldsymbol{A}_n^{n-1}(q_n) \tag{7.2}$$

where $\boldsymbol{A}_i^{i-1}(q_i)$ is the homogenous transformation matrix between consecutive links. Each homogenous transformation matrix can be completely defined by a single joint variable. One widely used method to determine the individual transformation matrices between links is the Denavit-Hartenberg (DH) convention (Denavit and Hartenberg 1955), the rules of which can be formalized in an operating procedure. An advantage of DH representation is that it requires the minimum number of parameters to describe the robot's kinematics.

A second approach to forward kinematics uses modern geometric techniques incorporating tools from classical screw theory. This approach uses an exponential description of rigid body motion where rotation matrices are represented with exponential coordinates. The forward kinematics are found with the product-of-exponential (PoE) formula where the key concept is to consider each joint as applying a screw motion to all the outward links (Lynch and Park 2017). Unlike DH representation, the PoE representation is not minimal. Its advantages are that it is intuitive owing to its geometric underpinnings and does not require assigning reference frames to each link. The reader is referred to the recent text by Lynch and Park (2017) for additional details on the PoE representation for forward kinematics.

Inverse Kinematics

A more fundamentally important problem for robotic manipulation in agricultural and field applications (e.g., robotic harvesting) is the inverse kinematics problem, which requires the determination of the joint variables corresponding to a given end-effector pose in the operational space. If the position of a crop in the Cartesian

coordinate system is known, what joint coordinates are required to place the harvesting end-effector at the correct position with the desired orientation? The inverse kinematics problem is much more complex than the direct kinematics problem for the following reasons (Siciliano et al. 2010):

- The equations are generally nonlinear and closed-form solutions may not exist
- Multiple solutions may exist
- Infinite solutions may exist

Because closed-form solutions only exist for certain classes of manipulators with simplified structures, general-purpose solution procedures require numerical methods. Some of the numerical methods developed include Jacobian transpose methods, pseudo-inverse methods, damped least squares methods, quasi-Newton methods, and neural net methods. The nonlinear optimization technique developed by Wang and Chen (1991) is commonly used because of its computational efficiency. It is not sensitive to the manipulator's singular configurations and is numerically more stable than other methods because it does not require matrix inversion. The technique uses the cyclic coordinate descent (CCD) method (Luenberger and Ye 2008) to rapidly find a feasible point that is near the true solution and then uses the Broyden-Fletcher-Goldfarb-Shanno (BFGS) variable metric method (Luenberger and Ye 2008) to obtain a solution at the desired degree of precision. The CCD method is a direct search method that cycles through n coordinate directions using each in turn as the search direction (Nocedal and Wright 1999).

7.4.1.2 Dynamics

The derivation of a manipulator's dynamic model is important for such activities as actuator selection, structural analysis, motion simulation, and controller design. Just like there are forward and inverse kinematics, there are also forward and inverse dynamics. The forward problem finds the robot's acceleration from its state variables (i.e., joint position and velocity) and joint forces and torques. Conversely, during the inverse dynamics problem, the joint torques/forces corresponding to the robot's state and the desired acceleration are found.

Two popular techniques are available for modeling the robot's dynamics and formulating the equations of motion, a system of second-order differential equations. With the Lagrangian formulation, the first step is to choose a set of generalized coordinates that describe the link positions of an n-DOF manipulator, which, for an open chain manipulator, is the vector of joint variables q. The generalized forces associated with the generalized coordinates include nonconservative joint torques, joint friction, and external forces from interaction with the environment. The equations of motion can be found with the well-known Lagrange's equations, available in mechanics textbooks, where the Lagrangian of the robot is the difference between the total kinetic and potential energies of the system. While the Lagrangian formulation's energy-based approach is conceptually simple for kinematic chains with a few degrees of freedom and also yields the equations of motion

in analytical form, the technique can become quite complex for higher-dimensional systems.

Whereas the Lagrangian formulation derives the robot's dynamic model using the system's total Lagrangian, the Newton-Euler formulation considers force balances on all manipulator links. For open kinematic chains, this leads to efficient recursive algorithms for both inverse and forward dynamics where a forward recursion starting at the base is used to propagate velocities and accelerations and a backward recursion starting at the end-effector is used to propagate forces and moments along the structure (Siciliano et al. 2010). An advantage of the Newton-Euler formulation is that it is computationally efficient. The texts by Siciliano et al. (2010) and Lynch and Park (2017) provide excellent overviews of using both the Lagrangian and Newton-Euler formulations to derive dynamic models of robots.

7.4.2 Path and Trajectory Planning

Robot motion planning is one of the central parts in the design of autonomous robotic manipulation and among the most difficult problems. The computational complexity of the best-known complete path planning algorithms grows exponentially with the number of internal DOF of the robot (Spong et al. 2006). The term motion planning incorporates a multitude of different aspects, like finding a collision-free path, handling the interaction of cooperating robots, impedance control, developing of grasping strategies, dealing with inaccurate or incomplete sensor information, and much more. Additionally, while fulfilling the aforementioned primary objectives, motion planning algorithms have to handle several physical constraints inherent to any technical system, like speed or power limitations of actuators (Baur 2015).

7.4.2.1 Collision-Free Path Planning

A heuristic A*-search algorithm is commonly used to calculate collision-free manipulator motions. The A*-search algorithm minimizes a measure of the length of the motion trajectory through the configuration space (Pearl 1984; Kondo 1991; Van Henten et al. 2003). Another interesting feature of the A*-search algorithm is that it only returns a solution if one exists. This means that if a goal configuration is not reachable without collisions caused by, for instance, too large links, this design can be heavily penalized. For the robot design optimization, the A*-search algorithm was modified to allow for both orthogonal and diagonal node expansion, as illustrated in Fig. 7.2, resulting in smoother motion trajectories than with orthogonal node expansion only (Van Henten et al. 2009).

Bac et al. (2016) proposed a sequence of algorithms consisting of three steps: calculation of the start and goal configuration, path planning, and path smoothing. Each step uses a collision detection module. The RRT (rapidly exploring random

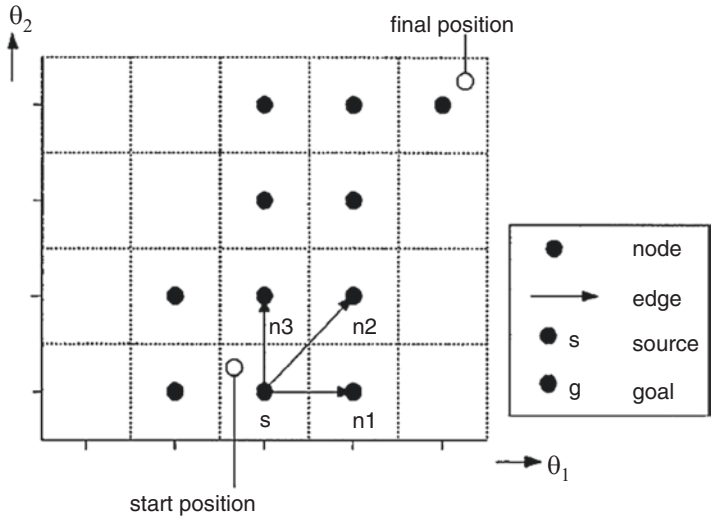


Fig. 7.2 Orthogonal and diagonal node expansion in the A*-search algorithm. (From Van Henten et al. 2009)

tree) is a well-known representative of single-query planners as stated by Kuffner and LaValle (2000). However, the paths found by RRT-based planning algorithms tend to be tortuous due to the random nature of such sampling algorithms. Therefore, a path smoothing algorithm was implemented following the path planning phase. Most path planning algorithms were not suitable for manipulators with a larger number of DOFs because planning time increases exponentially with the number of DOFs (Choset et al. 2005). However, the sampling-based planner implemented, a balanced bidirectional rapidly exploring random tree (bi-RRT) (Kuffner and LaValle 2000; LaValle 2006), is less affected by the number of DOFs. This planner randomly samples the configuration space using a user-defined sampling resolution. If the sample is collision-free, it is added to a growing tree. After a few iterations, the trees growing from the start and goal configuration may connect and a path is found.

Sampling-based motion planning algorithms have also proven extremely successful in addressing manipulation problems (Berenson et al. 2009; Diankov et al. 2008; Rusu et al. 2009). They are very efficient at finding collision-free paths in a high-dimensional state space. While they can quickly find feasible motion plans, they are often lacking in the quality of paths produced. Smoothing and post-processing is typically required before executing these plans on a robot (Hauser and Ng-Thow-Hing 2010), which can significantly add to the required computation time. Trajectory optimization techniques have recently been applied to motion planning for manipulation (Ratliff et al. 2009; Kalakrishnan et al. 2011). These techniques plan collision-free motions by minimizing a cost function over trajectories. Such approaches allow the user to include additional desirable objectives, such as trajectory smoothness, obstacle clearance, torque minimization, and constraint satisfaction. The use of optimization resolves the inherent redundancy in finding

collision-free configurations and trajectories, making the robot behave in a more predictable and repeatable fashion (Righetti et al. 2014).

7.5 Optimization: Putting It All Together

Using quantitative measures and nonlinear optimization techniques, optimal kinematic design of the manipulator can be pursued. Optimal robot design has received considerable attention in the engineering literature and recently in agricultural and field applications, in particular for greenhouse agriculture.

Van Henten et al. (2009) adopted a performance criterion scheme of combining both the measure of path length and the dexterity measure for autonomous cucumber harvesting in greenhouses. The optimal design combines small values of the measure of path length and large values of the dexterity measure. Once a performance criterion is defined, there are various ways to solve the optimization problem. Because we are dealing with a highly nonlinear optimization problem, the solution may contain several local minima (Paredis and Khosla 1995). Therefore, global optimization techniques are preferred for the solution. Local optima were detrimental to the performance of the SQP (sequential quadratic programming) algorithm and that differential evolution (DE) was most capable in dealing with the multi-modality of the optimization problem (Van Henten et al. 2009).

Bac et al. (2016) investigated the influence of five parameters, on success or planning time, including average stem spacing, fruit location at a side of the stem, dimensions of the end-effector, robot position, and sampling resolution (ε) for the robotic system used for sweet pepper harvesting in an obstacle-dense greenhouse. Average stem spacing was expected to influence goal configuration success and path planning success because it modifies the density of the obstacle map. Fruit location at a side of the stem was expected to influence goal configuration success and path planning success because, for instance, back-side fruit are further away from the robot and obstacles are more likely to obstruct a path than for front-side fruit. Changing the dimensions of the end-effector was expected to influence goal configuration success and path planning success because dimension of the end-effector influences the size of the collision-free workspace of the manipulator. Dimensions were precisely known and did not vary among harvest cases. Robot position influences dexterity of the manipulator (Klein and Blaho 1987), i.e., the range of motion toward a target object (e.g., fruit in plant canopies). In addition, variation in position and orientation of the target object influence the robot position required to reach an object. Sampling resolution (ε) is a parameter in the bi-RRT algorithm that determines the probability of finding a path. A smaller resolution increases this probability at the cost of additional iterations. A small resolution is typically required for crops surrounded by many obstacles (Bac et al. 2016).

7.6 Case Study: Robotic Apple Harvester

This case study is based on Silwal et al. (2017) that described individual components, an integrated system, and field evaluation results for a single-manipulator, single end-effector robotic system developed at Washington State University. Silwal et al. (2017) reported the design and field evaluation of a robotic apple harvester. The approach adopted was to use a low-cost system to assess required sensing, planning, and manipulation functionality in a modern orchard system with a planar canopy.

7.6.1 Mechanical Design

Based on orchard parameters and 0.94 m bed height of the mobile platform (John Deere, Moline, IL, USA) used for field studies, the third and fourth trellis wires of a modern apple orchard (Fig. 7.3) were selected as the working environment for the mechanical design.

The intent was to assess the overall approach and determine what additional functionality was needed, if any, before expanding the system to the entire tree. The shaded regions in Fig. 7.3 show the working environment in relation to tree geometry. The robotic harvester's mechanical design includes a custom, serial link manipulator with seven DOFs and a grasping end-effector.

7.6.1.1 Manipulator

To expand the number of permissible end-effector orientations at the target fruit's position and not constrain the approach path (e.g., horizontal only), a manipulator task space $\mathcal{T} \in SE(3)$ was desired. Monte Carlo simulations were conducted to determine link lengths for a six-DOF conceptual model. MATLAB's (MathWorks

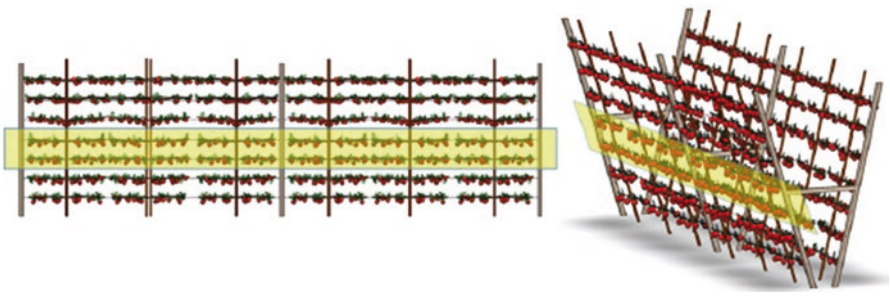


Fig. 7.3 Two different views of the system's working environment (shaded regions) in relation to tree structures

Inc., Natick, MA) random number function was used to generate a 100,000 × 1 vector of uniformly distributed random numbers for each joint within its respective joint limits. For each row of joint coordinates, the manipulator's forward kinematics was then used to determine the position of the end-effector. Simulation results were used to graphically verify that the resulting workspace bounded the region between the third and fourth trellis wires (Fig. 7.3) in the vertical direction. Further lab studies with the fabricated six-DOF arm and a replica apple tree (Davidson et al. 2016a) indicated the need for a seventh DOF.

Unlike a vertical orchard system, in the V-trellis system selected for field studies, apples on the far side of the canopy plane are not accessible from the adjacent row. A prismatic base was added to extend the depth of the reachable workspace into the tree canopy. The kinematic redundancy provided by the seventh DOF was used to help the manipulator avoid singularities during fruit picking. The kinematic model of the serial link manipulator is provided in Fig. 7.4.

The six-DOF manipulator is fastened to a prismatic base.

Table 7.1 provides the picking manipulator's DH parameters and physical joint limits where $\boldsymbol{a}$ is the link length, α is the link twist, d is the link offset, and θ is the joint angle.

7.6.1.2 End-Effector

The fruit removal technique selected during this research incorporates a grasping end-effector. An advantage of a grasping approach is control over the end-effector workspace dimensions. Unlike a vacuum and funnel design with a constrained

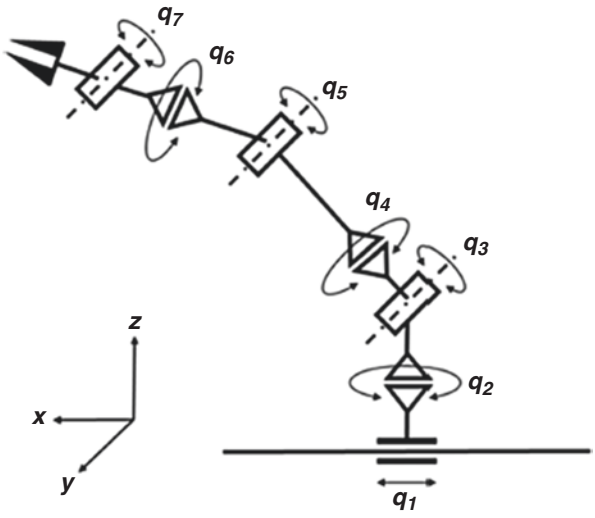


Fig. 7.4 Kinematic configuration and DH parameters of the custom manipulator

Table 7.1 DH parameters and physical joint limits of the picking manipulator

Joint	a (m)	α (°)	d (m)	θ (°)	Lower joint limit	Upper joint limit
1	0	−90	d_1	0	−0.21 m	0.21 m
2	0	90	0.189	θ_2	−90°	90°
3	0	90	0	θ_3	−270°	−135°
4	0	90	0.27	θ_4	−90°	90°
5	0	90	0	θ_5	−200°	−75°
6	0	90	0.231	θ_6	−270°	−95°
7	0.198	90	0	θ_7	0°	180°

opening span, for example, there is the potential for incorporating grasp planning for apples in especially cluttered environments, such as clustered fruit or those adjacent to obstacles. Development of the end-effector and fruit-picking motion was facilitated by a recent study of the hand-picking process (Davidson et al. 2016b). All information about the analysis, design, and fabrication process used to build the end-effector can be found in recent papers (Davidson and Mo 2015; Silwal et al. 2017).

The end-effector does not contain any sensors, and grasping is executed in a purely open-loop manner with feedforward control. Each finger actuator is preset to a torque value with a limiting stall current. Finger motion stops when the actuator's encoder indicates that the motor has stalled. Because the actuator is non-back drivable, reducing the motor current to a small amount is sufficient to ensure that tendon tension is maintained. After a grasp is initiated, the system waits for a small amount of time assuming that static equilibrium is reached with the grasped fruit. To release the apple, the direction of motor rotation is reversed and the finger is returned to a preset open position. This control mode was selected to reduce design costs and complexity as well as increase manipulation speed.

A pneumatic end-effector with bellows-type inflatable actuators for fingers was also investigated (Hohimer et al. 2019) as a possible approach to reduce the overall weight of the end-effector and improve robustness of the fingers when colliding with undetected branches near the fruit. This approach offers the ability to remove weight from the end of the manipulator thereby reducing the torque required of each motor in the kinematic chain but adds to the overall footprint of the systems with the need for a pneumatic air supply. It also attempts to capitalize on the advantages of soft more compliant systems to reduce damage to tree, fruit, and robot.

7.6.2 Harvesting Cycle Activities

After fruit localization, all apples identified in the scene are prioritized so as to optimize the picking sequence. The system then determines manipulator joint solutions for a set of waypoints along the path to each fruit. The end-effector's path to and

from each apple is linear. A limitation of the design used during field studies is that the potential provided by kinematic redundancy was not fully exploited.

7.6.2.1 Apple Prioritization

In order to maximize the efficiency of robotic arm's movement during harvesting, the identified apples had to be prioritized in a logical sequence. This research considered the task of apple prioritization during each harvesting cycle as a traveling salesman problem (TSP). The TSP is a well-known optimization problem in the class of non-deterministic polynomial-time hard (NP-hard) problems. In brief, the TSP aims to find a route from a known location that visits a pre-described set of locations, visiting each only once, and returns to the original location in such a way that the total traveling distance is minimum. The nearest neighbor algorithm (Kizilateş and Nuriyeva 2013) was used to solve the TSP where distances between apples were calculated using the 3D Euclidean distance equation with respect to the manipulator's starting location at the beginning of a harvesting cycle.

7.6.2.2 Path Planning

Path planning for a single fruit pick is described in this section. The origin of the world reference frame O (a right-hand coordinate frame) with unit vectors $\mathbf{i}$, $\mathbf{j}$, $\mathbf{k}$ is located at the arm's base where it is centered over the mobile platform (Fig. 7.5).

The world x-axis is oriented along the length of the mobile platform normal to the tree canopy, and the world z-axis is vertical. Using pure translation, the fruit's position vector is transformed from the camera's reference frame to the world reference frame O. Let $\mathbf{p} = (a + d)\mathbf{i} + b\mathbf{j} + c\mathbf{k}$ be the vector of coordinates of the fruit's center with respect to frame O where d is displacement of the prismatic base along the x-axis. At the beginning of inverse kinematics (IK) planning, the arm's base frame B with unit vectors $\mathbf{x0}$, $\mathbf{y0}$, $\mathbf{z0}$ (Denavit and Hartenberg 1955) is collocated and aligned with the world reference frame O (i.e., d is equal to 0). The motion planning algorithm determines the six-DOF arm's IK using the dual optimization technique proposed by Wang and Chen (1991). In brief, the cyclic coordinate descent (CCD) method (Luenberger and Ye 2008) is used to rapidly find a joint vector $\mathbf{q}$ near the true solution, which is then used as input to the Broyden-Fletcher-Goldfarb-Shanno (BFGS) variable metric method (Luenberger and Ye 2008) to obtain a solution at the desired degree of precision. The joint limits of the manipulator are used as boundary constraints, and the convergence tolerance of the objective function is set at 1E-6. For each apple in the harvesting cycle, IK solutions are found for three positions, the approach point, grasp point, and release point. The prismatic base is only displaced if the IK solver fails to converge to a solution for the desired end-effector position or if the manipulator's configuration at the apple's position is near a singularity. The algorithm then performs a search along the axis of the prismatic

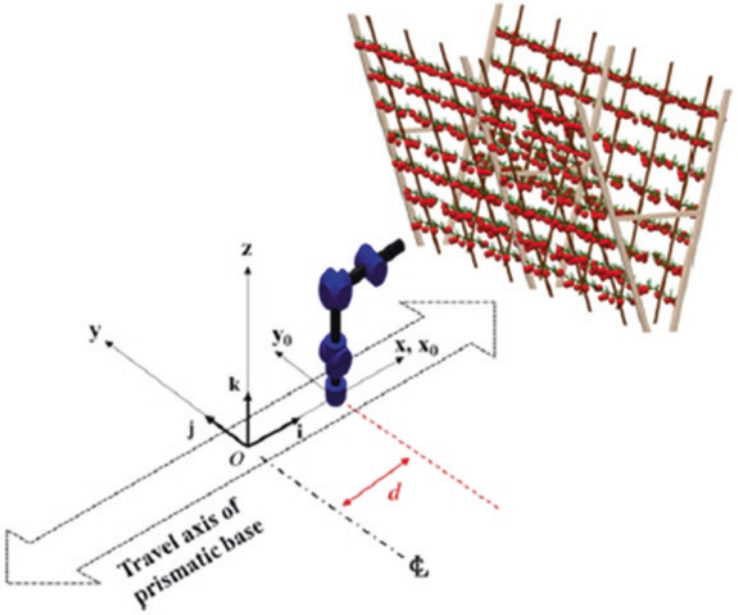


Fig. 7.5 The world reference frame of the autonomous harvesting system is located at the prismatic base's centerline

base to find a displacement *d* that leads to an IK solution, if any. The end-effector's path and velocity between the three positions are specified in the operational space.

7.6.2.3 Approach to Fruit

The approach point is 15 cm from the fruit. At the approach position, the orientation of the end-effector's normal vector $\boldsymbol{n}$ is partially defined by an azimuth angle α, which is the angle between the x-axis and the projection of **p** in the x-y plane (Fig. 7.6). So, the equation for the azimuth angle α is

$$\alpha = \tan^{-1}\left(\frac{b}{a+d}\right) \tag{7.3}$$

The path the end-effector follows to the apple depends on fruit height. If *c* is less than 0.55 meters, the end-effector makes a horizontal approach to the fruit. To expand the number of apples that can be picked, the end-effector follows a 45° inclined path to fruit that are vertically higher than 0.55 meters. For the horizontal approach, the rotation matrix $\mathbf{U}^{\mathbf{E}}$ relating orientation of the end-effector's frame *E* with respect to the manipulator's base frame *B* is obtained via the elementary rotation matrix about the z-axis $\mathbf{R}_z(\alpha)$. For the inclined case, $\mathbf{U}^{\mathbf{E}}$ is found by composition of elementary rotations about the y and z coordinate axes

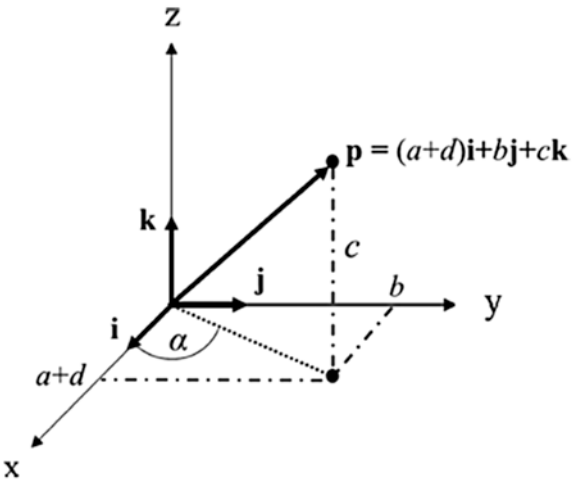


Fig. 7.6 The azimuth angle partially defines the orientation of the end-effector's normal vector at the approach position

$$\mathbf{U}^{\mathbf{E}} = \mathbf{R}_z(\alpha)\mathbf{R}_y(-45^\circ) \tag{7.4}$$

7.6.2.4 Fruit Grasp

Before selecting the joint vector $\mathbf{q}$ required for the end-effector to reach the apple, the motion planning algorithm first checks the kinematic configuration of the arm at the fruit's position. The geometric Jacobian ($\mathbf{q}$) of the differential kinematics equation defines a linear mapping (Siciliano et al. 2010)

$$\mathbf{v}_e = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{7.5}$$

between the arm's 6 × 1 vector $\dot{\mathbf{q}}$ of joint velocities and the 6 × 1 vector $\mathbf{v}_e = [\mathbf{l}_e\ \omega_e]^T$of linear $\mathbf{l}_e$ and angular end-effector velocities ω_e in the arm's base frame B. Because end-effector velocity from the approach point to the grasping point is defined relative to the end-effector frame E, the algorithm first computes the geometric Jacobian $\mathbf{J}^{\mathbf{E}}(\mathbf{q})$ in frame E using the relative rotation matrix $\mathbf{U}^{\mathbf{E}}$ whereby

$$\mathbf{J}^{\mathbf{E}}(\mathbf{q}) = \begin{bmatrix} \mathbf{U}^{\mathbf{E}} & 0 \\ 0 & \mathbf{U}^{\mathbf{E}} \end{bmatrix} \mathbf{J}(\mathbf{q}) \tag{7.6}$$

The motion planning algorithm then checks the condition number of the 6 × 6 matrix $\mathbf{J}^{\mathbf{E}}$. For an ill-conditioned Jacobian $\mathbf{J}^{\mathbf{E}}$, the algorithm updates the prismatic base position in an iterative fashion until the arm's configuration is optimized to produce higher end-effector velocities at the apple's position. The end-effector's velocity from the approach to apple position is 0.15 m/s along its normal vector $\mathbf{n}$, the orientation of which was defined by the azimuth angle α. So, for the horizontal and inclined approach, $\mathbf{v}_e^{\mathbf{E}} = [0.1500000]^T$ (m/s) where $\mathbf{v}_e^{\mathbf{E}}$ is the end-effector

velocity expressed in the end-effector frame *E*. The required joint velocities $\dot{\mathbf{q}}$ are found with the inverse differential kinematics equation using the vector of joint angles of the manipulator's configuration when the end-effector is at the approach position

$$\dot{\mathbf{q}} = \left(\mathbf{J}^{\mathbf{E}}\left(\mathbf{q}\right)\right)^{-1} \mathbf{v}_{\mathrm{e}}^{\mathbf{E}} \tag{7.7}$$

The algorithm used for computation of the system's kinematics at the grasping position is shown below. The initial base position *d* is the displacement of the prismatic base at the approach position. End-effector orientation and velocity are determined by the azimuth angle and fruit height. If a solution of real joint variables and a well-conditioned geometric Jacobian are not found, the base position is incremented alternating between positive and negative displacements. Between each iteration, the algorithm checks for acceptable results. Should a solution of real joint variables and well-conditioned Jacobian not be found within the range of the mobile base, which is approximately 0.44 m, the fruit is excluded from the picking cycle.

Algorithm 1.

Input: Fruit position **p**; relative rotation matrix $\mathbf{U}^{\mathbf{E}}$ for desired end-effector orientation; desired end-effector velocity $\mathbf{v}_{\mathbf{e}}^{\mathbf{E}}$; initial base position *d*

Output: 6×1 vector of joint angles **q**; 6×1 vector of joint velocities $\dot{\mathbf{q}}$; final base position *d*;

```
temp = 1;
for i = 1:12
    Call IK solver(p, U^E) → q
    Call FK solver(q, v_e^E) → J^E and q̇
    if q ∈ ℝ && cond(J^E) <= 40
        break;
    end
    d = temp*(.01*i);
    temp = -temp;
end
```

7.6.2.5 Release of Fruit

After grasping the apple, the manipulator moves the end-effector 12 cm back from the tree along a path coincident with the end-effector's normal vector **n** originally defined by the azimuth angle and fruit height. Selection of the 12 cm displacement was based on previous studies of fruit-picking dynamics (Davidson et al. 2016b) which showed that mean displacement to fruit separation from the onset of picking was approximately 7 cm. The end-effector's linear velocity $\mathbf{l}_e$ along the path to the release point has the same magnitude as the linear velocity from the approach position to the grasping position. During fruit detachment, there is also a 45° rotation (0.6 rad/s) around **n** in order to produce a combined pulling and twisting motion. The rotation matrix $\mathbf{U}_{\mathbf{R}}^{\mathbf{E}}$ relating orientation of the end-effector's frame *E* to the manipulator's base frame *B* at the release point is therefore

$$\mathbf{U}_{\mathbf{R}}^{\mathbf{E}} = \mathbf{R}_{z}(\alpha)\mathbf{R}_{x}^{\mathbf{E}}(-45^{\circ}) \tag{7.8}$$

for the horizontal approach/removal and

$$\mathbf{U}_{\mathbf{R}}^{\mathbf{E}} = \mathbf{R}_{z}(\alpha)\mathbf{R}_{y}(-45^{\circ})\mathbf{R}_{x}^{\mathbf{E}}(-45^{\circ}) \tag{7.9}$$

for the inclined approach/removal where $\mathbf{R}_{x}^{\mathbf{E}}$ is the elementary rotation matrix about the x-axis of frame *E* (i.e., the end-effector's normal vector **n**). The position and orientation at the release point are passed to the IK solver for calculation of the required manipulator joint angles **q**.

7.6.2.6 Field Test Results

Figure 7.7 shows the setup used during field testing of the robotic system. Of the 193 apples identified, 150 of the fruit were in the system's reachable workspace and were selected as the harvesting samples in this study.

Fig. 7.7 The robotic platform with the arm attached to end-effector, global camera system, and computer during field testing

The picking efficiency for the apples the system attempted to pick was 85% (127/150). Post inspection of the 127 fruit successfully harvested revealed no obvious evidence of bruising or surface damage. Because stem retention is desirable for fresh market apples, after each harvesting cycle, all apples were inspected to determine whether the stem was intact (86/127), had pulled out of the stem cavity (8/127), or had the spur attached (33/127).

One of the major objectives of this research was to use well-defined performance criteria and segregate and report execution time of each major task individually. Figure 7.8 displays the contents of timing parameters for all major procedures. Vision timing included the sum of all listed procedures including apple identification, inverse mapping, and depth filtering.

On average, 6.1 s was required to identify and localize all apples in an image, equivalent to 1.5 s of computation time per apple. IK time was measured as the time required to calculate joint velocities and angles for each path point for all apples as well as filter apples beyond the workspace. The mean computation time for path planning was 0.1 s per fruit. Picking time describes physical manipulation only and

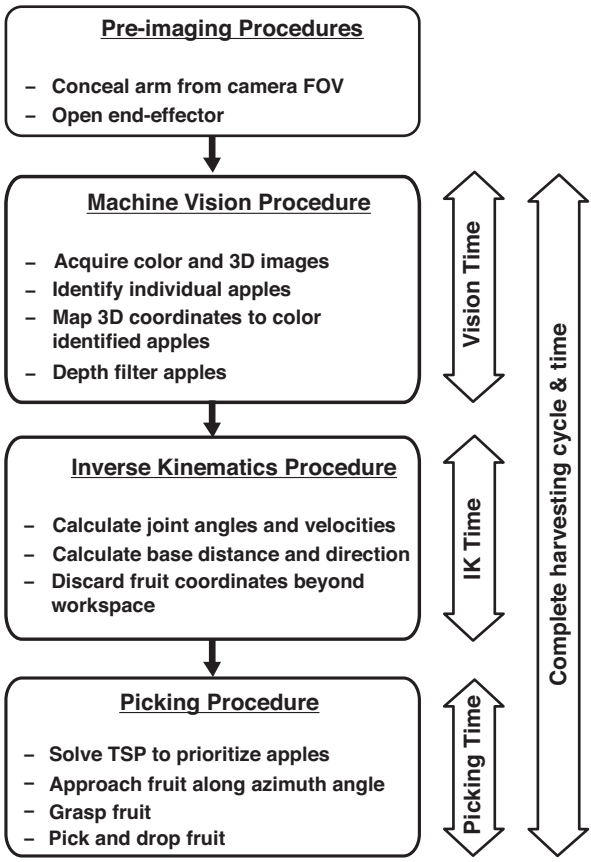


Fig. 7.8 Time required to complete each procedure. Total cycle time includes vision processing, path planning computations, and physical manipulation of the fruit

includes the time required to approach, grasp, remove, and drop an individual apple. The system's average picking time was 6.0 s. The total cycle time was the summation of vision processing, path planning computations, and manipulation of the fruit. Including the average vision time of 1.5 s, the total cycle time required to harvest a single apple was approximately 7.6 s. It should be noted that the cycle time reported here does not include the time required to reposition the vehicle between harvesting cycles. Because the largest movement was usually from the first apple in a cycle to the manipulator's waiting position, the average picking time for the first fruit was significantly higher than the remaining apples in a cycle. Table 7.2 compares the picking time of the first fruit in a cycle with the remaining fruit in that cycle.

7.6.3 Workspace Analysis for an Eight-DOF Apple-Picking Robot

After completing the first iteration of field studies, an additional prismatic joint was added to the robot in order to expand the system's workspace and increase the number of reachable fruit per harvesting cycle. The picking manipulator, which is shown in Fig. 7.9, has eight DOFs. The picking end-effector design is the same as that described in Silwal et al. (2017). In addition to the DH parameters and physical joint limits of the picking manipulator in Table 7.1, the parameter and joint limit for the y-axis prismatic joint are provided in Table 7.3.

7.6.3.1 Inverse Kinematics

With a *configuration space* of $\mathbf{q} \in R^8$ and workspace of $\mathbf{x_e} \in R^6$, the picking manipulator is kinematically redundant. For this work, the picking manipulator's inverse kinematics planner was revised. The inverse differential kinematics problem for the redundant manipulator is considered a constrained linear optimization problem. Using Lagrange multipliers to minimize a cost function, Siciliano et al. (2010) show that a solution to the inverse differential kinematics problem for an n DOF redundant manipulator is

Table 7.2 Comparison of average picking time, which measures physical manipulation during an apple pick, by cycle position

	Average Picking Time (s)	Standard Deviation (σ)
Overall	6.0	0.4
1st apple in a cycle	6.2	0.6
Remaining apples in a cycle	5.8	0.4

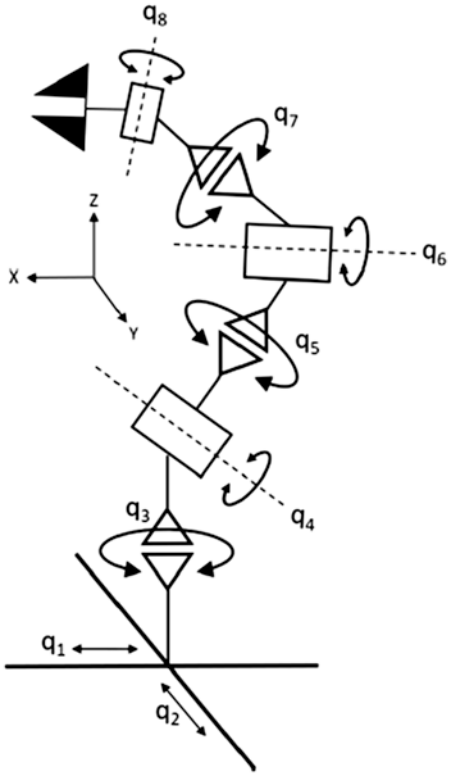


Fig. 7.9 Kinematic model of the eight-DOF configuration

Table 7.3 The parameter and joint limit for the y-axis prismatic joint

Joint	a (m)	α (°)	d (m)	θ (°)	Lower joint limit	Upper joint limit
2	0	90	d_2	−90	−0.34 m	0.34 m

$$\dot{\mathbf{q}} = \mathbf{J}^{\#}\mathbf{v}_{\mathbf{e}} + \left(\mathbf{I}_{\mathbf{n}} - \mathbf{J}^{\#}\mathbf{J}\right)\dot{\mathbf{q}}_{\mathbf{0}} \tag{7.10}$$

where $\dot{\mathbf{q}}$ is the n x 1 vector of joint velocities, $\mathbf{v}_{\mathbf{e}}$ is the 6 x 1 vector of the end-effector's linear and angular velocities, $\mathbf{J}$ (6 x n matrix) is the manipulator's geometric Jacobian, $\mathbf{I}_{\mathbf{n}}$ is the n x n identity matrix, $\dot{\mathbf{q}}_{\mathbf{0}}$ is an arbitrary n x 1 vector, and the matrix

$$\mathbf{J}^{\#} = \mathbf{J}^{\mathbf{T}}\left(\mathbf{J}\mathbf{J}^{\mathbf{T}}\right)^{-1} \tag{7.11}$$

is the right pseudo-inverse of $\mathbf{J}$. The matrix $(\mathbf{I}_{\mathbf{n}} - \mathbf{J}^{\#}\mathbf{J})$ is a null space projection matrix of $\mathbf{J}$. Therefore, the vector $\dot{\mathbf{q}}_{\mathbf{0}}$ can be chosen so as to satisfy secondary constraints, such as maximizing manipulability or the distance from an obstacle, without changing the end-effector's position and orientation $\mathbf{x}_{\mathbf{e}}$. For this work, $\dot{\mathbf{q}}_{\mathbf{0}}$ is specified as

$$\dot{\mathbf{q}}_0 = k\left(\frac{\partial w(\mathbf{q})}{\partial \mathbf{q}}\right)^{\mathrm{T}} \tag{7.12}$$

where k is a gain factor and w is an objective function. Similar to the approach used in earlier studies (Baur et al. 2012; Chaumette and Marchand 2001; Schuetz et al. 2014; Liegeois 1977), in this work, the objective function w includes a piecewise, polynomial function that penalizes joint positions $\mathbf{q}$ near their physical limits

$$w = \begin{cases} \sum_{i=1}^{n} \frac{\left(q_i - \overline{q}_{M,i}\right)^2}{\left(q_{M,i} - \overline{q}_{M,i}\right)^2} & \text{if } q_i > \overline{q}_{M,i} \\ \sum_{i=1}^{n} \frac{\left(q_i - \overline{q}_{m,i}\right)^2}{\left(q_{m,i} - \overline{q}_{m,i}\right)^2} & \text{if } q_i < \overline{q}_{m,i} \\ 0 & \text{else} \end{cases} \tag{7.13}$$

where $q_{M,\,i}$ ($q_{m,\,i}$) is the maximum (minimum) joint limit and $\overline{q}_{M,i}$ ($\overline{q}_{m,i}$) are joint thresholds. Thus, the picking manipulator's inverse kinematics algorithm uses kinematic redundancy for joint limit avoidance. To calculate the joint positions $\mathbf{q}$, Eq. (7.10) is numerically integrated in discrete time using the Euler method. A first-order, closed-loop inverse kinematics algorithm (Siciliano et al. 2010) that accounts for the *operational space error* $\mathbf{e}$ between the desired end-effector position and orientation $\mathbf{x_d}$ and actual end-effector position and orientation $\mathbf{x_e}$ was used to compensate for numerical drift. A time step of 0.01 sec was shown to produce accurate results with minimal position and orientation error at a sufficient computation time. All path planning and inverse kinematics calculations were completed in Microsoft Visual Studio C++ using the open-source Eigen linear algebra library.

7.6.3.2 Workspace Analysis

The workspace, i.e., the reachable region, of the eight-DOF manipulator can be calculated by the simulations of manipulator movement with specified joint values (Wang et al. 2018; Wang 2018). Each of the eight joints of the manipulator has specific limits on its movement. Seven points were chosen within each of the two prismatic joints' limits, and five points were chosen within each of the six revolute joints' limits, to obtain 765,625 grids of chosen joint parameters. Forward kinematics function is called to calculate the x, y, and z coordinates of the end-effector location from those joint parameters. All these positions are considered as reachable locations of the arm. Figure 7.10 depicts the eight-DOF picking arm's total workspace (pink-colored shape), and blue dots represent apple coordinates obtained

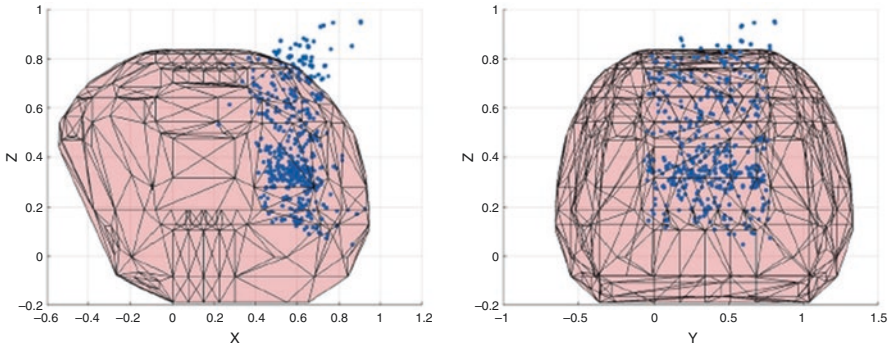


Fig. 7.10 Eight-DOF picking arm's workspace (Wang et al. 2018)

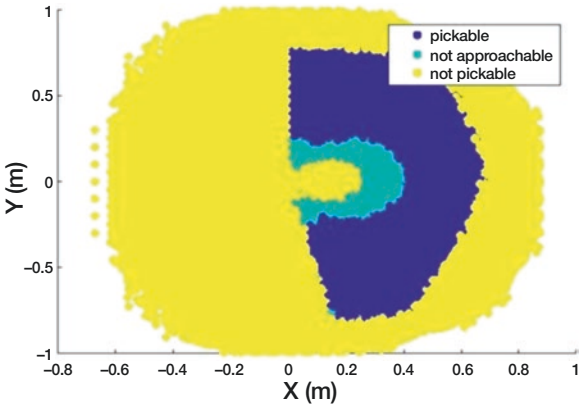


Fig. 7.11 Simulated workspace of the eight-DOF robotic arm; 765,625 points were plotted; yellow is not pickable, cyan is not approachable but pickable, and blue is pickable. (Wang 2018)

from field trial. All the simulation work was conducted in MATLAB (MathWorks Inc., USA) with the Peter Corke's Robotics Toolbox (Corke 2017).

It can be decided if these reachable locations of the manipulator can be reached by the inverse kinematics function. The algorithm that is used to decide if an apple is "pickable" is called to decide if all those supposedly reachable locations are sites of pickable apples. In that algorithm, the inverse kinematics function is called twice, once to see if the apple location is reachable and once for the approach position which is 15 cm away from the final location along the end-effector normal, to see if the approach position is also reachable. The reachability of each of these points is categorized accordingly (using the inverse kinematics solver): 1, if the final position cannot be reached; 2, if the final point is reachable but the approach position is not reachable; and 3, if both locations are reachable. Then all the points are plotted with three different colors to differentiate the three different situations. A cross section of the workspace at 0.4 m above the origin of the arm shows these three different situations (Fig. 7.11).

If the eight-DOF picking arm is changed to the five-DOF arm (Fig. 7.12), the workspace is shrunk as shown in Fig. 7.13.

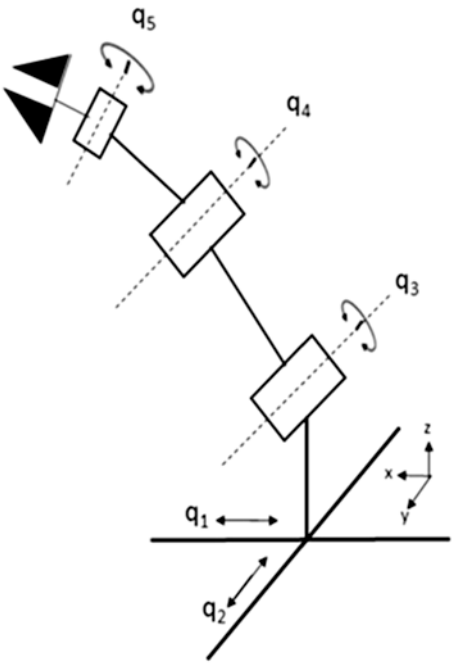


Fig. 7.12 Kinematic model of the five-DOF arm. (Hohimer et al. 2019)

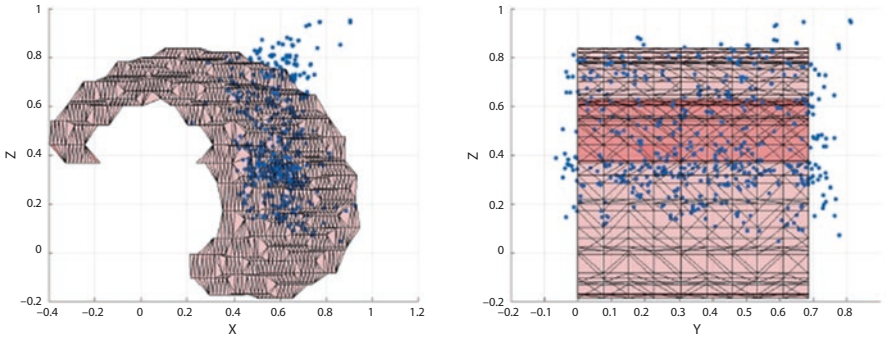


Fig. 7.13 Five-DOF picking arm's workspace. (Wang et al. 2018)

For all the 505 sets of converted apple coordinates, the eight-DOF arm can reach 76% of the apples, whereas the five-DOF arm can reach 63%. According to this workspace analysis, the absolute volume of the workspace is not as important as the effective workspace which is the overlap between the robot's workspace and the fruit space. Because of the unstructured environment of the apple orchard, the robot needs the ability to adjust to the fruit space to maximize the overlap. More details on optimizing the overlap are in Wang et al. (2018). By incorporating an adjustment algorithm that accounts for changes in the environment, the five-DOF configuration's performance is increased by 49% (reaching 94% of the apples) without any

increase in the actual workspace volume. Any increase in efficiency is important to make the system more economical (Wang et al. 2018).

7.6.4 *Optimization Process for the Robotic Apple Harvesting System as the Whole System*

The "whole system" does not only refer to the robotic arm itself but also the camera location, the size of the fruiting wall, and the time required to pick the whole fruiting wall (Wang et al. 2018; Wang 2018).

Given a fruiting wall of certain length and height and required time to pick all apples on that fruiting wall, we should be able to design a low-cost system that finishes the task within the required time. This is an optimization problem where we need to determine the optimal number and size of frames to divide the fruiting wall into, both vertically in the z-axis and horizontally along the y-axis.

In this system, assuming the apples are evenly distributed, then we have the following:

- The number of frames determines the size of each frame.
- The size of each frame determines the camera's field of view.
- The camera's field of view determines the optimal configuration of the picking arm.
- The configuration of the arm determines the time needed for each frame.
- The time needed for each frame plus the time needed to move between frames determines how many frames each arm can pick within the required time.
- The number of frames each arm can pick determines the number of arms needed for the whole fruiting wall.
- The number of arms needed and the arm's configuration together determines the total cost of the whole system.

All the above are variables in this optimization problem and can be symbolized and calculated in a function. The objective here is to minimize cost and finish picking the fruiting wall within the required time. The smaller the number of frames, the larger the frame size is. Then it takes longer for the arm to finish each frame but saves the time needed to move between frames. Thus, the total number of arms could be reduced, but the cost of each arm could be higher since it needs to pick a larger area. This problem might be solved iteratively. Several optimal or suboptimal solutions to this problem can be simulated to calculate the actual cost and compared.

7.7 Summary and Concluding Thoughts

The design principles and considerations for robotic manipulation and optimization for field agricultural applications are briefly discussed in this chapter. A case study with a robotic apple harvester along with workspace analysis is also introduced to provide a sequence of a design process including subsystems, an integrated system, sensing, planning, and manipulation functionality in a modern fruiting wall orchard. The manipulation and optimization of agricultural robotic systems for the highly unstructured environment is a difficult challenge, although there exists a clear need for the technology in today's economy. According to the field testing of a robotic system designed to harvest fresh market apples, horticultural practices play a critical role in the selection of functionality requirements.

Bloch et al. (2015) proposed that the robot would need be optimally designed for a specific task in a specific working environment to improve robot performance for agricultural tasks and decrease its cost. They also suggested the environment itself should also be optimized for optimal robot performance since the environment defines the robot's optimal kinematics (Bloch et al. 2018).

It is noted that the overlap of the manipulator's workspace and the target objects' space (e.g., fruit distribution in canopies for robotic apple harvesting) is the key to increase the reachability instead of only increasing the volume of arm's workspace (Wang et al. 2018). As such, a comprehensive workspace analysis should be preceded for optimal manipulator kinematic design. According to our prior work (Wang et al. 2018), the eight-DOF picking arm showed a bigger workspace than that of five-DOF arm; however, optimal kinematic design and motion planning made it possible to reach as many apples as the eight-DOF arm can reach. Also, robotic system engineers should collaborate with growers and horticulturalists in development of autonomous robotic systems for precision agriculture.

Noting that our camera and arm's origin are fixed together on the utility vehicle, there are challenges compared to the greenhouse environment. In greenhouse, the heating pipes are fixed, so the robotic manipulation can be optimized for the fixed distance between the robot and the plant stem. The problem lies in that every time the utility vehicle is parked, it cannot guarantee that the distance from the vehicle to the fruiting wall is optimal so that the maximum number of crops would be in the reachable range of the arm. This leads to some of the crops being either too close or too far in the view of the robotic arm. An autonomous vehicle with a LiDAR (Light Detection and Ranging) sensor or a visual camera may be used to keep the optimal distance from the fruiting wall.

For the case study (Sect. 7.6), for example, it would be ideal to be able to figure out an optimal distance from the vehicle to the fruiting wall at each vehicle stop to reach the maximum number of apples. In other words, to reach the maximum number of apples at each vehicle stop, we need to make the workspace of the arm include as many apples as possible, i.e., we need to move the vehicle around to move the arm's origin (Wang et al. 2018; Wang 2018).

From the workspace analysis (Wang et al. 2018; Wang 2018), we also can see that the eight-DOF configuration has larger volume, but most of this volume is wasted due to the shape of the fruit space when we superimpose them. The five-DOF showed less volume as expected, but its performance is not affected dramatically as long as we can optimize its overlap with the fruit space. Therefore, it is desirable to have a system that will be able to maximize its coverage over the fruit space but minimize any useless workspace. It should also save in terms of operation with less energy wasted on maintaining unnecessary workspace.

The design of the robot should match a specific task in a specific environment to be optimal (Bloch et al. 2018). According to their results, a three-DOF prismatic joint would satisfy such task since its workspace is rectangular prism shaped. Vougioukas et al. (2016) mathematically verified that 91% fruit in an orchard can be reached by linear-only motion. In order to be more efficient or implement hierarchical picking, we can actually install multiple vertical rails with x-axis arm on the y-axis rail, so we can start picking at multiple locations in a row or execute picking in a hierarchical approach, i.e., the first arm goes to pick first, and a second arm can go again to pick what's left and later a third arm as desired. This might be an approach to tackle the problem of fruit clustering in addition to implement obstacle and cluster detection in the algorithm (Wang et al. 2018; Wang 2018).

References

Bac CW, van Henten EJ, Hemming J, Edan Y (2014) Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. J Field Robot 31(6):888–911

Bac CW, Roorda T, Reshef R, Berman S, Hemming J, van Henten EJ (2016) Analysis of a motion planning problem for sweet-pepper harvesting in a dense obstacle environment. Biosyst Eng 146:85–97

Baur JT (2015) Agricultural manipulators: simulation, design and motion planning, PhD dissertation, Technical University of Munich (TUM)

Baur J, Pfaff J, Ulbrich H, Villgrattner T (2012) Design and development of a redundant modular multipurpose agricultural manipulator. In: 2012 IEEE/ASME international conference on Advanced Intelligent Mechatronics (AIM), pp 823–830. IEEE

Bechar A, Vigneault C (2016) Agricultural robots for field operations: concepts and components. Biosyst Eng 149:94–111

Berenson D, Srinivasa SS, Ferguson D, Collet A, Kuffner JJ (2009) Manipulation planning with workspace goal regions. In: 2009 IEEE international conference on robotics and automation, pp 618–624. IEEE

Bloch V, Degani A, Bechar A (2015) Machinery robotics and PA technologies (in section 4), precision agriculture '15, Wageningen Academic Publishers, pp 313–320

Bloch V, Degani A, Bechar A (2018) A methodology of orchard architecture design for an optimal harvesting robot. Biosyst Eng 166:126–137

Ceccarelli M, Carbone G, Ottaviano E (2005, March) An optimization problem approach for designing both serial and parallel manipulators. In: Proc. of MUSME 2005, the Int. Sym. on Multibody Systems and Mechatronics, pp 6–9

Chaumette F, Marchand É (2001) A redundancy-based iterative approach for avoiding joint limits: application to visual servoing. IEEE Trans Robot Autom 17(5):719–730

Choset HM, Hutchinson S, Lynch KM, Kantor G, Burgard W, Kavraki LE, Thrun S (2005) Principles of robot motion: theory, algorithms, and implementation. MIT Press

Corke P (2017) Robotics, vision and control: fundamental algorithms in MATLAB® second, completely revised, vol 118. Springer

Davidson JR, Mo C. (2015, November) Mechanical design and initial performance testing of an apple-picking end-effector. In: ASME 2015 International mechanical engineering congress and exposition. American Society of Mechanical Engineers Digital Collection

Davidson JR, Silwal A, Hohimer CJ, Karkee M, Mo C, Zhang Q (2016a, October) Proof-of-concept of a robotic apple harvester. In: 2016 IEEE/RSJ international conference on Intelligent Robots and Systems (IROS), pp 634–639. IEEE

Davidson J, Silwal A, Karkee M, Mo C, Zhang Q (2016b) Hand-picking dynamic analysis for undersensed robotic apple harvesting. Trans ASABE 59(4):745–758

Denavit J, Hartenberg R (1955) A kinematic notation for low pair mechanisms based on matrices. ASME J Appl Mech 22:215–221

Diankov R, Ratliff N, Ferguson D, Srinivasa S, Kuffner J (2008) Bispace planning: concurrent multi-space exploration. Proceedings of Robotics: Science and Systems IV, 63

Hauser K, Ng-Thow-Hing V (2010, May) Fast smoothing of manipulator trajectories using optimal bounded-acceleration shortcuts. In: 2010 IEEE international conference on robotics and automation, pp 2493–2498. IEEE

Hohimer CJ, Wang H, Bhusal S, Miller J, Mo C, Karkee M (2019) Design and field evaluation of a robotic apple harvesting system with a 3D-printed soft-robotic end-effector. Trans ASABE 62(2):405–414

Kalakrishnan M, Chitta S, Theodorou E, Pastor P, Schaal S (2011) STOMP: stochastic trajectory optimization for motion planning. In: 2011 IEEE international conference on robotics and automation, pp 4569–4574. IEEE

Kizilateş G, Nuriyeva F (2013) On the nearest neighbor algorithms for the traveling salesman problem. In: Advances in computational science, engineering and information technology. Springer, Heidelberg, pp 111–118

Klein CA, Blaho BE (1987) Dexterity measures for the design and control of kinematically redundant manipulators. Int J Robot Res 6(2):72–83

Kondo K (1991) Motion planning with six degrees of freedom by multistrategic bidirectional heuristic free-space enumeration. IEEE Trans Robot Autom 7(3):267–277

Kuffner JJ, LaValle SM (2000, April) RRT-connect: an efficient approach to single-query path planning. In: Proceedings 2000 ICRA. Millennium Conference. IEEE international conference on robotics and automation. Symposia proceedings (Cat. No. 00CH37065), Vol. 2, pp 995–1001. IEEE

LaValle SM (2006) Planning algorithms. Cambridge University Press

Liegeois A (1977) Automatic supervisory control of the configuration and behavior of multibody mechanisms. IEEE Trans Syst Man Cybern 7(12):868–871

Luenberger DG, Ye Y (2008) Duality. In: Linear and nonlinear programming. Springer, New York, pp 79–110

Lynch KM, Park FC (2017) Modern robotics. Cambridge University Press

Murray RM, Li Z, Sastry SS, Sastry SS (1994) A mathematical introduction to robotic manipulation. CRC Press

Nocedal J, Wright SJ (1999) Numerical optimization. Springer, New York/Berlin/Heidelberg

Paredis CJ, Khosla P (1995) Design of modular fault tolerant manipulators

Pearl J (1984) Intelligent search strategies for computer problem solving. Addison-Wesley

Ratliff N, Zucker M, Bagnell JA, Srinivasa S (2009, May) CHOMP: gradient optimization techniques for efficient motion planning. In: 2009 IEEE international conference on robotics and automation, pp 489–494. IEEE

Righetti L, Kalakrishnan M, Pastor P, Binney J, Kelly J, Voorhies RC et al (2014) An autonomous manipulation system based on force control and optimization. Auton Robot 36(1–2):11–30

Rusu RB, Şucan IA, Gerkey B, Chitta S, Beetz M, Kavraki LE (2009, October) Real-time perception-guided motion planning for a personal robot. In: 2009 IEEE/RSJ international conference on Intelligent Robots and Systems, pp 4245–4252. IEEE

Schuetz C, Buschmann T, Baur J, Pfaff J, Ulbrich H (2014, May) Predictive online inverse kinematics for redundant manipulators. In: 2014 IEEE international conference on robotics and automation (ICRA), pp 5056–5061. IEEE

Siciliano B, Sciavicco L, Villani L, Oriolo G (2010) Robotics: modelling, planning and control. London: Springer

Silwal A, Davidson JR, Karkee M, Mo C, Zhang Q, Lewis K (2017) Design, integration, and field evaluation of a robotic apple harvester. J Field Robot 34(6):1140–1159

Spong MW, Hutchinson S, Vidyasagar M (2006) Rigid motions and homogeneous transformations. In: Robot modeling and control, 1st edn. Wiley, Hoboken, pp 29–53

Tenev T, Stoyanov B (2000) On the performance indexes for robot manipulator. Problems of Engineering Cybernetics and Robotics 49:64–71

Van Henten EJ, Hemming J, Van Tuijl BAJ, Kornet JG, Bontsema J (2003) Collision-free motion planning for a cucumber picking robot. Biosyst Eng 86(2):135–144

Van Henten EJ, Van't Slot DA, Hol CWJ, Van Willigenburg LG (2009) Optimal manipulator design for a cucumber harvesting robot. Comput Electron Agric 65(2):247–257

Van Henten EJ, Schenk EJ, Van Willigenburg LG, Meuleman J, Barreiro P (2010) Collision-free inverse kinematics of the redundant seven-link manipulator used in a cucumber picking robot. Biosyst Eng 106(2):112–124

Vougioukas SG, Arikapudi R, Munic J (2016) A study of fruit reachability in orchard trees by linear-only motion. IFAC-PapersOnLine 49(16):277–280

Wang H (2018) Simulation and evaluation of a robotic apple harvesting system (Doctoral dissertation, Washington State University)

Wang LC, Chen CC (1991) A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. IEEE Trans Robot Autom 7(4):489–499

Wang H, Hohimer CJ, Bhusal S, Karkee M, Mo C, Miller JH (2018) Simulation as a tool in designing and evaluating a robotic apple harvesting system. IFAC-PapersOnLine 51(17):135–140

Yoshikawa T (1985) Manipulability of robotic mechanisms. Int J Robot Res 4(2):3–9

Chapter 8
End-Effector Technologies

Qingchun Feng

8.1 Introduction

An end-effector of a robotic system is the part that comes into direct contact with the terminal objects to perform required tasks. As it is the part that is directly engaged with and operates on the objects of interest, which is similar to the human hands, it is also called the robot hand, or sometimes just the gripper. Though there could be a wide variation of end-effectors designed and developed for specific tasks (to be discussed later), an end-effector, commonly, is composed of two or more fingers that can be opened and closed to achieve the desired functions, such as grasping, holding, and rotating objects. Considering the extensive need of robotic technology for labor-intensive tasks such as harvesting, pollination, pruning, spraying, milking, sterilizing, as well as cleaning during animals' breeding, end-effectors for agricultural and field applications are required to handle a great diversity of objects, such as fruit, leaves, flowers, stems, trees, stones, and even animal body parts.

Because end-effectors come into direct contact with the objects being manipulated (or being operated on), the end-effector efficiency greatly determines the performance of a robotic system. Therefore, physical characteristics of the target object (to be manipulated), including its shape, size, surface characteristics, and softness, should be investigated prior to the development of a suitable end-effector. Besides, the biological and chemical properties of the target objects should also be known in order to avoid possible damage to the object as well as to the end-effector during the operation.

Q. Feng (✉)
Beijing Research Center of Intelligent Equipment for Agriculture, National Research Center of Intelligent Equipment for Agriculture, Beijing, China
e-mail: fengqc@nercita.org.cn

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*, Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_8

As an example of long-lasting need of innovation in agricultural robotics, research and development on fresh-market fruit and vegetable harvesting comes in the forefront. Key technological features of a robotic fruit/vegetable harvesting machine essentially consist of acquiring the targets' visual information in a complex agricultural environment, flexible and nondestructive manipulation of biological matter, and integration of the production system by combining the machinery with agronomy/horticulture, which also happen to be the common technical bottlenecks limiting the adoption of current agricultural robotic systems.

The vast array of features related to fresh-market fruit presents some critical challenges in robotic fruit picking, which is also a representative case for various agricultural operations when it comes to application of robotics. Hence in this book chapter, end-effector technologies, including their principles and application, for fruit harvesting robots would be the main focus. The current harvesting robots (that are being developed around the world) are primarily utilized to harvest vegetable and fruit crops in greenhouses, as well as tree fruit crops in natural orchards. Among these crops, vegetable and fruit in greenhouses are mostly characterized by soft surfaces and small sizes. A few examples include tomatoes, strawberries, cucumbers, and eggplants. The tree fruit crops with the need for robotic harvesting primarily include apples, oranges, stone fruit, mangoes, and kiwi. In addition, automatic picking end-effectors for special commodities such as cabbage, pineapple, and mushrooms have also been researched.

This chapter will provide a comprehensive review on the existing end-effectors for fresh-market fruit harvesting including a brief introduction to various harvesting principles used in designing these end-effectors. The crucial issues for designing an effective picking end-effector will be discussed in detail. Finally, some concluding thoughts (including future trends) in technology advancement and potential solutions for overcoming the existing technical challenges will also be evaluated.

8.2 Functions of a Picking End-Effector

An acceptable fruit picking operation signifies that the fruit is successfully, efficiently and safely separated from the plant, as any damage to it would affect its quality. The one-fruit-at-a-time picking model, which is widely adopted for fresh-market fruit harvest, is composed of two steps. First and foremost, the target fruit or stalk is held, and then separated from the plant. Based on this picking principle, currently researched and developed picking end-effectors can be classified into three groups: (i) fruit-holding type, (ii) stem-holding type, and (iii) direct-separating type.

8.2.1 *Fruit-Holding End-Effectors*

Since the fruit-holding type picking end-effectors are in direct contact with the fruit, its key function is to exert suitable force on the object being picked, which is strong enough to stably hold the object (e.g., fruit), but is not too high to cause any damage to the object. Based on the techniques investigated in the past, the end-effectors designed to hold fruit can be divided into five categories: (i) elastic buffering, (ii) under-actuated fingers, (iii) flexible driving, (iv) end-effectors with clamping force feedback, and (v) air suction.

(a) *End-effectors with elastic buffering material*

In view of the potential hard contact between the rigid finger and fruit, an elastic buffering material, such as rubber, silicone, and polyfoam, can be used to cover the end-effector fingers in order to absorb the impact and present better frictional property. The two end-effectors illustrated in Fig. 8.1, namely, a three-finger citrus picking end-effector developed at the University of Florida (Fig. 8.1a; Hannan and Burks 2004) and a two-finger tomato picking end-effector developed by Kondo (Fig. 8.1b, Monta et al. 1998), both have an elastic buffering material attached to their fingers.

Elastic materials are placed, generally, on the inner side of the fingers to provide a clamping buffer, so as to avoid or minimize the chances of scathing the soft fruit. This method is low-cost, easy to implement, and it could also be used with harder (less delicate) fruits. However, it is not very useful for soft fruits, and the thick buffering material might even reduce the fingers' valid range of motion.

(b) *End-effectors with under-actuated fingers*

The under-actuated mechanism is defined as the process through which more than one joint could be driven by a single motor. This principle implies that the robot's under-actuated fingers are equally operated by only one driver, and the

Fig. 8.1 End-effectors with varying types of elastic buffering materials. (**a**) An apple picking end-effector (from Gauchel 2014). (**b**) A tomato picking end-effector (from Monta et al. 1998)

knuckles are self-adaptive to the shape and size of the target fruit. Considering the differences in the harvested fruit size and shape, end-effectors with under-actuated fingers are of tremendous utility. The under-actuated structure can be used for fruits with various shapes and positions, which sharply improves the end-effectors' error tolerance characteristic, and ensures uniform contact between the finger and the fruit.

One of the under-actuated end-effector was developed at Washington State University for their apple picking robot. The end-effector consisted of three fingers (Fig. 8.2a, Silwal et al. 2016) and each finger included two joints covered with soft rubber on the surface. When one of the fingers first touched the object of interest, the disc differential mechanism rotated to compensate for the displacement of the other two jaws, thereby adapting to the fruit's shape, which ensured a reliable grabbing performance with an under-actuated mechanism to equalize the contact force. Another apple picking end-effector was designed by FFRobotics (Haifa, Israel), which included three fingers (Fig. 8.2b; Courtney and Mullinax 2019), which were driven by a telescopic electric push-rod to ultimately hold the fruit. Subsequently, an unpowered joint in each finger rotated to match with the fruit's shape. An additional wrist joint of this end-effector was adopted to twist off the apple's stalk.

A tomato picking robot end-effector was developed by Kyoto University with four flexible fingers fixed to a nylon supporting plate by four sections of connected nylon hose (Fig. 8.3a; Kondo and Ting 1998 and Krikke 2005). A cable was fixed at the end of the finger through the hose. Whenever the cable was pulled, the finger would bend to grab the fruit. As the fingers were bent to adapt to the fruit, they could fit tightly with the fruit. With the similar concept, a four-finger tomato picking end-effector was created by Ling et al. (2004) at Ohio State University, which also had only one cable to drive the fingers simultaneously. The nylon hose was replaced with a harder ABS plastic tube in order to limit lateral movement during clamping (Fig. 8.3b), which was expected to provide a stronger holding force. Another tomato picking end-effector developed at National Ilan University, Taiwan, also included

Fig. 8.2 End-effectors with under-actuated fingers designed and developed for picking apples: (**a**) developed by Silwal et al. (2016); (**b**) developed by Courtney and Mullinax (2019)

Fig. 8.3 Tomato picking end-effectors with under-actuated fingers for picking tomato: (**a**) developed by Kondo and Ting (1998); (**b**) developed by Ling et al. (2004)

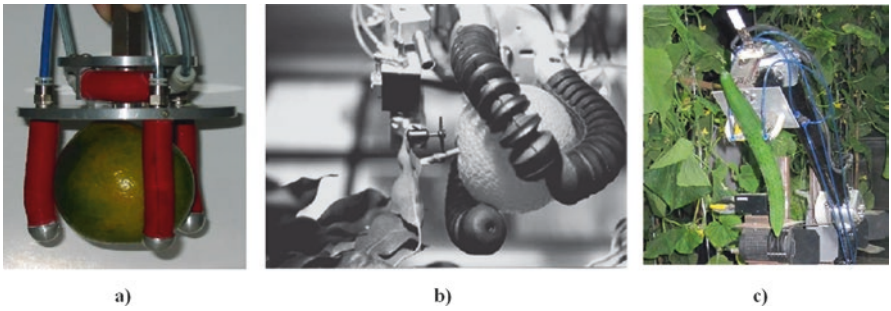


Fig. 8.4 End-effectors with pneumatically driven soft fingers: (**a**) an orange picking end-effector (from Yang et al. 2010); (**b**) a citrus picking end-effector (from Allotta et al. 1990); (**c**) a cucumber picking end-effector (Li et al. 2008)

four under-actuated fingers, consisting of four elastic joints, and a soft foam rubber on its surface. The main difference in this configuration was that every two fingers were driven by an electromagnet through an elastic plate.

(c) *End-effectors with flexible driving medium*

If a traditional rigid drive (such as a motor) is used in actuating an end-effector, the clamping apparatus' dynamic impact may easily damage fruit. Therefore, gas and liquid are used as flexible/soft driving media so as to prevent any risk of causing fruit blemish. The pneumatic flexible/soft picking end-effector developed at Zhejiang University of Technology consisted of one pneumatic torsion joint and three pneumatic flexible bending fingers (Fig. 8.4a, Yang et al. 2010; Jin 2010; and Bao et al. 2009). Compressed gas was filled into the torsion joint and the bending finger cavity, respectively, enabling these structures to make the fingers rotate and bend inward. Similarly, ARTS Laboratory (Scuola Superiore Sant'Anna, Italy)

developed a pneumatically driven flexible, three-finger clamping end-effector (Fig. 8.4b; Allotta et al. 1990), which had longer fingers steadily holding citrus fruit. Similarly, Li et al. (2008) at China Agricultural University used two groups of rubber fingers to clamp cucumbers (Fig. 8.4c). Due to varying thickness of rubber on two sides of the fingers, variation in deformation between two sides under the action of high-pressure gas led to overall bending of the fingers.

Differing from the flexible/soft fingers mentioned above, a tomato picking end-effector was studied by Feng et al. (2015) and Wang et al. (2016) at China National Research Center of Intelligent Equipment for Agricultural, using a circumferentially distributed airbag of constant pressure to clamp the fruit (Fig. 8.5). Internal pressure of the airbag was adjusted through a pressure reducing valve in order to establish a constant clamping force on the surface of the fruit.

Regarding the liquid actuation, Swedish Institute of Food and Biotechnology developed a magnetorheological fluid robot gripper (Fig. 8.6; Pettersson et al. 2010) which used a magnetic field to control the pressure of magnetorheological fluids, and then accomplished flexible/soft gripping of differently shaped fruits using an adequate gripping force.

(d)*End-effectors with clamping force feedback control*

As described above (a)–(c), based on their intrinsic flexibility, elastic material, under-actuated mechanism, and flexible driving medium were selected to safely clamp target fruit, which are all passive flexible operations. The clamping force on the fruit surface originated from the adaptive deformation of the clamping apparatus, which could not be dynamically and accurately sensed or controlled. Consequently, the actuator was predominantly used for fruits with similar shapes, small sizes, and compact surfaces, since the clamping function was not ideal for targets with very soft surfaces or of heavy weight.

Active flexible operation is focused on achieving active intervention on potential damage caused by the clamping mechanism/apparatus or fruit slippage from the fingers, by sensing and controlling the force exerted between the clamping

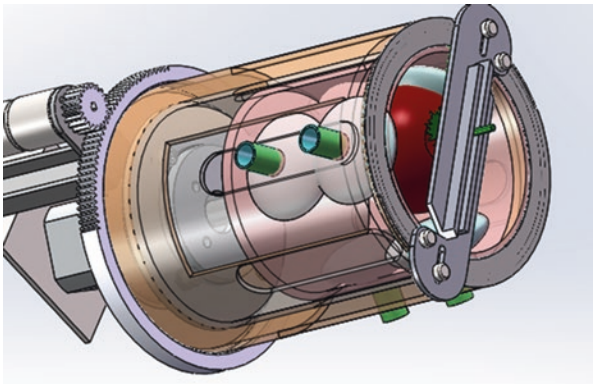


Fig. 8.5 Tomato picking end-effector with constant pressure airbag (form Feng et al. 2015)

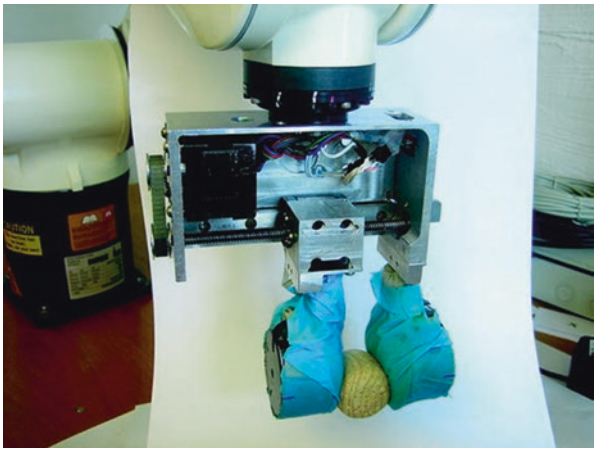


Fig. 8.6 End-effector with fingers driven by a liquid media (Pettersson et al. 2010)

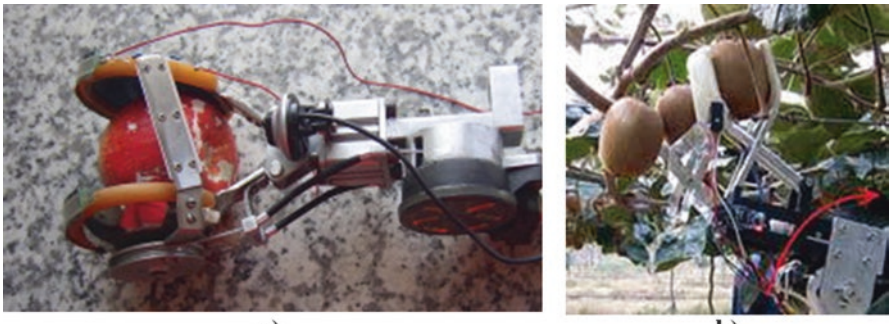


Fig. 8.7 End-effectors with pressure sensors on fingers: (**a**) an end-effector for picking apples (from Ji et al. 2014); (**b**) an end-effector for picking kiwi (from Zhang 2014)

apparatus and objects of interest. Active flexible clamping has stronger adaptability to the object being manipulated through feedback-based closed-loop control of the clamping force. Nowadays, the picking end-effector mainly relies on pressure or sliding sensors in the fingers and wrist joints, in order to improve the effectiveness of the machines in holding and separating/detaching fruit.

Dimeas et al. (2015) at the University of Patras constructed a three-finger picking end-effector, equipped with a pressure array sensor inside the finger. Moreover, Fuzzy control based on clamping force feedback was integrated. A notable drawback with this technique was that it took more than 10s to achieve stable clamping of strawberries.

A double-finger apple picking end-effector was developed by Ji et al. (2014) at Jiangsu University. The end-effector included a pressure resistor attached to the inner side of the finger to sense the real-time clamping force (Fig. 8.7a). Through the grasping torque control method based on generalized proportional-integral technique, the end effector achieved a successful clamping rate (without causing any

fruit damage) of 90% for apples and pears, and it took a total of 3 s from touching the fruit to completing the stable holding. A similar end-effector was developed and tested with kiwi fruit by Zhang (2014) at Northwest Agriculture & Forestry University (Fig. 8.7b). Pressure sensors were installed inside the arc surface of the fingers in order to detect the clamping pressure.

Researchers have also investigated the potential of sensing fruit stem using end-effector mechanisms. A flexible clamping end-effector was designed by a team at ARTS Laboratory (Scuola Superiore Sant'Anna, Italy; Allotta et al. 1990; and Muscato et al. 2005; Fig. 8.8). When the fingers of this end-effector held an orange and tightened the fruit stem backward, the force and torque sensors at the wrist of the claw detected the information, which was used to determine spatial position of fruit stems. The end-effector then used a cutting mechanism to remove the fruit from the plant.

Through installation and integration of a clamping force/pressure sensor, the clamping pressure exerted on the fruit surface could be dynamically measured, which was then used in various studies to avoid or minimize potential damage to the fruit during picking/detachment. However, since the fruit are composed of soft biological tissues, the interaction force between the fruit and the end-effector fingers exhibited a dynamic characteristic. It generally took a long time to attain a stable clamping state by using the force signal as a reference for dynamic control, which lowered the robot's picking efficiency.

(e) *End-effectors with air suction*

In this design, a suction component (such as a sucker) connected with an air source with negative pressure was approached close to the fruit in such a manner that the fruit was captured and fixed with the suction component under the negative pressure. Hence, the end-effector mechanism with fruit suctioning with negative pressure offered a larger tolerance to the positioning error of the end-effector (or caused by the error in the sensing/machine vision system), as well as variations of fruit size and shape, which made its application convenient and simple to realize.

Fig. 8.8 An fruit picking end-effector with force and torque sensors at the wrist (Muscato et al. 2005)

Christopher et al. (2017; Queensland University of Technology) successfully developed a pepper picking robot using this technique. The end-effector used consisted of a negative pressure sucker and a cutting knife. After the robot attached the fruit through the sucker, it was pulled away slightly from the plant, and the manipulator subsequently directed a cutting knife to cut the fruit stem. Similarly, Feng et al. (2012; the National Agricultural Intelligent Equipment Research Center, China) developed an end-effector consisting of a fruit suction cup and a stem grasper (Fig. 8.9). After being sucked and captured, the target fruit was retrieved from the plant by a telescopic cylinder, and the fruit stem was severed between the grasper's fingers. It is worth mentioning that collision interference between the end-effector and other fruits could be prevented by capturing and separating fruits using these types of suction mechanisms.

In addition to being used as an auxiliary component, negative pressure suction was also employed to separate fruits from plants, and eventually collect fruits. For instance, an apple picking end-effector developed by Abundant Robotics (Fig. 8.10; Thorne 2019) sucked in and pulled off fruit from the plants as the end-effectors approached them.

As discussed above, one of the major advantages of using suctioning mechanism is to enhance the picking end-effectors error tolerance, so that the picking robot acquires the adaptability to pick targets of random postures and varying shapes. Nonetheless, when a large amount of negative pressure is required, parameters such as the rotational speed of vacuum fans, the vacuum power, and the air flow rate are difficult to control, which give rise to issues such as high energy consumption and noise, and increased likelihood of fruit damage from sucking. When suctioning is used as an auxiliary device of picking end-effectors for clamping and separating the target, it is capable of pulling the fruit away from the plant, thus avoiding any sort

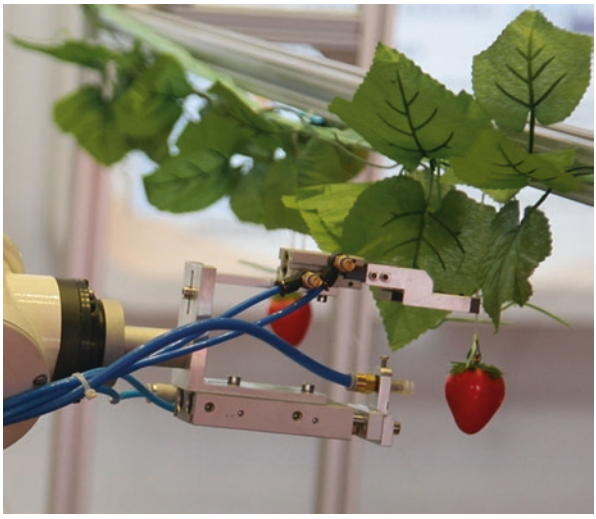


Fig. 8.9 An end-effector with air suction as auxiliary clamping unit designed for picking strawberries (Feng et al. 2012)

Fig. 8.10 Apple picking end-effectors designed with air suction as fruit separating unit (from Thorne 2019)

of collision between the fingers and the plant, which is necessary to optimize the success rate of picking.

8.2.2 Stem-Holding End-Effectors

Compared to the end-effectors designed to hold fruit (discussed in Sec. 8.2.1), the end-effectors designed to hold fruit stem have the advantage of avoiding direct contact and potential impact on fruits, which makes it more suitable for fruits and vegetables with slim and long stems, such as strawberries, cucumbers, and grapes. For example, the cucumber picking end-effector developed by Henten et al. (2002) at Wageningen University (Fig. 8.11a) consisted of two fingers for grasping the cucumber stems. After the cucumber was held in place by a suction mechanism, its stem was then grabbed by the end-effector. Subsequently, electrical heating in the upper part of the fingers was used to cut off the stem. Another end-effector with clamping fingers was designed for strawberry picking by Hayashi. This end-effector had the clamping fingers moving synchronously with the cutting blades, which were installed above a clamping claw. After the claw was closed, the fruit stem was held and separated from the plant. Feng et al. (2018) also designed a similar gripper with synchronous grasping and cutting claws for picking bunches of tomatoes (Fig. 8.11b).

Though these techniques were shown to be effective for picking long-stem fruit, end-effectors with stem-holding mechanisms were often found to be difficult to apply to fruits with short stalk/stem and/or those fruit growing in clusters due to the limited space available for grabbing and cutting operation.

8.2.3 End-Effectors for Direct Separation

Direct separation picking implies that the end-effectors directly separate the fruit or fruit stalk from the plant without clamping it, and then collects or transports it to a container. The apple harvesting end-effector designed by Abundant Robotics

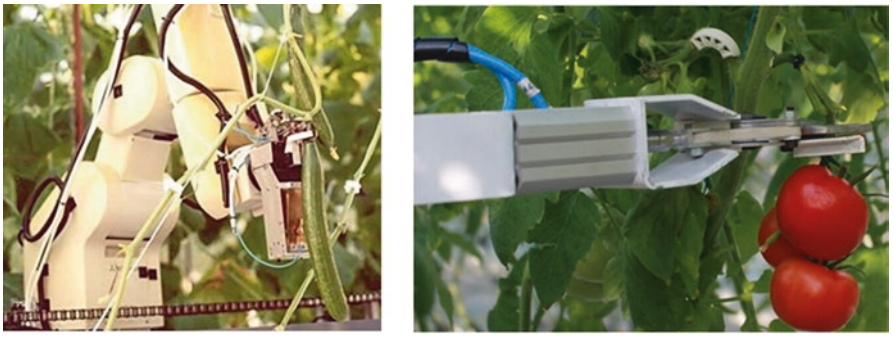


Fig. 8.11 Various end-effectors designed to hold and cut fruit stems: (**a**) a cucumber picking end-effector (from Henten et al. 2002); (**b**) a tomato picking end-effector (from Feng et al. 2018)

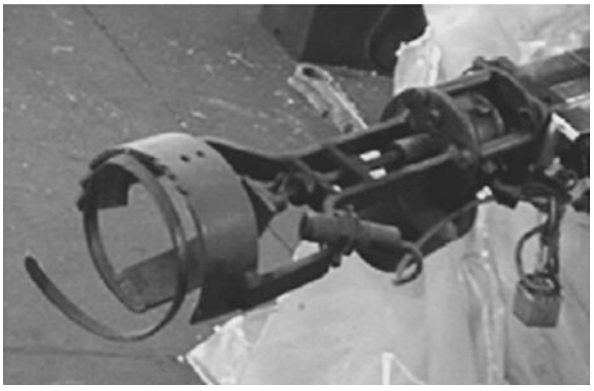


Fig. 8.12 A direct-separation type end-effector consisting of a spiral sleeve used to pick citrus (from Muscato et al. 2005)

discussed above (Thorne 2019; Fig. 8.10) picked and separated apples from their respective branches using strong negative pressure formed by a high-power vacuum generator and then transported them to a collection bin/container along a vacuum pipe. Muscato et al. (2005), on the other hand, designed a citrus picking end-effector adopting a spiral sleeve mechanism (Fig. 8.12). When the fruit entered the sleeve, the sleeve rotated to direct the fruit into the picking device along the spiral track. Next, the rotating cutter at the end severed the fruit stem to separate the fruit from its stem.

In addition to robotic picking of individual fruit as discussed in the previous paragraph, these types of end-effectors have been widely used in harvesting a bunch of fruit together including mass harvesting systems based on shaking or vibratory mechanisms. An automated wolfberry harvester designed by Peng et al. (2019) at Ningxia Academy of Agricultural and Forestry Sciences, China, was customized to hold wolfberry bunches into a container through manual assistance, and a vibrating device installed in the container caused the mature fruit fall off into the container

Fig. 8.13 End-effectors designed to directly separate bunches of fruit together: (**a**) an end-effector developed for wolfberry harvesting with vibration (from Peng et al. 2019); (**b**) a vibration beam end-effector used to harvest wine grapes (from Huffman 2010)

(Fig. 8.13a). The wine grape harvesting machine developed by Oxbo International used a beam end-effector (Huffman 2010; Fig. 8.13b) that applied vibratory signal to the vines to separate bunches of berries off of the canopies.

Because a single device was able to perform fruit separation, capturing, and collection, the direct-separation end-effectors (and associated mechanism) like the ones discussed in this subsection led to generally a compact structure and efficient harvesting operation. Nevertheless, due to the lack of auxiliary mechanisms to protect the fruit, these systems could cause high level of fruit damage from fruit-to-fruit and fruit-to-device impacts. Therefore, this type of end-effectors and associated harvesting systems are more appropriate for harvesting fruit with a thicker pericarp, or fruit used in processing market (and not those consumed fresh).

8.2.4 Other Novel End-Effectors

Different fruit crops pose different challenges for robotic picking and seek for novel end-effector technologies to be effective. Sweet pepper is one of such examples as it has a special shape as well as unique characteristics including large size, irregular posture, and short stem. To meet the needs of picking fruit with these unique characteristics, a novel end-effector was developed by Arad et al. (2019). The end-effector consisted of two-finger auxiliary positioning device to restrict swinging of sweet peppers during picking (Fig. 8.14a), and a saw-shaped cutting knife was continuously pushed forward until the fruit stem was cut off. Severed fruit fell onto a container placed below the end-effector. Sweet peppers are often produced using hanging wire cultivation mode and are currently picked manually. The end-effector designed in this study, which used three steps of constraint, cut, and collect fruit,

a) b) c)

Fig. 8.14 Some unique types of end-effectors studied for fruit and vegetable harvesting and handling: (**a**) a sweet pepper picking end-effector (from Arad et al. 2019); (**b**) a strawberry picking end-effector (from Preter et al. 2018); (**c**) a soft fruit gripper (from Anandan 2016)

was able to overcome the fact that the fruit and the fruit stem were difficult to hold. The design also offered simple and practical solution that also improved the tolerance on fruit positioning error of the robotic systems.

Preter et al. (2018; Octinion Company, Leuven, Belgium) designed a strawberry picking end-effector (Fig. 8.14b) incorporating a combination of fruit holding and stem pulling mechanisms. The end-effector, thus, prevented any residual stem on the fruit from scraping other fruits being collected together. The clamping fingers were made of elastic material with a hollowed-out structure using 3D printing technology. The fingers demonstrated satisfactory adaptive deformation to strawberry fruits so that the fruits could be reliably held without causing any damage.

Considering the need for automated sorting and packaging of fresh fruit and vegetable, Soft Robotics Company (Boston, USA) customized a fruit gripper with tentacle-shaped soft fingers based on the breakthrough in soft materials and structural design (Anandan 2016; Fig. 8.14c). This technology leads to an end-effector that is highly adaptable to fruits and vegetables with different shapes. Compared to the suction mechanism and multi-joint fingers, it possesses the obvious advantages of being more efficient and reliable. At the same time, with this technique, end-effector mechanism control is simple, the modular installation is easily applicable, and the material is both harmless and sanitary. Hence, this type of end-effectors is suitable for application in the fruit and vegetable industry, both for harvesting and post-harvest handling.

8.3 Design Considerations for Picking End-Effectors

8.3.1 Fruit Holding Mechanism

(a) *Finger configurations*

As mentioned previously, the current picking robots comprised various end-effectors, each offering its own advantages and disadvantages. Owing to its

relatively simple structure, flexible application, and convenient operation, the two-finger configuration is currently the most widely used picking end-effector. However, its versatility with respect to picking targets of complex shapes and varying postures is limited. The under-actuated end-effectors use the simple multi-joint flexible fingers to form a multi-finger holding device. The bending curve of fingers is smooth and has a certain compensation ability, which allows for proper adaptation of the end-effector to various fruit sizes. That said, this mechanism is considered highly under-actuated since its multi-joints are driven by only one motor. Whenever obstacles such as branches and leaves are encountered between the fruit and the end-effector, the flexible finger will bend, thus causing failed picking.

The end-effectors with multiple flexible fingers, based on the anthropomorphic multi-joint finger, pneumatic artificial muscle, shape memory alloys, ion exchange polymeric metal materials, are currently being researched actively in the field of robotics. However, they are still far from practical application, especially agricultural application, given their significant complexity and cost.

(b) *Number of fingers*

The more the fingers, the better the reliability of holding the target. It is worth mentioning that the increased number of fingers would also increase the complexity of the mechanism and the control system would also increase, and would cause more interference with fruit stalks, branches, and leaves. Spherical fruits can generally be held by two or more fingers. In instances when there are fewer fingers, the overall configuration of the end-effector would be compact and convenient to use.

(c) *Measuring physical properties of fruits*

The static-friction coefficient, which is the intrinsic property between materials touching each other, is a critical parameter to determine the grasping force on fruit/stalk in order to overcome the external force causing them to slide off. Stainless steel, PVC, and rubber are often the materials of choice in developing end-effectors. The static-friction coefficients between the fruit/stem and the end-effector materials can be calculated by measuring a series of maximum static-friction forces under different pressures (Feng et al. 2019).

A device for measuring the maximum static-friction force is depicted in Fig. 8.15. The contrast material is pasted on the presser and the base of the universal testing machine, and contacted with the measured fruit/stalk. After setting the pressure's intensity with which the presser holds the fruit/stalk, a pulling-force instrument is adapted to retrieve the fruit until it becomes freely movable. The maximum value displayed on the pulling-force instrument is noted as the double maximum static-friction force under the setting pressure, including frictions both on the top and bottom sides of the fruit/stem.

The friction coefficient is greater when the fruit/stalk comes into contact with the rubber compared with the other two kinds of contrast material, meaning that it would be easier to hold fruit or stalk with graspers made of rubber. Besides, the stalk's friction coefficient is less than that of fruit for all the three kinds of contrast

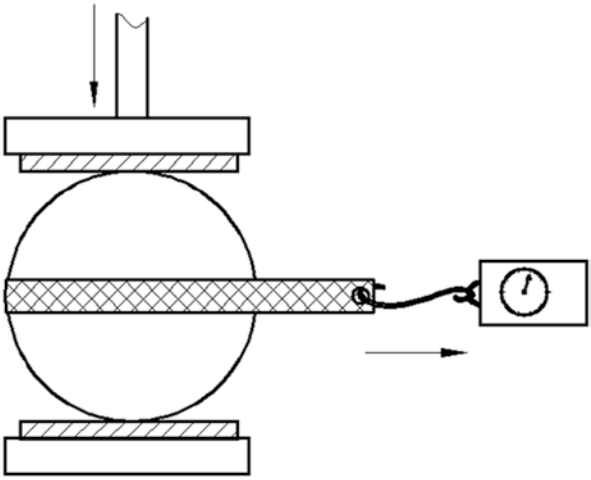
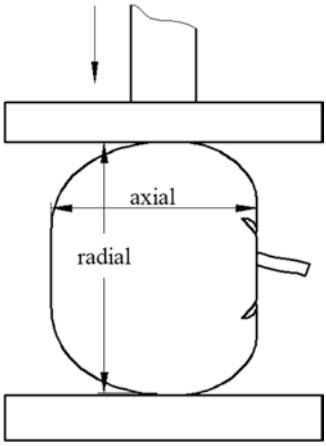


Fig. 8.15 A device to measure maximum static frictional force (from Feng et al. 2019)

Fig. 8.16 A device to measure pressure resistance (radial) of fruit (from Feng et al. 2019)



material, hence holding the stalk necessitates a stronger force compared with holding fruit.

Impact or pressure resistance capability of fruit is another important factor to consider when designing the flexible picking end-effectors. A device for measuring the resistance is delineated in Fig. 8.16. The fruit is set between the presser and the base. And the the presser extrude the fruit until the pressure on the fuite rises to the extremity, when the resistance force from fruit has a sharp decline. The maximum pressure value measured is recorded as the pressure a fruit can resist without being damaged.

As tomato fruit ripen, its pressure resistance capability declines by 60% from the green mature stage to full maturity (with complete color change). The pressure resistance ability of fruit along the axial direction is generally higher than the same

in the radial direction, irrespective of the sample's maturity stage. In other words, it is easier to crush the tomatoes by pressing radially rather than axially.

8.3.2 Separating Fruit from Plants

(a) *Stem cutting*

Using blades to directly cut off fruit stalks/stems is a simple and practical way to separate the fruit. At present, stem cutting end-effectors are mostly used for long-stem fruits such as strawberries, cucumbers, and tomatoes. However, since the blades are repeatedly used, it is possible to cause infection in plants by pathogens and fruit moisture loss at the incision site. In addition, the stem cutting mechanism may fail to work for fruits that grow in clusters due to relatively narrow working space and varying fruit postures, which creates challenging environment for cutting apparatus to get into the stems for cutting.

(b) *Separation at the abscission layer*

Given that the fruit stem abscission layer is naturally formed when the fruits mature, end-effectors acting on separating fruit at abscission do not need an additional stem separation mechanism. These end-effectors are capable of separating the fruit through pulling, folding, and twisting actions, while the fingers hold the fruit steadily. The major advantages of this kind of end-effectors include simple structure and versatility.

Due to the directional distribution features of the abscission layer tissues, the breaking forces in different directions vary substantially. If a fruit is pulled along the stem, it usually requires a larger pulling force to sever the tissue. Additionally, the forces exerted by the fingers on fruit surface will also rise to ensure that the fruit is securely grasped, which results in a significant increase in the actuation power, configuration size, and weight of the corresponding mechanism. In addition, excessive pulling force exerted on the fruit may cause the branches to break. On the other hand, if the force is tangent to the abscission layer, the required detachment force will be comparatively lower. Fruits with stem abscission layers, such as tomatoes, apples, and pears, all have this characteristic. It is also noted that humans also exploit this characteristic to achieve efficient picking of these fruits by twisting the stem rather than pulling fruit along the stem.

(c) *Thermal cutting*

As discussed above, stem cutting might cause disease infection in plants. Thermal cutting, therefore, provides an alternative that is considered a safe way for severing fruit stems. For example, cucumber (Henten et al. 2002) and sweet pepper (Bachche and Oka 2013) picking end-effectors were used to separate fruits from plants by thermal cutting. These end-effectors operate with the principle that when the two electrodes come into contact with the fruit stem, a high-frequency current is

generated, and the high temperature generated inside the stem breaks off its fibrous tissue. This method avoids the dilemma of incision pollution and water loss, but it require two electrodes to make reliable contact with the fruit stem to form a circuit pathway, which necessitates high positioning accuracy of the picking robot. As an alternative way of thermal cutting, Zhang et al. (2011) developed a strawberry harvesting end-effector using laser cutting. As the fruit stem was grabbed, the laser installed on the finger was energized, which generated sufficient heat to burn off the fruit stem. The prototype they developed, however, was effective for only small strawberry stems. Thermal cutting of thicker fruit stems with laser requires a more powerful laser generator.

(d) *Measuring physical properties of fruit*

A device used to measure cutting force is necessary to severe fruit stem as portrayed in Fig. 8.17. Two blades (NO.1 and NO.2) made of T12 steel were installed in two opposite sides of the device as shown in the figure. One blade is stationary and other the can be moved. During the test, fruit stem to be tested is held in place between two blades. Then, the movable plate is pressed down against the stem until the stem is cut off. The maximum force recorded by the device in this process is noted as the cutting force necessary to severe the stem. It is also noted that the cutting force needed to severe the stems with the double blade device discussed here was about 50% less than the force needed with the single blade device (Feng et al. 2019).

As fruits like tomatoes mature, the abscission layer formed at the stem-branch and stem-calyx can be readily separated, which is the optimum point for harvesting the fruit from the plant. Thus, obtaining its breakage property is pivotal for

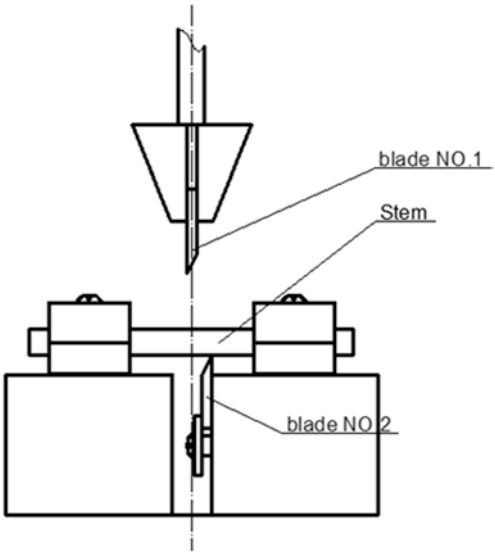


Fig. 8.17 A device to measure cutting force of fruit stem (Feng et al. 2019)

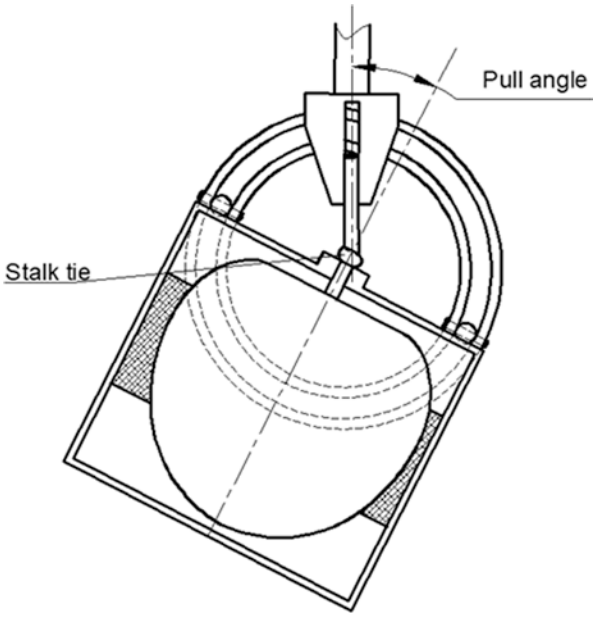


Fig. 8.18 A device to measure abscission layer breakage force of fruit (Feng et al. 2019)

designing the harvesting end-effectors. A device for measuring stem breakage force at abscission-layer is depicted in Fig. 8.18.

The stem is more easily broken at stem-branch layer when pulled along an angle of 0° than at any other angles. It has also been found that the difference in the breakage force by pulling along 30°, 60°, and 90° are negligible. Besides, as the fruit gets more mature, the value of the breakage force goes down. It has been reported to fall by roughly 44% as the fruit goes from green maturity to full maturity.

Compared to the force necessary to separate stem at stem-branch junction, the stronger force is necessary to separate fruit at the stem-calyx layer. The breakage force at this layer decreases with increasing pull force angle. The minimum value of the breakage force along an angle of 90° was 44% less than the maximum value along an angle of 0°. In addition, the breakage force also diminishes as the fruit get more mature, and the trend is slowed down after the maturity gets to the color change stage, because the development of the calyx abscission layer is complete by this stage (Feng et al. 2019).

8.4 Challenges

8.4.1 Complexity, Diversity, and Variability of Objects

The fruit is an important biological component of plants, and different plants produce diverse fruits. The picking end-effectors for different target fruit, therefore, are usually not interchangeable. For example, strawberry picking claws may not be used to pick tomatoes without a substantial modification. The development of reliable, unique end-effectors for picking various types of fruits is a necessary prerequisite for realizing robot picking operation.

Even for the same plant, due to the random growth of fruit population, their shape, posture, and position are different. Therefore, it is important to account for the difference in size and mechanical properties of fruits and fruit stem in designing end-effectors, so that the power and size of the end-effectors meet the requirements of most of the fruits. Taking tomato fruits as an example, the diameter of most fruits ranges between 50 mm and 90 mm, and there are also heteromorphic fruits that are either too large or too small. If the goal is to meet the needs of every fruit, the size and power of the end-effector will have an excessively vast range, and the development and application costs will rise accordingly.

The growth posture of the picking targets (fruits) is also a random variable. Optimal positioning of the end-effector to approach and capture the fruit/stem is necessary for successful picking. This need presents a huge challenge for positioning the end-effector to hold the fruit or stem, because it is difficult to locate the posture of spherical fruit and slim stems in the complex farming environment.

The aggregation of naturally grown fruits is also hard to address. In addition to individual fruits, fruits growing in clusters and/or bunches and overlap/occlusion of fruit and fruit stem are also quite common. Therefore, avoiding mechanical interference and accurately manipulating single fruit is also an immensely challenging issue.

8.4.2 Needs for High Precision Operations

As an integral part of a robotic operation in crop production, fruit picking end-effectors need to be able to prevent damage and achieve high-efficiency and low-energy picking operations. Due to the variation in fruit shape and size, it is necessary for the picking end-effector to have an adequate error tolerance and nondestructive, flexible operation characteristics to ensure high success rate of picking, and prevent or reduce picking failure and/or bruising, cuts, and other kinds of damages to fruit. There are three main sources of fruit damage during robotic picking: (i) Impact during grasping is the most common damage, which is caused by the collision between fingers (or other end-effector parts) and fruit under fast operation, which greatly increase the probability of fruit damage. (ii) The fruit is bruised, cut, or otherwise

damaged by other objects such as fruit rubs on the robot or plant during picking. (iii) The fruits are bruised when being released to a conveyer or container, primarily due to fruit-to-fruit impact/collision.

Due to the narrow working space within the plant canopies and the limited bearing capacity of the manipulator, lighter and smaller end-effectors have greater advantages in terms of working flexibility and are more conducive to saving energy. So, under the premise of meeting the requirements of picking operations, the ideal end-effector should be light and compact. This requirement, however, may create a contradiction. Despite the fact that the reliability of the picking end-effector can be improved through structural optimization, mechanical feedback and intelligent control, its complexity, size, and cost would increase accordingly, thus its practicability would decrease instead.

8.5 Summary and Concluding Thoughts

Application of the humanoid design concept

Traditional agricultural production conditions, with manual operation as the main body, do not match the requirements of robots for a standardized environment, which is an important factor limits the practical application of agricultural robots on farmland. Through analyzing and imitating human and animal behavior, the idea of bionic design is very necessary for improving the performance of the traditional end-effector device (Chen et al. 2015). So the actuator of multi-function, high-efficiency and humanoid characteristics would be an important trend of the end-effector R&D for agricultural robots.

Material technology breakthrough

With the advancement in material science and engineering in recent years, the emergence and application of soft materials such as SMA (Yang and Gu 2007), IPMC (Chatterjee et al. 2013), and FEA (Tarvainen and Yu 2017) are of great significance to solve the issue of flexibility faced by current agricultural robots. These kinds of materials offer outstanding advantages in terms of contact flexibility, motion control, and structure compactness, which can effectively overcome the problems of multi-joint drive, fruit damage caused by rigid contacts, and complex structure of the current clamping end-effectors. However, these materials still have shortcomings in stress, service life, and cost, which cannot fully meet the current needs for agricultural robots.

Integration of machinery and agronomy/horticulture

Compared with industrial robots, agricultural robots work with biological materials/objects with different shapes and sizes, constantly changing working environment, and random distribution of target positions and postures. Such uncertain agricultural conditions have become one of the important factors restricting the

commercial application of agricultural robots. The concept of systems approach combining agricultural machinery and agronomics/horticulture could be a powerful tool to increase production efficiency and reduce cost and to improve the adaptability of agricultural machinery. Such an approach also provides an important way to promote the wider application of innovative, intelligent equipment in agriculture. Some of the agronomic/horticultural improvement that might help adoption of robotic harvesting include: (i) adopting precise crop/canopy management techniques so that fruits growing at the same height range can be harvested at the same time; (ii) cultivating new varieties with longer fruit stems, which are more convenient for robotic picking; and (iii) Intensifying the abscission layer formation of mature fruits to reduce the difficulty in fruit separation.

References

Allotta B, Buttazzo G, Dario P (1990) A force/torque sensor-based technique for robot harvesting of fruits and vegetables. Proceedings of IEEE international workshop on intelligent robots and systems, towards a new frontier of applications, pp 231-235

Anandan T (2016) Robots help feed the World. Retrieved 27 Oct 2016, from https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Robots-Help-Feed-the-World/content_id/6292

Arad B, Balendonck J, Barth R, Shahar O, Edan Y, Hellström T, Hemming J, Kurtser P, Ringdahl O, Tielen T, Tuijl B (2019) Development of a sweet pepper harvesting robot. J Field Robot:1–13

Bachche S, Oka K (2013) Performance testing of thermal cutting systems for sweet pepper harvesting robot in greenhouse horticulture. J Syst Design Dynam 7(1):36–51

Bao G, Gao F, Xun Y, Yang Q (2009) Flexible end-effector based on flexible pneumatic actuator and its grasping model. Trans CSAE 25(10):121–126

Chatterjee D, Hanumaiah N, Bahramzadeh Y, Shahinpoor M (2013) Actuation and sensing studies of a miniaturized five fingered robotic hand made with Ion polymeric metal composite (IPMC). Adv Mater Res 740:492–495

Chen X, Chaudhary K, Tanaka Y, Nagahama K, Yaguchi H, Okada K, Inaba M (2015) Reasoning-based vision recognition for agricultural humanoid robot toward tomato harvesting. Proceedings of the IEEE international conference on intelligent robots and systems, pp 6487–6494

Christopher L, Andrew E, Christopher M, Adam W, Tristan P (2017) Autonomous sweet pepper harvesting for protected cropping systems. IEEE Robot Auto Lett 2(2):872–879

Courtney R, Mullinax T (2019) Washington orchards host robotic arms race. Retrieved 2 Dec 2019, from https://www.goodfruit.com/washington-orchards-host-robotic-arms-race/

Dimeas F, Sako D, Moulianitis V (2015) Design and fuzzy control of a robotic grpper for efficient strawberry harvesting. Robotica 33(5):1085–1098

Feng Q, Wang X, Zheng W (2012) A new strawberry harvesting robot for elevated-trough culture. Int J Agri Biol Eng 5(2):1–8

Feng Q, Wang X, Wang G, Li Z (2015) Design and test of tomatoes harvesting robot. Proceedings of IEEE International Conference on Information and Automation, (ICIA 2015), pp 949–952

Feng Q, Zhang M, Xu R, Zhang C, Wang X (2018) Design and test of robotic harvesting system for cherry tomato. Int J Agri Biol Eng 11(1):96–100

Feng Q, Wang G, Wang S, Wang X (2019) Tomato's mechanical properties measurement aiming for auto-harvesting. IOP Conf Series Adv Mater Sci Eng 585(1):012120

Gauchel W (2014) Freshly picked – harvesting with robots. Retrieved 26 Sept 2014. https:// www.festo.com/group/en/cms/10382.htm

Hannan M, Burks T (2004) Current developments in automated citrus harvesting. American Society of Agricultural and Biological Engineers Annual International Meeting
Henten J, Hemming J, Tuijl B, Kornet J, Meuleman J, Bontsema J, Os E (2002) An autonomous robot for harvesting cucumbers in greenhouses. Auton Robot 13(3):241–258
Huffman W (2010) The status of labor-saving mechanization in fruits and vegetables. Staff general research papers archive 31630, Iowa State University, Department of Economics, pp 11–13
Ji W, Luo D, Li J, Yang J, Zhao D (2014) Compliance grasp force control for end-effector of fruit-vegetable picking robot. Trans CSAE 30(9):19–26
Jin Y (2010) End-effector of apple picking based on flexible pneumatic actuator FPA. Master's thesis, Zhejiang University of Technology, Hangzhou, Zhejiang
Kondo N, Ting K (1998) Robotics for bioproduction systems. American Society of Agricultural Engineers, pp 30–38
Krikke J (2005) Robotics research exploits opportunities for growth. IEEE Pervasive Comput 4(3):7–10
Li W, Wang K, Tan Y, Yang Q, Gao F, Yuan T, Ren Y, Hou M, Feng Q (2008) Study on the technology of cucumber harvesting robot in greenhouse. In: Proceedings of 4th national advanced manufacturing equipment and robotics summit forum
Ling P, Ehsani R, Ting K (2004) Sensing and end-effector for a robotic tomato harvester. In: Proceedings of the ASAE annual meeting paper
Monta M, Kondo N, Ting K (1998) End-effectors for tomato harvesting robot. Artif Intell Rev 12(1):11–25
Muscato G, Prestifilippo M, Abbate N (2005) A prototype of an orange picking robot: past history, the new robot and experimental results. Indus Robot J 32(2):128–138
Peng Y, Zhang Z, Liu Y, Xu T, Zhang L, Wang R (2019) Kinematics and vibration mode analysis of vibrating harvesting for wolfberry. Mech Res Appl 32(02):8–13
Pettersson A, Davis S, Gray J (2010) Design of a magnetorheological robot gripper for handling of delicate food products with varying shapes. J Food Eng 98(3):332–338
Preter A, Anthonis J, Baerdemaeker J (2018) Development of a robot for harvesting strawberries. IFAC-PapersOnLine 51(17):14–19
Silwal A, Davidson J, Karkee M, Mo C, Zhang Q, Lewis K (2016) Design, integration, and field evaluation of a robotic apple harvester. J Field Robot 34:1104–1159
Tarvainen T, Yu W (2017) Pneumatic multi-pocket elastomer actuators for metacarpophalangeal joint flexion and abduction-adduction. Actuators 6(3):27–49
Thorne J (2019) Apple-picking robots gear up for U.S. debut in Washington State. Retrieved 13 May 2019, from https://www.geekwire.com/2019/apple-picking-robots-gear-u-s-debut-washigton-state
Wang X, Wu P, Feng Q, Wang G (2016) Design and test of tomatoes harvesting robot. J Agri Mech Res 38(4):94–98
Yang K, Gu C (2007) Research on novel shape memory alloy multi-fingered humanoid hand. Proc Inst Mech Eng C J Mech Eng Sci 221(9):1131–1140
Yang Q, Jin Y, Qian S, Bao G (2010) Research on end-effector of apple picking based on new flexible pneumatic actuator. Trans CSAM 41(09):154–158
Zhang F (2014) Research and design on the non-destructive end-effector of Kiwifruit harvesting robot. Doctor's thesis, Northwest A&F University, Yanglin, Xi'an
Zhang K, Yang L, Zhang T (2011) Design and experiment of picking mechanism for strawberry harvesting robot. Trans CSAM 42(9):155–161

Chapter 9
Control Techniques in Robotic Harvesting

Siddhartha Mehta and Maciej Rysz

9.1 Introduction

Food security and global market competition can be regarded as major factors fuelling the interest and need for agricultural robotics. In addition, the agricultural sector in many developed countries is operating at reduced manpower due to the lack of seasonal labor availability, high insurance costs, and rising wages. According to Gongal et al. (2015), the labor-intensive and injury-prone working conditions experienced in specialty crop harvesting are leading to a decline in skilled labor availability and the increase in harvesting costs. For example, consider the case of Florida citrus, which had harvesting costs of about 2–4 times that of Brazilian harvesting costs in the period from 1979 to 2009. In 2008–2009, the delivered-in cost of Florida orange was $2.359 USD per kg solids, while that of the Brazilian orange was $1.598 per kg solids. The harvesting cost alone for Florida orange in 2015–2016 was $0.2985 per m^2 compared to the production cost of $0.5752 per m^2. As reported by major economic studies, the harvesting cost of Florida citrus must be reduced by 50% to maintain global competitiveness and its long-term viability (Brown 2002). In this respect, mechanization of fruit harvesting via automated or semiautomated mass harvesting systems is highly desirable. However, mechanized fruit harvesters have limitation for soft fruit harvesting due to excessive bruising and mechanical damage to the harvest. Fruit damage is typical to mass harvesting systems since they are based on the principle of shaking or knocking fruit out of trees. As a result, the harvest obtained via mechanical harvesting is more suitable for the juice market. Another alternative to manual labor is to develop autonomous robotic harvesting systems, which have potential to improve productivity and reduce harvesting costs. Advanced robotic end-effector designs employing state-of-the-art technology, such

S. Mehta (✉) · M. Rysz
University of Florida, Gainesville, FL, USA
e-mail: siddhart@ufl.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_9

as force feedback sensors, can greatly reduce and even eliminate fruit damage, thus making robotic harvesters attractive for the fresh fruit market. However, the harvesting time for this selective harvesting technology can be significantly greater than that of the mass harvesters. Therefore, a viable approach could be to use mechanical harvesters for the juice market and robotic harvesting systems for the fresh fruit market. Collectively mechanical and robotic harvesting technologies, by the manner of complementing each other, can aid in realizing sustainable agriculture practices that improve economic viability and support the global mission of food security.

Control systems are ubiquitous in modern day agriculture as we increasingly seek autonomous and robotic solutions to the traditionally manual agricultural operations. And the demand for advanced control techniques that offer performance and robustness guarantees is growing rapidly to realize highly efficient autonomous operations. In reference to robotic harvesting, the role of a control system can be to guide a robotic system, typically a robotic manipulator, from its current position and orientation (i.e., pose) to a desired pose corresponding to the location of a fruit or vegetable to be harvested using observations or measurements provided by sensors. *Visual servo controllers* or *vision-based controllers* rely on cameras or imaging devices to provide image measurements of an object to the underlying control systems. One of the advantages of using a camera as a sensor is that even a relatively inexpensive monocular camera can provide exceptionally rich information, including shapes, colors, and textures, of the scene being viewed. As a result, vision-based control is the most popular control technique used in robotic harvesting. The cameras can be monochromatic, color (RGB), or color with depth (RGB-D), also known as depth cameras. Both monochromatic and RGB cameras form an image by projecting the three-dimensional scene onto the camera's two-dimensional image plane. On the other hand, RGB-D cameras, in addition to capturing the two-dimensional image, provide depth information for every pixel in the image. RGB-D cameras hold great promise but are currently limited by the available technology. Although the control techniques discussed in this chapter hold for any of these cameras, we will limit our discussions to monocular RGB cameras for the sake of simplicity.

In this chapter, we will go over the basics of vision-based control and review the progress made in visual servoing in robotic harvesting. Subsequently, we will study the process of controller design by presenting a simple visual servo control system. Stability and performance of control systems, which are the key influencers in determining harvest efficiency, can be affected by uncertainties in the system and environment. With this motivation, we will present robust and adaptive visual servo controllers that are resilient to these uncertainties. Finally, we state open problems that are aimed at improving harvest efficiency and harvesting times via improvements in the performance of visual servo control systems.

9.2 Basics of Visual Servo Control

As discussed earlier, a visual servo controller uses image feedback from a camera to control the motion of a robotic system. Specifically, the camera provides relative position of an object being viewed in the camera's coordinate frame, which can be used to obtain control signals (e.g., velocity, acceleration) for the robot. Visual servo controllers can be classified into three architectures—position-based, image-based, and 2.5D control system—depending on how the image feedback is used to obtain control input. In position-based control system, the control input is computed in the 3D Cartesian space based on the estimated 3D pose of an object with respect to the camera. The 3D pose of the object can be recovered from its 2D images using the knowledge of a model of the object. In image-based control system, the control input is computed in the 2D image-space using the image coordinates of the object. Roughly, 2.5D control systems can be considered as a combination of the position-based and image-based control systems. In 2.5D control systems, the control input is expressed in part in the 2D image-space using image coordinates of the object and in part in the 3D Cartesian space using the partial 3D pose of the object. This 3D pose information is only limited to determining the rotation and *scaled* translation of the object with respect to the camera, which can be obtained from the images using photogrammetric techniques. Therefore, 2.5D control systems do not rely on the model knowledge of the object.

Independent of the choice of visual servo control architecture, usually the first step is to process the images obtained from a camera to provide the image-space coordinates of the object being viewed. This is accomplished by detecting certain features of the object. The commonly used features include edges, corners or points of interest, and blobs. Advances in computer vision and image processing have led to the development of several feature detectors such as canny (Canny 1986), sobel (Sobel and Feldman 1968), FAST (Rosten and Drummond 2006), and Shi and Tomasi (Shi 1994), which offer relatively straightforward and computationally tractable means of processing images to detect the desired features. In robotic harvesting, it is natural to use the centroid or geometric center of a fruit or vegetable as a feature, which first requires that the fruit or vegetable be detected from the background in the image. For brevity, we will restrict our discussion in this chapter to fruit harvesting. The process of separating fruits from the image background is referred to as *fruit detection*. Most fruits can be detected based on their shape, color, or reflectance. For example, oranges can be detected via color thresholding accompanied by circular shape detection methods. Fruit detection typically yields blobs corresponding to the position of the fruits in the image, and the centroid or geometric center of the blobs provide the 2D image-space coordinates of the fruits. The knowledge of 3D position of the fruits is highly desirable in path planning and control of the robot. However, when using monocular camera as the only sensor, the challenge is to recover the 3D scene information from its 2D images, i.e., to obtain the depth information that is lost during projection of the 3D scene on the 2D image plane. To this effect, various depth recovery techniques based on monocular, stereo,

or multiple camera configurations can be used to estimate the 3D fruit position. Alternatively, additional sensors such as laser rangefinders and ultrasonic transducers can also be used to provide depth information. The process of obtaining 3D coordinates of the fruits by measuring or estimating the unknown depth is known as *fruit localization.*

A vision system can be configured as eye-in-hand, also known as camera-in-hand, or eye-to-hand, which is largely dictated by the application. In eye-to-hand camera configuration, a camera mounted external to the robot provides image measurements of the robot's pose. For example, a fixed camera mounted in a greenhouse can view and navigate a wheeled mobile robot. On the other hand, in eye-in-hand configuration, a camera is attached to the robot such that it provides image measurements of the environment as the robot moves. Roughly speaking, the former provides image measurements encoding information on the pose of the robot and the latter provides that of the object being viewed with respect to the camera. Robotic fruit harvesting systems widely employ the eye-in-hand camera configuration, where a camera can be mounted on the robot's end-effector. The control objective in this case becomes to drive the robot towards a fruit to be harvested while regulating the image coordinates of the fruit to a desired set point (e.g., the image center).

9.2.1 Progress in Vision-Based Control in Robotic Harvesting

The objective of vision-based control in robotic harvesting is to autonomously position the robot in relation a fruit for successful detachment using measurements provided by a vision system. This section outlines different control techniques used in robotic harvesting. For comprehensive overview of robotic systems and vision-based control in agriculture, readers are referred to Tillett (1993), Sarig (1993), Hannan and Burks (2004), Li et al. (2011), Bac et al. (2014), and Zhao et al. (2016). Vision-based control approaches in robotic harvesting can be broadly divided into *open-loop control* and *closed-loop control*.

Open-loop vision-based control systems combine visual sensing and robotic manipulation using the principle of "look-then-move," in which a scene is interpreted to determine the position of the fruits and then control is executed to position the robot at these pre-determined fruit locations. This principle leads to simple controller design but puts burden on image processing to accurately obtain the fruit position. Since open-loop control does not require continuous image feedback, computational demands are greatly reduced and higher manipulation speeds can be obtained. Below is some of the notable research that uses open-loop control strategy for robot positioning. Grand d'Esnon (1985) and Grand d'Esnon et al. (1987) developed a vision-based three degrees-of-freedom (DOF) manipulator—MAGALI—for golden apple harvesting, where a monocular camera detected fruits during a vertical scan. Subsequently, the telescopic arm translated along the optical beam until it reached the fruit, which was sensed by a photoelectric sensor. Levi et al. (1988) developed a vision-based cylindrical manipulator system for robotic citrus

harvesting. In AGROBAT project, Buemi et al. (1996) used a stereo-vision camera for tomato localization, which provided measurements for open-loop control of a six DOF robotic arm for harvesting. Kondo et al. (1996) relied on a similar approach but using pseudo-stereo vision to localize cherry tomatoes by triangulation. In an orange harvesting robot, Recce et al. (1996) used two robotic arms, each with a monocular camera within its end-effector, to form a pseudo-stereo sensor. Neural network-based approach was used to match the stereo images and map the image coordinates of the fruits to joint-space coordinates of the robot. In a robotic cucumber harvester, Van Henten et al. (2002) proposed a cooperative camera scheme that uses two monocular cameras; one mounted a rail and another on top of the end-effector. Since both the cameras can move, pseudo-stereo images can be obtained from each of the cameras to localize fruits. The rail-mounted camera enables robot motion planning and coarse guidance, while the camera mounted on the end-effector is used for final approach to the fruit. Cho et al. (2002) considered open-loop control architecture for lettuce harvesting robot, where the camera measurements were used to control the force of the gripper. Baeten et al. (2008) relied on a single eye-in-hand monocular camera for robotic apple harvesting. Two open-loop controllers were employed; a rotation controller to bring the target fruit to the center of the image and a translation controller to move the gripper along the optical axis by a distance determined through triangulation process. Tanigaki et al. (2008) employed a laser-based 3D vision sensor in eye-in-hand configuration to obtain 3D position of the fruits for a cherry harvesting robot. Hayashi et al. (2010) used a standard stereo-vision setup for 3D localization of strawberries, which were harvested using a three DOF manipulator.

Open-loop control systems are attractive due to their simplicity. However, these systems may suffer from excessive positioning errors in outdoor agricultural environments since continuous image feedback is not opted to verify and rectify the position of the robot with respect to the fruit. Measurement errors in the fruit's position, unmodeled dynamics of the robot, and environmental disturbances (e.g., fruit motion) are some of the factors that may adversely affect the performance of open-loop control systems, which may result in unsuccessful pick-cycles and reduced harvest efficiency. Closed-loop vision systems overcome these limitations by employing continuous image feedback. A dynamic "look-and-move" approach is the most widely implemented closed-loop visual servo control framework. In this architecture, the vision system is separated from the robot dynamics. As a result, knowledge of the complex, nonlinear robot dynamics is not required, which significantly simplifies controller development. The vision system in this framework provides set-point kinematic inputs, often position or velocity of the end-effector, to a robot controller, which tracks these set-point commands by computing appropriate joint inputs for the manipulator. In contrast, the direct visual servo control framework eliminates the robot controller entirely and directly provides joint inputs using vision alone. Therefore, direct visual servo controllers require accurate knowledge of the robot dynamics.

The first reported closed-loop visual servo control in robotic harvesting appears in Harrell et al. (1985), where a monochromatic vision system was implemented to

track citrus fruits at 60 Hz. Subsequently, the authors developed a direct visual servo controller in Harrell et al. (1989) to obtain joint inputs based on the image position of the fruit centroid with respect to the principle point (i.e., center of the image). However, the control structure resulted in geometric increase in control gains raising concerns about stability of the system and requiring an ad hoc gain scheduling procedure. In Harrell et al. (1990) the authors developed a vision controller that computed joint velocity inputs to regulate a fruit at the principle point. The robot approached the fruit with predefined velocity until an ultrasonic transducer mounted in the end-effector detected the presence of fruit. The closed-loop vision control improved robustness to fruit motion, but it was largely limited by the image feedback rate. Similar to Van Henten et al. (2002), Edan et al. (2000) used a cooperative sensing framework consisting of two monochromatic cameras serving as a far-vision sensor and a near-vision sensor. The far-vision camera detected oncoming melons, whereas the near-vision camera mounted on the gripper provided image feedback to reach centrally over a target melon until a proximity sensor detected the soil surface. In eggplant harvesting, Hayashi et al. (2002) developed a vision-based fuzzy controller to determine the forward, vertical, and angular motion of the end-effector based on the feedback from a monocular camera mounted in eye-in-hand configuration. Bulanon et al. (2005) considered an end-effector mounted camera and laser ranging sensor for apple harvesting robot. A proportional controller was developed to provide camera position inputs based on the error between the image position of an apple and the image center. Foglia and Reina (2006) also considered a monocular camera system in eye-in-hand configuration for radicchio harvesting, where the closed-loop system was aimed to compensate for positioning errors of the gripper due to unexpected speed variations of the carrier. In robotic apple harvesting, De-An et al. (2011) developed a proportional controller that related the image-space (pixel) errors to joint angle increments to position a fruit at the image center. A standard IBVS controller was used in sweet pepper harvesting in Barth et al. (2016) using a single monocular camera in eye-in-hand configuration, where the center of gravity image coordinates of the largest segmented sweet pepper served as the image feature. The feature depth in the image Jacobian was obtained using simultaneous localization and mapping.

As evident from the literature, the general idea is to use vision feedback from an eye-in-hand camera to align a fruit with the image center using closed-loop control and move the robot towards the fruit with a constant velocity until an additional sensor (e.g., proximity sensor) detects the presence of fruit. However, it must be noted that the dynamics of the fruit in the image plane are coupled with the motion of the camera along the optical axis. Controllers designed without taking into account this coupling can at best guarantee semi-global stability. In other words, the control gains must be selected sufficiently large to compensate for the change in the fruit position due to motion of the robot along the optical axis. Otherwise, it will result in instability of the closed-loop system and failure to maintain the fruit at the image center. Furthermore, the change in the fruit position in the image is "amplified" as the camera gets closer to a fruit thus requiring even larger control gains. This can lead to excessive camera motion, which may result in the fruit leaving the

camera's field-of-view (FOV). To improve stability of robotic harvesters, Mehta and Burks (2014) presented a hybrid control framework that uses principles of IBVS and PBVS to control the 3D translation of the camera. The image measurements of the fruit provide camera velocities along the image axes, whereas the camera velocity along the optical axis is obtained using the depth estimates of the fruit. Subsequently, the hybrid control design is extended to develop robust and adaptive visual servo controllers in Mehta et al. (2016) and Mehta and Burks (2016), respectively, to compensate for the unknown fruit motion that may arise due to environmental disturbances.

9.2.2 A Basic Visual Servo Control System

In this section, a closed-loop visual servo controller is presented to regulate a robot end-effector to a target fruit location. The controller is based on the dynamic look-and-move principle. A cooperative vision system consisting of a fixed camera and an eye-in-hand camera is incorporated. The fixed camera provides a global view of a tree canopy, while the eye-in-hand camera, due to proximity, provides high resolution fruit images. The fixed camera is primarily responsible for high-level tasks such as building global fruit map, optimal harvest planning, and robot path planning. While the image feedback from the eye-in-hand camera enables the end-effector to reach the desired fruit location. The 2D image position of the fruits in the eye-in-hand camera and the depth recovered in fruit localization form the basic feedback structure of this visual servo controller as shown in Fig. 9.1.

In this chapter, we will use a computationally inexpensive model-based approach (Mehta and Burks 2014) for depth estimation. The controller development presented in this section is modular with respect to depth estimation technique. Hence, the controller holds for different depth estimation methods, such as the ones using stereo-vision (Buemi et al. 1996; Kondo et al. 1996; Recce et al. 1996; Van Henten et al. 2002, 2003) and range measurement sensors (Harrell et al. 1989, 1990; Bulanon et al. 2005). As discussed earlier, harvest efficiency is one of the most

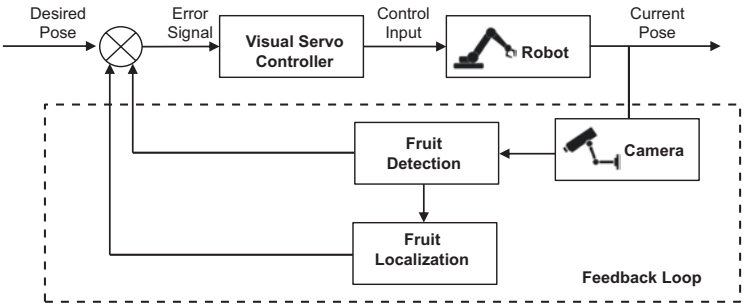


Fig. 9.1 Visual servo control scheme in robotic fruit harvesting

influential factors in robotic harvesting economics, which depends on the stability and performance of closed-loop control systems. Therefore, specific attention is given to rigorous controller formulation. A rotation controller is developed to orient the robot end-effector towards the target fruit such that the target fruit enters the FOV of the eye-in-hand camera. This enables harvesting the fruits that were not initially visible to the eye-in-hand. Subsequently, the developed visual servo controller regulates the end-effector to the fruit location. Lyapunov-based stability analysis guarantees global exponential stability of the closed-loop system such that the desired transient performance can be obtained by appropriately selecting control gains.

Euclidean Reconstruction

Consider the orthogonal coordinate frames F, F_f, and F_b as shown in Fig. 9.2. The time-varying coordinate frame F is attached to an eye-in-hand camera, i.e., a camera held by a robot end-effector. The coordinate frame F_f is attached to a fixed camera, for example, a stationary camera mounted in the workspace of a robot, and the coordinate frame F_b is attached to the stationary base of a robot. $O^* \in \mathrm{R}^3$ denotes the fruit position measured in the base frame F_b. The unknown Euclidean coordinates of the fruit center, $\bar{m}(t), \bar{m}_f \in \mathbb{R}^3$, expressed in terms of F and F_f, respectively, are given as

$$\bar{m}(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix}^T, \bar{m}_f = \begin{bmatrix} x_f & y_f & z_f \end{bmatrix}^T \tag{9.1}$$

where $z(t)$, $z_f \in \mathrm{R}$ denote the unknown depth of the target fruit expressed in F and F_f, respectively.

The Euclidean-space is projected onto the image-space, so let $m_i(t)$ and m_{fi} denote the corresponding normalized Euclidean coordinates of the fruit center as

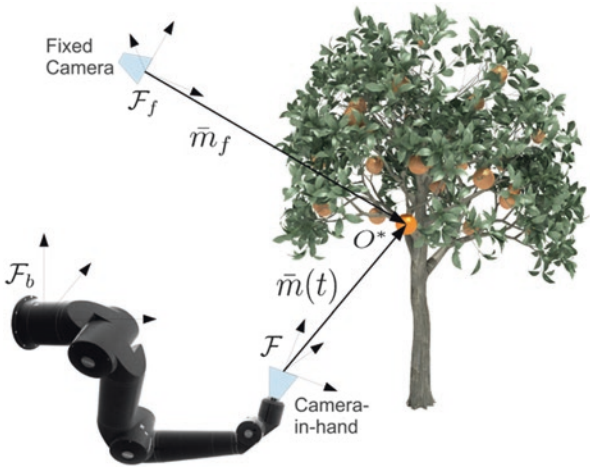


Fig. 9.2 Cooperative camera framework showing coordinate frame relationships, where the time-varying frame F is attached to the eye-in-hand camera, F_f corresponds to the fixed camera, and F_b is attached to the stationary base of the robot

$$m(t) = \frac{\bar{m}(t)}{z(t)} = \begin{bmatrix} \frac{x(t)}{z(t)} & \frac{y(t)}{z(t)} & 1 \end{bmatrix}^T, m_f = \frac{\bar{m}_f}{z_f} = \begin{bmatrix} \frac{x_f}{z_f} & \frac{y_f}{z_f} & 1 \end{bmatrix}^T. \tag{9.2}$$

Assumption 9.1 *In (9.2), it is assumed that the unknown depths $z(t)$, $z_f > \varepsilon$, where $\varepsilon \in \mathrm{R}_{>0}$ is a constant. This is a standard assumption in visual servo control, which physically means that the target is always in front of the camera.*

In addition to having normalized task-space coordinates, the target point will also have pixel coordinates acquired by the eye-in-hand camera and the fixed camera. Let $p(t)$, $p_f \in \mathrm{R}^2$ denote the pixel coordinates of the target center expressed in F and F_f, respectively, as

$$p(t) = \begin{bmatrix} u(t) & v(t) \end{bmatrix}^T, p_f = \begin{bmatrix} u_f & v_f \end{bmatrix}^T. \tag{9.3}$$

Since the normalized Euclidean coordinates in (9.2) cannot be measured directly, a global invertible transformation (i.e., the pinhole camera model) is used to determine the normalized Euclidean coordinates from the corresponding pixel information as

$$\begin{bmatrix} p^T & 1 \end{bmatrix}^T = Am, \begin{bmatrix} p_f^T & 1 \end{bmatrix}^T = A_f m_f. \tag{9.4}$$

where A, $A_f \in \mathrm{R}^{3\times 3}$ are the known, constant, invertible, intrinsic camera calibration matrices for the eye-in-hand camera and the fixed camera, respectively.

Leveraging on our efforts in Mehta and Burks (2014), the depth of a fruit can be estimated via perspective transformation by assuming known geometry of the fruit variety as[1]

$$\hat{z}_f = \frac{f_f\left(\lambda_{xf} d_{ox} d_{iy} + \lambda_{yf} d_{oy} d_{ix}\right)}{2 d_{ix} d_{iy}} \tag{9.5}$$

$$d_{ix} = 2\left(\frac{A_p}{\pi\sqrt{1-e^2}}\right)^{1/2}, d_{iy} = 2\left(\frac{A_p\sqrt{1-e^2}}{\pi}\right)^{1/2} \tag{9.6}$$

where $\hat{z}_f \in \mathbb{R}$ denotes the estimated Euclidean depth of a fruit measured in F_f; d_{ox}, $d_{oy} \in \mathrm{R}$ denote the sample mean major and minor axes, respectively, of an ellipsoidal fruit, d_{ix}, $d_{iy} \in \mathrm{R}$ denote the major and minor axes, respectively, in the image plane, $A_p \in \mathrm{R}$ denotes the area of the fruit in the image plane of the fixed camera (in pixels), $e \in \mathrm{R}$ is the known eccentricity of the ellipse, the constant $f_f \in \mathrm{R}_{>0}$ represents

[1] In the presence of partial occlusions or clustered fruit, advanced methods such as perimeter detection and shape analysis techniques (Plebe and Grasso 2001; Hannan et al. 2010) can be used to directly obtain the image-space diameters d_{ix}, d_{iy} of the fruit to get $\hat{z}_f$ using (9.5).

the focal length in pixels for the fixed camera, and λ_{xf}, $\lambda_{yf} \in \mathrm{R}_{>0}$ are the scaling factors in the image x and y directions of the fixed camera, respectively.

Remark 9.1 *Any inaccuracy in estimating the fruit size A_p affects the major and minor axes, d_{ix} and d_{iy}, respectively. As stated in Remark 1 in* Mehta and Burks (2014)*, it can be shown that d_{ix}/d_{iy} is constant and hence the unknown depth ratio $\hat{z}_f / z_f$, denoted by $\gamma_z \in \mathrm{R}_{>0}$, is also constant.*

Controller Development

The objective is to locate the robot end-effector to the target fruit position for harvesting, i.e., to regulate the eye-in-hand camera coordinate frame F to the target fruit in the sense that $O_{F(t)} \to O^*$, where $O_{F(t)} \to \mathrm{R}^3$ denotes the time-varying position of the frame F measured in F_b. The control objective can be achieved by regulating the time-varying fruit pixel coordinates $p(t)$ to the desired image coordinates, and regulating the end-effector to the desired fruit depth. Hence, mathematically, the control objective can be stated as

$$p(t) \to p_d, p_d = \begin{bmatrix} u_0 & v_0 \end{bmatrix}^T \text{and} z(t) = z_d \tag{9.7}$$

where $z_d \in \mathrm{R}_{>0}$ denotes the maximum desired depth of the fruit in F, and u_0, $v_0 \in \mathrm{R}$ denote the pixel coordinates of the principal point (i.e., the intersection of an optical axis with the image plane) of the eye-in-hand camera.

As discussed earlier, the fixed camera can view an entire or part of a tree canopy. Using (9.5) and (9.6), the fixed camera can obtain a global fruit map and the corresponding harvesting sequence such as in Edan et al. (1991). However, the fruit to be harvested may not be visible to the eye-in-hand camera, say, because the eye-in-hand camera is pointing away from the fruit. Therefore, a nonlinear rotation controller is developed to orient the eye-in-hand camera such that the target fruit enters its FOV. The rotation controller uses the estimated fruit position obtained by the fixed camera to determine the desired orientation of the eye-in-hand camera. Once the target fruit is visible to the eye-in-hand camera, the translation controller regulates the end-effector to the target fruit using image feedback from the eye-in-hand camera.

(a) *Rotation Controller*

In this section, a controller is developed to orient the robot end-effector such that the target fruit enters the FOV of the eye-in-hand camera. Using (9.2), (9.5), and (9.6), let the estimated Euclidean position of the fruit in the fixed camera F_j be denoted by $\widehat{\bar{m}}_f \in \mathbb{R}^3$. The Euclidean coordinates $\widehat{\bar{m}}_f$ can be expressed in the eye-in-hand camera as $\widehat{\bar{m}}(t)$ (see (9) and (12) in Mehta and Burks 2014). Therefore, the objective is to align $\widehat{\bar{m}}(t)$ along the direction of the camera's optical axis $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$.

The rotation error $e_\omega(t) \in \mathrm{R}^3$ defined as orientation mismatch to bring the target fruit in the FOV of the eye-in-hand camera can be represented in terms of angle-axis representation as

$$e_\omega = u\theta \tag{9.8}$$

where $u(t) \in \mathrm{R}^3$ represents a unit axis of rotation such that $u(t) = \hat{\bar{m}}^{\cdot}(t) \wedge [0 \quad 0 \quad 1]^T$, and $\theta(t) = \cos^{-1} \hat{\bar{m}}(t), [0 \quad 0 \quad 1]^T \in \mathbb{R}$ denotes the rotation angle about $u(t)$ that brings $\hat{\bar{m}}(t)$ along the optical axis, such that $0 \leq \theta(t) \leq \pi$. In (9.8), $\hat{\bar{m}}(t) \in \mathbb{R}^3$ represents a unit vector along $\hat{\bar{m}}(t)$.

Based on the rotation error in (9.8), the angular velocity $\omega_c(t) \in \mathrm{R}^3$ of the camera can be designed using the following PD controller:

$$\omega_c = -k_{p\omega} \left(I_3 + k_{d\omega} L_\omega \right)^{-1} e_\omega \tag{9.9}$$

where $k_{p\omega}$, $k_{d\omega} \in \mathrm{R}_{>0}$ are the proportional and derivative control gains, respectively. Various loop tuning methods, such as Ziegler-Nichols and manual (trial-and-error), can be adopted to determine the control gains. In (9.9), I_3 denotes a 3 × 3 identity matrix, and $L_\omega(t) \in \mathrm{R}^{3\times 3}$ is a measurable Jacobian-like function defined as

$$L_\omega = I_3 - \frac{\theta}{2}[u]_\times + \left(1 - \frac{sinc(\theta)}{\sin c^2 \left(\frac{\theta}{2} \right)} \right) [u]_\times^2 \tag{9.10}$$

where sinc(θ) is the unnormalized sinc function, and $[u]_\times$ is the skew-symmetric matrix of $u(t)$. The determinant of $L_\omega(t)$ is $det(L_\omega) = 1/\mathrm{sinc}^2(\theta/2)$, thus being singular only at $\theta = 2k\pi \; \forall \; k \in N_{>0}$, i.e., outside of $0 \leq \theta(t) \leq \pi$.

Theorem 9.1 *The angular velocity control input in* (9.9) *ensures global exponential regulation of robot end-effector such that the target fruit is in the FOV of the eye-in-hand camera in the sense that*

$$e_\omega(t) = \zeta_0 \exp\{-\zeta_1 t\} \tag{9.11}$$

where ζ_0, $\zeta_1 \in R$ *denote positive bounding constants.*

Proof *For detailed stability analysis, readers are referred to* Mehta and Burks (2014).

(b) *Translation Controller*

The objective of the translation controller is to regulate the eye-in-hand camera to the target fruit position. Based on the control objective, the translation errors $e_{v1}(t) \in \mathrm{R}^2$ and $e_{v2}(t) \in \mathrm{R}$ can be defined as

$$e_{v1} = p_d - p \tag{9.12}$$

$$e_{v2} = z_d - \alpha \hat{z} \tag{9.13}$$

The error $e_{v1}(t) \in \mathrm{R}^2$ corresponds to regulating the fruit to the image center, and the error $e_{v2}(t) \in \mathrm{R}$ is designed to regulate the end-effector to the target fruit depth.

In (9.13), $\hat{z}(t) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \widehat{\bar{m}}(t)$ is the estimated depth of the target fruit from the eye-in-hand camera, and $\widehat{\bar{m}}(t)$ is the estimated fruit position in the eye-in-hand camera coordinate frame. The estimated fruit depth $\hat{z}(t)$ is assumed to be a continuous function of time. In (9.13), $\alpha \in \mathrm{R}_{>0}$ denotes a scaling factor such that $z < \alpha \hat{z} \forall t$. The constant α is selected based on an upper bound on the Euclidean depth estimation error, and, as a rule of thumb, α can be selected arbitrarily high to ensure that the robot reaches the target fruit despite any estimation errors. Since $z < \alpha \hat{z}$, the robot may overshoot the target fruit, and hence the end-effector is equipped with an infrared proximity sensor to stop once the fruit is reached.

Let $v_c(t) \triangleq \begin{bmatrix} v_{cx}(t) & v_{cy}(t) & v_{cz}(t) \end{bmatrix}^T$ be the linear velocity of the eye-in-hand camera and define $v_{c1} \triangleq \begin{bmatrix} v_{cx}(t) & v_{cy}(t) \end{bmatrix}^T$. Taking time derivative of (9.13), the linear velocity $v_{cz}(t)$ along the optical axis of the camera can be obtained as

$$v_{cz} = -k_{pv2} \left(1 + \alpha k_{dv2}\right)^{-1} e_{v2} \tag{9.14}$$

where $k_{pv2}, k_{dv2} \in \mathrm{R}_{>0}$ are the proportional and derivative control gains, respectively.

Taking the time derivative of the first expression in (9.4), the velocity of the eye-in-hand camera can be related to the velocity $\dot{p}(t) \in \mathbb{R}^2$ of the target centroid in the image frame as

$$\dot{p} = -\frac{1}{z} J_v^{'} v_{c1} + \frac{1}{z} J_v^{''} v_{cz} \tag{9.15}$$

where $J_v^{'}(u, v) \in \mathrm{R}^{2\times 2}$ and $J_v^{''}(u, v) \in \mathrm{R}^2$ are measurable image Jacobians. Since no orientation change is required during translation control, the image dynamics in (9.15) are obtained considering $\omega_c(t) = 0$..

Using (9.15), the velocity $v_{c1} \in \mathrm{R}^2$ along the x and y-axis of the eye-in-hand camera can be designed as

$$v_{c1} = \left(I_2 + \frac{k_{dv1}^{'}}{\hat{z}} J_v^{'} \right)^{-1} \left(-k_{pv1} e_{v1} + \frac{k_{dv1}^{'}}{\hat{z}} J_v^{''} v_{cz} \right) \tag{9.16}$$

where $k_{pv1}, k_{dv1}^{'} \in \mathrm{R}_{>0}$ are proportional and derivative control gains, respectively, such that $k_{pv1} = k_{pv11} + k_{pv12}$. In (9.16), the facts that $\hat{z} / z = \gamma$ and $z \leq \alpha \hat{z}$ are used.

Theorem 9.2 *The translation velocity control inputs* $v_{c1}(t)$ and $v_{cz}(t)$ *in* (9.16) *and* (9.14)*, respectively, ensure global exponential regulation of robot end-effector to the desired fruit depth in the sense that*

$$e_{v1}(t) \leq \zeta_2 \exp\{-\zeta_3 t\} \tag{9.17}$$

$$\left| e_{v2}(t) \right| = \zeta_4 \exp\{-\zeta_5 t\} \tag{9.18}$$

where $\zeta_2, \zeta_3, \zeta_4, \zeta_5 \in \mathrm{R}$ *are positive bounding constants.*

Proof *For detailed stability analysis, readers are referred to* Mehta and Burks (2014).

Experimental Validation

The performance of the developed visual servo controller was demonstrated using a 7 degree-of-freedom Robotics Research K1207 manipulator shown in Fig. 9.3. The indoor experiment comprised of an artificial citrus tree and two color CCD cameras (KT&C, KPCS20-CP1) with focal length of 4.3 mm and resolution of 640 × 480 that served as the fixed camera and the eye-in-hand camera. The fixed camera was mounted on stationary base of the manipulator as shown while the eye-in-hand camera was attached to the robot end-effector as shown in Fig. 9.3. Images from both the cameras were digitized using USB frame grabbers. The image processing workstation (IPW) was used to identify fruits from the captured images using the method described in Hannan et al. (2010). The robot control workstation (RCW) hosted a lower level controller to generate joint torque commands based on the control input from the IPW. Also, the RCW broadcasted the joint position feedback along with the end-effector position and orientation to the IPW using a real-time communication network.

The experiment was repeated several times for different robot and fruit positions. The actual fruit position O^* was measured using forward kinematic analysis by positioning the end-effector at the center of the fruit. Figure 9.4 shows the polar plot of the Euclidean distance error between $F(n)$ (red $\diamondsuit$) and O^* (blue $\triangle$). To assist in visualizing the results, the fruit is shown as an ellipse of axes $\{d_{ox}, d_{oy}\}$. From the fact that regulation error is less than the radius of the fruit and that the fruit was harvested successfully during each trial, the preliminary results indicate satisfactory performance of the developed visual servo controller.

Fig. 9.3 (**a**) Robotic Research K1207 manipulator with the fixed camera mounted on the stationary base of the robot and (**b**) the eye-in-hand camera located inside the robot end-effector

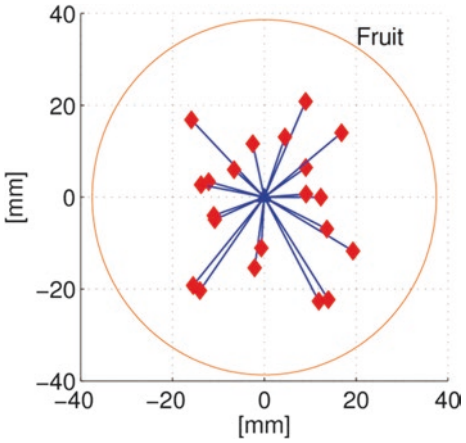


Fig. 9.4 Plot showing Euclidean distance error between the final position of eye-in-hand camera $F(n)$ (red) and the fruit centroid O^* (blue △) for the obtained 21 observations

9.3 Advanced Control Techniques

The performance of visual servo controllers in robotic harvesting is affected by endogenous as well as exogenous uncertainties. The uncertainties in the vision system specifications (e.g., camera calibration parameters), unmodeled robot dynamics, and uncertain fruit depth measurements can be regarded as endogenous factors. Whereas, environmental factors such as uneven and varying illumination, fruit occlusion, clustered fruits, obstacles, and fruit motion contribute to exogenous uncertainties. In this chapter, we will particularly focus on the issue that a fruit may not be stationary as reported in the early works of Harrell et al. (1989). Exogenous disturbances such as wind gusts, fruit detachment forces, canopy unloading, and robot-tree contact may cause an unknown time-varying fruit motion. If not considered during control system development, the fruit motion could result in unsuccessful pick cycle and reduced harvest efficiency. Robotic harvesters employing open-loop control or dead-reckoning are highly vulnerable to fruit motion since the fruit position is not updated during the reaching stage. The existing approaches to closed-loop visual servo control either do not consider fruit motion or implicitly rely on high-gain controllers (Mehta and Burks 2014; Barth et al. 2016) in conjunction with or separate from using high-frequency visual feedback (Harrell et al. 1989, 1990) for fruit motion compensation. Approaches using high-gain controllers are susceptible to measurement noise as the noise inadvertently gets amplified along with the feedback signal, which could lead to high-bandwidth actuation (causes chattering of the end-effector) and system instability. In the field conditions, where a fruit can easily be partially occluded or clustered, additional image processing becomes necessary to robustly identify the fruit to improve fruit detection rates, which limits the rate of image feedback. Passive approach to fruit motion compensation using high frequency image feedback may not be viable with the need for robust image processing (Muscato et al. 2005) and the desire to process higher resolution imagery for improved positioning accuracy. Additionally, as stated earlier,

passive approaches that do not take into account disturbance dynamics fail to guarantee stability and performance of the closed-loop harvesting system. Various researchers (Fortuna et al. 1994; Hayashi et al. 2002; Muscato et al. 2005; Mehta and Burks 2014; Bac et al. 2014) have expressed the need for advanced control methods to improve harvest efficiency when uncertainties in fruit detection and tracking arise. In this section, we will present robust and adaptive visual servo controllers that guarantee stability of the closed-loop harvester system in the presence of unknown fruit motion.

Robust and adaptive control approaches deal with uncertainties. Robust control methods do not require model knowledge of the uncertainty. However, a bound on the uncertain parameters is required to be known a priori. The robust controller is then designed for the "worst case" scenario using the bound. As a result, control energy expenditure in robust controllers can be high. On the other hand, adaptive control methods are based on the notion of adapting to a controlled system whose parameters are uncertain. Since the unknown parameters are adapted online, these methods can be computationally more expensive than robust control, but they require less control energy.

9.3.1 Robust Visual Servo Controller

The robust controller presented in this section uses the same cooperative camera framework presented in Sect. 9.2.2. The model-based fruit localization approach provides 3D position of the fruits in the fixed camera as well as the eye-in-hand camera, and the fruit is assumed to undergo an unknown but bounded fruit motion. The objective of robust controller is to position the eye-in-hand coordinate frame to the target fruit position in the presence of unknown fruit motion.

Controller Development

The above control objective can be achieved by regulating the time-varying fruit image coordinates $p(t)$ to the desired image coordinates, and driving the end-effector to the desired fruit depth. Hence, mathematically, the control objective can be stated as

$$p(t) \to p_d, p_d = \begin{bmatrix} u_0 & v_0 \end{bmatrix}^T \text{and} z(t) = z_d \tag{9.19}$$

where $z_d \in \mathrm{R}_{>0}$ denotes the maximum desired depth of the fruit in F, and u_0, $v_0 \in \mathrm{R}$ denote the pixel coordinates of the principal point (i.e., the intersection of an optical axis with the image plane) of the eye-in-hand camera.

(a) *Rotation Controller*

The rotation error in (9.8) depends on the relative orientation between a moving camera and a moving fruit. Therefore, the open-loop error system obtained by taking the time derivative of (9.8) contains two terms as

$$\dot{e}_{\omega} = L_{\omega}\omega_c + d_{\omega} \tag{9.20}$$

where $L_{\omega}(t) \in \mathrm{R}^{3\times 3}$ is defined in (9.10) and $d_{\omega} \in \mathrm{R}^3$ is the exogenous disturbance as a result of fruit motion. The disturbance is assumed to be bounded such that $|d_{\omega}| \leq \gamma_{\omega}$, where $\gamma_{\omega} \in \mathrm{R}_{>0}$ and $|\cdot|$ denotes the L^2 vector norm. This is a valid assumption because the fruit moves with a finite velocity.

Based on the open-loop error dynamics in (9.20) and the subsequent stability analysis, the angular velocity of the camera can be designed as

$$\omega_c = -k_{\omega}e_{\omega} - \frac{e_{\omega}\gamma_{\omega}^2}{e_{\omega}\gamma_{\omega} + \varepsilon_{\omega}} \tag{9.21}$$

where $k_{\omega} \in \mathrm{R}_{>0}$ is the control gain, and $\varepsilon \in \mathrm{R}_{>0}$ is chosen to be arbitrarily small.

Theorem 9.3 *The rotation controller orients the end-effector such that the target fruit appears arbitrarily close to the center of the eye-in-hand camera. Formally, the rotation control input developed in* (9.21) *ensures uniformly ultimately bounded regulation of the end-effector in the sense that*

$$\left|e_{\omega}(t)\right| \leq \zeta_0 \exp\{\zeta_1 t\} + \zeta_2 \tag{9.22}$$

where ζ_0, ζ_1, *and* $\zeta_2 \in \mathrm{R}$ *denote positive bounding constants.*

Proof *For detailed stability analysis, readers are referred to* Mehta et al. (2016).

(b) *Translation Controller*

Assuming the estimated fruit depth $\hat{z}(t)$ to be a continuous function of time, the open-loop error dynamics for depth regulation can be obtained by taking time derivative of (9.13) as

$$\dot{e}_{v2} = \alpha\xi v_{cz} + d_z \tag{9.23}$$

where $d_z(t) \in \mathrm{R}$ is the component of fruit motion along the optical axis, such that $|d_z(t)| \leq \gamma_d$ for $\gamma_d \in \mathrm{R}_{>0}$, and ξ denotes the constant depth ratio $\hat{z}/z$. γ_d indicates an upper bound on the fruit velocity along the optical axis due to disturbance. Based on (9.23), the linear velocity $v_{cz}(t)$ of the eye-in-hand camera along the optical axis can be designed as

$$v_{cz} = k_z e_{v2} + \frac{e_{v2}\gamma_d^2}{e_{v2}\gamma_d + \varepsilon_z} - w\left|e_{v1}\right|^2 \tag{9.24}$$

where $k_z = k_{z1} + k_{z2} \in \mathrm{R}_{>0}$ is the constant control gain, $\varepsilon_z \in \mathrm{R}_{>0}$ is an arbitrarily small design constant, and $w \in \mathrm{R}_{>0}$ is the user defined weight on the term $|e_{v1}|^2$. The control gain k_z determines response of the controller to the input error $e_{v2}(t)$; however, large gains should be avoided in the presence of measurement noise.

To design the linear velocity $v_{c1}(t)$, an open-loop error system can be obtained by taking the time derivative of (9.12) as

$$\dot{e}_{v1} = -\frac{1}{z} J_v \dot{v}_c - \frac{1}{z} J_v \ddot{v}_{cz} + \dot{d}_p. \tag{9.25}$$

Based on the open-loop error system in (9.25) and the subsequent stability analysis, the linear control velocity $v_{c1}(t) \in \mathrm{R}^2$ of the eye-in-hand camera can be designed as

$$v_{c1}(t) = J_v^{-1} \left(\alpha \hat{z} k_p e_{v1} - J_v \ddot{v}_{cz} + \frac{\alpha \hat{z} e_{v1} \gamma_p^2}{e_{v1} \gamma_p + \varepsilon_p} \right) \tag{9.26}$$

where $k_p \in \mathrm{R}_{>0}$ is the constant control gain, and $\varepsilon_p \in \mathrm{R}_{>0}$ is an arbitrarily small design constant. Similar to k_z, the control gain k_p determines response of the controller to input error $e_{v1}(t)$.

Theorem 9.4 *The translation controller guarantees that the end-effector is placed arbitrarily close to the target fruit location. Formally, the translation control inputs developed in* (9.24) *and* (9.26) *ensure uniformly ultimately bounded regulation of the end-effector in the sense that*

$$\begin{aligned} \left| e_{v1}(t) \right| &\leq \zeta_3 \exp\{\zeta_4 t\} + \zeta_5 \\ \left| e_{v2}(t) \right| &\leq \zeta_6 \exp\{\zeta_7 t\} + \zeta_8 \end{aligned} \tag{9.27}$$

where $\zeta_3, \zeta_4, \zeta_5, \zeta_6, \zeta_7, \zeta_8 \in \mathrm{R}$ *denote positive bounding constants.*

Proof *For detailed stability analysis, readers are referred to* Mehta et al. (2016).

Experimental Validation

We used an artificial citrus fruit suspended in the air to imitate a fruit attached to a stem. The fruit was manually perturbed in the experiment, and the performance of the controller with and without robust feedback elements was recorded. The displacement of the fruit centroid was ≈120 mm. Figure 9.5 shows the rotation and translation errors as a function of time. It can be seen that the robust controller offers improved compensation to fruit motion as compared to a high-gain controller. The motion of the fruit centroid in the image plane of the eye-in-hand camera is shown in Fig. 9.6, where it can be seen that the fruit is satisfactorily regulated to the image center even in the presence of fruit motion.

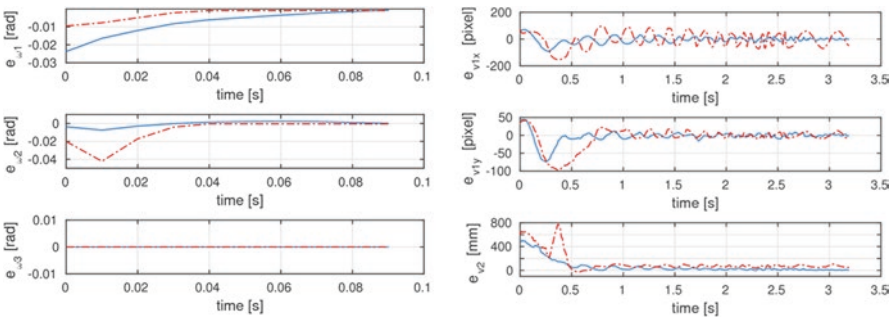


Fig. 9.5 Rotation error $e_\omega(t) = [e_{\omega 1}(t) e_{\omega 2}(t) e_{\omega 3}(t)]^T$ and translation error $e_v(t) = \left[e_{v1}^T(t) e_{v2}(t)\right]^T$ where $e_{v1}(t) = [e_{v1x}(t) e_{v1y}(t)]^T$ using the proposed robust controller (blue line) and without robust feedback terms (red dash-dot line)

9.3.2 Adaptive Visual Servo Controller

The robust control approach presented in the previous section compensates for the unknown fruit motion by including robust feedback terms that were designed to upper bound a nonlinear disturbance. The designed controller guaranteed that a robot can be regulated arbitrarily close to fruit in the presence of fruit motion (see Mehta et al. (2016) for details). The motivation behind using direct adaptive control approaches is to "learn" and compensate the fruit motion in real time. With the gained knowledge of the fruit motion, effective compensation can be offered as against using robust disturbance bounding terms. On the downside, learning requires that the functional form of the fruit motion is known. Instead of considering an arbitrary fruit motion as in Mehta et al. (2016), it is assumed that the suspended fruit follows the motion of a simple pendulum, which is a mild assumption for fruits with long stems such as orange. The fruit motion is analyzed in the image plane and along the optical axis of the camera using a second-order spring-mass system. By linearly parameterizing the motion dynamics, an adaptive update law is designed to identify the unknown fruit motion, and the developed adaptive visual servo controller regulates the robot to a target fruit. To account for modeling uncertainties, robust feedback elements are also included in the control structure.

Fruit Motion Modeling

A suspended fruit, similar to simple pendulum, is considered to move in x, y, and z directions with respect to F. Using the small angle approximation (i.e., angular displacement $\ll$1 rad), a pendulum can be considered a harmonic oscillator. The fruit motion in the image-space is analyzed as a combination of two second-order spring-mass systems.

For a fruit modeled using a spring-mass system as shown in Fig. 9.7 and observed by a stationary camera, its motion can be analyzed by separating the motion in the

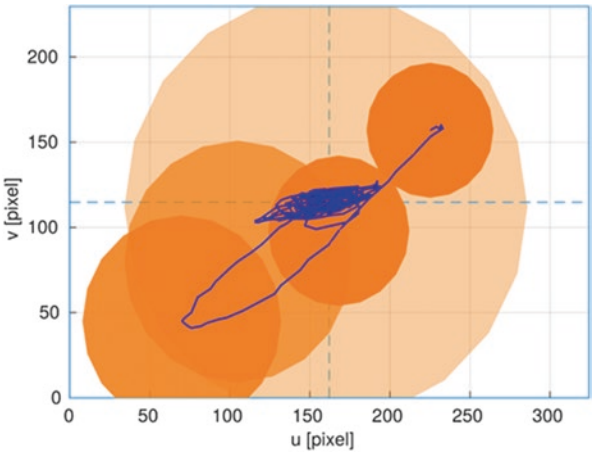


Fig. 9.6 Time-varying fruit position in the image plane of the camera-in-hand. The trajectories in blue refer to the observed fruit position during rotation and translation control. The relative size of the fruit in the image plane at different time instances during translation control is also shown

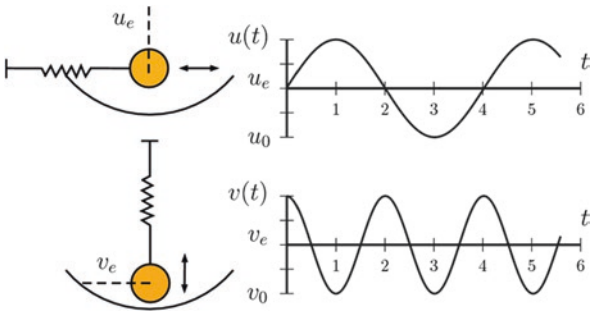


Fig. 9.7 Fruit motion in the image plane is modeled as a spring-mass system

image plane $(\dot{u}(t), \dot{v}(t))$ from the out-of-plane motion $\dot{z}(t)$ along the optical axis of the camera as

$$\begin{aligned} \dot{u} &= u_0 \omega \cos(\omega t) \\ \dot{v} &= -2v_0 \omega \sin(2\omega t) \\ \dot{z} &= z_0 \omega \cos(\omega t) \end{aligned} \tag{9.28}$$

where u_0, $v_0 \in \mathrm{R}$ is the unknown amplitude of motion in the image plane, $z_0 \in \mathrm{R}$ is the unknown amplitude of motion along the optical axis, and $\omega \in \mathrm{R}$ is the unknown angular frequency of motion.

Using polynomial approximation of cyclic functions in (9.28) and segregating the unknown parameters u_0, v_0, and ω from the known functions of time, i.e., after linearly parameterizing the uncertainty, the image dynamics can be obtained as

$$\dot{p} = -\frac{1}{z} J_{v1} v_c^{'} + \frac{1}{z} J_{v2} v_z + J_\omega \omega_c + Y_p \Theta_p + R_p \tag{9.29}$$

where $Y_p(t) \in \mathrm{R}^{2\times 2n}$ is the regression matrix of known functions of time, $\Theta_p \in \mathrm{R}^{2n}$ is the vector of constant unknown parameters, and $R_p \in \mathrm{R}^2$ is defined as

$$R_p = \begin{bmatrix} u_0 \omega R_1 \\ -2 v_0 \omega R_2 \end{bmatrix} \tag{9.30}$$

such that $|R_p| \leq \gamma_p$ for any $\gamma_p \in \mathrm{R}_{>0}$. Similarly, the depth dynamics can be obtained as

$$\dot{z} = -v_z + Y_z \Theta_z + R_z \tag{9.31}$$

where $Y_z(t) \in \mathrm{R}^{1\times n}$ is a vector of known functions of time, $\Theta_z \in \mathrm{R}^n$ is the vector of constant unknown parameters, and $R_z = z_0 \omega R_1$ such that $|R|_z \leq \gamma_z$ for any $\gamma_z \in \mathrm{R}_{>0}$.

Controller Development

The rotation controller to regulate the orientation of the eye-in-hand camera to view the target fruit is the same as given in (9.21). In this section, we will design an adaptive translation control law. The regulation error in the image plane $e_p(t) \in \mathrm{R}^2$ and along the optical axis $e_z(t) \in \mathrm{R}$ can be defined as

$$\begin{aligned} e_p &= p - p_d \\ e_z &= z - z_d \end{aligned} \tag{9.32}$$

where the fruit depth $z(t) \in \mathrm{R}$ is assumed to be known (e.g., using triangulation).

Based on the error dynamics and the stability analysis, the velocity of the robot along the optical axis $v_z(t) \in \mathrm{R}$ can be designed as

$$v_z = k_z e_z + Y_z \hat{\Theta}_z + \frac{e_z \gamma_z^2}{e_z \gamma_z + \varepsilon_z} \tag{9.33}$$

where $k_z \in \mathrm{R}_{>0}$ is the control gain, $\hat{\Theta}_z(t) \in \mathbb{R}^n$ is the time-varying estimate of the unknown parameter vector Θ_z, and $\varepsilon_z \in \mathrm{R}_{>0}$ is an arbitrarily small design constant. The estimate $\hat{\Theta}_z(t)$ in (9.33) is obtained using the following parameter update law:

$$\hat{\Theta}_z = proj\left(\Gamma_z Y_z^T e_z\right) \tag{9.34}$$

where $\Gamma_z \in \mathrm{R}^{n\times n}$ is the positive definite gain matrix, and *proj* denotes the normal projection algorithm, which ensures that the elements $\hat{\Theta}_{zi}(t) \forall i = 1, 2, \cdots, n$ of $\hat{\Theta}_z(t)$ are bounded as $\underline{\Theta}_{zi} \leq \hat{\Theta}_{zi}(t) \leq \overline{\Theta}_{zi}$, where $\underline{\Theta}_{zi}, \overline{\Theta}_{zi} \in \mathbb{R}$ denote the known constant lower and upper bounds of $\Theta_{zi}(t)$, respectively.

Similarly, the camera velocity $v_c^{'}(t) \in \mathrm{R}^2$ in the *xy*-plane of *F* can be obtained as

$$v_c^{'} = z J_{v1}^{-1} \left(k_p e_p + \frac{1}{z} J_{v2} v_z + J_\omega \omega_c + Y_p \hat{\Theta}_p + \frac{e_p \gamma_p^2}{e_p \gamma_p + \varepsilon_p} \right) \tag{9.35}$$

where $\hat{\Theta}_p(t) \in \mathbb{R}^{2n}$ is the estimate of the unknown vector Θ_p that is obtained using

$$\hat{\Theta}_p = proj\left(\Gamma_p Y_p^T e_p\right) \tag{9.36}$$

where $k_p \in \mathrm{R}_{>0}$ is the control gain, $\varepsilon_p \in \mathrm{R}_{>0}$ is an arbitrarily small design constant, and $\Gamma_p \in \mathrm{R}^{2n \times 2n}$ is a positive definite adaptation gain matrix.

Theorem 9.5 *The adaptive visual servo controller in* (9.33)–(9.36) *guarantees that the end-effector is placed arbitrarily close to the target fruit location. Formally, the adaptive controller ensures uniformly ultimately bounded regulation of the fruit in the sense that*

$$\begin{aligned} e_p(t) &\le \zeta_0 \exp\{-\zeta_1 t\} + \zeta_2 \\ |e_z(t)| &\le \zeta_3 \exp\{-\zeta_4 t\} + \zeta_5 \end{aligned} \tag{9.37}$$

where $\zeta_0, \zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5 \in \mathrm{R}$ *denote positive bounding constant.*

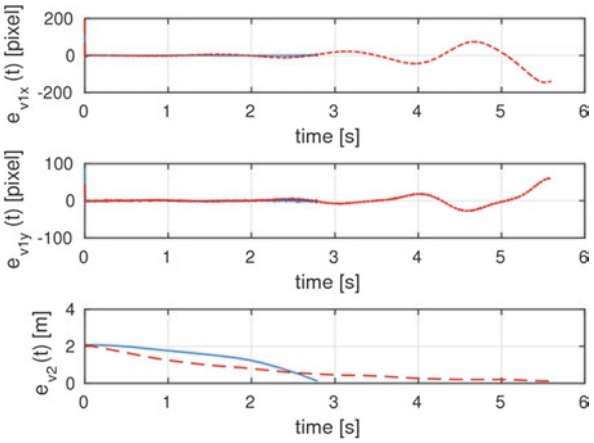
Proof *For detailed stability analysis, readers are referred to* Mehta and Burks (2016).

Simulation Results

A numerical simulation was performed to demonstrate the performance of the adaptive controller. A non-vanishing disturbance was assumed to perturb the fruit with velocity causing the fruit centroid to oscillate with amplitude of $\approx d_c = 210$ mm. The image coordinates of the fruit were assumed to be affected by a zero-mean Gaussian noise of standard deviation one pixel. The performance of the developed adaptive controller was compared with a pure high-gain controller.

Figure 9.8 compares the regulation error of the adaptive controller with a high-gain controller. For clarity, the transient response of the adaptive controller using larger time scale is shown in Fig. 9.9. It can be seen from Fig. 9.9 that the regulation errors in the image plane approach zero as the adaptive controller learns the fruit motion, thus offering excellent disturbance rejection.

Figure 9.10a shows the image-space trajectory of the fruit centroid during the closed-loop operation using the adaptive controller, and the corresponding image-space trajectory using the high-gain controller is shown in Fig. 9.10b. The adaptive controller regulates the fruit close to the image center (blue line) as shown in Fig. 9.10a while the high-gain controllers demonstrate poor performance and potentially unstable closed-loop operation (refer to Fig. 9.10b).



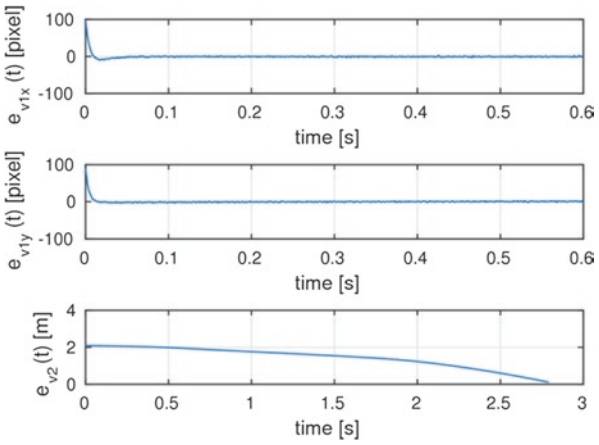


Fig. 9.8 Error comparison between the adaptive controller (blue line) and the high-gain controller (red dotted line)

Fig. 9.9 Error $e_v(t) = \left[e_{v1}^T(t) e_{v2}(t)\right]^T$ for d_c = 210 mm, where $e_{v1}(t) = [e_{v1x}(t) e_{v1y}(t)]^T$ for the proposed adaptive controller

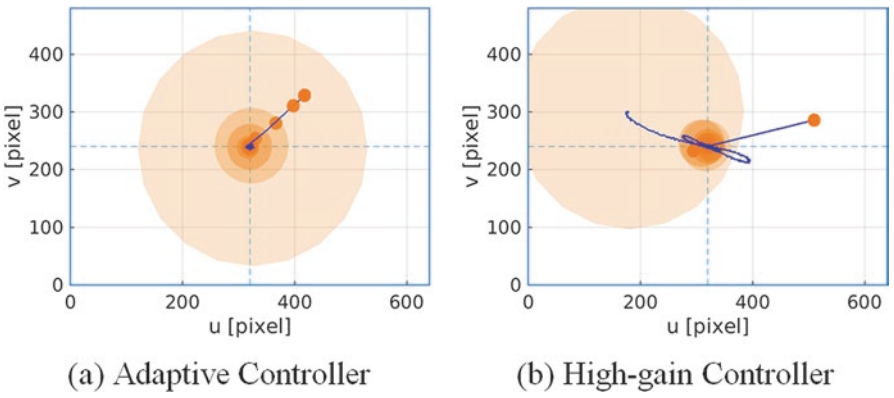


Fig. 9.10 Time-varying fruit position (blue line) and relative fruit size in the image plane of the camera

9.4 Summary and Concluding Thoughts

As discussed earlier, the stability of robotic harvesters is affected by uncertainties in the system and environment including fruit motion, obstacles, illumination changes, and uncertain camera parameters. Although stability is utmost important and must be ensured, the control methods in robotic harvesting should also take into account the performance of the system. The performance can be quantified in several ways, such as the total control energy or the convergence time of a controller. Controllers guaranteeing stability as well as performance will improve harvesting economics. To this end, below are some of the future research directions that are aimed at developing robust and performance-oriented controllers.

Finite-Time and Fixed-Time Controllers
For the robotic harvesting systems to be able to replace labor force, it is necessary that the pick-cycle times of a robotic harvester are comparable to that of the manual operation. A significant portion of the pick-cycle time is the time required for the robot to reach the fruit, which depends on the performance of the controller. Robotic harvesters relying on constant velocity during the approach phase can reduce pick-cycle times by increasing the forward velocity. However, higher forward velocity in the vicinity of fruit is more likely to result in unsuccessful harvest attempt. The closed-loop 3D translation controllers developed in prior sections guarantee stability of the closed-loop system, i.e., the robot will be successfully regulated to the constant or time-varying fruit position, but the convergence time is not well-defined. The controller in Sect. 9.2.2 provides the rate of convergence (i.e., the translation error decays exponentially), but theoretically it takes infinite time to reach the exact fruit location. It may not a concern in practice since the robot is only be required to reach in the vicinity of the fruit for the end-effector to successfully harvest the fruit. This is because the end-effectors can usually accommodate small positioning errors. The robust and adaptive controllers designed in Sect. 9.3 guarantee UUB stability, i.e., the position errors decay exponentially to a ball of known radius at the origin and remain inside the ball. The size of the ball can be designed to ensure that the end-effector successfully picks the fruit once the errors are reduced to this ball. As a result, the controllers developed in this chapter guarantee that the robot reaches in the *vicinity* of the fruit in finite time. The time required depends on the control gains as well as the initial position of the robot with respect to the fruit.

The future directions in controller development should include development of robust finite-time and fixed-time controllers. The finite-time controllers can guarantee that the position of the robot is regulated to the exact fruit position, not in the vicinity of the fruit, in finite time. Among others, the finite convergence time depends on the initial position error between the robot and the fruit. Whereas, fixed-time controllers guarantee fixed convergence time that is independent of the initial position of the robot with respect to the fruit. Moreover, robust finite-time or fixed-time controllers can ensure robustness with respect to environmental and system uncertainties. This blend of stability and performance is important to improve harvest efficiency and reduce pick-cycle times.

Human-Robot Collaboration

The performance of autonomous and robotic systems can severely deteriorate in complex agricultural environments due to unmodeled effects and unforeseen events. While autonomous systems are preferred due to cost-effectiveness and computational benefits, given the brittleness of automation in complex environments, human involvement becomes crucial to improve harvesting efficiency. Therefore, the growing interest is in augmenting the capabilities of autonomous robotic systems with the expert knowledge of humans through human-robot collaboration. Human collaborators with their reasoning skills and expert knowledge gained through technical practices, training, and experience can overcome the shortfalls of autonomous systems pertaining to perception, planning, and mobility. To this effect, human collaborators may interact with the autonomous system by providing information in response to undocumented situations or by correcting the behavior of the system when deemed erroneous or inefficient.

Based on the nature of interaction, continuous or intermittent, various collaborative control architectures, such as shared control and supervisory control can be explored. In supervisory control, a human supervisor monitors the activity of the robot and intermittently issues commands to modify the behavior of the robot. The supervisor often issues broad task-level commands instead of controlling the motion of the robot at an execution-level. For example, if an imminent collision is perceived while the robot is approaching a fruit, the supervisor can invoke "obstacle avoidance control" which would then be executed by the robot. The separation between human intervention and autonomous execution in supervisory control is overcome with shared control that enables both the autonomy and the human to participate in execution-level control. The control input to the robot can be obtained by appropriately fusing the human and the autonomy control inputs (Ton et al. 2018). With reference to the above example, the human operator can directly issue control inputs to the robot to avoid obstacles while the robot approaches a target fruit. The shared control architecture is particularly beneficial in agricultural robotics as it can fuse robust and optimal control inputs to appropriately balance the performance and robustness for improved harvest efficiency.

References

Bac C, van Henten J, Hemming J, Edan Y (2014) Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. J Field Robot 31(6):888–911

Baeten J, Donné K, Boedrij S, Beckers W, Claesen E (2008) Autonomous fruit picking machine: a robotic apple harvester. Field Serv Rob:531–539

Barth R, Hemming J, van Henten J (2016) Design of an eye-in-hand sensing and servo control framework for harvesting robotics in dense vegetation. Biosyst Eng 146:71–84

Brown K (2002) Mechanical harvesting systems for the Florida citrus juice industry. In: 2002 ASAE annual meeting

Buemi F, Massa M, Sandini G, Costi G (1996) The AGROBOT project. Adv Space Res 18(1):185–189

Bulanon M, Okamoto M, Hata I (2005) Feedback control of manipulator using machine vision for robotic apple harvesting. In: 2005 ASAE annual meeting
Canny J (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 6:679–698
Cho I, Chang J, Kim Y, An J (2002) AE-automation and emerging technologies: development of a three-degrees-of-freedom robot for harvesting lettuce using machine vision and fuzzy logic control. Biosyst Eng 82(2):143–149
De-An Z, Jidong L, Wei J, Ying Z, Yu C (2011) Design and control of an apple harvesting robot. Biosyst Eng 110(2):112–122
Edan Y, Flash T, Peiper M, Shmulevich I, Sarig Y (1991) Near-minimum-time task planning for fruit-picking robots. IEEE Trans Robot Autom 7(1):48–56
Edan Y, Rogozin D, Flash T, Miles E (2000) Robotic melon harvesting. IEEE Trans Robot Autom 16(6):831–835
Foglia M, Reina G (2006) Agricultural robot for radicchio harvesting. J Field Robot 23(6–7):363–377
Fortuna L, Muscato G, Nunnari G, Pandolfo A, Plebe A (1994) Application of neural control in agriculture: an orange picking robot. Acta Horticulturae 406:441–450
Gongal A, Amatya S, Karkee M, Zhang Q, Lewis K (2015) Sensors and systems for fruit detection and localization: a review. Comput Electron Agric 116:8–19
Grand d'Esnon A (1985) Robotic harvesting of apples. Agri-Mation 1:210–214
Grand d'Esnon A Rabatel G, Pellenc R, Journeau A, Aldon M (1987) MAGALI: a self-propelled robot to pick apples. In: Proceedings of American Society of Agricultural Engineers, pp 87–1037
Hannan W, Burks F (2004) Current developments in automated citrus harvesting. In: 2004 ASAE annual meeting
Hannan W, Burks F, Bulanon M (2010) A machine vision algorithm combining adaptive segmentation and shape analysis for orange fruit detection. Agri Eng Int CIGR EJournal, 1–17
Harrell C, Adsit D, Slaughter C (1985) Real-time vision-servoing of a robotic tree fruit harvester. ASAE Paper 85:1–15
Harrell C, Slaughter C, Adsit D (1989) A fruit-tracking system for robotic harvesting. Mach Vis Appl 2(2):69–80
Harrell C, Adsit D, Pool A, Hoffman R (1990) The Florida robotic grove-lab. Trans ASAE 33(2):391–399
Hayashi S, Ganno K, Ishii Y, Tanaka I (2002) Robotic harvesting system for eggplants. Jpn Agri Res Q JARQ 36(3):163–168
Hayashi S, Shigematsu K, Yamamoto S, Kobayashi K, Kohno Y, Kamata J, Kurita M (2010) Evaluation of a strawberry-harvesting robot in a field test. Biosyst Eng 105(2):160–171
Kondo N, Nishitsuji Y, Ling P, Ting C (1996) Visual feedback guided robotic cherry tomato harvesting. Trans ASAE 39(6):2331–2338
Levi P, Falla A, Pappalardo R (1988) Image controlled robotics applied to citrus fruit harvesting. In: 7th international conference on robot vision and sensory controls
Li P, Lee H, Hsu Y (2011) Review on fruit harvesting method for potential use of automatic fruit harvesting systems. Proced Eng 23:351–366
Mehta S, Burks F (2014) Vision-based control of robotic manipulator for citrus harvesting. Comput Electron Agric 102:146–158
Mehta S, Burks F (2016) Adaptive visual servo control of robotic harvesting systems. IFAC-PapersOnLine 49(16):287–292
Mehta S, MacKunis W, Burks F (2016) Robust visual servo control in the presence of fruit motion for robotic citrus harvesting. Comput Electron Agric 123:362–375
Muscato G, Prestifilippo M, Abbate N, Rizzuto I (2005) A prototype of an orange picking robot: past history, the new robot and experimental results. Indust Robot Int J 32(2):128–138
Plebe A, Grasso G (2001) Localization of spherical fruits for robotic harvesting. Mach Vis Appl 13(2):70–79

Recce M, Taylor J, Plebe A, Tropiano G (1996) Vision and neural control for an orange harvesting robot. In: Proceeding of neural networks for identification, control, robotics, and signal/image processing international workshop, pp 467–475

Rosten E, Drummond T (2006) Machine learning for high-speed corner detection. In: European conference on computer vision, pp 430-443

Sarig Y (1993) Robotics of fruit harvesting: a state-of-the-art review. J Agric Eng Res 54(4):265–280

Shi J (1994) Good features to track. In: Proceedings of IEEE conference on computer vision and pattern recognition, pp 593-600

Sobel I, Feldman G (1968) A 3x3 isotropic gradient operator for image processing. In: A talk at the stanford artificial project, pp 271–272

Tanigaki K, Fujiura T, Akase A, Imagawa I (2008) Cherry-harvesting robot. Comput Electron Agric 63(1):65–72

Tillett D (1993) Robotic manipulators in horticulture: a review. J Agric Eng Res 55(2):89–105

Ton C, Kan Z, Mehta S (2018) Obstacle avoidance control of a human-in-the-loop mobile robot system using harmonic potential fields. Robotica 36(4):463–483

Van Henten J, Hemming J, Van Tuijl J, Kornet J, Meuleman J, Bontsema J, Van Os A (2002) An autonomous robot for harvesting cucumbers in greenhouses. Auton Robot 13(3):241–258

Van Henten J, Hemming J, Van Tuijl J, Kornet G, Bontsema J (2003) Collision-free motion planning for a cucumber picking robot. Biosyst Eng 86(2):135–144

Zhao Y, Gong L, Huang Y, Liu C (2016) A review of key techniques of vision-based control for harvesting robot. Comp Electron Agri 127:311–323

Chapter 10
Guidance, Auto-Steering Systems and Control

Riikka Soitinaho and Timo Oksanen

10.1 Introduction

This chapter describes the fundamentals of automatic guidance systems, including methods for positioning, navigation, and control. We begin with a short introduction to some central terms and concepts that are the basis of understanding the technical descriptions later on. We also introduce a brief history of automatic guidance in agriculture and highlight some of the motivations and requirements for modern automatic guidance systems before proceeding to explain the basics of the technical solutions in more detail.

10.1.1 Terminology and Concepts

Automatic guidance: Sometimes also referred to as automated guidance, auto-guidance, or guidance. Exact definition may vary across literature. Here we prefer the term automatic guidance. Automatic guidance comprises the tasks of positioning, navigation, control, and actuation, a system that is capable of autonomously guiding a vehicle to perform a task where the vehicle needs to move from one place to another.

Auto-steering: As opposed to manual steering, automation controls the steering actuator, not a human being, by turning the steering wheel. A system that makes it possible to steer a vehicle without a human driver, including software and components for actuation.

R. Soitinaho · T. Oksanen (✉)
Technical University of Munich, München, Germany
e-mail: timo.oksanen@tum.de

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_10

Control: A means to regulate a process variable. A controller manipulates the process variables. To apply a correction to a process variable based on the deviation of measured and desired signal with the help of actuators.

GNSS (Global Navigation Satellite System): A general term for global coverage satellite-based navigation utility systems that provide users with geo-spatial positioning and timing information, for example, GPS, GLONASS, Galileo and BeiDou.

GPS (Global Positioning System): The Global Positioning System is a satellite-based navigation utility system, owned by the United States, that provides users with positioning, navigation, and timing services.

Guidance directrix: A guidance directrix, or sometimes called just a *directrix*, is a course or direction for the guidance to follow.

Guidance Objective: A guidance objective is a general term for a target that is set for the automatic guidance system. In practice, a guidance objective could be formulated as "follow the furrow."

Implement: An agricultural implement is a tool that is used to carry out agricultural processes. For operation, implement is connected to a tractor using either hitch, hook, or drawbar. There are different types of implements, for example, for irrigation, cultivation, planting, and harvesting.

Motion planning: Motion planning is the task or process of finding a sequence of valid configurations to move a robot (for example, an autonomous vehicle) from one place to another. The term valid is used in two regards: *valid as in* the robot is capable of moving to such a position and orientation (pose, configuration), considering all possible constraints resulting from the kinematics and dynamics of the robot itself, but also *valid as in* the solution does not result in any collisions with the environment.

Navigation: A typical definition of navigation usually includes at least the task of *finding a way from one place to another*. Sometimes, also correcting the course, orientation, and speed to arrive at the desired location is considered to be part of navigation.

Positioning: Determining position (and orientation) in reference to some defined reference system (for example, a geodetic datum).

Position, location, orientation, heading, course, track: In a three-dimensional space, a rigid body has 6 degrees of freedom, which means 6 values are needed to describe the placement of the object. Position refers to the 3D coordinates of the object, whereas orientation (angular position) describes the 3D rotations of the object. To be more precise, each rotation is measured as a displacement from a defined reference orientation and with respect to a specified rotation axis. In summary, we need both position and orientation to fully describe how an object is placed in space. Same definitions apply to a two-dimensional space. The terms position and location are interchangeable to some extent, but here we prefer the term position. The term heading is typically used to indicate the direction of the front of a vehicle (which might differ from the travel direction of the vehicle). The terms course and track are used synonymously to indicate the travel direction of the vehicle. In the context of satellite navigation, heading is usually

defined as the compass direction in which the device (for example a GPS receiver) is travelling.

Path: A path only considers geometric constraints, i.e., how to move from pose A to pose B (position and orientation). Therefore, a path is a sequence of poses from pose A to pose B.

Path planning: Path planning is the task or process of finding a valid (collision-free) path (only considering geometric constraints) from pose A to pose B.

Path tracking: Path tracking is the task or process of determining the right actions in order to minimize the deviation of the true (actual) pose from the planned pose while travelling a path. With certain vehicle applications, we consider, for example, lateral deviation and deviation in heading.

Semiautonomous vehicle: A vehicle that is not fully autonomous but contains many automated functions and most of the time can work in autonomous manner but still under supervision of human being. It usually means that a human driver is on-board but vehicle navigates automatically.

Sensors: An accelerometer is a device that measures acceleration, that is, the rate of change of the velocity of an object. A three-axis accelerometer can measure acceleration in the direction of three different axes. An accelerometer senses both static and dynamic forces of acceleration, including gravity. A compass is a device that can indicate direction. The term is typically used to refer to a magnetic compass that indicates a direction relative to the magnetic field of the Earth. A GPS receiver is a device that can receive signals from GNSS satellites and based on the information calculate its own geographical location. A gyroscope is a device that can measure orientation and angular velocity. An inertial measurement unit (IMU) is a device that combines accelerometers, gyroscopes, and sometimes magnetometers (a compass) to provide measurements of orientation, angular rate and gravitational force. A lidar (originally an abbreviation for *Light Detection and Ranging*, sometimes also called a laser scanner), sometimes called a laser scanner or a laser rangefinder, is a device that can compute the distance to objects by emitting a laser light and measuring the travel time of the reflection. An odometer is a device that can measure the distance travelled by a vehicle, for example, a car.

Steering actuator: A steering actuator is a mechanism to electronically control the steerable components of the vehicle, usually realized by electro-hydraulic mechatronics. An actuator is typically defined as a device that can convert energy into motion. A steering actuator is an actuator for steering commands, for example, an electric motor that provides the rotation for a wheel.

Trajectory: A trajectory (as opposed to a path) also considers time constraints when moving from pose A to pose B (position and orientation over time).

Trajectory planning: Trajectory planning is the task or process of planning a valid (collision-free) trajectory (considering also timing constraints) from pose A to pose B.

10.1.2 History, Motivation, and Requirements for Automatic Guidance

Automatic guidance, sometimes also referred to as *automated guidance*, *auto-guidance*, or *guidance*, is a system that is capable of autonomously guiding a vehicle to perform a task where the vehicle needs to move from one place to another. For example, autonomous spraying of a field requires an automatic guidance system.

In the future, agriculture will be increasingly more automated. This poses technological challenges but also great opportunities to implement more precise, efficient, and environmentally friendly ways of farming and cultivation. Knowledge and understanding of how to automate agricultural processes are essential in order to design and maintain these systems. Like in many other industries, where a similar trend of automation has been seen over the past few decades, one essential motivation for agricultural automation is to allow human workforce to attend to other tasks that require capabilities that automation cannot yet provide. For example, automatic guidance can free the human operator from adjusting the course of the agricultural vehicle and allow them to concentrate on operating the equipment.

Automatic guidance can provide multiple benefits for crop production as well as for the environment. Even with declining workforce, productivity can be increased with automation. Sometimes also cost benefits can be achieved, and the fast progress of automation can eventually provide added safety and reliability. Precision and efficiency are increased by reducing overlaps and minimizing gaps in field operations, thus also reducing the amount and cost of fertilizer and other chemicals. This in turn minimizes the environmental impact and helps protecting the surrounding habitats. Automation, including automatic guidance, can also enable variable rate applications and other applications that are difficult without automation and modern sensor systems. In some cases, an automatic guidance system can work in conditions that are difficult for a human driver.

Former surveys of the topic provide a wide overview of the early ideas and development of automatic guidance in agriculture. For example, Wilson (2000), Reid et al. (2000), and Keicher and Seufert (2000) provide great summaries of the relevant research until the beginning of the new millennium. For the interested reader, these surveys provide an excellent reference to earlier development of automatic guidance.

Wilson (2000) concluded that especially computer vision and the Global Positioning System (GPS) offer the best technical solution to vehicle guidance. This, in general, seems to have become reality by the 2020s. Beginning from early mechanical solutions, the development of automatic guidance in agriculture has advanced with the advent of various sensors based on different principles such as ultrasonic sensors, magnetometers, laser ranging devices, accelerometers, GPS receivers, and imaging sensors such as cameras, combined with novel algorithms to extract information from the data and act accordingly.

By now, consumer products have entered the market, many of them based on GPS navigation, and are capable of traversing an open field semiautonomously,

including fully automatic turning. Some of these systems are able to offer a centimeter-level accuracy and precision. A modern automatic guidance system may include various sensor and measurement devices, including a GPS receiver for absolute positioning, but also IMU, gyroscope, accelerometer, and compass for additional measurements. Especially in research, relative positioning approaches have been applied as well, utilizing, for example, cameras or laser ranging devices, combined with different algorithms to extract positioning information from the environment. Some additional computational capabilities might be required for some of these applications. Finally, to control the vehicle with automation, it has been necessary to implement the system with actuators and feedback sensors for the actuation. An example of some of the typical hardware components of a modern automatic guidance system is outlined in Fig. 10.1.

To determine a suitable guidance strategy and to implement a suitable guidance system (both software and hardware), we need to address individual conditions and limitations that stem from different practices in crop production. Four different agricultural landscapes are presented in Fig. 10.2. For example, in an orchard (a) or between two rows of crop (b), it is possible to obtain a directrix for the guidance by observing and analyzing visual cues in the environment. A system based on computer vision might be able to provide a majority of the necessary information for positioning. On an open field (c), however, it is much more challenging to establish a directrix based on camera imaging or laser ranging sensors alone. In this case, for example, GNSS-based positioning can offer a means of navigating in the field. In addition to the availability of positioning information in the environment, we also need to consider challenges related to the terrain. For example, additional considerations might be needed for terrace farming (d) or other difficult terrain. Finally,

Fig. 10.1 An example of some typical hardware components of a modern automatic guidance system. (Illustration: Riikka SOITINAHO)

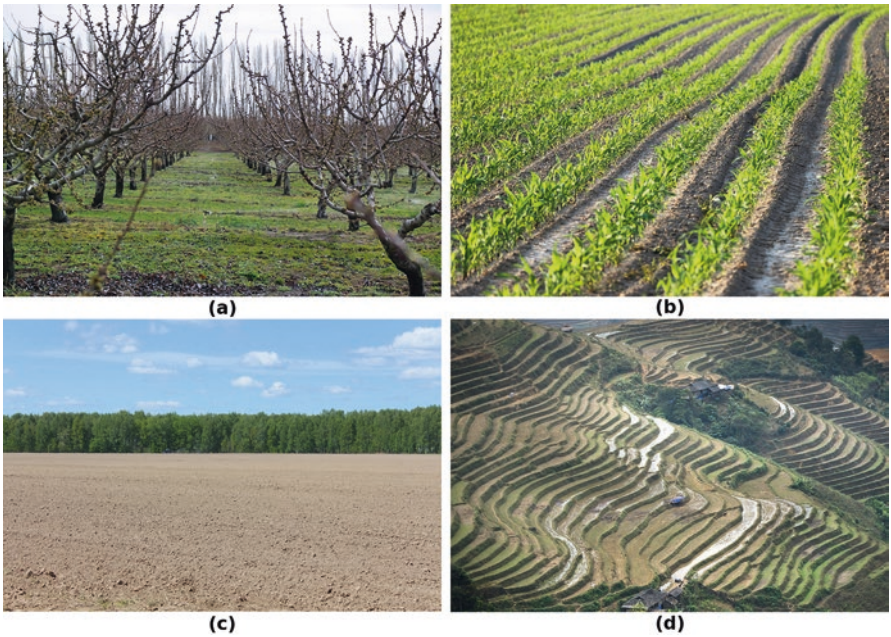


Fig. 10.2 Agricultural landscapes: (**a**) orchard, (**b**) row crops, (**c**) open field arable land, (**d**) terrace farming

Fig. 10.3 Automatic guidance and related concepts. (Illustration: Riikka SOITINAHO)

since our knowledge of the environment might be imperfect or incomplete, a means of observing the surroundings might be necessary to avoid any unforeseen obstacles.

The framework of an automatic guidance system often comprises the following components: path planning, positioning, navigation, control, and actuation. The hardware part of the system consists of sensors and actuators, where the sensors contribute information of the environment and the system itself, and the decisions made based on this information are converted into control signals that are transmitted to the actuators that change the course and orientation. The software part, obtaining input from the sensors, is responsible for combining all the information into control signals to operate the actuators.

In the rest of this chapter, we apply the following structure to introduce and explain the key components of an automatic guidance system, as shown in Fig. 10.3. First of all, we need positioning methods to obtain information about the location

and orientation of the vehicle. This information is an input to navigation which uses it to find a plausible path from one place to another. Control is then concerned with how to feasibly satisfy the navigation plan, given the case specific limitations of the real world. Finally, actuation addresses the specific electrical, mechanical, hydraulic, and pneumatic systems that are employed to control and move the vehicle. Before concluding the chapter, we briefly visit the topic of standardization related to automatic guidance.

10.2 Automatic Guidance

The most common form of guidance is related to parallel swathing. In parallel swathing, the guidance objective is to keep fixed distance to another path that is either preplanned (a set of paths) or the trace of any earlier swaths in the field to be tracked, sometimes called *contours*.

Straight paths, often called A-B lines, are the basic element of any guidance system. Basic path planning is rather easy, as it only requires filling the field with a set of straight lines, a fixed distance apart from each other. Advanced paths, often called contours, are anything that has a more complex shape compared to a straight line.

Guidance refers to guiding a vehicle, such as a mobile robot or a tractor, along a path or to a target. Also manual guidance, using human perception and human steering control, is classified as guidance, but this is out of the interest of this book. In many field operations, the main noise to be compensated by automatic guidance is related to wheel slip. Lateral slip of the steering wheel is directly influencing the accuracy of guidance, but also wheel slip in direction of travel influences in curves because it causes bias for the actuation as the desired curvature is not realized. Due to unavoidable wheel slip in the field, closed-loop control is the only means to realize automatic guidance.

10.2.1 Guidance Objectives

The task for the guidance system depends on the field operation. Therefore, it is crucial to make a distinction between various guidance objectives in order to cover precision of operation. The most common guidance objective is parallel swathing, which aims at seamless field operation in an open field when it comes to operations like harrowing or sowing. In this case, the traces made by the implement may neither overlap nor is it allowed that any gaps remain between adjacent swaths. The goal is a seamless field with no visible gaps or overlap between the swaths (Fig. 10.4).

With small compromises, this very generic guidance objective can be applied also to several other field operations like open field spraying, fertilizer spreading, grass mowing, and combine harvesting. However, furrow ploughing is an example

Fig. 10.4 The guidance objective in sowing is seamless parallel swathing. (Image: Timo OKSANEN)

of an operation where the guidance objective is to guide the vehicle along the furrow by keeping the wheels in the furrow—a physical feature that was created in the field by the previous pass. Similar track-a-feature guidance objectives appear in potato field operations (and other row crop operations) both during the season and harvesting.

Vehicle combinations, either mechanically connected or independent, require us to define further guidance objectives. For instance, a trailed sprayer connected to a tractor used for crop protection with multiple steering actuators is an example of this. During the growth season, the vehicle combination should cause as little damage to the crops as possible, and therefore it is intended that the wheels of the sprayer follow the tracks of the tractor precisely to minimize the area where pressure and compaction is applied by the wheels.

Furthermore, a subsequent series of operations offer various guidance objectives that are somehow interconnected. For instance, in hay making, a typical workflow is mowing, tedding, raking, and baling. In this work, especially raking and baling are interrelated—the guidance objective of baling is to track the windrows created during raking. In addition, proper raking operation is dependent on the maneuverability of the baling vehicle.

The examples of guidance objectives provided here are to give the reader an idea of how a guidance objective is typically defined. The set of possible field operations in different regions and crops is somewhat unlimited, and therefore an exhaustive list of guidance objectives cannot be defined.

10.2.2 Sensors for Guidance

Commercial success in order to realize automatic guidance was largely based on economically available GNSS positioning components. Automatic guidance was certainly possible prior to GPS, but the systems never became popular due to

complexity and high cost. GNSS receivers and easy-to-use correction services have allowed users to deploy automatic guidance systems in increasing numbers.

Depending on the guidance objective, it might or might not be possible to rely entirely on GNSS positioning. In order to follow the ridges in a potato field, the traces need to be precisely recorded during planting to form the paths for the subsequent operations. In general, this is possible and offered by many commercial automatic guidance systems, but due to human errors it might be difficult to use in practice. The most common challenges are related to forgetting to set the recording on, losing the recorded log files, or partially losing the accuracy of positioning during the planting work.

Some guidance objectives are related to tracking some features in the field that are created in situ. For instance, windrows have certain size and shape based on the material properties, yield, and the characteristics of the machine. For this sort of guidance objectives, accuracy benefits can be gained by using local sensors that measure directly the location of the feature in the field relative to the location of the vehicle. For example, local row feelers, based on either mechanical or noncontact sensing, are used in row crop machinery to follow the crop rows. Contactless sensors, including sensors detecting features optically, are used to guide machines; these are typically based on vision with one or multiple cameras in different wavelengths or 2D or 3D lidars.

10.3 Positioning and Navigation

10.3.1 Positioning

Positioning is the task or ability to determine one's location (including orientation) with relevant accuracy and precision. In some regards, all positioning can be considered to be relative, whether it is relative to a satellite constellation or to a marker found in the surrounding environment. However, sometimes it makes sense to make a distinction between positioning methods that yield a location in a geographic coordinate system (relative to the surface of the Earth) and positioning methods that yield a location relative to a landmark, a beacon, or any kind of a reference point in the vicinity. In this case, we would say the former methods provide *absolute positioning* whereas the latter ones provide *relative positioning*.

Sometimes it is not enough to know one's position (coordinates in 3D space), but also knowledge of one's orientation (rotation in 3D space) is of essence. Other important variables include speed, heading, and course. In general, heading and course are not the same thing—heading means the direction of the front of a vehicle, whereas course means the travel direction of the vehicle. Instead of speed and course we can also use the term velocity.

Field navigation is often planned on projected 2D space even if the fields are not strictly flat. The main reason for this is mathematical simplification for path

tracking, path planning, and other subsystems of the guidance system. In gentle slopes, the inaccuracy resulting from this simplification is often acceptable. The other reason for using 2D instead of 3D is the lack of precise information about the field slopes (or the terrain model), as fields tend to change along the years due to erosion and natural altitude changes over the season. After ploughing, the field surface is relatively higher. In order to do 3D path planning and guidance, a sufficient terrain model should be acquired.

To follow and guide a vehicle along the desired path, the positioning system should give a good estimate of where the path is relative to the vehicle and its components (like a trailer), or vice versa, where the vehicle and its components are relative to the path. In 2D space, for a simple single body system, two main variables are to be considered when it comes to path tracking: angular deviation and cross-track error to the path. In simplified cases, only the cross-track error might be sufficient, but in general case orientation error is needed for conversion of the cross-track error in various locations of the vehicle (like the front axle or intake of a harvester), not only where the GNSS receiver is located.

For a multibody vehicle, like a tractor with a trailer, each component should be considered in the estimation. For instance, a trailed seed drill follows the tractor in trailer-like behavior during curves, but also on side slopes and soft soil, the implement tends to slide due to wheel and other slip. Estimation of the relative orientation of the components in multibody systems can be done with three main approaches: indirectly by using the motion and kinematic model of the system, by measuring the relative angles with sensors, or by multiple GNSS receivers. All these options are used in commercial solutions.

To calculate the angular deviation of the vehicle and the path, the *heading* of the vehicle needs to be estimated. Heading is the real orientation of the vehicle (one angle in 2D space) in the global coordinate system. Heading should not be mixed with *course*, which refers to the travel direction of the vehicle in the global coordinate system. Under strong side forces and wheel slip, heading and course may differ remarkably. Also, in case of true four-wheel steering, the heading and course differ intentionally.

10.3.1.1 Relative Positioning

Relative positioning is feasible when position can be measured relative to a reference point. Instead of measuring position as coordinates in a global coordinate frame, we measure position relative to, for example, the coordinate frame of the vehicle. Various sensors and methods exist for relative positioning, including odometry, camera and machine vision systems, stereo vision, laser rangefinders, ultrasonic sensors, artificial marker systems, and mechanical feelers. Hague et al. (2000) give a good introduction to ground-based sensing systems that provide relative positioning, including odometry, artificial landmarks, and local feature detection. For example, a mechanical feeler can be used to estimate the position of a furrow, and

camera systems and lidar sensors can detect the location of crops and obstacles. With odometry, we can estimate change in position using data from motion sensors.

These sensors can be classified into two categories: sensors that measure the internal state of the system and sensors that make external observations. For example, inertial sensors like accelerometers and gyroscopes can be used to measure the state of the agricultural vehicle. A geomagnetic compass can indicate the direction of the vehicle relative to the magnetic field of the Earth. Encoders that measure wheel rotation can be used to estimate travelled distance, given that the circumference of the wheel is known. On the other hand, sensors that make external observations include cameras, laser rangefinders, ultrasonic sensors, and mechanical feelers. Finally, combining data from multiple sources to achieve higher accuracy positioning is possible with sensor fusion methods like Kalman filter (Kalman 1960).

Systems based on artificial landmarks typically measure the distance to known markers in the environment, for example, by a radar or a laser device and time-of-flight calculation. Given three (or more) markers, the position of the vehicle can be calculated based on trilateration. If the absolute locations of the markers are known, this can be considered a method of absolute positioning. A disadvantage of these systems is that the markers need to be set up, and a line of sight needs to be available independent of the location and tilt of the vehicle.

More common nowadays are positioning systems based on cameras or laser rangefinders that detect naturally occurring objects in the environment. Several studies describe systems that are based on a camera, a stereo camera, or a laser rangefinder alone. These sensors are used to observe local environment features that are distinct from their surroundings, especially crop rows and crop edges.

Often some form of image or data processing is required in order to use the sensor information for positioning. Extracting meaningful information from camera images or laser scanner data can be demanding compared to the relative ease with which humans can identify objects they see. Some assumptions need to be made about the objects that are observed in the environment. For example, a typical assumption in vision-based positioning is that a crop row can be distinguished from the surroundings based on its green color. Separating the green plants from the soil and other surroundings is done by image segmentation. The goal of image segmentation, in this case, is to obtain a binary image where the crops are typically represented with white pixels and the background with black pixels. One way of performing automatic image thresholding is the Otsu method (Otsu 1979), which was used, among others, by Guerrero et al. (2013). Also near-infrared images can be used to detect vegetation (for example, Åstrand and Baerveldt 2005).

Another commonly used sensor for relative positioning is a laser rangefinder, or a lidar. For detecting objects like crops with a lidar, it is required that the crops (crop rows, crop edges, windrows) can be distinguished from the ground level. Another typical assumption is that the crops grow in relatively straight rows. For example, Fleischmann et al. (2013) present an approach for observing field structures, like windrows, for positioning and guidance of agricultural vehicles. The positioning relies exclusively on distance data from a laser scanner. Also Malavazi et al. (2018)

propose a lidar-only based approach for positioning. A line detection algorithm for 2D point clouds is applied to detect the crop rows.

Varying outdoor illumination is a challenge especially for camera imaging. Also blurring of the images might be caused by the vehicle movement. Vehicle movement needs to be considered with certain types of laser scanners as well, since so called rolling shutter artifacts can result from measurement during movement. A rotating laser scanner takes multiple asynchronous distance measurements during one rotation. Because the vehicle is moving, each measurement is recorded in a different location. Therefore, unless a correction is applied, a distortion can occur in the constructed scene.

Other types of sensors, for example, IMU and odometry, are sometimes used for positioning in combination with a camera or a lidar scanner. For example, Blok et al. (2019) combined lidar scanner, IMU, and encoder data for positioning using both a Particle filter and a Kalman filter.

In certain field operation, it is possible to make marker lines to the field in order to do parallel swathing. For instance, in sowing, the seed drill may be equipped with side markers which draw a small furrow into the soil that is not affecting the agronomic result but is a visible target line for the next swath, to be followed with an eye. This marker may be also followed by lidar or machine vision. Detection of single furrow is challenging as the field may have similar lines naturally, for instance after preceding harrowing. One option is to improve the marker with additional features, like diagonal lines combined with the marker line. Kannas et al. (2008) presented an example of this, realized at the edge of harrow, to create special featured diagonal line to be detected and tracked in the parallel swath, pattern is presented in Fig. 10.5.

To provide positioning information for automatic guidance, the positioning sensors that make external observations need to be installed in a known location on the vehicle so that a transformation can be made between the coordinate frame of the vehicle and the sensor.

10.3.1.2 Absolute Positioning

Absolute positioning in a fixed coordinate system is necessary to work in open fields with no features. In the early days, in the era before satellite navigation, absolute positioning was possible, for example, using local beacon systems, triangulation of signals, or with an optical theodolite or a tachymeter. Even today, a robotic tachymeter is a valid precision localization tool, but the setup time and cost are not suitable for agricultural applications. Thanks to their versatility and global performance, GNSS receivers have outnumbered other absolute positioning technologies, to an extent that at the time of writing other technologies have only historical relevance.

GNSS (Global Navigation Satellite System) is an umbrella term for all satellite navigation systems for absolute positioning. Originally the term was used to refer to GPS (USA), but parallel systems have been introduced later, including GLONASS

Fig. 10.5 A harrow with marker device that created improved feature to be detected with higher confidence with machine vision. (Image: Timo OKSANEN)

(Russia), Galileo (Europe), and BeiDou (China). Each satellite system has equal or more than 24 satellites in different orbits. GPS was originally designed for 24 satellites, but both GPS and other systems have been augmented with additional satellites for redundancy. Modern satellite navigation radios on the Earth are capable of tracking all systems with a single antenna, and therefore multiple constellations can be used simultaneously to improve availability. Satellites broadcast signals in frequencies typically between 1.1 and 1.6 GHz.

Mobile receivers have to receive a signal at least from four satellites in order to calculate their own position. The positioning is based on trilateration of distances to each satellite, but since the receiver clocks are relatively inaccurate, a fourth signal is required to estimate the time shift. In the original GPS constellation, the maximum number of visible satellites at any given time was 12. Some shadow regions are therefore possible when the minimum of four satellites at a time cannot be seen by the receiver. Multi-constellation systems have improved the situation, and tracking at least four satellites at a time is more accessible nowadays.

A GNSS receiver natively tracks only GNSS satellites, and positioning is based purely on trilateration of pseudo-distances to each tracked satellite. Therefore, a GNSS receiver in basic mode works autonomously, without any additional dependency or two-way communication. However, due to signal distortion in the

atmosphere of the Earth, the pseudo-distances are not precisely correct. This makes positioning relatively inaccurate. Typical absolute accuracy of an autonomous GNSS receiver is in the scale of 10 m.

Autonomous GNSS does not provide sufficient accuracy for agricultural guidance and navigation. Only in exceptional cases, it can be used for rough operations. Typically, GNSS is augmented with free SBAS (Satellite-based Augmentation System) services that provide regional corrections over other geostationary satellite systems. SBAS is a general term for regional systems, like WAAS (USA) or EGNOS (Europe). Using SBAS usually improves the accuracy so that basic navigation can be achieved with accuracy of 1 m in good conditions. This accuracy is sufficient for fertilizer spreading but not for precision planting.

Similar DGNSS (differential GNSS) can also be realized with local base stations where a dedicated receiver is stationary, and differential correction data is transferred to a moving rover by some radio technology; a dedicated local radio link or over cellular data network.

The next level in accuracy requires real-time kinematic (RTK) technology. This requires tracking the carrier phase of the signals transmitted by the satellites, not only the pseudorandom data in the carrier. The goal of carrier-phase tracking is to calculate multiples of wavelengths between the satellite and the rover and add the phase difference. If the receiver is able to solve the number of whole cycles to each satellite, the accuracy is in the level of centimeters. The wavelength of the basic GPS signal (L1) is about 19 cm. Phase tracking in the cycle improves the theoretical accuracy in parts of the wavelength. In practice, RTK GPS requires a base station (in a known location) that records the observations with the same principle. This information is transmitted in real time to the rovers in the neighborhood. By using a base station and multiple signals on different frequencies transmitted by the satellites, the deviations caused by the atmosphere can be eliminated. However, the distance from the base station to the rover should not be too long in order to achieve the best accuracy. As a rule of thumb, a good radius is 20 km—the longer the distance, the more inaccuracy.

Since RTK correction requires a local base station and radio communication between the base and the rover, the usability in practice is not the best. To improve usability and convenience, service providers can be used to obtain the correction for the rover. Some service providers use a ground network over cellular data, and some use dedicated satellite broadcasting. Most of the services are commercial and require a commercial license.

Heading estimation can be achieved either by using a dual antenna GNSS receiver and calculating the orientation from their relative position output or by combining the GNSS data with inertial and/or magnetic sensors. A magnetic sensor (an electronic compass) provides direct information about the orientation but is prone to magnetic field disturbances caused by the vehicle itself or stationary objects in the field. To solve the disturbances, an advanced compensation methodology is

required. Inertial sensors provide information about relative movement, and binding the relative information to absolute positioning in movement is required to estimate the heading by using sensor fusion methodologies: the GNSS receiver provides the course when the vehicle is moving, and this is valuable information only when the vehicle is moving with known bias between course and heading.

In addition, the point of interest is the position of the vehicle, not the position of the GNSS receiver itself. The navigational point of the vehicle is rarely the point where the GNSS receiver is installed. Typically, the origin of the tractor coordinate system is on the center of the rear axle, either on the ground level or on the level of the axle. Due to visibility reasons, to provide a good line of sight to all directions, the GNSS receiver is usually mounted on the highest point of the vehicle. Thus, the distance between the GNSS receiver and the origin of the tractor coordinate system is often more than 1 m, or up to about 3 m in large size tractors. When the tractor travels on a slope, the error caused by the inclination must be compensated by tilt sensors, or by using more sophisticated estimation methods to estimate purely the position of the vehicle in global coordinate system, instead of GNSS receiver. An example of the configuration of coordinate systems is illustrated in Fig. 10.6.

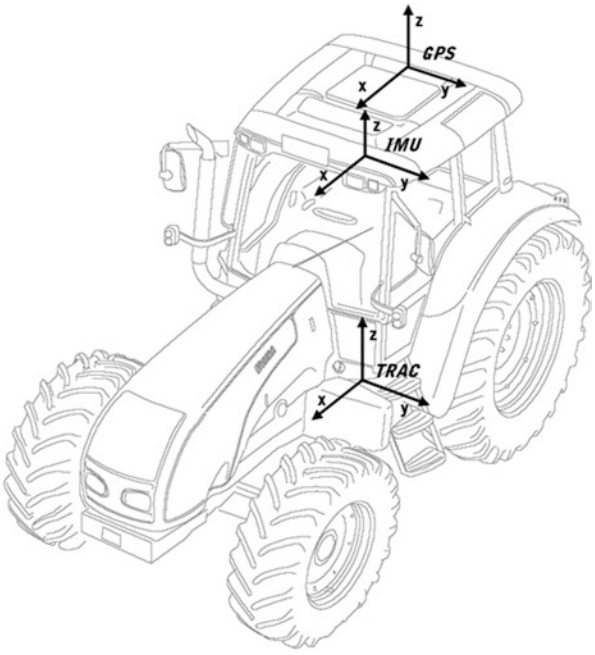


Fig. 10.6 An example of coordinate systems in tractor positioning and navigation; including tractor coordinate system origin which is on the rear axle, IMU and GPS. Coordinate transformations between coordinate systems are needed. (Oksanen et al. 2005. Image: Timo OKSANEN)

10.3.2 Navigation

The exact definition of navigation varies from one source to another, but a typical definition usually includes the task of *finding a way from one place to another*, which essentially means planning a path (or a trajectory) from point A to point B. Sometimes, also monitoring one's past position is considered part of navigation, whereas determining one's position is most often considered to be a positioning task. Some definitions also consider applying the necessary corrections to course, orientation, and speed to arrive at a desired position to be a navigation task, whereas it can also be seen as a separate task related to control. Therefore, depending on the definition, some overlap may exist between positioning and navigation, as well as between navigation and control.

Here we distinguish the tasks of positioning, navigation, and control so that positioning is mostly concerned with answering the question "where am I?," navigation should answer the question of "where should I be?" and "how do I go there?," and the task of control is to decide "how do I track the path?." There is an obvious dependency between positioning and navigation, but navigation also needs to consider the limitations related to the ability to control the vehicle.

There are various applications in agriculture where the ability to navigate accurately and reliably plays a significant role. Sometimes, instead of finding a way from point A to point B, we want to pass over every point of an area (or volume) of interest. This task is called coverage path planning, and it is in fact a typical task in agriculture, for example, in crop cultivation (think of harrowing, sowing, or similar tasks). For some applications, it makes sense to make a long-term plan, from start to finish, whereas for other applications, it is enough to react and plan according to short-term information. For example, to follow a crop row based on camera images or lidar data does not require a preplanned path, nor is it necessarily helpful to have one. Instead, the path can be computed online, based on the relative location to the crops, which is estimated from the sensor information. For some other tasks, it is difficult to find visual cues from the environment to provide a directrix for the navigation. In those situations, navigating based on absolute positioning methods makes sense, so as it allows to plan the full path in advance. Of course one should still consider if it is necessary to have access to real-time situational awareness in any case. For example, a dynamic or partially unknown environment might necessitate such information.

One more important aspect of navigation is to plan feasible paths. First of all, a feasible path takes into account the geometry of the vehicle and its surroundings. But, in addition, navigation needs to be aware of the vehicle kinematics, dynamics, and actuation, since not every path can be followed by every vehicle. This is to say that we should always plan a path that can actually be tracked, given that there are vehicle specific constraints. For example, a car-like non-holonomic vehicle cannot make a 90 degree turn in place. These limitations in vehicle maneuverability pose interesting challenges to navigation planning.

Finally, we distinguish between navigation based on relative and absolute positioning, because it is worth underlining the central differences between the basic principles of these two approaches. As outlined in Sect. 10.3.1., the most prominent absolute positioning method is based on satellite systems that provide a geographical location on the Earth's surface. Most other methods, including vision-based and laser ranging methods, provide relative positioning. However, also combination approaches can be used for navigation, and they are worth a mention in their own right.

It is also important to note that there are at least two different ways of using a laser ranging device for positioning and navigation. First, a lidar can be used to measure distance to any objects that reflect the laser pulse emitted by the device, so that the distance can be calculated by analyzing the time-of-flight when the returning reflection is detected by the device. This captures the surface shape of the objects in the surroundings, and also means that these objects can be a part of the environment by nature. Since the exact location of these objects is not known, we can only navigate relative to these objects. Another way is to place *artificial* markers in the surroundings, and then the distance to these markers is measured with laser ranging (or another means). Three markers are enough to perform trilateration calculation. Notice that if the absolute (geographical) coordinates of these markers are known, this can be considered a method of absolute positioning.

10.3.2.1 Navigation Based on Relative Positioning

Certain features in the environment can be detected and observed, for example, with a camera or a laser rangefinder. For instance, the green color of some crops, or the height of the stems or the crop line, stands out from the surrounding environment. Therefore, one of the elementary assumptions in visual and laser ranging-based applications is that there is a demarcation line in the environment that can be detected and followed. The demarcation line might be readily found in the environment (for example, crop rows), or something that is created during the process itself (for example, mowing), or sometimes the guideline is established by a previous task (for example, windrowing).

Based on the features that are detected from the sensor data, we can estimate the relative location of the crop line or the windrow. The location of the crops is used as a reference for path planning and navigation. To establish a directrix for the guidance, this reference needs to be combined with knowledge about the nature of the task, for example, that the vehicle should drive between two rows of crop or that the vehicle should follow the windrow. The guidance objective is the key to determining where the path for the vehicle should be placed. For example, the vehicle might be required to drive exactly in the middle of two adjacent rows of crop, or at a certain distance from the crop line.

A simple way of determining the navigation line is to apply linear regression to the points that were estimated to belong to the crop line. The idea is the same independent of the data being obtained from a camera, a laser rangefinder, or ultrasonic

sensors. However, the field of view of different devices might differ remarkably. A camera can capture a significant part of the environment at once, whereas a single ultrasonic sensor measures the distance to one point in the environment at a time. Therefore, we need different ways of processing the data to provide meaningful information for the navigation. For example, we may need to combine current and past data to identify a crop line with a laser scanner.

After determining the location and orientation of the crop row, one more step to decide the navigation line is to determine if there should be a lateral distance between the crop line and the vehicle path. Either we calculate the center of two adjacent crop rows if we want the vehicle to drive in the middle or we calculate the navigation line as parallel to the crop line but at a desired distance. Notice that the line is typically calculated based on the current camera view, and therefore it may change (both relative and absolute location) each time a new frame is obtained from the camera.

Different methods of detecting crop rows (or other linelike formations), and thereon establishing the navigation plan, include, for example, linear regression, Hough transform, blob analysis, stereo vision, and horizontal strips. There is a vast amount of applications and research on relative navigation, and therefore this introduction is by no means exhaustive. Instead, we present the reader with an overview and a starting point for further reading.

Linear regression in combination with a camera-based vision system has been applied to detect crop rows in various instances. For example, Billingsley and Schoenfisch (1997) applied a line fitting method to locate crop rows. The crop row detection is a part of a vision guidance system for agriculture. Søgaard and Olsen (2003) estimated the crop row lines by weighted linear regression without performing image segmentation. Montalvo et al. (2012) used a linear regression approach where the line parameters are estimated from image pixels that were considered to belong to the crop rows. Underlying knowledge of the amount and expected location of the crop rows was used to determine the crop row pixels from the images. Han et al. (2004) used K-means clustering to automatically find a threshold for row detection. Also Zhang et al. (2018) and Ospina and Noguchi (2019) applied linear least squares to identify the center lines of crop rows.

Another method of extracting line features from camera images (or laser rangefinder data) is the Hough transform (Hough 1959), to which Duda and Hart (1972) proposed an alternative parametrization that later became popular. For example, Rovira-Más et al. (2005) applied Hough transform to camera images of crop rows to obtain navigation directions. A forward-viewing camera was used to capture the view in front of the tractor. Ji and Qi (2011) proposed an algorithm based on random Hough transform for detecting the center of crop rows. Winterhalter et al. (2018) applied Hough transform to detect crop rows from both camera and laser scanner data, which they first transformed into feature maps that describe the location of the plants in 2D space.

A stereo vision-based approach was applied, for example, by Kise et al. (2005), who calculated an elevation map based on the images from a stereo camera that was installed on a tractor. A navigation point was extracted from the elevation map, and

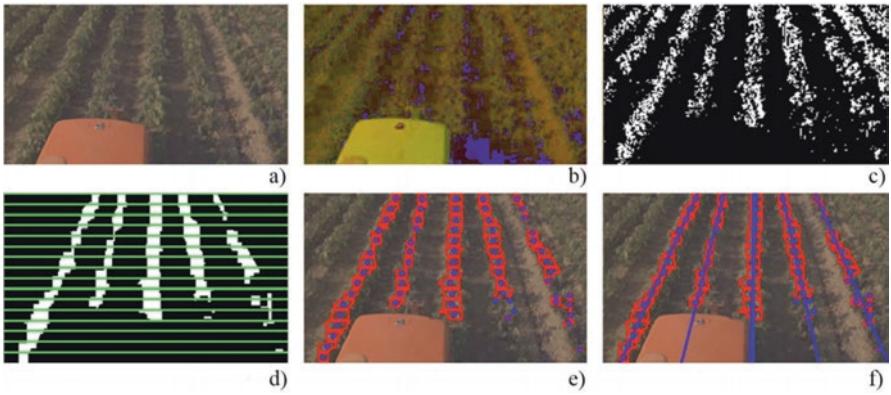


Fig. 10.7 Crop row center line detection of Ospina and Noguchi. (Adapted from Ospina and Noguchi 2019)

the steering angle to correct the offset was computed from the navigation point coordinates in the tractor coordinate frame. Yun et al. (2018) developed a guidance line detection system based on stereo vision. The guidance line was extracted with linear regression from the boundary points between a ridge and a furrow. Also Kneip et al. (2019) used a stereo vision system to obtain an elevation map of the crops. The crop edge, the edge between the standing crop and the ground, was assumed to be linear, and the crop edge model was calculated using linear regression.

The crop row center line detection process of Ospina and Noguchi is shown in Fig. 10.7. Wide angle images were used for crop row detection. The input image is shown in (a), an HSV image from the input image in (b), and a binary image in (c). The calculated row centers are shown in (e), and the center lines based on least squares regression in (f). This method can be used for automatic guidance systems for agricultural machinery (Ospina and Noguchi 2019).

Navigation methods based on positioning data from other types of sensors are often based on the same ideas; only the sensor data is different and so are the methods to process the data. We may use, for example, a laser rangefinder or ultrasonic sensor. The working principle of these devices is similar: they measure distance to objects based on time-of-flight. A laser rangefinder uses a laser beam, whereas an ultrasonic sensor uses ultrasonic waves. The sensor measures the time between sending and receiving a pulse, either light or sound, and the proximity to the object is determined based on the duration and the propagation speed of the pulse in the medium (in this case air).

The navigation task, then, is to establish an image of the environment and determine a meaningful path based on this information and the guidance objective. For example, to guide a vehicle along the windrow, we must calculate the center of the windrow and determine the lateral and angular deviation of the vehicle from this line. Finally, a path tracking controller is required in order to drive the vehicle towards the path, by controlling the actuators that steer the vehicle.

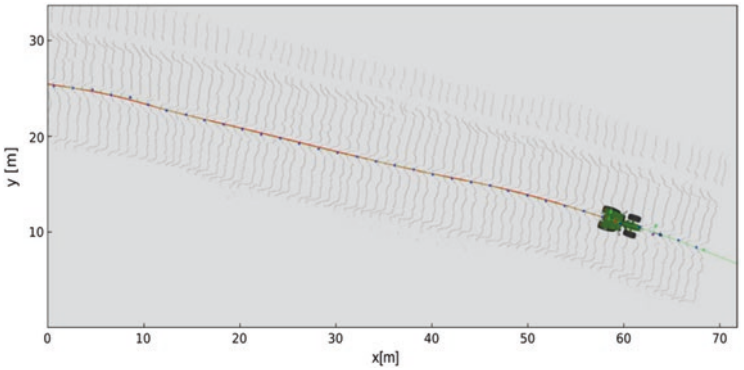


Fig. 10.8 Detected swath (green line with blue points) based on laser scanner data. (Adapted from Fleischmann et al. 2013)

For example, Fleischmann et al. (2013) presented an approach that is exclusively based on distance data from a laser scanner. A windrow was detected from the laser scanner data based on a particle filter approach and geometric modelling of the windrow. The detected swath from the example by Fleischmann et al. (2013) is shown in Fig. 10.8. The windrow detection result can be used as a waypoint for the path planning and navigation. Using a laser scanner also makes the detection more robust against variations in illumination.

Methods based on relative positioning typically apply local, short-term planning, where the solution is reactive with regards to newly obtained information about the environment. For example, the exact location of the navigation directrix might change between each camera frame. One part of planning and navigation is how the directrix is calculated; however, where the directrix is located is decided only on the field once the features are observed in real time.

10.3.2.2 Navigation Based on Absolute Positioning

Another approach, especially when features in the environment are difficult to detect (or there *are no features* to detect), is to navigate based on absolute positioning. Using absolute positioning like GPS gives some added flexibility when planning a path or a turn. For example, at the end of a crop row, we might temporarily lose sight of the features that we need for relative positioning and navigation. Given that the satellite visibility is sufficient, the vehicle can still be guided based on GPS. However, GPS-based navigation comes with its own disadvantages as well: satellite visibility might sometimes be blocked or hindered by tree lines or buildings. It is also important to notice that relying solely on satellite-based navigation poses some limitations. In an unknown environment, satellite-based navigation alone is not feasible. In addition, unless the exact location of the crops is known, we cannot make sure that the vehicle does not run over the plants.

Fig. 10.9 An example of a field boundary, path plan, and navigation outcome based on GPS positioning. (Image: Riikka SOITINAHO)

Again, the guidance objective is the key to determining how the path should be planned. A typical guidance objective that can be solved with absolute navigation is a coverage task related to open field operations such as harrowing or seeding. As opposed to many of the relative navigation approaches, the idea with navigation based on absolute positioning is that a precise path plan can be determined beforehand. However, if we rely entirely on satellite navigation, the assumption is that the environment is known and static. A map of the environment is necessary to plan a path in the correct coordinate frame.

An example of a GPS-based navigation plan and the outcome is shown in Fig. 10.9. The data was recorded as part of the project for cooperative agricultural work for multiple tractors (Soitinaho 2018). A coverage path was planned for a tractor in a local coordinate frame, which was then transformed to GPS coordinates. The coverage path was planned inside an artificially defined rectangular field boundary that is shown in Fig. 10.9a. The path consists of an ordered sequence of GPS coordinate points called waypoints. Sections with higher curvature have higher density of waypoints. In turns, the distance between two waypoints is around 0.1–0.2 m, whereas in the straight sections two successive waypoints are up to 80 m apart. Both the planned path (black) and the true path (red) are shown in Fig. 10.9b. The waypoints were sent to the vehicle individually, and the true path was recorded with the vehicle GPS receiver at 200 ms intervals.

At its simplest, a coverage plan for an open field operation consists of straight driving lines that are placed parallel to each other at an equal distance. This distance is decided based on the width of the implement, for example, a trailed harrow. In an ideal case, if the harrow is 3 m wide, the interval between the driving lines is 3 m as well. If we had sensors without noise and a perfect controller, all of the field would be harrowed by the time each driving line has been completed. In reality, some gaps and overlaps are observed due to various sources of imprecision in the system.

Just like with any relative navigation approach, we measure the deviation from the planned path. In addition, we need to keep track of the progress to know where

to measure the deviation from, which means that we also need a method to determine when a navigation point has been successfully reached.

10.4 Path Tracking

In order to keep the vehicle on the path, a path tracking system is required to command the steering actuator. Path tracking can be considered as a control design problem, where a cross-track is the process variable and needs to be regulated to zero by using steering angle as the control variable.

Path tracking can be also realized with little knowledge on control science, by using simple direct geometrical approaches like pure pursuit. In pure pursuit method and its variations, the principal idea is to take a fixed "look-ahead" distance in front of the vehicle and find the intersection with the path to be tracked. The intersection point is converted either to curvature of the arc interconnecting this intersection point and tangent to true rotational point of the vehicle or the angle of the vector connecting vehicle true rotational point to the intersection. Finally, the derived conversion is injected to the steering actuators with certain gain. Also, the look-ahead distance is considered as another tuning variable along with the gain (Snider 2009).

Cross-track error can be calculated from different parts of the vehicle body. In the Stanley method, the cross-track error is calculated from the center of the front axle to the nearest point in the path. This results cross-track error and a vector from the front axle to the path. The control method utilizes both the angle of the vector and the cross-track distance to compute the control in a nonlinear manner. The control law incorporates a tunable gain parameter for adjusting reaction speed (Snider 2009).

In order to improve derivation of a more precise control law, a kinematic model needs to be derived for the vehicle, to generalize usability. Most agricultural tractors utilize two-wheel steering. Bicycle models are used to simplify four wheels to two wheels and one rigid frame between these, which usually describe motion in a sufficient way. A tractor using the Ackermann steering principle can be converted directly to the bicycle model. The kinematic model is usually presented as a set of differential equations where the state variables are coordinates of the vehicle origin in global coordinates and the heading angle; control or input variables are steering angle or curvature and the velocity of the vehicle.

A kinematic model can be used in path tracking in many ways to derive the control law. One approach is to convert the kinematic model working in global coordinates to another form where the motion is expressed with respect to the path. This path relative model can be used in a more direct way for path tracking, by using a kinematic controller. Furthermore, the bicycle model can be improved to incorporate dynamics, including forces and mass of the vehicle. Including dynamics is beneficial in higher velocities to keep the vehicle control stable. The derived model may be linearized in order to use a linear feedback control law, or use other control methods capable utilizing nonlinear plant model (Snider 2009).

Model predictive control (MPC) is a numerical method for solving optimal control problems over a certain horizon in an iterative manner. Model predictive control is limited to linear plant models. A set of various nonlinear MPC methods have been developed to handle also nonlinear plants, but these are computationally more expensive and require fast processors to be solved in real-time applications for path tracking. The clear benefit of these methods is direct utilization of the kinematic model with dynamics. The specific challenge in modeling and control design is related to nonlinearities of steering actuation, which includes rate limits, saturation, and dead time delays due to hydrostatic transmission of power. Therefore, appropriate control design must compensate also control delays and performance limits of the steering actuator.

Another way to utilize kinematic model is to use it in inverse way to linearize the path tracking problem to lateral cross-track error and heading error. This approach makes the design of control laws easier to tune, and simple PID control approach may be used with feed forward compensator (Oksanen and Backman 2013).

10.5 Actuation and Steering Actuator

Modern machinery for arable farming utilized hydrostatic power steering mechanism. Most tractors are either front wheel steered or articulated, and a hydraulic cylinder is used to change the steering angle. Combine harvesters and other self-propelled vehicles may also use rear wheel steering, and some special tractors use four-wheel steering; also these are usually hydrostatically controlled, except the models manufactured in developing countries. Vehicles with full tracks use various types of steering actuation: in early models steering was usually based on brakes, but in modern transmissions the speed of each track is controlled to cause steering action.

The most common mechatronics in modern agricultural vehicles for steering is based on hydrostatic steering with no direct mechanical link between the steering wheel and the wheel angle. A special hydraulic valve contains feedback with an integrated metering unit. The steering unit connects the physical steering wheel in the cabin to the hydraulic lines of the steering cylinder. Steering actuation of articulated tractors requires higher force and torque compared to front wheel steering. Therefore, the energy requirement of articulated tractors is about 3.5 times higher compared to Ackermann steering (Renius 2020).

Later, to enable smooth automatic steering, the steering unit has been augmented with an additional proportional valve that allows electronic control of the hydraulic flow. A valve augmented with electronics may enable also other advantages, such as quick steering or driving speed-dependent amplification by electronic means. Automated steering requires also a sensor for the steering wheel that measures position and/or torque. This sensor serves as a safety function, for quick disengagement of automated steering. In addition, the automated steering actuator is also closely coupled to the motion controller or sensors for motion, as the safety standards

require that automated steering must be disengaged if certain speed threshold, typically 25 km/h, is exceeded. Automated steering is also disengaged automatically if the speed becomes zero.

By using the appropriate communication method, like CAN bus, the guidance system is able to command and control the steering actuator. In addition to controlling the steering angle, the guidance system needs to know the necessary constraints: mainly the maximum steering angles and the maximum achievable rate of steering. Also, proper calibration needs to be done for the zero angle. This can be adjusted either in the steering actuator or in the guidance system, or both.

10.6 Standardization

Commercially available automatic guidance systems can be divided into two categories: OEM (original equipment manufacturer)-installed integrated systems and retrofit kits. During the transition to the era of automatic guidance, retrofit kits were a popular means to convert old tractors and other agricultural vehicles to be capable of (usually GNSS based) automatic guidance. Today, OEM installations are typical for new machinery as it allows better integration and usability. In addition, the cost of selecting at least a ready-for-guidance option is lower than before. Ready-for-guidance usually means a system that has a preinstalled cable harness and steering actuators, but a GNSS receiver, other sensors, and the guidance controller are not included.

The interface between the steering actuators and the guidance system has been standardized in ISO 11783-7 by single frame messages. The behavior of the interface itself is not well standardized, only the message to command the steering and to record the actual steering orientation. The standardized property is curvature, as it is a generic measure to abstract different vehicle kinematics, from front wheel steering to four-wheel steering and articulated steering. However, using this standard interface is not straightforward, as vendors have different interpretations of the needs for other communication around it. For instance, AEF (The Agricultural Industry Electronics Foundation) has defined the TIM (tractor implement management) standard, which first requires authentication in order to enable commanding messages. In order to authenticate, the guidance system vendor must follow all the guidelines and pass the conformance test that verifies compatibility.

Also testing procedures for positioning and guidance systems in agriculture have been a subject of standardization. In ISO, there has been a working group TC 23/SC19/WG7 to prepare the standard series ISO 12188. Part 1 defines dynamic testing of GNSS receivers. The test procedure tests how a receiver behaves in various curves and curvatures and laid in certain orientation to north. The test defines test speeds from 0.1 to 5.0 $m \cdot s^{-1}$, which is the relevant speed range for agricultural field applications. Simulation of signal loss is also considered. The purpose of standardized tests is to provide ground for comparison of various products and to output comparable numbers. Also terminology and calculation of absolute, relative, and

dynamic accuracies are defined. Part 2 defines path tracking performance testing during straight and level travel.

Further standardization to augment ISO 11783-7 capability for steering actuation has been proposed. Two main functions missing are: a) four-wheel steering with two degrees of freedom and b) implement steering. The four-wheel steering with no fixed equation between front and rear steering axles is not possible within the current ISO 11783-7, but it could be added in a rather simple way, by introducing a course offset in addition to the curvature, which is already standardized (Oksanen 2017).

Implement steering suitable for various guidance objectives has also been suggested for the scope of standardization by Oksanen and Backman (2016). The proposal is based on the idea of an abstract implement, which is a skeleton model of a multibody implement with steering capability, in similar way as bicycle model is abstracting kinematics of a tractor or a car. The abstraction is possible for hitch mounted implements with side shift, trailed implements with drawbar offsetting, trailed implements with steerable drawbar, and trailed implements with steering wheels or steerable coulters, or any combination of these to allow two degrees of freedom.

Ultimately, the standardized implement steering enables commercial multi-brand guidance systems where a single guidance system is able to command both steering of the tractor and the implement (see Fig. 10.10). The commercial guidance systems designed for implement steering are either independent of tractor guidance or work only together with the tractor of the same manufacturer (Oksanen and Backman 2016).

Furthermore, the current ISO 11783-7 lacks a means to communicate the steering actuator parameters from the steering actuator to the guidance system, like maximum capable curvature, maximum rate of steering, or any delay or latency

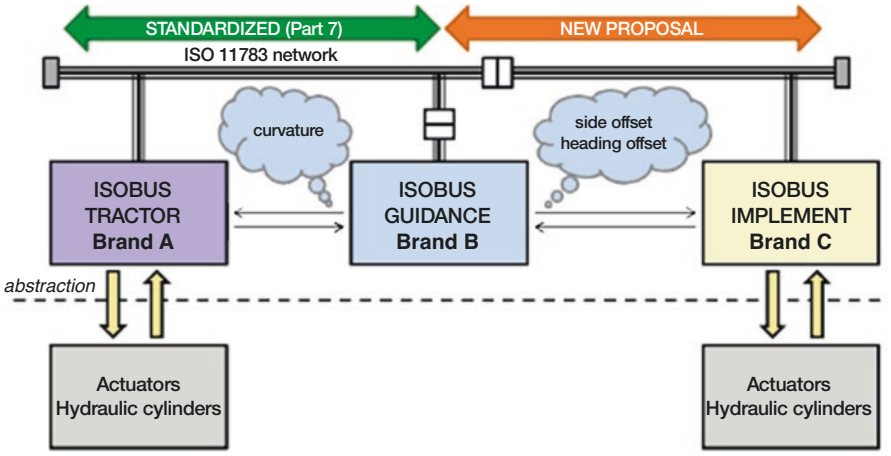


Fig. 10.10 The architecture of combined guidance system. (Oksanen and Backman 2016. Image: Timo OKSANEN)

parameters. Lack of delivery of these parameters make plug-and-play guidance system installation laborious as tuning procedure must be done completely manually by installation personnel (Oksanen 2017).

10.7 Summary and Concluding Thoughts

This chapter describes some fundamental topics and concepts in automatic guidance and control in agriculture. To obtain a comprehensive understanding of automatic guidance, it is important to familiarize oneself with the topics of positioning, navigation, control, vehicle engineering, and path planning. This chapter presents a general view of the relevant tools and methods. Agricultural applications also require some special considerations when it comes to outdoor environments and off-road conditions. The methods that are applied should be evaluated and chosen case by case. Therefore, having a good understanding of the specific application is always of essence. The guidance objective is the key to selecting the right methods and hardware. Properly designed automatic guidance has to be taken into account in the overall vehicle design process. In the future, the level of autonomy is expected to increase and new functionalities will be innovated. For instance, automated turning of agricultural vehicles in conjunction with parallel guidance will be more commonly used and interlinking guidance objective of sequential operations will be introduced commercially.

References

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

Billingsley J, Schoenfisch M (1997) The successful development of a vision guidance system for agriculture. Comput Electron Agric 16(2):147–163

Blok M, van Boheemen K, van Evert K, IJsselmuiden J, Kim H (2019) Robot navigation in orchards with localization based on Particle filter and Kalman filter. Comput Electron Agric 157:261–269

Duda O, Hart E (1972) Use of the Hough transformation to detect lines and curves in pictures. Commun ACM 15(1):11–15

Fleischmann P, Föhst T, Berns K (2013) Detection of field structures for agricultural vehicle guidance. KI Künstliche Intelligenz 27(4):351–357

Guerrero M, Guijarro M, Montalvo M, Romeo J, Emmi L, Ribeiro A, Pajares G (2013) Automatic expert system based on images for accuracy crop row detection in maize fields. Expert Syst Appl 40:656–664

Hague T, Marchant A, Tillett D (2000) Ground based sensing systems for autonomous agricultural vehicles. Comput Electron Agric 25(1–2):11–28

Han S, Zhang Q, Ni B, Reid F (2004) A guidance directrix approach to vision-based vehicle guidance systems. Comput Electron Agric 43(3):179–195

Hough C (1959) Machine analysis of bubble chamber pictures. Proceedings of the international conference on high energy accelerators and instrumentation, 590914, pp 554–558
ISO (2010) ISO 12188:1 tractors and machinery for agriculture and forestry—test procedures for positioning and guidance systems in agriculture—part 1: dynamic testing of satellite-based positioning devices. Geneve, Switzerland
ISO. (2012). ISO 12188:2 tractors and machinery for agriculture and forestry—test procedures for positioning and guidance systems in agriculture—part 2: testing of satellite-based auto-guidance systems during straight and level travel. Geneve, Switzerland
Ji R, Qi L (2011) Crop-row detection algorithm based on Random Hough Transformation. Math Comput Model 54(3–4):1016–1020
Kalman E (1960) A new approach to linear filtering and prediction problems. Trans ASME J Basic Eng 82(D):35–45
Kannas K, Oksanen T, Visala A (2008) Machine vision based parallel navigation for tractor-harrow. CIGR international conference of agricultural engineering
Keicher R, Seufert H (2000) Automatic guidance for agricultural vehicles in Europe. Comput Electron Agric 25(1–2):169–194
Kise M, Zhang Q, Rovira Más F (2005) A stereovision-based crop row detection method for tractor-automated guidance. Biosyst Eng 90(4):357–367
Kneip J, Fleischmann P, Berns K (2019) Crop edge detection based on stereo vision. In: Proceedings of the 15th international conference IAS-15, pp 639–651
Malavazi P, Guyonneau R, Fasquel B, Lagrange S, Mercier F (2018) LiDAR-only based navigation algorithm for an autonomous agricultural robot. Comput Electron Agric 154:71–79
Montalvo M, Pajares G, Guerrero M, Romeo J, Guijarro M, Ribeiro A, Ruz J, Cruz M (2012) Automatic detection of crop rows in maize fields with high weeds pressure. Expert Syst Appl 39(15):11889–11897
Oksanen T (2017) Extending ISO 11783 for four wheel steering and implement steering. VDI Landtechnik AgEng 2017 (VDI-MEG 2017)
Oksanen T, Backman J (2013) Guidance system for agricultural tractor with four wheel steering. Proceedings of IFAC AgriControl 2013, pp 124–129
Oksanen T, Backman J (2016) Implement Guidance model for ISO 11783 standard. IFAC-Papers OnLine 49(16):33–38
Oksanen T, Linja M, Visala A (2005) Low-cost positioning system for agricultural vehicles. Proceedings of 2005 IEEE international symposium on computational intelligence in robotics and automation, pp 297–302
Ospina R, Noguchi N (2019) Simultaneous mapping and crop row detection by fusing data from wide angle and telephoto images. Comput Electron Agric 162:602–612
Otsu N (1979) A threshold selection method from gray-level histograms. IEE Trans Sys Man Cyber 9(1):62–66
Reid F, Zhang Q, Noguchi N, Dickson M (2000) Agricultural automatic guidance research in North America. Comput Electron Agric 25(1–2):155–167
Renius T (2020) Fundamentals of tractor design. Springer International Publishing, Cham, p 287
Rovira-Más F, Zhang Q, Reid F, Will D (2005) Hough-transform-based vision algorithm for crop row detection of an automated agricultural vehicle. Proc Inst Mech Eng Part D J Automob Eng 219(8):999–1010
Snider J (2009) Automatic steering methods for autonomous automobile path tracking. Report CMU-RI-TR-09-08. Carnegie Mellon University, Pittsburgh
Søgaard T, Olsen J (2003) Determination of crop rows by image analysis without segmentation. Comput Electron Agric 38(2):141–158
Soitinaho R (2018) Cooperative path planning for multiple agricultural field machines performing sequentially dependent tasks. Master's thesis, Aalto University, Espoo, Finland
Wilson N (2000) Guidance of agricultural vehicles—a historical perspective. Comput Electron Agric 25(1–2):3–9

Winterhalter W, Fleckenstein V, Dornhege C, Burgard W (2018) Crop row detection on tiny plants with the pattern Hough transform. IEEE Robot Autom Lett 3(4):3394–3401

Yun C, Kim J, Jeon W, Kim H (2018) Stereovision-based guidance line detection method for auto-guidance system on furrow irrigated fields. IFAC-PapersOnLine 51(17):157–161

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

Chapter 11
Automated Infield Sorting and Handling of Apples

Zhao Zhang and Renfu Lu

11.1 Introduction

Apple is the third largest fruit in the world in terms of production tonnage, following watermelon and banana (STATISTA 2017). Consumers like apples because they have distinct texture, flavor, and aroma, as well as their nutritional and health benefits, such as reducing risk of some cancers and cardiovascular diseases (Boyer and Liu 2004). While apples are mainly consumed as fresh fruit, apple juice is also popular in many regions of the world (e.g., North America, China, Europe, and Japan), which is just behind orange juice in annual consumption per capita (Produce for Better Health Foundation 2015).

United States is the second leading apple producing country after China; it produced 5.1 million metric tons of apples in 2016, which generated about $3.5 billion farm-gate revenue (wholesale value). Apples are grown commercially in 32 states in the United States, with Washington, New York, Michigan, Pennsylvania, and California accounting for 65%, 10%, 10%, 4%, and 2% of the total production, respectively (USDA 2017). Harvest and postharvest storage and packing operations account for more than 50% of the total production cost in the United States (Gallardo et al. 2010; Gallardo and Galinato 2012). Increased labor cost, shortage of domestic labor supply, and low profitability are threatening the future of the apple industry in the United States (Baugher et al. 2009) as well as in other parts of the world. Hence, automation is crucial to improving productivity and addressing labor cost and

Z. Zhang
North Dakota State University, Fargo, ND, USA

R. Lu (✉)
U.S. Department of Agriculture, Agricultural Research Service,
East Lansing, MI, USA
e-mail: renfu.lu@usda.gov

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_11

availability issues, in harvest and postharvest handling, thus lowering the overall production cost for the apple industry.

11.1.1 Current Status in Apple Postharvest Handling and Presorting

Currently, apples are still harvested manually and once over for most cultivars. Harvested apples of mixed quality grades (i.e., fresh and processing) are hauled to the storage facility for short-term cold or refrigerated storage (i.e., for a few weeks only) or, otherwise, for long-term controlled atmosphere (CA) storage (from a few months to a year or even longer). Upon receipt of orders from the market, fruit are removed from storage and then graded, sorted, and packaged before being delivered to the retailers (Zhang et al. 2017a). Sorting and grading of apples at packinghouses in the United States and other developed countries is largely automated. Machine vision, including both color and near-infrared spectral imaging, is now widely used for sorting fruit for color, size, shape, and/or surface defects. In recent years, we have also seen increased use of near-infrared spectroscopic technology in many packinghouses for detecting internal defects and/or for sorting fruits based on soluble solids content. These commercial sorting systems are highly automated and extremely efficient (sorting at a rate of 10–15 fruits/s per lane), but they require high initial capital investment and are also expensive to operate.

Postharvest storage and packing operations typically account for 1/3 or more of the total apple production cost (Wunderlich et al. 2007). For instance, a packinghouse in Michigan would charge growers $30/MT and $80/MT for cold and CA storage, respectively, while the packing cost (including sorting, grading, and packaging) can be as high as $290/MT, regardless of the quality grade of apples. Since processing apples are usually sold at a fraction of the price of fresh ones (Lehnert 2015), growers may not break even when there is a high percentage of processing or cull apples in the whole lot (Schotzko and Granatstein 2005). For this reason, some growers would rather dump processing apples in orchard to reduce the postharvest handling expense (Lehnert 2013). Moreover, inferior, defective fruits are susceptible to disease and pest invasion during storage, which may spread to good apples in the same lot, thus presenting risk for huge economic loss for growers and packinghouses. Currently, only some large packinghouses in the United States presort or pre-grade apples, which are usually done shortly after harvest and prior to long-term CA storage, while most packinghouses do not adopt this practice, mainly because of cost concerns.

Instead of presorting apples at packinghouse, it would be more advantageous to presort apples in orchard so that growers can send inferior or processing apples directly to the processor for making juice or other apple products (e.g., fresh cut and apple sauce). This would effectively lower the postharvest storage cost because these apples only need be kept in cold or refrigerated storage for short term instead

of the more expensive long-term CA storage (Mizushima and Lu 2011). Presorting in orchard would be less demanding in terms of sorting requirements, as it only needs to sort apples into two or three grades (i.e., fresh, processing, and cull), compared to the packinghouse machine vision-based sorting system, which needs to sort apples into as many quality grades as possible at high sorting speed. However, presorting in orchard entails several unique technical challenges, which include, but are not confined to, limited space for the sorting system, the requirements for the system to be compact and low in cost, with easy integration with the existing or new harvesting platforms, and the accompanying capabilities needed for automated handling of harvested apples and bins, both empty and full. Currently, no commercial infield sorting system is available for apple growers, because no cost-effective technology has been developed, and R&D costs are too high for horticultural equipment manufacturers, most of which are small in scale with limited technological expertise in sensing and automation.

11.1.2 Current Status of Handling Harvested Apples in Orchard

Handling of harvested apples is an essential part of an infield sorting system, which is, hereinafter, referred to the process from the time right after apples are detached from trees until they are placed in the bin. While apples are still manually harvested using the traditional method of buckets and ladders, we have seen increased use of self-propelled harvest-aid platforms in the United States and in European countries (Zhang et al. 2016). These harvest-aid platforms, or simply harvest platforms, are usually self-propelled and have limited automation features for handling harvested fruit. Workers can stand on the fixed or adjustable stages of the harvest platform to pick apples from trees at different heights from the ground, which eliminates the use of ladders and, possibly, buckets (mainly for those units made in Europe) and reduces the time needed for transporting harvested apples to the bin, thus decreasing the working strength and increasing the overall harvest productivity. There are mainly two types of commercial harvest platforms, i.e., those equipped with harvest conveyors (either mechanical or vacuum-based) and bin fillers (hereinafter called WBF), and the other using buckets and with no bin fillers (NBF). The bin filler plays an important role in automatic handling of harvested apples; it receives apples from conveyers and then places them into the bin evenly, without causing bruising damage to the fruit.

With the NBF platforms (Fig. 11.1), which are popular in the United States, workers no longer use a ladder to reach to fruit up on the tree, but still need to carry a picking bucket for holding harvested apples temporarily. One advantage of using the picking bucket, which can hold up to 20 kg of apples, is that workers can conveniently place apples in the bucket without making large upper body movements during picking, thus enhancing the picking efficiency. Since there are no harvest

a) b)

Fig. 11.1 Two commercial apple harvest platforms with no bin filler, on which workers still need to carry picking buckets for holding picked apples temporarily, until the buckets are full and dumped to the bins

a) b)

Fig. 11.2 A commerical harvest platform with mechancial harvest conveyors (**a**) and a vacuum-based harvest platform (**b**). For both platforms, workers directly place harvested apples on the belt conveyors or vacuum tubes, which are then transported to the bin via a bin filler (not shown here). (Permission for using the photo in (**a**) is granted from LaGasse Works, Inc.)

conveyors and bin filler(s), NBF platforms are simple to construct; they usually have few or limited automation features and thus are relatively low cost.

With the WBF platforms (Fig. 11.2), harvested apples are transported to the bin by means of belt conveyors with fingers or similar structures, or vacuum tubes. Harvest platforms with belt conveyors are popular in Europe, while the platforms using vacuum tubes (Fig. 11.2) currently have very limited adoption by the industry because of cost and productivity concerns. With the mechanical belt conveyors, pickers directly place apples onto individual conveyors, which are then transported via one or more conveyors to the bin filler, which places the fruit into the bin. The use of mechanical harvest conveyors relieves workers from the burden of carrying a picking bucket. One major issue with the harvest conveyors is that workers would need to make larger upper body and/or arm movements, as these conveyors are not

always placed at the most convenient position relative to the workers, which would decrease the picking efficiency and may also pose occupational health risks to the workers. With the vacuum tube transporting system, pickers can conveniently place picked apples into the tube (Weinstock 2016). The system thus enables workers to maintain high picking efficiency. Depending on fruit shape and size as well as cultivar, individual apples may exhibit different aerodynamic behavior during transport in the vacuum tubes. Hence, it is important that proper vacuum pressure and tube configuration be maintained to ensure smooth transport of fruit in the vacuum tube to avoid fruit collision. Moreover, upon reaching the destination, the fruit need to go through a deceleration process before being released to the bin filler. Overall, the vacuum-based platform requires a more sophisticated system design with greater power consumption for vacuum transport of fruit, and fruit bruising could be an issue.

The bin filler, as discussed above, plays a critical role in all WBF platforms. Over the years, many different types of bin fillers have been developed for use with various mechanical harvest systems. A comprehensive review of bin fillers for postharvest packinghouse use and for infield use is given in Zhang et al. (2018a). There are different forms of bin fillers that are currently being used with the commercial WBF platforms. Figure 11.3 shows two commercial bin fillers used with the WBF harvest platforms with mechanical harvest conveyors (Fig. 11.3a) and the vacuum harvest platform (Fig. 11.3b). The bin filler used with the mechanical harvest conveyors consists of a vertical fingered conveyor and a rotary brush for bin filling. During filling, the bin is rotating to achieve even distributions of apples. The bin filler, shown in Fig. 11.3b, mainly consists of a pinwheel installed with soft pads. As apples arrive at the pinwheel from the decelerators, they are released from the rotating pads in all directions into the bin to achieve even distributions of fruit. This bin

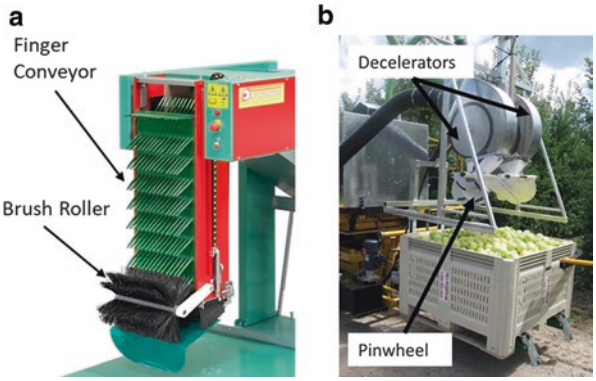


Fig. 11.3 Two bin fillers used for the commercial harvest platforms with mechanical (**a**) and vacuum (**b**) harvest conveyors. The bin filler (**a**) uses a vertical conveyor with fingers coupled with a soft brush roller to transport apples to the bin, which is rotating during filling to ensure even distributions of apples in the bin, while the bin filler (**b**) uses a spinning wheel installed with soft flexible pads for distributing apples in the bin evenly. (Permission for using the photos was granted by LaGasse Works, Inc. (**a**) and Phil Brown Welding Corp. (**b**))

filler design overall is simpler and also more compact, compared to other forms of bin fillers. As discussed in Zhang et al. (2018a), most of the bin fillers are still too complex in design, bulky in size, and expensive; they have low level of automation and require human assistance or intervention during the operation. Hence, further R&D is needed in the development of a low-cost, compact, simpler bin filler with full level of automation.

11.1.3 Current Status of Handling Fruit Bins in Orchard

Proper handling of empty and full bins during harvest in orchard is also critical to the harvest productivity when using a self-propelled harvest platform with or without the bin filler. In the United States, apple bins are typically made of wood or plastic, and each bin can hold 400–500 kg of apples, depending on the actual bin size. For the NBF platforms, like the one shown in Fig. 11.1a, four or five empty bins are first loaded onto the platform. After all bins have been filled, they are discharged to the bin trailer altogether from rear of the platform. During the unloading of full bins and subsequent loading of empty bins, the harvest crew ceases picking activities temporarily. For a harvest crew of 6–8 workers, it would need about 30–40 min to fill five bins and then take 2–3 min to unload the full bins and load empty bins. This means that workers, on average, would lose 5–10% of their picking time due to the time breaks needed for handling full and empty bins. Some other NBF platforms (such as the one shown in Fig. 11.1b) carry one or two bins; only one bin is being used for fruit filling in any given time. Once that bin is full (which would take about 10–15 min for a crew of four), it is discharged to the ground, and the empty bin is then moved in place. Workers need to cease picking fruit during the handling of bins (1–2 min). Thus, the harvest productivity could be reduced by 10–20%, due to the less efficient bin handling procedure.

The productivity loss caused by bin handling is also of concern for the WBF platform. Workers on the WBF platform need to suspend picking activities for 1–2 min, during the unloading of the full bin and loading of an empty bin, which would reduce the overall harvest productivity by about 10% (Jones 2015). Moreover, some of the WBF platforms require a dedicated person to handle the bin switching, which could further reduce the overall productivity by 10–15%.

In summary, bin handling for the current commercial harvest platforms is not fully automated, and it still needs human involvement, either partially or fully. Moreover, during the bin handling, workers need to cease picking activities. Altogether, the productivity loss attributed to the bin handling would be between 10 and 25%, depending on type of platform used, which offsets much of the productivity gain from using the platform. Hence, it is important that the bin handling operations (i.e., loading and unloading) be fully automated and do not interfere with the worker's picking activities.

11.2 Overview of New Automated Apple Infield Sorting Technology

Motivated by the potential benefits of presorting apples in orchard and in view of technical shortcomings and lack of automation with the current apple harvest platforms in handling harvested fruit and bins, our research team at USDA Agricultural Research Service (ARS) initiated a new project several years ago on the development of new automated infield sorting technology. To ensure that the new infield sorting system would meet the needs of apple growers, surveys were conducted with apple growers and packinghouses in Michigan to better understand their needs and concerns on apple harvest and infield sorting. Analyses were also performed on the potential economic benefits resulting from the adoption of infield presorting and harvest-assisting technology (Mizushima and Lu 2011; Zhang et al. 2017b). Furthermore, we also evaluated various machine vision systems currently being used in commercial packinghouses for sorting and grading apples and other fruits to determine whether we could adopt or modify some of the system design features for infield use. In addition, field evaluations were performed on the productivity of different commercial harvest platforms currently being used by growers. An extensive review was also conducted of various bin filling technologies that have been developed over the past 60+ years for packinghouse and harvest use (Zhang et al. 2018a). Through these reviews and evaluations, it was clear that the existing packinghouse sorting and bin filling technologies are not appropriate for infield presorting use, and substantial innovation is thus needed. It was further determined that the research needed be focused on addressing these key technical issues, i.e., *automated sorting and grading of apples*, *automatic handling of harvested fruit*, and *automatic handling of bins*. In the initial phase of the project, the infield sorting system was developed for working with a commercial harvest trailer, which was hauled by a tractor. However, such design was not well received by growers, because the infield sorting system needed a dedicated person to drive the tractor/trailer and the automatic bin handling function was not integrated with the system. Hence, in the second phase of the project, further effort was made on the development of a self-propelled harvest platform with an automatic bin handling function. The new apple harvest platform was constructed and field tested in 2016 harvest season, followed with further improvements and field testing in 2017.

Figures 11.4 and 11.5 show a schematic and picture of the new apple harvest and infield sorting machine prototype. This machine allows six workers standing on the ground or elevated stages to pick apples from trees and then place the fruit onto harvest conveyors. The fruit on the six harvest conveyors will then converge to the main conveyor, through which they are transported to the machine vision-based sorter for quality inspection (Pothula et al. 2018). Apples are sorted into two grades (i.e., fresh and processing) and finally sent to the respective bins by a custom-designed sorting mechanism (Lu et al. 2018). The sorting system has several innovative design features; it is simple in design, compact, and efficient (able to handle 6 fruits/s). There are three bin fillers installed with the sorting system. These bin

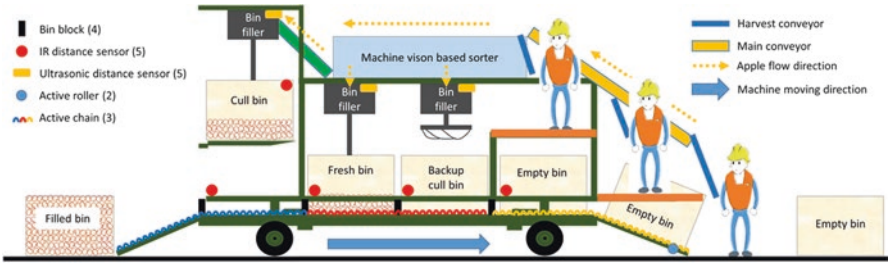


Fig. 11.4 Schematic of the apple harvest and infield sorting machine with four major design features, i.e., harvest-aid stages with conveyors, machine vision-based sorting system, automated bin fillers, and automated bin handling system. (Reproduced with permission from Zhang et al. 2017b)

Fig. 11.5 Apple harvest and infield sorting machine being tested in a commercial orchard in Michigan during the 2017 harvest season

fillers are compact and fully automated. For any given time, two bin fillers are at working status; one is handling fresh apples, while the other cull apples. The third bin filler is used as a backup for handling cull apples when the regular cull bin is full and is being unloaded from the machine, so that the harvest crew does not cease picking activities during the replacement of cull bins. When a fresh or cull bin is full, the sensor installed on the bin filler would send a signal to the computer, which then triggers the automated bin handling system to unload the full bin and load an empty bin at the same time. As the machine travels forward in the orchard, it automatically picks up empty bins that have been placed on the ground in advance. The picked-up empty bins are then pulled into the right positions on the machine, using a computer-controlled hydraulic system, coupled with chain conveyors, active rollers, and sensors.

In the following sections, we give a further, detailed description of the automated apple sorting system (Sect. 11.3), the new bin filler (Sect. 11.4), and the automatic bin handling system (Sect. 11.5).

11.3 Automated Grading and Sorting of Apples

Much research has been reported over the past 40 years on the development of machine vision systems, including both hardware and software, for quality inspection of apples (Rehkugler and Throop 1986; Blasco et al. 2003; Cubero et al. 2011; Unay et al. 2011; Sofu et al. 2016; Moallem et al. 2017). Machine vision is now widely used commercially for sorting and grading apples and other fruits in packinghouses. Some of the major fruit sorting equipment manufacturers include Compac Sorting Equipment Ltd. based in New Zealand, Durand Wayland in the United States, Greefa in the Netherlands, and Unitec in Italy. While each system is different in design, a typical machine vision-based sorting system needs to perform these tasks: (1) *fruit singulation*, which separates or parses the incoming, often disorganized fruit and arranges them in a specific, consistent pattern, so that each fruit can be identified and tracked throughout the entire packing process until it reaches the final destination, (2) *fruit transport*, which moves unsingulated and singulated fruit from one place to the next, (3) *fruit rotation,* which is needed for imaging the entire surface of fruit by the vision system, (4) *fruit imaging and grading,* which is done by a color and/or spectral imaging system to determine quality grade based on the analysis of fruit surface color, size or shape, and presence/absence of blemishes, and (5) *fruit sorting*, the final step of sending the graded fruit to their destinations. Many commercial sorting lines are also installed with in-line load cells to weigh each fruit during sorting and grading, so that precision packaging can be realized. A commercial packinghouse needs to handle a large volume of apples. Hence, efficiency and high throughput are critical to commercial packinghouse sorting lines, while space and size for the sorting system are not important considerations in the overall system design. For most commercial sorting lines, fruit singulation, rotation, and sorting are performed in sequence and, in most cases, by separate systems or machines. These machines cannot be directly adopted or adapted for infield use, due to their complexity, large size, and high cost. Therefore, a new sorting system needs be developed to meet the specific requirements for infield use.

An infield sorting system should have the same basic functions, as those described above for packinghouse use. The system must also be compact and low in cost, so that it can be integrated with the existing or new apple harvest platform. Since the system only needs to accommodate a harvest crew of 6–8 workers, an overall throughput of 6–8 fruits/s is needed (assuming each worker, on average, can pick one apple per second) (Zhang 2015). With such low throughput and less stringent sorting requirement, it is possible to develop a low-cost, compact infield sorting system. Based on these considerations, a new automated infield sorting system was developed, which consists of a compact multi-function mechanism for fruit

singulation, rotation, and transport, a color vision system with in-house–developed sorting algorithms, and a compact sorting mechanism for directing graded fruit to the right destinations or bins (Fig. 11.6) (Lu et al. 2018). A detailed description of these major subsystems is given in the following subsections.

11.3.1 Screw Conveyor for Fruit Singulation, Rotation, and Transport

As described above, the sorting system needs to perform fruit singulation, rotation, and transport functions, so that apples coming from the main conveyor would form in line (three lanes in the current design) with proper distance separations and rotate continuously (on the stem-calyx axis or randomly) for imaging the entire surface of the fruit by the computer vision system, while moving forward at the same time. In addition, the sorting system should be able to keep track of individual fruit on the conveyor in real time after the fruit enters the vision inspection chamber, so that the graded fruit can be sent to the right destinations. In commercial sorting lines, bi-cone conveyors and cups are commonly used for transporting and rotating fruits (Throop et al. 2001, 2003). These designs were considered in the initial phase of research but were not adopted because they overall are too complex and take too much space to be a viable solution for infield use. Blasco et al. (2003) and Cubero et al. (2011) developed a robotic vacuum cup to hold and rotate individual fruit under the camera, so that the majority of the fruit surface can be imaged by the

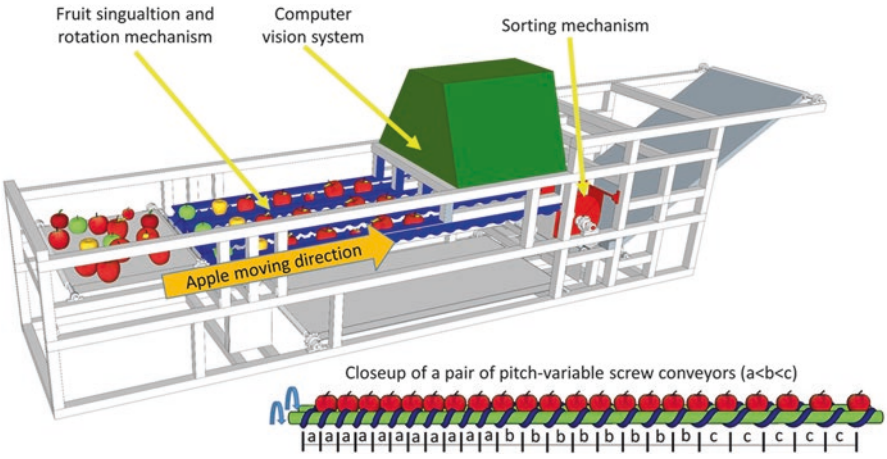


Fig. 11.6 Overview of the automated apple infield sorting system, consisting of a pitch-variable screw conveyor for fruit singulation, rotation, and transport, a computer vision system, and a rotary sorting mechanism. The vision system covers three lanes, each running at a sorting rate of up to 2 fruits/s

camera. This design, however, has low throughput and is not practical for infield grading use.

Subsequently, we developed a new multi-function screw conveyor system, with which fruit singulation, rotation, and transport are accomplished altogether (Fig. 11.6). The screw conveyor system consists of three pairs of screw shafts; the metal shafts are covered with soft material, and the helical strip for driving fruit forward is also made up of soft synthetic material to prevent bruising damage to the fruit during transport on the conveyor. The system is simple, compact, cost-effective, and reliable. Apples first arrive from the main conveyor in disorganized pattern, and they are then organized into three lines. For the beginning section of the screw conveyor, apples in the lines are moving at slow pace and are close to, or even in touch with, each other, due to small pitches for that section of the screw shafts. The fruits are moving faster with the increasing pitch of the helical strip until they are being completely separated (see the close-up of Fig. 11.6). As the fruits are moving forward, they are also rotating, driven by the pair of shafts. Once the apples enter the imaging chamber, the separation distance between the neighboring apples maintains approximately constant, because the pitch of the screw shafts for that section is fixed. The vision system takes multiple images of each rotating fruit to ensure that the entire fruit surface is inspected.

11.3.2 Computer Vision System and Sorting Algorithms

The computer vision system was designed to grade apples at a rate up to 6 apples/s for size and color (Mizushima and Lu 2013a). There are several imaging configurations that have been adopted for different machine vision systems (Fig. 11.7). With

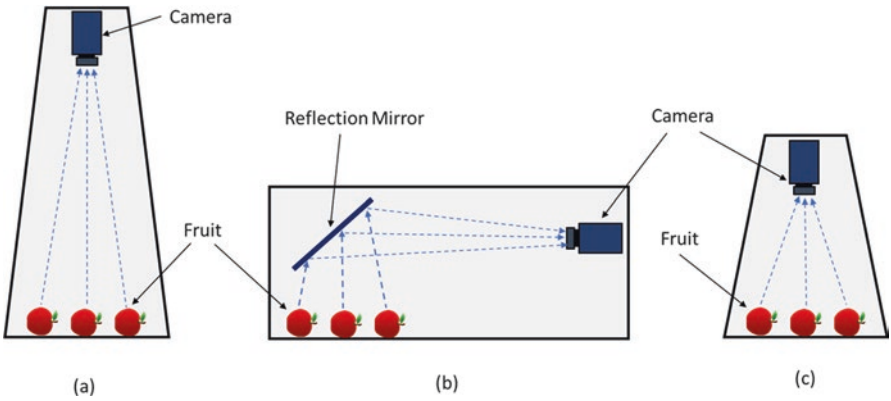


Fig. 11.7 Three camera mounting configurations: (**a**) vertical mounting of the camera with a narrow field of view; (**b**) horizontal mounting of the camera with a reflection mirror arranged in approximately 45° of angle, so that the camera can take images of fruit from the mirror; (**c**) vertical mounting of the camera with a large field of view to reduce the vertical distance to the fruit

the conventional imaging configuration commonly used in the packinghouse, cameras with a narrow field of view (or a long focal length) are mounted directly above the moving fruit (Fig. 11.7a). This imaging configuration is easy to set up and can minimize image distortion, but it requires a relatively large distance (or height) between camera and fruit, which is not desirable for the infield sorting system due to the overall height limitation. To reduce the height of the imaging chamber, Cubero et al. (2010) proposed using a mirror to reflect the fruit images to the camera, which is mounted in a plane parallel to the moving direction of the fruit (Fig. 11.7b). This configuration effectively lowers the overall chamber height, but it is not desirable for infield use, since the sorting system needs to work in a rugged orchard environment, where dust and machine vibration cannot be avoided, which could negatively affect the system's performance.

Instead of using the narrow field-of-view configuration, we chose a wide field-of-view camera (Fire-I, Unibrain, Inc., San Ramon, CA, USA) for the imaging system (Fig. 11.7c), to limit the overall chamber height, while keeping the system simple (Mizushima and Lu 2013a, b). Accompanied with this camera mounting configuration is large distortion in the acquired images, due to great variations in the viewing angle and distance from the fruit objects to the camera. Figure 11.8 shows that the size estimation error increases as the fruit is farther away from the camera. Several camera calibration methods have been developed for correcting image distortion (Zhang 2000; Hartley and Zisserman 2004), but they were found unsatisfactory for the current camera setup. Hence, a new distortion correction method of using reference balls of 63.5 mm and 76.2 mm (representing a typical size range of apples) was proposed.

First, a quadratic equation between the location of the ball along the fruit moving direction (x) and the number of pixels for the ball images obtained from the raw images was developed, as shown in Fig. 11.9, where the x-axis is the apple moving direction and y-axis indicates the total pixels obtained from the raw images. As the ball centroid moves further away from the camera center, the measured area became smaller, resulting from the increased distance. The correction curves shown in

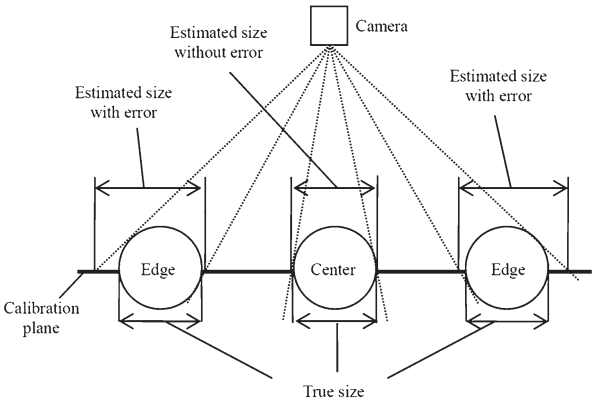


Fig. 11.8 Errors in fruit size (or weight) estimation caused by varied distances of objects from the camera. (Reproduced with permission from Mizushima and Lu 2013a)

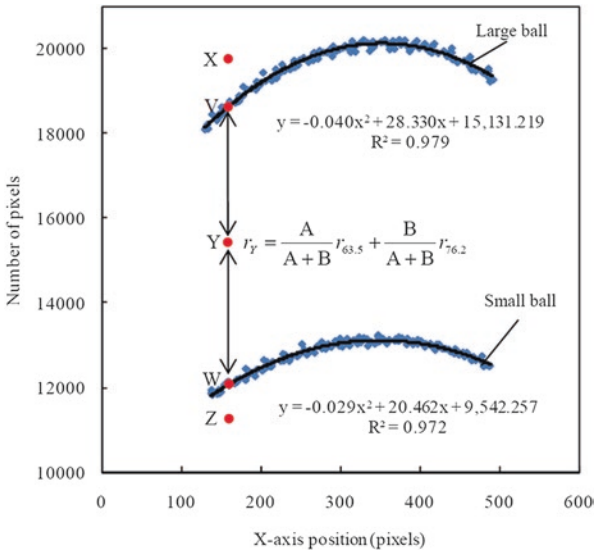


Fig. 11.9 Relationship between the number of pixels obtained from the images of the small (63.5 mm) and large (76.2 mm) reference balls and their centroid position x (x along the fruit moving direction). (Reproduced with permission from Mizushima and Lu 2013a)

Fig. 11.9 are unsymmetrical, mainly because the camera was not mounted at the center of the chamber along the fruit moving direction.

Second, the variable pixels per unit dimension could be calculated using Eq. 11.1.

$$r(x) = \sqrt{\frac{\pi\left(\frac{d}{2}\right)^2}{ax^2 + bx + c}} \tag{11.1}$$

where *r(x)* is the variable pixels per unit dimension function; *d* is the actual diameter of the calibration ball (mm); *x* is the object centroid in the image (pixels; along the fruit moving direction); *a*, *b*, and *c* are the coefficients corresponding to the obtained quadratic equation for the calibration balls (Fig. 11.9). Third, the variable pixels per unit dimension for different conditions were obtained. If the measured number of pixels is larger than the large ball (76.2 mm) (X in Fig. 11.9) or smaller than the small ball (63.5 mm) (Z in Fig. 11.9), the variable pixels per unit dimension function for the large ball and small ball would be used, respectively. Otherwise, the variable pixels per unit dimension function would be recalculated based on its ratio of distance VY (A in Fig. 11.9) and YW (B in Fig. 11.9).

The USDA standards (USDA 2002) define apple size based on the maximum equatorial diameter, but few researchers have actually used this criterion to grade fruit, owing to the difficulty of detecting the stem/calyx and orientation of moving apples (Throop et al. 2003; Pla et al. 2001). Mizushima and Lu (2013a) developed two algorithms (i.e., stem and symmetry detection) to calculate fruit maximum

equatorial diameter. If a stem is detected in the image, a radius function (number of contour points and the distance between the contour points and centroid) is used to determine the orientation of the apple, after which the maximum equatorial diameter could be calculated (Fig. 11.10). If a stem is not detected in the image (some apples lost stems during harvest), the symmetry detection algorithm is used to estimate the orientation of the symmetry axis. Perpendicular to the symmetry axis, the maximal equatorial diameter can be calculated. Furthermore, an automatic adjustable algorithm for apple color evaluation was developed for color sorting of apples, using support vector machine, coupled with Otsu's thresholding method (Mizushima and Lu 2013b).

An image processing software program for inspecting fruit quality (mainly size and color, with the defect detection algorithm being added later) was developed using Visual Studio 2010 Compiler and Qt and OpenCV library (Version 2.3). The software program, consisting of four algorithms, assesses apples by extracting 12 features from each image (Table 11.1).

11.3.3 Sorting Mechanism

The sorting mechanism directs the apples into different bins based on the grading results from the computer vision system. The sorting mechanism should be compact and reliable, and meet the throughput of 6 apples/s. As discussed earlier, the bi-cone and cup conveyors are widely used in commercial sorting systems. In the initial

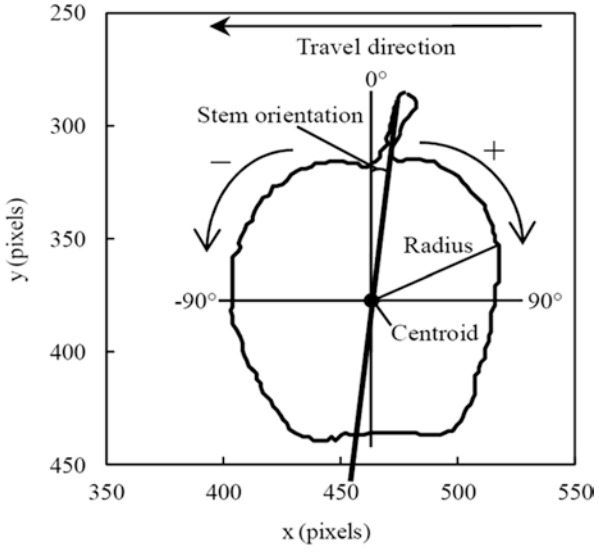


Fig. 11.10 Contour points extracted to determine fruit orientation using the detected stem and centroid. (Reproduced with permission from Mizushima and Lu 2013a)

Table 11.1 Features extracted from a single fruit image

Algorithm	Features extracted
Moment algorithm	Major orientation
	Maximum diameter perpendicular to major orientation
	Minor orientation (perpendicular to major orientation)
	Maximum diameter perpendicular to minor orientation
Contour extraction algorithm	Area
	Diameter estimated from area
Stem detection algorithm	Stem orientation
	Maximum diameter perpendicular to stem orientation
Symmetry detection algorithm	First symmetry orientation
	Maximum diameter perpendicular to first symmetry orientation
	Second symmetry orientation
	Maximum diameter perpendicular to second symmetry orientation

Reproduced with permission from Mizushima and Lu (2013a)

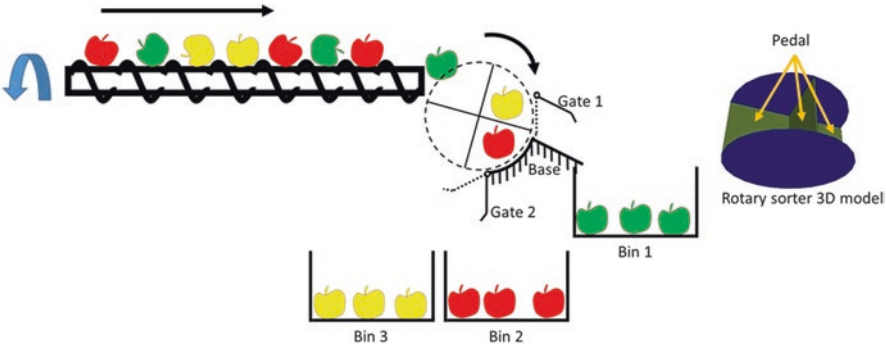


Fig. 11.11 Apple sorting mechanism consisting of a rotary sorter with four compartments and two gates controlled by solenoids

phase of the project, a special cup conveyor design was used for transporting and sorting fruit. But it was found too complex and not quite reliable for field use. Hence, that design was abandoned in the next phase of development. Instead, a new rotary sorting mechanism was proposed and built (Lu et al. 2018). The rotary sorter consists of four compartments, separated by disks or pedals, and two gates (Fig. 11.11) for each lane. The sorter is synchronized with the screw conveyor; when the screw shafts complete one revolution, the sorter would rotate 90° covering one full compartment. As a graded apple is coming out from the imaging chamber at the end of the screw conveyor, it falls into one of the sorter's compartments. After the apple arrives at Gate 1, the gate (solid line) would open to allow the apple, if graded "cull," to exit from the compartment, by gravity and centrifugal force, and go to bin 1. Otherwise, the gate remains closed (dotted line of Gate 1), and the sorter

continues transporting the apple to the next gate (Gate 2). Gate 2 would open (solid line) to allow the apple to exit to bin 2, if it is graded "fresh." Otherwise, the gate stays in close status and the apple continues moving to the next, final bin (i.e., bin 3). The two gates are controlled by solenoids, which are in turn controlled by computer. Hence, this rotary sorter, in principle, can sort apples into three grades (e.g., bins 1, 2, and 3). However, for practical considerations (i.e., easy handling of bins and actual needs by growers for two-grade presorting), apples are only sorted into two grades (i.e., fresh and cull), and bin 3 is used as a backup bin for culls (see further discussion later). It should be mentioned that by increasing the diameter of the disk with more compartments, the rotary sorter's design concept can be used for sorting apples into multiple grades, if such sorting is desired.

11.3.4 Performance Evaluation

To ensure that the computer vision-based sorting system, along with the in-house–developed software, would perform satisfactorily for infield sorting of apples, laboratory and field tests were conducted to evaluate the screw conveyor, the vision system along with the software, and the rotary sorter, as well as the system's bruising damage to apples.

The screw conveyor system was tested and evaluated for its ability of singulating, transporting, and rotating apples (Pothula et al. 2018). The conveyor system should enable the computer vision system to take enough images of each fruit, as it is passing the imaging chamber, so that the entire surface of the apple would be imaged at least once. As such, six apples, two for each of the three sizes (i.e., small, medium, and large), for each of "Golden Delicious" and "Delicious" varieties, were selected. Each quadrant of the test apple's surface was painted with different colors (i.e., blue, red, green, and yellow) (Fig. 11.12). As these painted apples were passing through the vision system at the three throughputs of 1, 2, and 3 apples/s, the camera took images of each apple at 15 fps (frames per second). The acquired images were then analyzed to determine the percent of the entire surface area that had been

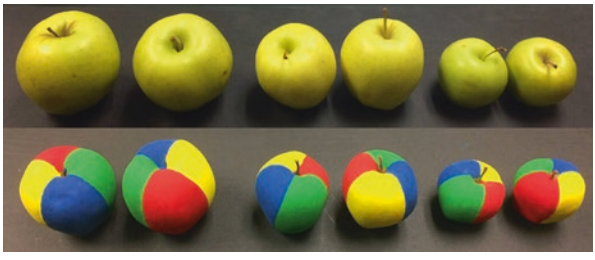


Fig. 11.12 Six "Golden Delicious" apples, two for each of the three sizes, painted with different colors for assessing the screw conveyor design in rotating each apple for imaging; original apples before (top) and after each quadrant of the apples painted with different colors (bottom)

imaged by the vision system. Results showed that as the throughput of the screw conveyor increased from 1 to 3 apples/s per lane, the number of times for which the entire apple surface was inspected decreased from 6 to 2. As expected, higher conveyor speed means faster linear moving speed for apples, thus fewer images taken by the vision system. At the maximum speed of 3 apples/s (or 9 apples/s for the sorting system), the entire surface of each apple was imaged at least twice. Hence the screw conveyor design was successful in singulating, rotating, and transporting apples for infield presorting application.

The computer vision system was able to acquire and process one image (several apples in one image) within 50 ms (image capture, size estimation, and color assessment needed approximately 12 ms, 20 ms, and 15 ms, respectively). Thus, the vision system can meet the requirement of 6 apples/s for infield presorting. The relatively short imaging and processing times provide room for further implementation of additional sorting functions, such as defect detection. The vision system was also tested for size grading of "Delicious" and "Golden Delicious" apples (Mizushima and Lu 2013a). Approximately, 86% of the test fruits were correctly identified for the stem-calyx orientation, within ±20° accuracy. The system was able to estimate fruit maximum equatorial diameters, with the overall root mean squared error of 1.79 mm, which is considered acceptable for practical use. Furthermore, the vision system was also evaluated for color sorting capabilities; it achieved superior results, when compared with a commercial sorting system, for the color sorting of 300 "Delicious" apples of three commercial grades (i.e., Extra Fancy, Fancy, and Cull based on the USDA grading standards), with overall classification errors of being less than 2% (Mizushima and Lu 2013b).

The sorting system was further evaluated for sorting accuracy and bruising damage at the throughputs of 2, 2.5, and 3 apples/s lane in single apple feeding mode. Results showed that overall the system performed satisfactorily, with sorting accuracies for these speeds all above 90% (achieving 99% and 100% at the rate of 2 apples/s in the two setups). Bruising damage evaluation showed that more than 95% of the apples, after having run through the sorting system, were graded as Extra Fancy, which has met the industrial requirement for commercial use (< 5% of apples downgraded from Extra Fancy) (Peterson et al. 2010). It was, however, noticed that when the system ran at a higher speed (2 fruit/s or higher), some apples in the rotatory compartments did not have enough time to roll out of the gates, which was most likely due to the improper timing for opening and closing the gates. Better timing of the gates and improved design of the compartments should help to address these issues when the sorter is running at a speed greater than 2 fruit/s lane.

11.4 Automatic Handling of Harvested Apples

The bin filler is the most critical part for automatic handling of harvested apples. Bin fillers currently are not popularly used for the harvest platforms made in the United States, mainly because the existing bin fillers are still not satisfactory in

performance, too complicated, and not fully automated (Zhang and Heinemann 2017; Zhang et al. 2018a). To presort apples in the field, multiple bin fillers (at least two) are needed for handling cull and fresh apples. It is thus critical that these bin fillers are compact and simple in design, so that they would be cost-effective and can be easily integrated with the harvest platform. It is also important that these bin fillers meet the performance requirements (i.e., being safe and reliable, achieving uniform distributions of fruit in the bin, and causing minimal or no bruising damage during fruit handling). Finally, the bin filler needs be fully automated, so that it does not need human assistance or intervention and would cause no or little disruption to the harvest activities during the bin replacement process. An extensive review of the existing and past bin fillers indicated that none of the existing bin filling technologies would meet these requirements without substantial modification. Therefore, considerable effort was taken on the development of a new generation bin filler for incorporation with the apple harvest and infield sorting machine.

11.4.1 Development of New Bin Fillers

The bin filler needs to fill each bin with 400–500 kg of apples, and it should thus meet certain dimension requirements. A typical bin used in the United States has a height of 0.86 m and horizontal dimensions of 1.22 m by 1.02 m. Hence, the new bin filler must be able to move a vertical distance of no less than 0.86 m, with enough horizontal dimensions to fill the bin evenly. Moreover, the bin filler should handle at least 6 apples/s to meet the overall harvest capacity requirement. In the overall system design for the apple harvest and infield sorting machine (Figs. 11.4 and 11.5), it was necessary that the bin fillers be mounted directly beneath the apple sorting system (Sect. 11.3). This means that the bin filler needs to receive apples released from the rotary sorter (Sect. 11.3.1) and then place them in the bin, with varying distances between 0.86 and 1.72 m (two times the bin height).

Two types of commercial bin fillers for apple harvest use are presented in Sect. 11.1.2. Though the spinning pinwheel design is simple and low in cost, it requires the use of decelerators to lower apple speed and then release the fruit onto the pinwheel at sufficiently low speed (Lehnert 2010). The decelerators are not suitable for infield sorting because of the large size. After the evaluation of different bin fillers, a new design concept was proposed, which allows apples from the rotary sorter to fall freely to the bin filler (with the falling distance varying with the filling level in the bin) (Fig. 11.13). To better control and also decelerate the freely falling apples before arriving at the pinwheel, two pairs of foam rollers, spinning in opposite directions, were used. The first pair, fixed on the sorting system, receives apples from the rotary sorter and then releases them in vertical direction to ensure that the apples fall directly into the gap of the lower pair of rollers, and the second, lower pair of foam rollers, mounted on the bin filler frame just above the pinwheel, catches the falling fruit and then releases the apples onto the pinwheel beneath at a controlled speed. The use of two pairs of foam rollers greatly simplifies the overall

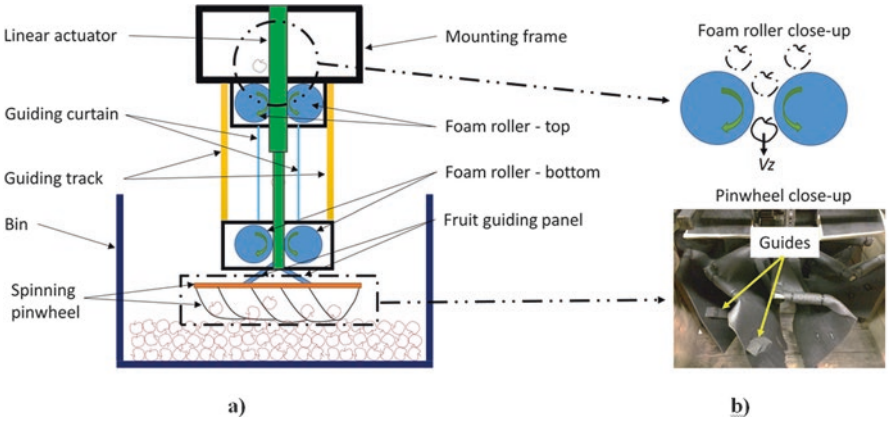


Fig. 11.13 Schematic of the bin filler (**a**) and close-up of foam rollers and pinwheel with multiple soft pads (**b**)

design of the bin filler and makes the system simple and compact. The bin filler, mainly composed of a pair of foam rollers and the pinwheel, moves vertically by using a linear actuator controlled by an onboard microcontroller. A detailed description of the major design features of the bin filler is given in Zhang et al. (2018b).

To realize the automatic bin filling process, an IR sensor and a Hall Effect sensor are used together to monitor the relative distance between the pinwheel and apple level in the bin and the pinwheel spinning speed, respectively. As the bin is being filled with more apples, the relative distance between the apple level and the pinwheel decreases and, at the same time, the pinwheel spinning speed tends to decrease due to increased force exerting on the motor for rotating the pinwheel. Two thresholds (one for relative distance and one for spinning speed) are preset in the microcontroller algorithm. When both real-time detected values are smaller than the preset thresholds, the linear actuator would be triggered to lift the pinwheel and foam roller for a specific distance. The bin filler control algorithm flowchart is shown in Fig. 11.14.

In testing and evaluating the initial version of bin filler which was installed with four soft pads, it was noticed that apple collisions occurred at the foam rollers and pinwheel pads when the system was running at the full capacity (i.e., 6 fruit/s). In addition, uneven distributions of apples in the bin during filling were also observed. Subsequently, improvements were made by increasing the number of soft pads from four to nine, so that each compartment of the pinwheel would not have more than one apple in any given time (Fig. 11.13). This effectively reduces or avoids the occurrence of apple collision at the pads. In addition, foam guides were also added at the end of pads to decelerate the rolling apples before exiting the pads as well as improve fruit distributions in the bin.

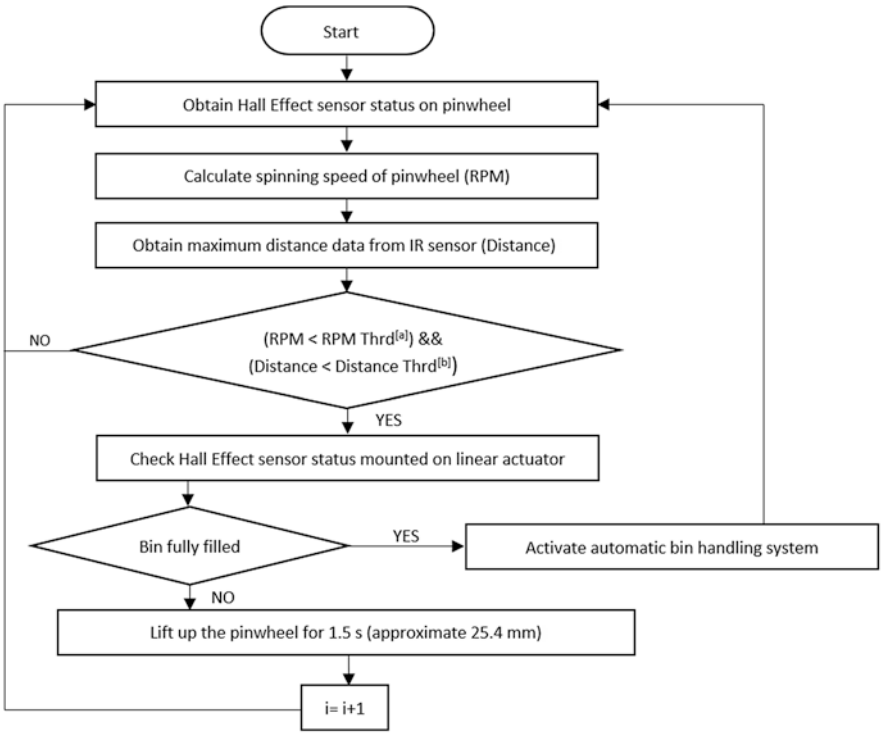


Fig. 11.14 Flowchart of the software program for automatically monitoring, recording, and controlling the bin filler. (Reproduced with permission from Zhang et al. 2017a)

11.4.2 Performance Evaluation

After incorporation into the apple harvest and sorting machine (Fig. 11.5), the new bin filler was tested and evaluated in both laboratory and orchard. Field tests were conducted in 2017 to evaluate the bruise damage by the bin filler for two varieties of apple ("Gala" and "Blondee") in a commercial orchard in Michigan. Four pickers, two on the ground and two on the platform, picked apples and then placed them onto the harvest conveyors. After the fresh bin was fully filled, the computer-controlled hydraulic system was triggered to unload the bin onto the ground. Four hundred apples for each cultivar ("Gala" and "Blondee") were randomly collected from the bin for bruise evaluation using the USDA fresh market standards (Table 11.2). Results indicated that 99% and 98% of "Gala" and "Blondee" apples were graded "Extra Fancy," confirming that the automatic fruit handling system has met the industrial bruising requirement.

Laboratory tests were intended to evaluate the fruit distribution and bruising by the bin filler. Currently, visual observation is commonly used for the evaluation of a

Table 11.2 Classification of bruise damage (Peterson et al. 2010)

Class	USDA fresh market standards	Bruise specification
1	Extra Fancy	No bruising
2	Extra Fancy	Bruise diameter ≤ 3.2 mm (1/8 in.)
3	Extra Fancy	Bruise diameter 3.2 to 6.4 mm (1/ 8 to 1/4 in.)
4	Extra Fancy	Bruise diameter 6.4 mm (1/4 in.) to 12.7 mm (1/2 in.) or area of several bruises <127 mm^2
5	Fancy	Bruise diameter 12.7 to 19 mm (1/2 to 3/4 in.) or total area of multiple bruises <283 mm^2
6	Downgraded	Bruises larger than the tolerances in "Fancy"
7	Downgraded	Cuts or punctures of any size

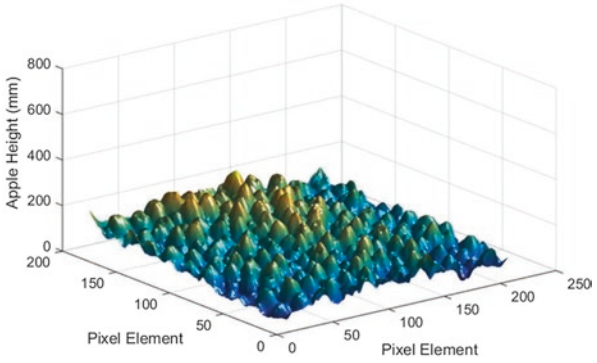


Fig. 11.15 A depth image for apple-in-the-bin distributions acquired by the Kinect-v2 camera. (Reproduced with permission from Zhang et al. 2018b)

bin filler for fruit distributions in the bin, which is subjective, inaccurate, and time-consuming. Hence, we proposed an objective and cost-effective approach, using a depth camera (Kinect-v2, Microsoft Corp., Redmond, WA, USA), for quantitative evaluation of fruit distributions in the bin (Zhang et al. 2018b). The Kinect-v2 camera was calibrated, and the absolute difference between the camera-based height and actual height (manually measured) ranged from 0.1 to 11.6 mm, with the mean absolute error of 1.3 mm (Fig. 11.15), which is much smaller than the size of a typical apple. Hence, the depth camera was considered appropriate for the evaluation of apple-in-the-bin distributions.

The performance of the bin filler for fruit distributions was tested at a throughput of 4 apples/s. During a field test, it was noticed that the center lane of the three-lane sorting system tended to have more apples than the two side lanes. Hence, in the laboratory evaluation, both uniform and nonuniform fruit feeding patterns (designated as UF and NF respectively) were considered. For UF, each lane would handle the same number of apples (1.33 fruit/s), while for the NF, the center lane would handle twice the number of apples as each side lane did. When the bin was

approximately 1/3, 2/3, and 7/8 filled, depth images were taken to quantify the fruit distributions in the bin. Standard deviation (SD) of the measured height (3 replications) was used as a measure of fruit distributions. The smaller the SD value is, the more even the fruit distribution is. Results showed that when the bin was about 1/3 filled, the SDs were not significantly different between the two feeding patterns. This is because the bin floor is flat, and apples were easy to move from the landed place to the final destination (Fig. 11.16). With the bin being filled with more apples (2/3 or 7/8 of the bin height), the later arrived ones tended to dwell at the landed place, thus resulting in a higher standard deviation (Fig. 11.16) or less even fruit distributions. Overall, feeding pattern (i.e., UF and NF) did not have a significant effect on the distributions of fruit in the bin, which is important as the fruit feeding pattern in the real condition may vary, depending on the picking speed of individual workers and number of workers working on the platform.

To further understand spatial variations in the fruit filling height, the depth images were divided into 8 × 8 blocks (total 64 blocks) at the three levels of filling. In each image, the absolute differences between the base block (the one with the largest height value among the 64 blocks) and non-base blocks were calculated, and they were categorized into four height groups. Group 1 for those blocks with the absolute difference being less than or equal to 70 mm, group 2 for blocks with the absolute difference between 71 mm and 140 mm, group 3 for blocks between 141 mm and 210 mm, and group 4 for blocks of larger than 210 mm. Figure 11.17 shows the 2D display of fruit distributions, in which darker blocks represent the larger absolute differences in height. The bin filler overall had satisfactory performance in fruit distribution in the bin under both uniform and nonuniform feeding conditions. The fruit distributions were less even around the four corner areas of the bin, as indicated by darker blocks in Fig. 11.17, because the circular pinwheel had

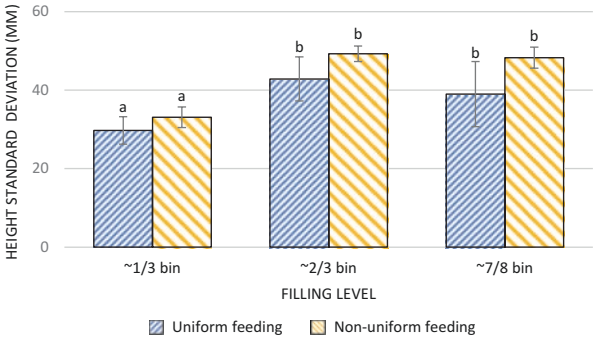


Fig. 11.16 Apple distributions in the bin as measured by the surface height standard deviation in mm, under the uniform and nonuniform feeding patterns. The fruit distributions were measured at the three levels of filling in the bin (i.e., 1/3, 2/3, and 7/8 bin filled). Whiskers associated with individual bars stand for two standard deviations calculated from three replications. Bars for each group (i.e., 1/3, 2/3, and 7/8 bin filled) with the same letters are not significantly different by Tukey Test at the level of 0.05. (Adapted with permission from Zhang et al. 2018b)

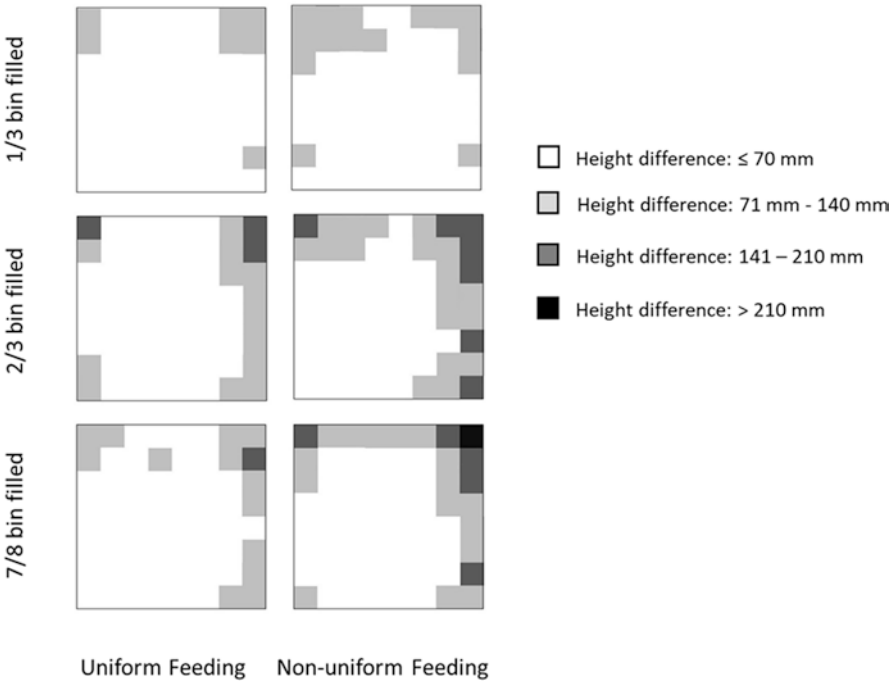


Fig. 11.17 Relative height variations of the fruit distribution in the bin under the uniform and nonuniform feeding conditions. The whole bin area was divided into 8 × 8 blocks or areas, with respect to the highest block measured at the three filling levels (1/3, 2/3, and 7/8 bin filled). (Adapted with permission from Zhang et al. 2018b)

difficulties moving apples to these corners. Overall, the bin filler has met our goals in terms of simplicity and compactness in design, cost, performance, and automation; it can handle fruit at a capacity of 6 fruits/s or higher.

11.5 Automatic Handling of Fruit Bins

As discussed earlier, the current bin handling process for the commercial harvest platforms would result in a significant loss (i.e., 10–25%) in harvest productivity, because it still needs human involvement and the pickers have to cease picking activities during the bin handling. Hence, it is important that the apple harvest and infield sorting machine can automatically handle full and empty bins without human intervention or involvement so as to minimize the downtimes for the harvest crew during the bin handling. This means that the system should be able to automatically monitor the bin filler and/or bin filling status. Once the bin is full, the bin filler would alert the system, so that it would automatically unload the full bin and load an empty bin, and once the latter is in place, the bin filler would automatically insert

into the new bin and begin filling again. In addition, since the bin replacement process could take some time (typically 1–2 min in the current commercial harvest platforms), it is desirable to use a backup bin for apple filling during the transitional period, so that the harvest crew does not need to cease picking activities during the bin replacement process. Based on these considerations, we developed an automatic bin handling system for the apple harvest and infield sorting machine.

To integrate the automatic bin handling function, the harvest and sorting machine was designed such that it can carry four bins at the same time; all four bins are lined up on the chain conveyor, which is driven by a computer-controlled hydraulic system. The automatic bin handling system consists of multiple IR and ultrasonic sensors, active rollers, and hydraulic-driven chains (Fig. 11.4). To effectively implement the automatic bin handling function, empty bins need to be pre-placed in the orchard at appropriate spacing intervals, which is already being practiced by many growers in the United States. As the machine is moving forward, the active chain and rollers in the front of the machine are both turned on. The tilted front forks would enter beneath the bin and lift the bin partially, until after it is being engaged with the active rollers and hydraulic-driven chain, which pull the bin to another horizontal chain. Once the empty bin is loaded on the horizontal chain, its movement is being monitored by the multiple IR sensors. After the bin has been moved to the proper position (Fig. 11.4), the IR sensor sends a signal to the computer, which activates one of the bin blocks, which are installed next to the chain tracks on the main frame of the harvest and sorting machine, from the horizontal to vertical position, thus stopping the bin from further horizontal movements. Following this procedure, additional bins are loaded one after another until all four bins are in place. Thereafter, the computer would trigger the hydraulic system to lift the first bin to the elevated position for handling culls. The cull and fresh bin fillers then begin to lower automatically to the bin floor. After completion of automatic initialization for the two bin fillers, the system is ready to perform normal harvesting and sorting operations.

After the fresh bin is full, the bin filler sends signal to the computer, which suspends the sorting system and harvest conveyors, followed with activation of the hydraulic chains to push the fresh bin out to the ground (Fig. 11.18). At the same time, a new empty bin is moved into the position for fresh apples. The fresh bin filler then automatically lowers into the new empty fresh bin, and the whole system then resumes normal harvesting and sorting operations.

In a second situation, the cull bin at the elevated position is filled to the full level. The sorting system then directs all new culls to the backup cull bin. Meanwhile, the hydraulic system is triggered to lower the cull bin to the chassis, which is then unloaded by a chain conveyor to the ground. During the unloading of the cull bin, normal harvesting and sorting operations are not disrupted. After the fresh bin is filled again, it is unloaded to the ground, and the backup cull bin is then moved to the elevated cull bin position, and a new empty bin is moved into the fresh bin position (Fig. 11.19). The system returns to the same working status as that shown in Fig. 11.18.

The bin handling system has been tested off-season for several times, and the overall performance was satisfactory. However, it was found that the IR sensors

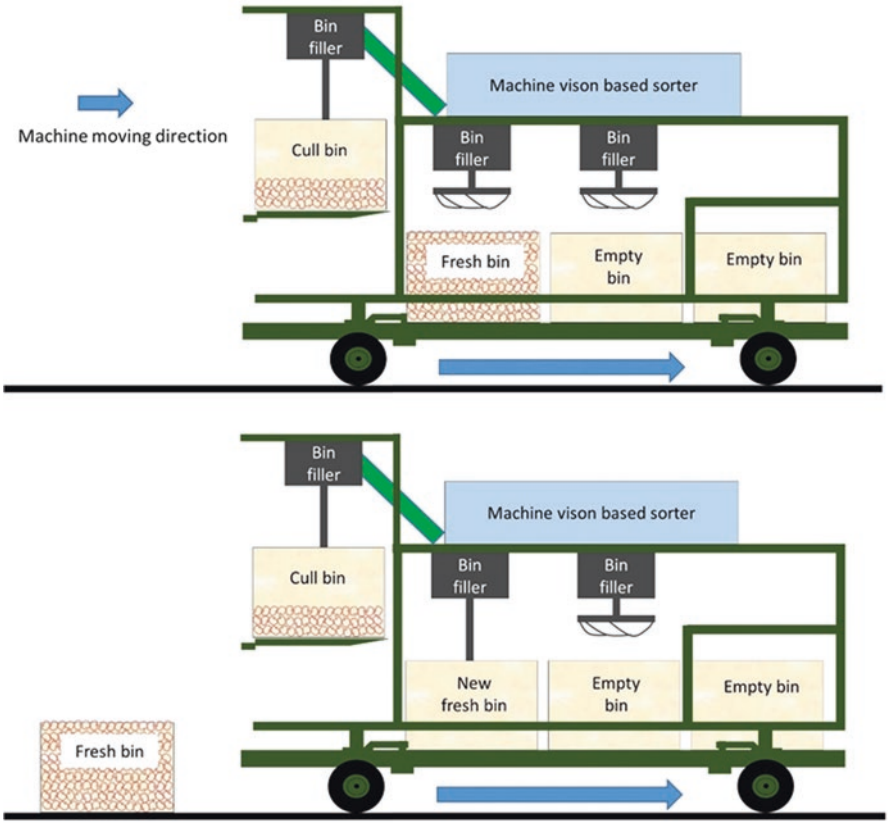


Fig. 11.18 Automatic handling of a fresh bin after it is full (top). The full fresh bin is unloaded automatically to the ground, and at the same time, a new empty bin is moved into the fresh bin position (bottom). After the bin filler is lowered to the proper position, the system resumes normal harvesting, sorting, and filling operations

sometimes could not provide accurate bin position data, due to interference from natural lighting, and the sensors were also prone to moisture. Field testing of the machine also suggested that all sensors used in the automatic bin handling system should be water, dust, and dirt proof. For the current bin handling system, the harvest crew still needs to cease picking activities for 1–2 min during the replacement of fresh bins. Further improvements in reducing the crew downtimes can be made by utilizing the backup cull bin for fresh apple filling, when the regular fresh bin is filled, which can be done by modifying the computer program without any hardware change.

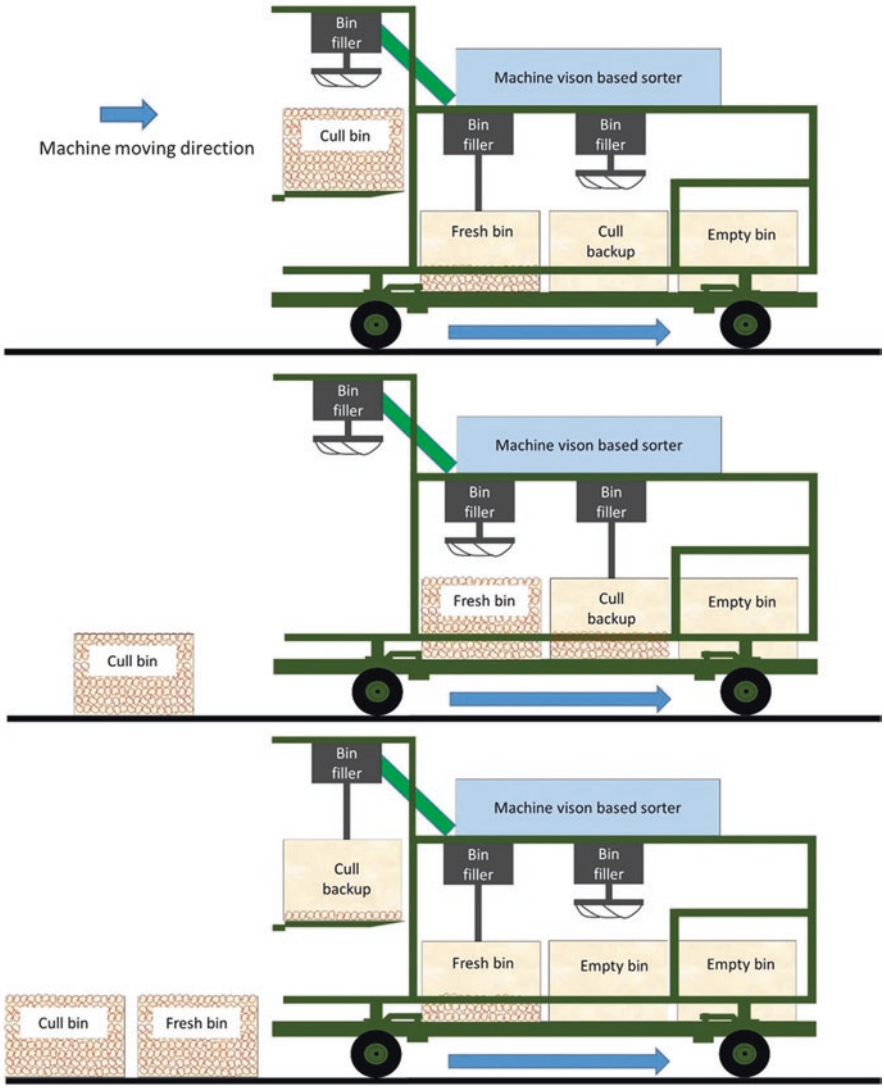


Fig. 11.19 Automatica handling of a full cull bin. After the cull bin has been filled (top), all culls are directed to the backup cull bin (middle). At the same time, the full cull bin is being unloaded to the ground automatically (middle). After the fresh bin is filled, it is unloaded and the backup cull bin is moved to the elevated cull bin position (bottom). The normal bin filling process, as shown in Fig. 11.18, resumes

11.6 Summary and Concluding Thoughts

Commercial harvest platforms have gained popularity in recent years as a means for improving harvest productivity and the working condition for workers, but their overall level of automation is still low, thus limiting them from achieving optimum harvest productivity. This chapter reviewed the current status of apple harvest-aid technology, fruit postharvest handling and presorting, and handling of harvested fruit and bins in orchard. It then described the development of a new apple harvest and infield sorting machine to enhance harvest productivity and achieve cost savings for postharvest handling. The new system had several innovative automation features, which include automated grading/sorting, handling of harvested fruit, and fruit bin handling. The patented infield presorting system utilized a simple, compact pitch-variable screw conveyor design to accomplish the fruit singulation, rotation, and transporting functions simultaneously. By using a wide field-of-view imaging configuration, coupled with appropriate system calibration procedures, a low-cost, compact imaging system was constructed, which enables to inspect and grade each fruit based on color and size. The graded apples are then sent to the cull and fresh bins by a compact rotary sorter. The new automated bin filler consists of two pairs of foam rollers, allowing to catch and deaccelerate freely falling apples and then release them to the pinwheel installed with multiple soft pads, which distributes the fruit in the bin evenly with minimum bruising. Laboratory and field tests showed that the machine vision-based sorting system and the bin filler have met the requirements for infield presorting use. Moreover, the automatic bin handling function has been incorporated into an apple harvest and sorting machine, which helps minimize the downtimes for the harvest crew during the bin replacement process, thus enhancing the overall harvest productivity.

With the current imaging system, apples are graded and sorted based on color and size. Defect sorting is important for presorting and hence should be incorporated in the imaging algorithm in the future. The sorting system was designed for handling a throughput up to 6 apples/s. A new sorter has been developed recently to replace the rotary sorter, which allows to sort apples into two grades (cull and fresh) at a speed of up to 10–12 fruit/s. Research is also being taken in the development of a new picking-aid technology for incorporation into the system, which is expected to further improve harvest productivity. Hence, improvements to the imaging-based sorting system and the bin filler are needed to meet the higher throughput requirement (i.e., 9 apples/s or higher). Moreover, additional functions, such as yield and quality mapping, should be considered in the future for the apple harvest and sorting machine, to help growers and packinghouses better monitor and keep track of harvested fruit from the field to the retailing market. Ultimately, it would be highly desirable to incorporate robotic technology with automated sorting technology in one machine to achieve total automation in fruit harvest and infield sorting. Finally, while the current machine is developed for apples, the technology, with some modification, should also be applicable for other fruits, such as peaches and pears.

Disclaimer Mention of commercial products in the chapter is only for providing factual information for the reader and it does not imply endorsement or recommendation by USDA over those not mentioned.

References

Baugher T, Schupp J, Travis J, Hull L, Ngugi H, Krawczyk G et al (2009) Specialty crop innovations: Progress and future directions. Pennsylvania State University Extension, University Park

Blasco J, Aleixos N, Moltó E (2003) Machine vision system for automatic quality grading of fruit. Biosyst Eng 85(4):415–423

Boyer J, Liu R (2004) Apple phytochemicals and their health benefits. Nutr J 3:5

Cubero S, Moltó E, Gutiérrez A, Aleixos N, García-Navarrete O, Juste F, Blasco J (2010) Real-time inspection of fruit by computer vision on a mobile harvesting platform under field conditions. Prog Agric Eng Sci 6(1):1–16

Cubero S, Aleixos N, Moltó E, Gómez-Sanchis J, Blasco J (2011) Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables. Food Bioprocess Technol 4(4):487–504

Gallardo K, Galinato SP (2012) 2012 cost estimates of establishing, producing, and packing Red Delicious apples in Washington. Washington State University, Pullman

Gallardo K, Taylor M, Hinman H (2010) 2009 cost estimates of establishing and producing Gala apples in Washington. Washington State University, Pullman

Hartley R, Zisserman A (2004) Multiple view geometry in computer vision, 2nd edn. Cambridge University Press, Cambridge, UK

Jones R (2015, February 23) The state of mechanical apple harvesting. Growing Produce. https://www.growingproduce.com/fruits/the-state-of-mechanical-apple-harvesting/

Lehnert R (2010, December 1) Vacuum harvester passes bruising tests. Good Fruit Grower

Lehnert R (2013) In-orchard sorting. Good Fruit Grower. http://www.goodfruit.com/in-orchardsorting/

Lehnert R (2015) Processed apples have a future. Good Fruit Grower 66(3):14–17

Lu R, Pothula A, Mizushima A, Van Dyke M, Zhang Z (2018) Device and system for sorting apples in the orchard. U.S. Patent no. 9,919,345. Washington, DC

Mizushima A, Lu R (2011) Cost benefits analysis of in-field presorting for the apple industry. Appl Eng Agric 27(1):33–40

Mizushima A, Lu R (2013a) A low-cost color vision system for automatic estimation of apple fruit orientation and maximum equatorial diameter. Trans ASABE 56(3):813–827

Mizushima A, Lu R (2013b) An image segmentation method for apple sorting and grading using support vector machine and Otsu's method. Comput Electron Agric 94:29–37

Moallem P, Serajoddin A, Pourghassem H (2017) Computer vision-based apple grading for golden delicious apples based on surface features. Info Process Agri 4(1):33–40

Peterson DL, Tabb AL, Baugher TA, Lewis K, Glenn DM (2010) Dry bin filler for apples. Appl Eng Agric 26(4):541–549

Pla F, Martí JMS, Sánchez JS (2001) An integral automation of industrial fruit and vegetable sorting by machine vision. In Proceedings of 2001 8th IEEE international conference on emerging technologies and factory automation conference, 2, 541–546

Pothula A, Zhang Z, Lu R (2018) Design features and bruise evaluation of an apple harvest and infield presorting machine. Trans ASABE 61(3):1135–1144

Produce for Better Health Foundation (2015) State of the plate, 2015 study on America's consumption of fruit and vegetables. Accessed on 15 May 2020 from http://www.pbhfoundation.org/pdfs/about/res/pbh_res/State_of_the_Plate_2015_WEB_Bookmarked.pdf

Rehkugler GE, Throop JA (1986) Apple sorting with machine vision. Trans ASAE 29(5):1388–1397

Schotzko RT, Granatstein D (2005) A brief look at the Washington apple industry: past and present, Project report SES 04–05. Washington State University, School of Economic Sciences, Pullman, WA. http://www.agribusiness-mgmt.wsu.edu/agbusresearch/docs/SES04-05_BRIEF_LOOK_WAFTA.pdf

Sofu MM, Er O, Kayacan MC, Cetişli B (2016) Design of an automatic apple sorting system using machine vision. Comput Electron Agric 127:395–405

STATISTA (2017) Global fruit production in 2017. https://www.statista.com/statistics/264001/worldwide-production-of-fruit-by-variety/. Accessed 25 Mar 2019

Throop JA, Aneshansley DJ, Upchurch BL, Anger B (2001) Apple orientation on two conveyors: performance and predictability based on fruit shape characteristics. Trans ASAE 44(1):99–109

Throop JA, Aneshansley DJ, Anger WC, Peterson DL (2003) Conveyor design for apple orientation. In: ASAE paper #036123. ASABE, St. Joseph

Unay D, Gosselin B, Kleynen O, Leemans V, Destain MF, Debeir O (2011) Automatic grading of bi-colored apples by multispectral machine vision. Comput Electron Agric 75(1):204–212

USDA (2002) United States standards for grades of apples. USDA, Washington, DC

USDA (2017) Chapter V: Agricultural statistics 2017. In: Statistics of fruits, tree nuts, and horticultural specialties, 1(59). USDA-NASS, Washington, DC. http://www.nass.usda.gov/

Weinstock D (2016, March 18) A gentler journey from bucket to bin. Good Fruit Grower. https://www.goodfruit.com/a-gentler-journey-from-bucket-to-bin/

Wunderlich L, Klonsky KM, DeMoura RL (2007) Sample costs to establish and produce apples: Fuji variety, AP-IR-07. University of California Cooperative Extension, Davis, CA. Retrieved from http://cecentralsierra.ucanr.edu/files/60510.pdf

Zhang Z (2000) A flexible new technique for camera calibration. IEEE Trans Pattern Anal Mach Intell 22(11):1330–1334

Zhang Z (2015) Design, test, and improvement of a low-cost apple harvest-assist unit. Doctoral dissertation, Department of Agricultural and Biological Engineering, Pennsylvania State University, University Park, PA

Zhang Z, Heinemann P (2017) Economic analysis of a low-cost apple harvest-assist unit. HortTechnology 27(2):240–247

Zhang Z, Heinemann P, Liu J, Baugher T, Schupp J (2016) Development of mechanical apple harvesting technology – a review. Trans ASABE 59(5):1165–1180

Zhang Z, Pothula A, Lu R (2017a) Economic evaluation of apple harvest and infield sorting technology. Trans ASABE 60(5):1537–1550

Zhang Z, Pothula A, Lu R (2017b) Development and preliminary evaluation of a new bin filler for apple harvesting and infield sorting machine. Trans ASABE 60(6):1839–1849

Zhang Z, Pothula A, Lu R (2018a) A review of bin filling technologies for apple harvest and post-harvest handling. Appl Eng Agric 34(4):687–703

Zhang Z, Pothula A, Lu R (2018b) Improvements and evaluation of an infield bin filler for apple bruising and distributions, ASABE paper no. 1800921. ASABE, St. Joseph

Chapter 12
Modeling, Simulation, and Visualization of Agricultural and Field Robotic Systems

Brian L. Steward, Mehari Z. Tekeste, Jingyao Gai, and Lie Tang

12.1 Introduction

Agricultural machinery technology needs to adapt to the demands for increased food production to support a growing world population. While at the same time, more volatile climatic and pandemic conditions make it necessary for machinery to accomplish agricultural field operations more quickly and efficiently. Emerging technologies are rapidly advancing, enabling greater autonomy and control of field operations.

Because of the importance of agricultural machines in perceiving cropping conditions and implementing management decisions, they will be a key part of any smart, digital agriculture scheme. Additionally, agricultural and field robots will likely be a part of the agricultural machine portfolio of the future, because autonomous machines enable a step change in the degree of perception and control that can be accomplished when human operator limitations are a constraint. Nevertheless, this type of autonomy in agricultural robotics can only be achieved through the use of multiple layers of technology (Fig. 12.1; Han et al. 2015). As a foundational technology layer, agricultural robotics must have machine architecture with both hardware and software components required for the robot's function. Then the robot must be aware of its location and the environment and cropping system around it. This awareness capability is provided by perception, localization, and monitoring technologies found in the machine awareness layer. Then to act upon the world, control technology is required to navigate and control implement motion. Fourthly, machine behavior needs to be planned and then supervised with technology in a machine behavior layer.

B. L. Steward (✉) · M. Z. Tekeste · J. Gai · L. Tang
Iowa State University, Ames, IA, USA
e-mail: bsteward@iastate.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_12

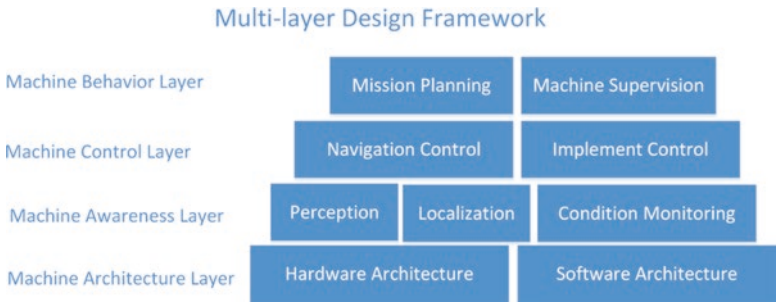


Fig. 12.1 A multilayer design framework for agricultural robotics has four technology layers. (Han et al. 2015)

All of these layers of technology means that agricultural and field robotics are highly complex and will need to be flexible and robust in their operation. Developing this high degree of functionality can only be accomplished with simulation complemented with field testing. Simulation requires the development of models of the robot along with models associated with the soil and organisms with which the robot will interact as well as human operators or supervisors. Additionally, the results of simulation are only useful when they are presented in a way that makes sense to the user of the simulation. Thus visualization of the robot and its virtual world are very important to this simulation ecosystem.

This simulation approach has been taken in the development of self-driving cars. For example, one company, Waymo, recently reported self-driving more than ten million miles on the road. However, they were simulating that amount of driving each day (DeBord 2018; Nelson 2018).

12.2 Simulation Requirements

A challenge for any complex system, like an agricultural robot, is to develop software algorithms that enable the autonomous behavior that is required for the designed function of the robot. For an efficient development process, work on the software needs to take place simultaneously with the development of the robot mechanisms. An even more complicating factor is that agricultural robots need to operate in outdoor agricultural fields, and their availability is determined by weather conditions as well as the crop growing season. If development progress is limited by field conditions, then it will happen at a very slow pace (Shamshiri et al. 2018).

However, simulation technology enables software development in a simulation environment, placing the algorithm development in a loop with the other physical parts of the robot modeled and simulated. This approach is called X-in-the-loop simulation where X is the controller at various stages of development, placed in a closed loop with simulation of the dynamic system or plant to be controlled. Depending on the stage of the development, X can be a model of the controller,

called Model-in-the Loop (MIL) simulation, or just the software running on a general purpose computing platform, like a PC, for Software-in-the-Loop (SIL) simulation. The final stage of XIL is Hardware-in-the-Loop (HIL) simulation where the embedded controller hardware and software is used along with a dynamic simulation of the plant (Rosique et al. 2019). This XIL approach is common in the development of cyber-physical systems today.

To do an XIL simulation of an agricultural robot, several components are required to implement the simulation system (Fig. 12.2). A closed-loop simulation system is first divided up into two parts. There are the algorithms which will be used to implement the autonomous behavior of the robot (left side of Fig. 12.2), and there are the models which the simulator must solve to represent the behavior of the robot (right side of Fig. 12.2).

In the multilayer design framework (Fig. 12.1), the three top technology levels are all algorithm-based, meaning that they are technologies that are implemented in software. Machine awareness, for example, requires that electrical signals produced by sensors are processed in a way that makes sense of the signals so that they are representing something in the physical world to the robot. Machine control involves the control of robot actuators to enable motion for the functions that the robotic system is designed to perform. Machine behavior is the planning and supervision of the robot in order to enable the robot to function with autonomy. All these are needed to be part of the XIL simulation system either as models, software, or hardware. For ease of deployment to the physical robot, it is important that they can be easily transferred to the robot.

To represent the dynamic robot system being controlled, the hardware needed for the simulation must be modeled as well as the environment in which the robot will be operating. So in addition to a model of the robot, the model of the virtual world is required to show how the robot will interact with its surroundings. This is very important to an agricultural robot because it will be interacting with biological systems around it to carry out its intended function. Also important are the models of

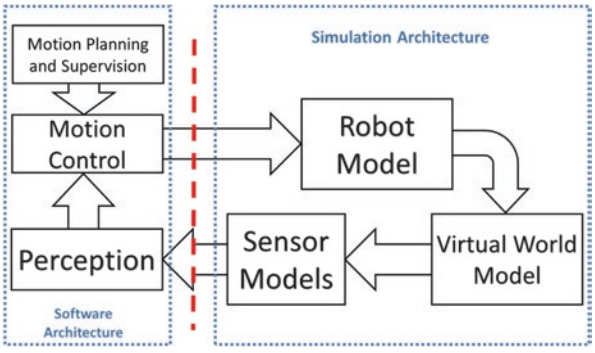


Fig. 12.2 Required elements of a full XIL robot simulation include both the algorithms (the X being either a model, software, or hardware), which will be used to determine robot behavior and the simulation of the robot and environment in which it will operate

the sensors used by the robotic system. In the case of agricultural systems, robot sensors may be cameras or distance sensors, among others. All of these models form the background upon which any simulation must be built.

12.3 Modeling Background

With any dynamic simulation, the physics governing the real-world entity of interest must be represented mathematically (Cellier 1991). This representation of the real-world entity is called the model of the entity. So modeling, the development of the mathematical representations of real-world entities, goes together with dynamic simulation, which involves solving the model equations and representing the behavior of the entity as a function of time and, in some cases, space.

Reality is complex and so making a model of some piece of reality will involve some kind of simplification. In other words, it is impossible to model all of the complexity of the real world, so a model should only represent the reality that results in the behavior of interest to the modeler. Additionally, modeling too much complexity will result in more model development time needed for troubleshooting and validating the model against experimental results. Too much complexity also has an implication at the time of simulation as overly complex models will take too much computational capacity resulting in long simulation times. In this context, it is important to consider two aspects of model development: model scope and model fidelity, as both are important and involve decisions that the modeler must make about their work.

Model scope is the boundary around what is included in the model and thus determines what physical reality is outside of the model. The physical reality inside the scope of the model is called the system; that which is outside is the environment (Fig. 12.3). The environment is the context in which the system operates, and there are mutual influences between the environment and the robotic system. The system inputs are physical signals that have influence on the system. They are causal, meaning that they cause the system behavior to change. The system also has an influence on the environment through its output signals.

Model fidelity is defined differently by various authors. Choi et al. (2017) defined model fidelity "as the degree of output similarity between a reference system and a model that abstracts the reference system." So model fidelity, in this sense, is mainly a faithfulness to produce an input-output response when simulated, which is consistent to that which is observed in the real system. Williams and Alleyne (2014) offered a similar definition of model fidelity being "the extent to which a model can replicate the actual physical event." It is "the degree to which a model reflects the behavior of a real system." However, they later indicated that fidelity is measured by the number of dynamic equations for a system that are included in a model. Thus model fidelity is associated with the quantity of correct dynamic systems equations that are included in a model so that when simulated, the model more faithfully

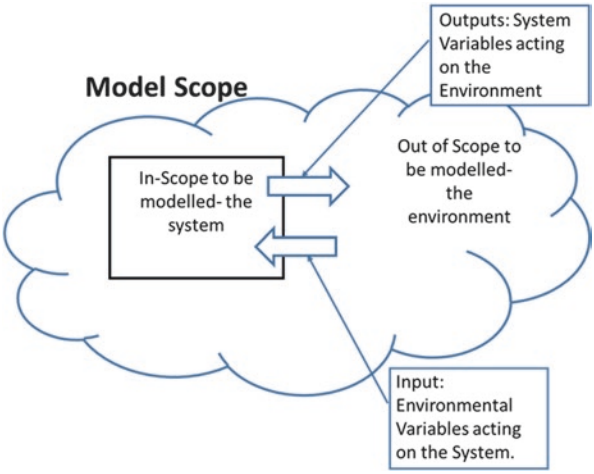


Fig. 12.3 The model scope is the demarcation between the system and the environment. It is determining that which will be modeled, which is called the system, and that which will be left unmodeled, the environment

provides responses that reflect the behavior of the real-world system over a wide range of inputs.

Model fidelity is how much of the system dynamics that a modeler is capturing in their representation of the real-world system. In modeling, not all of the complexity of the real world is captured, but as long as the behavior of interest is represented by the model, it has utility. George Box (1987) is famously quoted, "All modes are wrong, but some are useful." In the development of a model, the modeler needs to decide what behavior is particularly important to capture in the model, and what behaviors are not particularly important to model. For example, Karkee and Steward (2010) modeled the steering system of a tractor and cart system with three models containing various degrees of fidelity of the tire-soil interaction. The simplest model was called a kinematic model in which the wheels moved in the direction that they were pointed. The other models were called dynamic models in which the vehicle was turned based on the lateral forces acting on the wheel based on the interaction between the tires and the soil. Even these "higher" fidelity models were a simplification of the tire-soil interaction which is quite complex (Taheri et al. 2015; see below). When these vehicle system models were simulated, it was shown that at slow speeds, the lower fidelity model was sufficient to capture the steering behavior of interest. In other words, if developers are interested in simulating the steering behavior of a tractor-cart system at low speeds, the additional complexity of the higher fidelity models would have added no value.

For each model needed for an agricultural robot simulation, decisions need to be carefully made about the scope and fidelity of each model. These model attributes need to be selected so the behavior of interest is captured in the model and unnecessary model scope and fidelity are not included. Additional complexity, time, and

effort would be needed for modeling unnecessary system behavior. The scope and fidelity of the model are also related to the computational resources associated with the simulation platform. Computational capability to solve the equations of motion of the models may also influence choices about the scope and fidelity of the models because of the effect on computational speed, which determines how long it will take before the simulation is completed.

12.4 Dynamical System Modeling

Dynamical system models are mathematical representations of system behavior using differential algebraic equations (DAE). Models of the kinematics and dynamics of the mechanical systems associated with a robot are one type of dynamical system models, but electrical, fluid, and thermal physical systems are also modeled as dynamical system models as well as sensor and actuators at the interface between these different engineering systems. Control system dynamics can also be modeled.

Dynamical system models required for most robotic applications are typically lumped parameter models or "0D" models. This label means zero-dimensional because spatially distributed system characteristics are lumped into a topology of interconnected discrete elements. Thus the spatial aspect of the system is moved from partial differential equations into the topology of the model components leaving a system of ordinary differential equations. 0D models typically provide a fidelity that is sufficient for most agricultural robot applications, in comparison with 1D, 2D, or 3D models which include spatial variables in the equations.

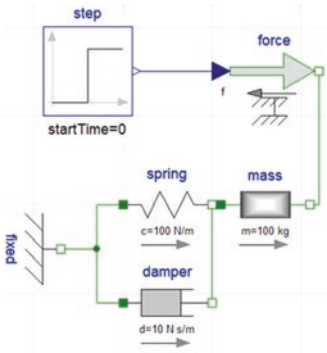
One challenge for modelers is how to represent the system in a form that can be used by computers. The mathematics associated with dynamical system models can be represented in several long-standing forms such as nth-order input-output differential equations, transfer functions, or state-space models consisting of n first-order differential equations often in a matrix representation, where n indicates the order of the system or the number of derivatives required to represent the dynamics of the system (Fig. 12.4). Systems can also be modeled graphically with the modeler representing the system via a user input to the computer that facilitates the construction of a diagram. Such graphical models include block diagram models or connection diagrams with physical modeling. Block diagrams flow out of the analysis in the Laplace transform domain and the representation of systems with transfer functions. As such, systems are represented by blocks with connections showing the flow of signals from one block to another (Mathworks 2019a). Block diagrams have been used for decades particularly in the controls system area.

Another approach to graphically representing physical systems as a model is called physical modeling (Fritzson 2015, p. 6). In physical modeling, graphical objects represent physical objects, and their physical connections are represented through the use of connection diagrams or physical networks (Mathworks 2019a). This approach is acausal as neither direction nor causality is specified in the

$$m\frac{d^2x}{dt^2} + c\frac{dx}{dt} + kx = F(t)$$

$$\begin{bmatrix}\dot{x}\\ \dot{v}\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\frac{k}{m} & -\frac{c}{m}\end{bmatrix}\begin{bmatrix}x\\ v\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}\mathrm{F(t)}$$

$$\frac{X(s)}{F(s)} = \frac{1}{ms^2 + cs + k}$$



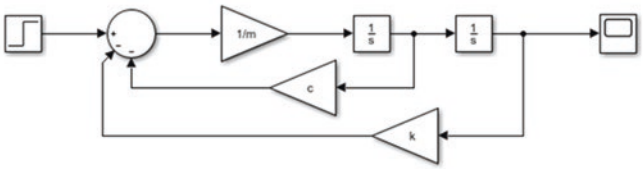


Fig. 12.4 Dynamical systems can be represented in different models. For example, a mass-spring-damper system can be represented (starting in top left and moving CCW) as a differential equation, a state space model, a transfer function, a block diagram, or a physical model

connections. The connections generally represent the paths in which energy is transmitted from one component to another.

Two commonly used physical modeling approaches are Simscape and Modelica. Simscape is the physical network blocks and connection approach embedded in the Matlab Simulink graphical modeling platform (The Mathworks, Natick, MA). It is a good example of dedicated modeling objects that are specifically used only within a specific simulation platform. Modelica, on the other hand, is an open modeling language that can be used to simulate dynamical systems in many modeling and simulation environments. Currently 12 environments are available including Dymola, MapleSim™, Wolfram System Modeler®, and OpenModelica (Modelica 2019).

12.5 Simulation and Visualization Platforms

There are generally two classes of simulation platforms that can be used for simulating agricultural robots. These two classes of platforms have different advantages relative to one another. The choice of a particular simulator depends on the purpose of the simulation. One use-case of a simulator is to have an XIL environment in which software development can take place with testing done by simulating the physical plant. Such an approach reduces the need to go to the field to do testing in

the early development. Another purpose of doing simulation is to evaluate the performance of a component or subsystem of a robot in completing some tasks. To achieve this purpose, a higher fidelity model may be needed for the simulation so that the performance metrics coming from simulation can be trusted with some level of confidence. Another purpose could be to explore the design space very early in the design process in which conceptual robot architectures are simulated.

The first class of simulators for robots are scientific computing platforms that have been developed for and are used for system-level modeling and simulation. These tools were mentioned above, which include Matlab/Simulink/SimScape and Modelica-based packages. These simulation environments are strong in providing a means for developing high fidelity models of the multibody structure of robots, including robot actuators. The Matlab environment also supports model-based software design so that algorithms can be simulated along with modeled robot plants.

The second class consists of robot simulators that have been developed specifically for simulation of robots. Examples of robot simulators include Gazebo, V-Rep, and WeBots, among others (Shamshiri et al. 2018). Gazebo, for example, is widely used for robot simulation as it easily interfaces with the Robot Operating Systems (ROS) enabling the development of software in the context of testing the developed code with a simulated robot and environment. The description of the robot and environment are entered through an XML file which describes the bodies. There are also simulators for autonomous vehicles (AV), typically focusing on the on-road vehicle sector including Carcraft by Waymo, Metamoto, and CARLA, which is an open-source layer over Unreal Engine 4, a commercially available graphics engine (Rosique et al. 2019).

There are trade-offs between these two approaches. Typically, the scientific computing platforms enable the development of higher-fidelity models that are inclusive of the multiple engineering systems encompassed in a mechatronics design. So, for example, not only the mechanical system can be modeled and simulated, but also the electrical or fluid power actuation and sensors. These platforms also enable dynamic system model representations, are readable, often in a graphical format, and thus offer some transparency between the modelers and users of the models. Modelica, for example, enables an object-oriented modeling approach so that physical components and subsystems can be abstracted into classes, from which specific objects are instantiated. In addition, this modeling language enables different views of the model including text, where all of the equations can be explicitly read, or graphical views, in which the connections between different components can be easily observed (Fritzson 2015; Tiller 2001). These platforms thus enable robust modeling tools coming out of the dynamic systems discipline. They also offer visualization of the robot or any other mechanism as it is moving during simulation. A limitation, depending on the tool, may be in the ability of the platform to include representations of the world with which the robot is operating and interacting. This is particularly important for agricultural and field robots because of their close coupling with the field space through both their actuation systems and their sensors. Some tools have explicit means for developing virtual world representations. Other may not, and then any interaction between the robot and the world would need to be

modeled explicitly within the system model. Yet another approach would be to coupling the dynamic simulation to graphical engines in which the virtual world and robot can be modeled, simulated, and visualized.

By contrast, robotic simulators mainly provide means for representing the mechanical systems associated with robots through a multibody representation consisting of sets of joints and bodies. Often the representation is in a textual file format, often in XML format, and may not offer a very readable model to the user. They generally lack capabilities for explicitly modeling other engineering systems, which may make representing high fidelity models of actuators difficult. Sensor models may be available, but their models are not typically very transparent, but are rather a plug-in that was written in some computer language. Because they are robot simulators, there are means to represent detailed robot worlds for the robots to operate in and typically provide very good computer graphics for simulation and visualization. In addition, interactions between the robot and the world are typically handled through collision detection and responses.

12.6 Robot Models

The backbone of a robot is a mechanical system, but it is more than a mechanical system; rather it is a mechatronic system, meaning it is a combination of mechanical systems along with other engineering systems, particularly electronic, electrical, and fluid systems, mostly in the form of actuators and sensors. Nevertheless, at the most fundamental level, the mechanical components and mechanisms of a robot must be modeled. The actuation system of the robot should also be modeled, but the level of fidelity depends on the goals of the simulation.

First, models of the mechanical components need to be developed. For models of the three-dimensional shapes of parts, solid models are typically used and can be developed using several available solid modeling computer aided drafting (CAD) software packages (e.g., Solidworks, Creo, Inventor, OnShape). *Solid models* are developed through sketches of geometrical shapes on a two-dimensional plane and then additively or subtractively growing those shapes in the third dimension to form a solid. Thus, geometric operations are used to form the 3D model of a part. Then other material properties associated with each part are added to the solid model such as material density, Young's modulus, and surface properties. With these properties in place, other engineering calculations and simulations can be carried out. Solid models are typically represented using one of two means: constructive solid geometry and boundary representation. Constructive solid geometry (CSG) representations consist of trees of geometric additive or subtractive primitive operations required to make the part (Foley et al. 1996). CSG is typically used as the format of the native files associated with solid modeling software. Boundary representation (B-rep) of solid models consists of descriptions of the edges or triangulated faces of parts. B-reps are often used in standard file formats that are used for file interchange between different software applications. For example, the STL (for

stereolithography) file format uses a triangulated surface B-rep of components and is used to transfer solid models to 3D printing software (Grimm 2004). Similarly, STEP (Standard for Product Data Exchange) files are using an ISO standardized file format for the exchange of 3D solid models. The standard is much broader than just the exchange of geometric information, but also includes interchange of a product data across the entire product life cycle (Pratt 2001; ISO 10303).

For simulation, component models must be imported into the simulation package. For example, STL files or STEP files can be imported and used to represent the shape of bodies of Simscape Multibody models in the Matlab environment. Also, Dymola will accept DXF files and ASCII STL files (Dymola 2018). DXF files are a file format developed by Autodesk for exchange of models between AutoCAD and other programs (Autodesk 2012).

In a CAD solid modeling software, assembly modeling is used to model how individual components are located relative to one another for a primary purpose of visualization, but also to manage all of the components that are assembled into a product. Physically, components are constrained by fasteners or may be able to move relative to one another. Assembly models contain a list of the components that are needed to build the assembly along with their relationship to the coordinate system in the assembly space or their relationship with other parts which are determined with mating conditions. For example, two flat surfaces of two different components can be mated to be coincident to each other. Robot simulation, however, requires multibody dynamic models which can come from assembly models.

Multibody dynamic (MBD) models are similar to the assembly models, but are focused on representing the interconnection of rigid or flexible components (or bodies) to enable simulation of their motion under the influence of external forces. Thus the main primitives for representing multibody systems are bodies and joints. Bodies are just the models of rigid components with dimensional properties, and can come from the solid models. Joints define how bodies are connected together and how their motion is constrained relative to one another. For example, a prismatic joint constrains motion to only translational motion along one translational axis. Similarly, a revolute joint constrains motion to only rotational motion about one rotational axis. Through the system of bodies and joints, the degrees of freedom for the system are determined. A degree of freedom is a possible independent movement of a body or system of bodies. In 3D space, a body will have six degrees of freedom, three translational motions, and three rotational motions. Joints constrain motion reducing the number of degrees of freedom for a system depending on the constraints associated with the joint (Shabana 2020).

In the Matlab platform, MBD models can be developed in Simscape Multibody; the representation of the bodies and joints can be represented graphically with references to STEP files exported from a CAD software. It is also possible to export an assembly model from a CAD software through the Simscape Multibody Link Plug-in, which will then take the assembly model and make it into an MBD model. In Modelica, there is an MBD library in the Modelica Standard Library from which MBD robot models can be built.

Robot simulators represent the multibodies that make up a robot using a format that is specific to the simulator. For example, in Gazebo, the robot elements are described in an XML file in the Universal Robotic Description Format (URDF). URDF files are used in ROS to specify the component and joint configurations of the mechanisms that make up a robot. Gazebo actually uses SDF (Simulation Description Format) files, which specify everything needed for the robot simulation including both the robot and the world. There is a way to use, however, a single URDF file for both ROS and Gazebo by using <gazebo> tags for parameters that are used by Gazebo, but are ignored when the file is used by ROS (Gazebo 2019; Bipin 2018).

In URDF, there are two main elements that define the multibody system of the robot: links and joints. Links are the solid elements that can be defined as simple geometric shapes or through the solid models. The solid models can be represented using STL files or COLLADA files. The COLLADA file format was designed for exchanging 3D graphics between graphics software; as such they can provide better surface feature representations than STL files. When defining the links, both visual and collision models are defined. The visual model will contain more details, so the fidelity of the visual appearance is maintained; the collision model will be a lower fidelity geometric model to reduce the computational demand of the collision detection algorithm (described below; Bipin 2018). Joints are the definition of the connection between the links and specify the possible relative motion between the links. Gazebo joint types include: fixed, continuous, revolute, prismatic, planar, and floating.

As an example of another robot simulator, in the V-REP robot simulator, robots can be defined using URDF files, or STL files can be imported and placed in the V-REP scene through the use of a graphical user interface (Shamshiri et al. 2018).

12.7 Virtual World Models

Since the robot will be operating in a real-world environment and will be acting upon that environment, a model of the world will be needed. Like the robot model, virtual world models also consist of solid models of the objects in the world, as well as their relationship to one another. In addition, for visibly realistic representations of the objects in the world, their surface properties including textures must be represented in the model. Then in the simulation, the 2D scenes will be rendered based on the models provided to the simulator. Matlab accomplishes this using the Simscape 3D Animation package which uses virtual world models represented as Virtual Reality Modeling Language (VRML) or eXtensible 3D (X3D) files. The virtual world can be imported from URDF or Simulation Description Formal (SDF) into a VRML file and edited with a 3D editor. In Gazebo, the virtual world can be edited within the Gazebo GUI editor and will be saved as a file with a ".world" extension which is in the SDF format. The SDF format was developed to provide a

complete description of everything needed to run a simulation in Gazebo across robot to virtual world levels (Gazebo 2019).

12.8 Collision Detection

An important part of the robot's interaction with the virtual world comes from the simulator's capability of determining when a robot component geometrically intersects or "collides" with an object in the world. Once a collision is detected, then the response to that collision must be determined. In Matlab/Simulink or Modelica, if the physics of the system are completely modeled in the dynamic model, then collision detection is handled in the system equations which must explicitly track interactions between objects. A bouncing ball can be used as a prototypical example problem to illustrate hybrid dynamic systems which have both continuous dynamics and discrete states. For this example, a gravitation force operates on the ball leading it to accelerate in the downward direction. The position of the ball above a horizontal surface is monitored, and when the ball comes into contact with the surface, the continuous simulation is stopped and then reinitialized with the velocity of the ball now directed upward with a magnitude that was the same as the downward velocity scaled by the coefficient of restitution (Fig. 12.5).

Another approach is to push the collision detection operation into the model of virtual world. One virtual world representation format is X3D (Extensible Three Dimensional; www.web3D.org), which is an ISO standard for defining three-dimensional computer graphics for virtual worlds and robots operating in those worlds. A feature of X3D is the capability for detecting collisions between objects

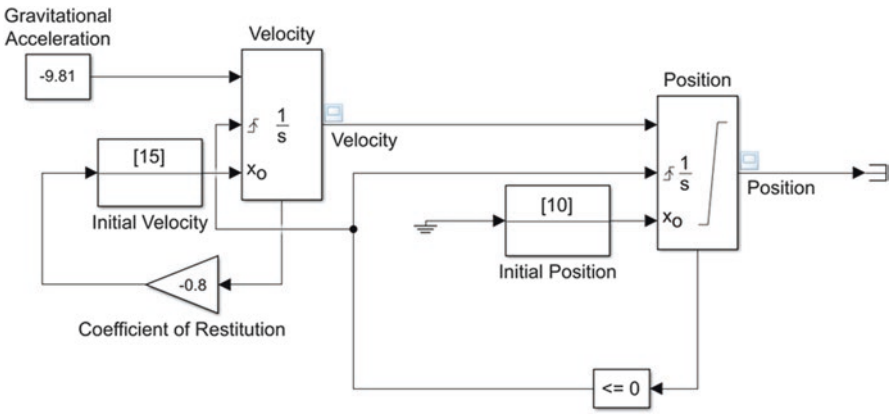


Fig. 12.5 Simulink model of a bouncing ball with two states for velocity and position. Collisions between the ball and surface are detected when the position is less than or equal to 0, and when the system is restarted with an initial upward velocity which is the product of the velocity upon collision and the coefficient of restitution. (Source: Mathworks 2020a)

with picking sensor components placed in the X3D world model. The term "picking" is a 3D graphics term that refers to testing for object collision. X3D sensors are used in the Simulink 3D animation tool, in which each X3D sensor indicates through a Simulink VR Source block when collisions occur. Responses to those collisions are determined through response models represented as Simulink blocks, which then provide input to a VR Sink block and modify the objects in the virtual world. This type of interaction between the virtual world and the robot enables transparency and flexibility to the people developing the models and software to control the robot (Fig. 12.6).

In Gazebo, like other robot simulators, the links associated with the robot and those associated with the virtual world are provided by a physics engine such as Open Dynamics Engine (ODE), Bullet, Simbody, or DART (Shamshiri et al. 2018; Mondesire et al. 2016; Erez et al. 2015; Lee et al. 2018; Sherman et al. 2011). The physics engine determines the motion of the multibody system, which makes up the robot as well as internally detects collisions and determines the response to those collisions.

The general collision detection and response process determines at each time step which bodies are in contact with each other. For each contact point, a temporary joint is established, and based on the stiffness, damping, and friction models between the two bodies, the resulting joint forces or impulses are calculated. Then based on these impulses, bodies' velocities and positions are updated (Fig. 12.7), the set of temporary joints is removed, and the process repeats (ODE 2019; Coumans 2015; Moore and Wilhelms 1988; Kavan 2003). Robot simulators will all take a similar approach to collision detection and response using the algorithms associated with their physics engine. It is important to note that this approach to collision detection largely emerged from the computer animation and graphics community

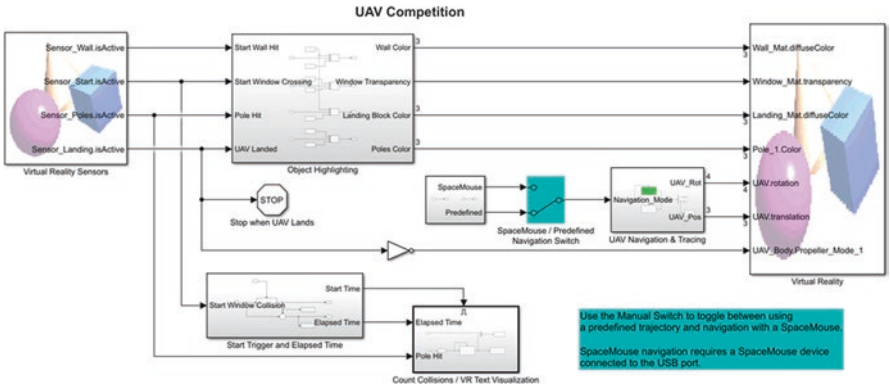


Fig. 12.6 Example of Simulink interface with virtual world in which sensors defined in the X3D format indicate when collisions occur. Collision signals come from the VR Source block labeled Virtual Reality Sensors in the upper-left corner are handled by the Object Handling block in the center, which provides signals to the VR Sink block on the right which implements changes in appearance to objects in the virtual world. (Source: Mathworks 2020b)

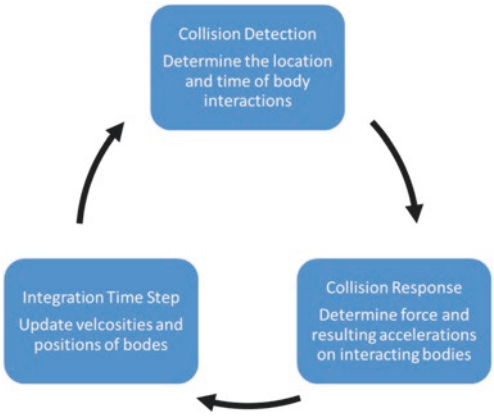


Fig. 12.7 Simulation loop for rigid body motions with collision detection and response. (Inspired by Coumans 2015)

with a common application of simulating the motion of objects in video games. Thus, while physics principles are applied, the goal has not been to maintain high physical fidelity, rather high visual fidelity to the gamer. The simulation has to appear visually correct, not necessarily be physically correct.

12.9 Sensor Models

Agricultural and field robotics depend on the use of sensors for machine awareness through perception of its location and interaction with the environment. Because the information from those sensors will ultimately be used to control the robot, simulation of the sensors is important for the overall simulation systems. In Matlab, some simple sensors are available in using Simulink 3D animation. For example, an X3D picking sensor can be placed in the virtual world which will provide an indication of colliding geometry. It is also possible to develop Simulink models with camera sensors acquiring images from photo-realistic environments like those produced with Unreal engine (Matlab 2019b). In Gazebo, sensor models can be explicitly defined in the world file using SDF (Gazebo 2020). The sensor is mated to the robot in the URDF file that also defines the robot's link geometry and inertia properties. The sensor's transduction of virtual world signals and signal processing are modeled in sensor model code and parameter definitions, which is called through a plug-in in the world and robot model. Sensor model plug-ins can be customized using the Gazebo Application Programming Interface (API), in which motion, force, and vision sensors can be modeled.

12.10 Integration with the Software Architecture

When operating with simulation, a major purpose is to test software that has been written for the application. In the Matlab Simulink environment, algorithms can be written in a variety of languages or graphically connecting blocks, many of which preexist from Matlab toolboxes. Additionally, code can be developed to run on the embedded hardware which will be used for the robot.

Since Gazebo has been historically connected to the Robot Operating System (ROS), the middleware already exists for implementing the robot code, which is then hooked into Gazebo. ROS is an open-source middleware that provides a framework for connecting many different software components that are needed to develop complex robotic applications. ROS manages a graph-like peer-to-peer network of processes which can be distributed across multiple machines. The processes are loosely coupled using the ROS communication protocols. Different communication styles have been implemented, including synchronous request/response communication over "service," asynchronous data streaming over "topics," and shared parameter storage named "parameter server" (ROS 2019).

When using Gazebo with ROS, Gazebo loads a series of ROS plug-ins, which turns Gazebo into a ROS node. The plug-ins provide message and service publishers for interfacing Gazebo with ROS. The ROS plug-ins implement functions to simulate force, motion, and perception sensors and actuate motors and dynamically reconfigure parameters in the simulation using the Gazebo API. Once the rest of the ROS nodes for robot control were developed and tuned in the simulation, they can be migrated into the real-world robot to control directly or with minor modification.

12.11 Comparison Between Robotic Simulation Techniques

As the development of robotic systems accelerates, different platforms for simulating robotic systems are being developed and tend to come from different disciplinary spaces. One area is the dynamics systems modeling and control space. The focus here is on developing models of physical systems that can provide high fidelity physics-based representations of the engineered systems. These platforms provide opportunity to model systems from different domains, so for robots this could include detailed models of actuators, mechanisms, and power trains, among others. There has been substantial activity in this area to develop standards for exchanging models between simulation platforms and co-simulation through the functional mock-up interface (FMI 2014). These developments have largely come out of large European research projects with participants largely from the dynamic systems community. Therefore, within this space, Modelica can be used to develop high fidelity robot models that could be exported to be part of a robot simulator, which

provided the additional needed capability of world modeling and visualization, sensor modeling, and linkage to the software architecture.

Similarly the scientific computing platforms produced by Mathworks have also largely come out of the controls and dynamic systems community. Mathworks' Matlab and Simulink platforms provide commercial solutions to a wide range of scientific computing applications, and their expertise encompasses a wide range of technical specialties. Mathworks provide tools that enable:

1. Physical modeling of robots to a high degree of fidelity across multisystems
2. Implementation of control system algorithms and porting algorithms to code for target electronic control hardware
3. Representing and visualizing visual worlds with collision detection and some limited sensing ability

Robotic simulators, on the other hand, have largely emerged from the robotics community and leverage physics engines and graphics engines, which emerged from the computer graphics and animation community. Therefore, these platforms tend to have another set of strengths as compared to the other ones discussed above. For this chapter, the focus has been on Gazebo, which has the following set of strengths:

1. Ability to define fundamental robot multibody properties.
2. Robust definition of virtual world.
3. Leveraging physics engines for multibody dynamics, collision detection, and response.
4. Connection to ROS, robotics middleware where algorithms can be developed and simulated with HIL.
5. Several available sensor models to facilitate interaction with the virtual world. These models have limited documentation.

Simulations that require a high level of physical fidelity will tend to be better suited by the scientific computing platforms, which also offer means for transparently representing models of components making up a robot. On the other hand, if interest is more in high fidelity modeling of the world that the robot will be operating in, the robot simulators may be a better choice. These types of tradeoffs with the different simulation platform options need to be considered in the decision process of selecting a simulation system for a specific application at hand.

12.12 Modeling Agricultural Field Elements

Modeling agricultural and field robots requires representing the system starting with the physical components that make up the robot, their connectivity to each other, and their dynamic behavior including all of the control elements along with sensors and actuators. These machine models are generally straightforwardly modeled with existing modeling elements and established methods. However, the modeling of

robot interactions with the environment where it will operate and carry out agricultural field operations is an area with more open research questions. Three important interactions are:

1. Between the machine and soil through traction systems (tire or tracks) and ground-engaging tools through which the robot will be acting on the environment
2. Between sensors and objects (usually biological) in the environment used to perceive the location of the robot both in a global and local level as well as obstacles to avoid or product to be manipulated (e.g., harvesting fruit, or grain and picking rocks)
3. Between operators and machines when operators play a role in robot operation, which depends on the level of autonomy and supervision of the robotic system

Discussion on some aspects of machine-human interactions and interfaces can be found in Chap. 15 and the same with machine-canopy interactions can be found in Chap. 16.

12.12.1 Modeling and Simulating Soil-Machine Interactions

For agricultural robotics, the soil-machine interface is critical to evaluate tractive performance of the machine to autonomously accomplish the required agricultural/field tasks. Under external loading, the soil-machine systems can be broadly categorized as (1) load bearing processes where soil supports load bearing capacity and develops tractive forces or (2) load loosening processes where soil exhibits deformation in the form of compression, shear, excessive displacement, or their combinations (e.g., tillage or blade cutting applications). In either of the load bearing or load loosening processes, soil reaction to applied external loads can cause soil compaction (change in volume); soil distortion (shear) combined with compaction; soil distortion at constant volume (plastic flow); expansion (dilation) that could occur with post-shear failure and tensile failure (Gill and Vanden Berg 1968; Koolen and Kuipers 1983). In a pre-failure phase, soil deformation often comprises elastic strain (recoverable) and plastic strain (irrecoverable) proportions with the elastic strains accounting for only small fractions of the total soil strain (Shen and Kushwaha 1998).

Agricultural and field robots must be able to propel themselves across a field surface in an off-road environment; in this case, soil-to-tire/wheel interactions should be modeled to predict the tractive forces in the longitudinal direction of the vehicle as well as the motion resistance that occurs as a result of soil and tire deformations. In addition, the relationship between side-slip and lateral forces is important in modeling the steering function of the robot.

Soil responds to normal and tangential forces applied from tractive devices as an elastic medium, or with plasticity in a combination of elasto-plastic deformations (Koolen and Kuipers 1983). The assumption of soil responding as a perfectly soil shear at constant volume (plastic flow, without dilation) has been used successfully to estimate the maximum traction force for the development and evaluation of

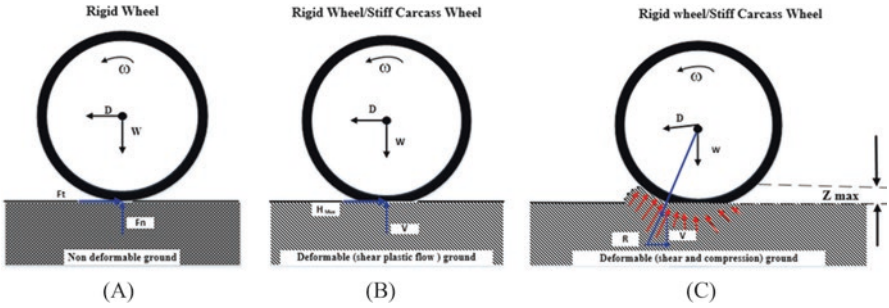


Fig. 12.8 Modeling soil-tire interaction can take on three levels of fidelity from left to right: rigid wheel on non-deformable soil (**a**), rigid wheel on deformable soil (Janosi's shear stress to soil deformation (shear; plastic flow assumption)) (**b**); and rigid wheel or stiff carcass tires on deformable soil (compression (sinkage) and shear) assumptions based on Bekker-Janosi approach (**c**). Either of these can be implemented in a rigid body simulation loop using collision detection of contact elements and explicit solutions of the soil-tire forces using finite difference technique. Examples of these approaches are implemented in the NATO-Mobility Model (Wong 2010) and Wismer-Luth-Brixius-Zoz (Upadhyaya 2009) wheel numerical tire performance models

off-road vehicles (Wong 2010). Assumptions of rigid wheels applying loads on soil that behaves as coulomb-friction (Fig. 12.8a) or soil responding in shear distortion in the form of plastic flow (Fig. 12.8b) can be integrated into robotic simulators or real-time vehicle performance models (Wong 2010).

For simulating robot mobility performance on off-road ground conditions in real time, the tractive element to ground surface interaction is modeled primarily as a dynamic interaction between the robot tractive elements and the soil surface. At each time step, the contact collision detection algorithm determines if the traction and soil elements are in surface contact with each other and makes a note of it. Then the collision response algorithm uses the relative position of each contacting elements and determines the resulting forces based on the material stiffness and damping properties of the two interacting elements and iteratively computes resulting forces. For low vehicle speeds, strain-dependent damping properties can be assumed to be negligible (Wong 2010). The friction behavior at the interface can be modeled as coulomb friction where the friction force can be calculated as the product of the normal force and a friction coefficient between the tractive element and the soil surface. This rigid soil surface assumption is typical for robotic simulators using rigid body collision detection and response algorithms. However, on cohesive-frictional soils where soil deformation exhibits mainly in the form of perfectly shear-distorted plastic flow, the soil-to-traction element interactions can be assumed to obey Mohr-Coulomb theory that governs the shear (tangential) force calculations at the point or surface where they are interacting. Janosi (1962) assumed that traction forces can be estimated from the tangential shear stresses developed at the tire to soil interface. Using Janosi's approach, the tractive element (rigid wheel or stiff carcass tires) are assumed non-deformable.

According to Janosi (1962), the soil shear stress can be calculated using the following equation:

$$S = \left(c + p\tan\left(\varphi\right)\right)\left(1 - e^{-\frac{j}{k}}\right) \tag{12.1}$$

where

S = shear stress
c = soil cohesion
p = pressure normal to the shear plane (normal stress)
φ = soil-to-soil internal friction angle
j = shear deformation
k = modulus of a soil shear stress-strain curve

The soil parameters can be estimated from laboratory direct shear tests at different normal stress levels. The magnitude of normal stresses applied to obtain the soil parameters should consider the range of vertical static or dynamic normal loads and contact area associated with the robot application. The values of j and k must be in the same units. At the maximum shear strain, Janosi's equation (Eq. 12.1) expresses the Mohr-Coulomb failure equation (S = c + p tan (φ)).

For analysis of the soil-robot tractive systems where the soil deformation consists of compression and shear (Fig. 12.8c), tractive performance of off-road vehicles or robots will require force-equilibrium analysis of motion resistance in addition to soil shear-induced horizontal gross tractive force (Wong 2010). Generally, robots will not have draw-bar pull, so the tractive effort is used to overcome both motion and terrain grade resistances. Analysis of robot tractive performance on soft and compressible soil conditions (e.g., wet soil) should account for the normal pressure and sinkage relationship, and shear stress and strain relationship according to Bekker-Janosi's approaches (Wong 2010). Further details on vehicle track or rolling wheel performance analysis using the semi-empirical approaches by Bekker's approach for modeling the plate pressure-sinkage and Janosi's shear stress strain relationships are available in Wong (2010). For tractive force prediction under slip (velocity reduction), Wong (2010) and Upadhyaya (2009) explained semi-empirical predictive equations with slippage parameter added to the tractive force calculations. In agricultural and field robotic applications without draw-bar pull requirements, free wheel rolling without slip can be assumed at the soil-tractive element interaction. The Wismer-Luth-Brixius-Zoz empirical approach (Upadhyaya 2009) by defining wheel numeric parameters to the soil cone index in the 0–150 mm soil layer has also been applied for tractive forces performance of large agricultural tractor tires. Further investigation may be required to investigate the robustness of Wismer-Luth-Brixius-Zoz approach for small robots because the numerical model parameters were developed for large agricultural tires and loads from high horsepower tractors. Soil-to-tire/wheel interaction modules based on the above soil deformation theories or semi-empirical approaches can be integrated into

rigid-body simulation modeling (Fig. 12.9) as opposed to utilizing simple coulomb friction analysis at the tractive elements and hard surface planes.

Soil interactions with deformable, pneumatic tires, however, are known to be characterized based on the knowledge that tires exhibit deflection due to the flexible nature of the carcass material inflated with a compressible fluid, air, and knowledge that soils are deformable with elasto-plastic material behaviors. In such tire-soil systems, contact between the two elements is the intimate continuous contact of the tire lugs engaging in normal load, shear, and penetration with the deformable soil material. In applications where both tire and soil deform, physics-based, high fidelity modeling of the tire-soil force systems is essential (Fig. 12.10). High-fidelity

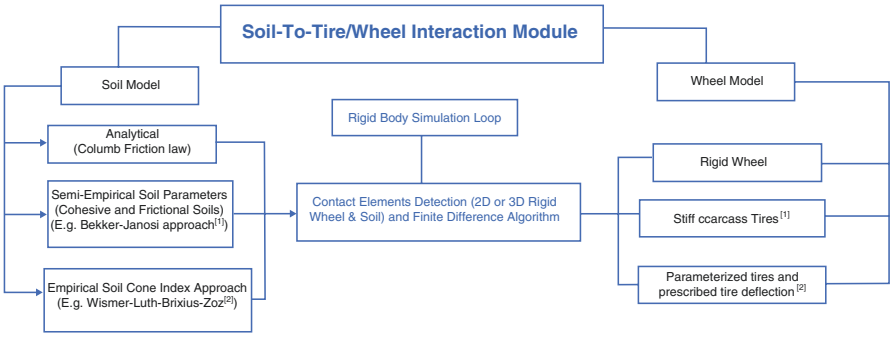


Fig. 12.9 Modeling the interaction between the soil and tire can have differing levels of fidelity for soil (left) as well as those for the wheel or stiff carcass tire model (right). Details about soil-to-tire/wheel using semi-empirical soil models according to Bekker-Janosi approach[1] are available in Wong (2010). Similarly Upadhyaya (2009) explained the application of empirical models that uses soil cone index for parameterized soil-to-wheel performance analysis according to the Wismer-Luth-Brixius-Zoz approach[2]

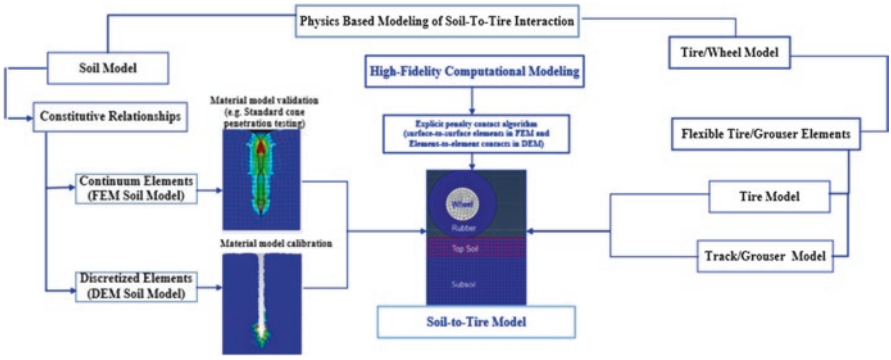


Fig. 12.10 Physics-based modeling of soil-to-tire interaction for tire/grouser elements on deformable soil in the continuum elements (e.g., finite element method (FEM)) and discontinuum elements (discrete element method (DEM)). Soil material models often need to be validated for FEM modeling (e.g., CAM-Clay and Drucker-Prager material models) and for DEM modeling (e.g., Hertz-Mindlin (HM) contact model) of simple soil-to-tool interaction, for example, standard soil cone penetration testing

modeling of soil responses to tire loading requires stress and strain constitutive soil models and computational methods such as finite element method (FEM) or discrete element method (DEM).

In finite element method (FEM) or discrete element method (DEM) techniques, soil models are entered as stress-strain constitutive relationships or force-overlap displacement contact laws, respectively. With recent advances in computer technology and explicit computational techniques, the FEM and DEM approaches can be integrated into the performance analysis of tire-soil interactions for robotic applications. Understanding the effect of tire inflation pressure on tractive forces and motion resistance, for example, requires transient performance of tire-soil interaction using these high fidelity computational modeling techniques. When evaluating tire or track performance on various soil conditions, high fidelity analysis of soil-tire interactions using soil models applicable for field soil conditions are necessary.

In mesh-based explicit FEM analysis, deformable tires can be modeled as elasticity, hyperelasticity, and viscoelasticity material, and soils can also be assumed to behave as elasto-plastic material (Abaqus 2013). The interaction produces tractive forces in the longitudinal direction of the vehicle; soil develops motion resistance acting against the direction of the tractive effort as a result of normal and shear stress distribution on the soil as well as lateral forces that are related to the steering of the vehicle. Deformation of tires, the stress distribution at the tire-soil contact surface, and the soil strain (deformation) can also be solved obeying the surface-to-surface friction behavior dependent shear stress calculations at the tire and soil contacting elements, and the material (tire and soil) constitutive stress-strain relationships. The DEM technique, being a meshless and explicit numerical analysis, can theoretically simulate the interaction of soil-to-soil and soil-to-geometry (tire or track elements). In either FEM or DEM approaches, understanding the soil material models and methods to determine the soil material model parameters are very important. The sections below attempt to explain the typical material models that can be relevant for robot interaction with agricultural soils.

12.12.1.1 FEM Soil Modeling

Soil behaviors under loading are generally considered having nonlinear elastic-plastic properties with large strain deformation (Upadhyaya et al. 2002). Formulation of elasto-plastic theories requires the definition of elastic stress and strain relations and yields criteria that mathematically define the stress conditions under which plastic deformation, stress hardening, or strain softening behaviors occur (Chen and Mizuno 1990). Yielding in soils defines the onset of plasticity or the point at which elastic behavior ceases. The elastic portion of stress and strain can be defined using Young' modulus, Poisson's ratio, shear modulus, and bulk modulus of elasticity. Numerous yield criteria have been proposed for the plasticity constitutive soil models and have been integrated into FEM commercial codes, e.g., (Abaqus 2013). Mohr-Coulomb failure, Drucker-Prager's, and Cam-Clay are among the widely used yield criteria in soil mechanics (Chen and Mizuno 1990; Wood 1990). Soil

models with Drucker–Prager cap yield criterion have been successfully applied to predict traction forces developed from rigid-wheel interaction on sandy soils (Varghese et al. 2013). Soil material parameters to define the soil stress and strain relationships in soil models can be estimated using soil triaxial and direct shear tests or validated from simulation of typical soil-to-tool interaction (Tekeste et al. 2009; Fig. 12.10) prior to use for soil-to-wheel simulation using FEM techniques (Chiroux et al. 2005). The theory of constitutive soil models and their application for FEM analysis are explained in detail in the Abaqus theory manual (Abaqus 2013).

12.12.1.2 DEM Soil Modeling

DEM models consist of particles with micro-mechanics constitutive relationships between forces and overlap displacement from the colliding particle-to-particle and particle-to-object geometry. The constitutive relationships between force and overlapping displacement are formulated using spring stiffness-damper originally on the bases of Hertzian and Mindlin theories between two elastic-collision of elements (Cundall and Strack 1979). Recently, advanced models such as cohesive and elasto-plastic contact models have been implemented in commercial DEM codes (EDEM 2011). The DEM contact forces (normal and tangential) as a function of overlap between two elements in contact can be calculated on the basis of Hertz-Mindlin (HM) contact theory (Tsuji et al. 1992; EDEM 2011). The tangential contact force is limited by Coulomb friction law and depends on the coefficient of static friction. Both normal and tangential damping forces as a function of normal and tangential components of the relative velocity related by the damping coefficients are calculated according to Tsuji et al. (1992). The rolling friction contact model (Eq. 12.1) in EDEM, a commercial DEM software, as described in Sakaguchi et al. (1993), is used to calculate torque from rolling resistance at the contacting surface from normal contact forces.

$$\tau_{\mathrm{i}} = \mu_{\mathrm{r}} F_{\mathrm{n}} R_{\mathrm{i}} \omega_{\mathrm{I}} \tag{12.2}$$

where:

τ_{i} = Torque applied to the contacting surfaces from element (sphere) i
μ_{r} = Coefficient of rolling friction
F_{n} = Hertzian normal contact force
R_{i} = The distance of the contact point from the element (sphere) i center of mass
ω_{i} = The unit angular velocity vector of the object at the contact point

The normal Hertzian contact force, F_{n}, as a function of normal overlap, δ_{n}, is calculated according to:

$$Fn = \frac{4}{3} E^{*} \sqrt{R^{*}} \delta_{n}^{3/2} \tag{12.3}$$

where E^* is the equivalent Young's Modulus and R^* is the equivalent radius.

The equivalent Young's Modulus is defined in terms of individual shear moduli (E_i and E_j) of contacting sphere i and sphere j, respectively, and Poisson's ratio (ν_i and ν_j) of each sphere using the relationship:

$$\frac{1}{E^*} = \frac{\left(1-\upsilon_i^2\right)}{E_i} + \frac{\left(1-\upsilon_i^2\right)}{E_i} \tag{12.4}$$

The equivalent radius is defined in terms of the individual sphere radii (R_i and R_j) using:

$$\frac{1}{R^*} = \frac{1}{R_i} + \frac{1}{R_j} \tag{12.5}$$

The HM contact and damping forces depend on the material properties of Poisson's ratio, solid density (particle density), shear modulus; and interaction model parameters of coefficient of restitution, coefficient of static friction, and coefficient of rolling friction (EDEM 2011).

The contact forces between two particles and the resultant motion of each particle after each contact collision are calculated at an explicit time step based on the equations of motion (Cundall and Strack 1979). Simulation-based DEM analysis workflow consists of generating a CAD geometry surface mesh, numerical approximating natural soil particle into DEM primitive shapes (e.g., spheres, clumped-sphere or other complex ellipsoidal shapes), assigning material model properties, setting the explicit DEM solver setting, and postprocessing.

Unlike FEM soil model development where the stress-strain constitutive relationship are length scale invariant, DEM soil material properties are expressed as stiffness (*F/L*, where *F* is force quantity and *L* is length quantity) that are not scale invariant. In addition, measurement of individual granular particle-to-particle or particle-to-geometry micro-mechanics model parameters is practically impossible for soil materials with soil particle sizes less than 2 mm.

For analysts to simulate soil-machine systems (e.g., tire-soil interaction for high fidelity simulation of robots), it is essential to evaluate if the continuum basis (using FEM) or discontinuum basis (using DEM) techniques fit to the idealized soil-machine system. In applications where soil deformation due to external loading from machine components comprises a combination of compression and shear, or dilation in discontinuum forms, FEM analysis are limited in handling excessively large strains and distortion of elements. Examples of such applications may include design of track grouser shapes, tire rolling-induced prediction of soil bulldozing motion resistance, and accurate prediction of tire-soil ground contact stress distribution. The DEM technique is, thus, a preferred numerical technique to predict such complete soil behavior and could also be coupled with multibody dynamic (MBD) models for wheel dynamic motion. DEM soil models require calibration technique to determine the material properties for the micro-mechanics contact laws. For

some tire-soil applications, for example, tire-soil interaction modeling for predicting gross traction and motion resistance on non-dilatant soils, explicit FEM analysis can successfully predict tire performance (Varghese et al. 2013). For robot interaction with soil, the framework example shown in Fig. 12.10 can be considered for physics-based modeling of soil-to-tire interactions. Explanation about tire material models including rubber or other carcass elements are beyond the scope of this chapter.

12.12.1.3 Simulation-Based Comparison of Soil Modeling with DEM and FEM

Determining soil parameters for DEM models continues to be a challenge and is an active research area in soil-machine academic communities (Upadhyaya 2009). Similarly for FEM soil models, direct determination of all the soil model parameters from standard soil testing do not necessarily provide accurate prediction of the soil-to-tire interactions (Chiroux et al. 2005). Experience in developing soil models for agricultural soils have shown that validating DEM or FEM soil material models to approximate standard in situ tests such as using soil cone penetrometer are better alternatives. As an exercise to compare computational demands in typical FEM and DEM soil-machine modeling frameworks, simulation of soil cone penetrometer insertion into soil using FEM and DEM approaches is presented here.

Soil was modeled as 4-node brick element (Abaqus 2013) for FEM analysis and as a single sphere for DEM analysis of soil cone penetration testing. Solver settings and the explicit calculations for the two commercial codes, Abaqus for FEM Drucker-Prager simulation (Tekeste et al. 2009) and EDEM for DEM soil constitutive simulations, were set to predict soil reaction forces on the standard conical tip of a cone penetrometer (ASABE 2018). The cone penetrometer was modeled with a 30° conical tip and a cone base diameter of 1.28 cm. A rigid cone body using the RAX2 2-node linear axisymmetric element was used in the FEM and a CAD solid model in the DEM. The cone body was driven vertically downward into the soil body at 30 mm/sec. Cone penetration through a soil discontinuum model in DEM and through adaptive meshed deformed soil brick elements in FEM show how the cone interacts with the soil media in an expected manner (Fig. 12.11). The FEM results showed significant stretching of the soil material and a concavity that would not be expected at the top; whereas the DEM results seemed to be consistent with observations of soil deformation as a discontinuous granular media.

Analysis of the computational effort required for the adaptively changing explicit time step solver used for the FEM in Abaqus and the fixed time step solver for the DEM in EDEM showed that the DEM simulation had a computational cost of 21,000 times the CPU/simulation time required to solve the FEM analysis (Table 12.1). The benefit of the DEM was that it captured the typical soil deformation from indentation by the conical tip. The FEM formulation, however, took numerous iterative exercises to avoid instability from geometry nonlinearity and excessive soil element distortion. Element distortion has not been an issue in the

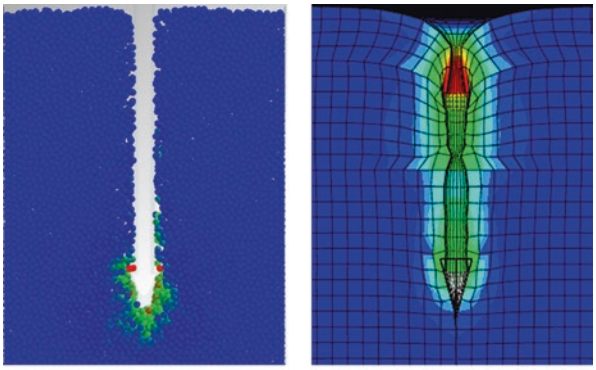


Fig. 12.11 Soil cone penetration simulation using a 3-mm diameter single sphere DEM soil model in EDEM (**a**) and using a 5-mm RAX2–2-node FEM element with adaptive technique in Abaqus (**b**)

Table 12.1 DEM and FEM simulation parameters and performance metrics from the soil cone penetration simulations

Method	Particle size (mm)	Number of particles	Time step (s)	Simulation time (s)	Estimated total CPU (hr)	CPU/ simulation time (hr/s)
DEM (Hertz-Mendlin soil model)	3 (single sphere)	86,157	1e-06	5	210 (8.75 days)	42
	5 (single sphere)	17,307	1e-06	5	167 (6.96 days)	33.4
FEM (Drucker Prager soil model)	5 (RAX2–2-node element)	1024 (RAX2–2-node elements)	4.23–05	5	0.01 (50.3 s)	0.002

DEM simulation, rather the method to determine the DEM soil material properties is very challenging.

After performing the comparative analysis of soil cone penetration in DEM and FEM frameworks, FEM analysis was done for large wheel soil analysis to predict tractive forces in a computationally efficient manner. The wheel geometry (overall diameter and rim diameter) were obtained from Raper et al. (1995) and Varghese et al. (2013). A stiff external wheel element was added in addition to the wheel rim to demonstrate the rigid wheel and soil deformation. A layered soil column cone penetration test on Norfolk sandy loam soil was used to develop soil model properties for the Drucker-Prager Cap soil model with hardening at two soil density conditions (1.32 Mg/m^3 loose and 1.64 Mg/m^3 dense states) (Tekeste et al. 2009). Abaqus-Explicit analysis of soil-to-wheel interaction (Fig. 12.12) was performed

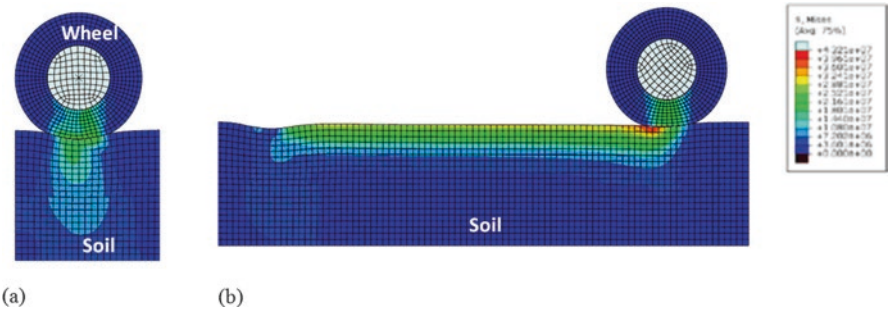


Fig. 12.12 FEM rigid wheel-soil formation using 2D plane strain analysis in Abaqus. The soil was modeled using Drucker-Prager with hardening material model. 2D rigid wheel similar to the model in Varghese et al. (2013) was used for this demonstration under gravity loading (**a**) and free-wheel rolling examples (**b**)

consisting of two steps: (1) gravity loading (5.6 kN) of the wheel on soil (A) after Raper et al. (1995) and (2) free rolling of the wheel (B) at 0.15 m/s after Varghese et al. (2013) on Norfolk sandy loam soil (72% sand, 17% silt and 11% clay). A surface-to-surface contact formulation with penalty method (Abaqus 2013) was used for friction behavior between the wheel and the soil. The results showed soil deformation under gravity loading underneath the wheel with the highest stress (Von-Misses) at the top soil surface and permanent (plastic) deformed soil elements (see left edge of the rolled wheel domain; Fig. 12.12b).

Modeling the interaction between soil and traction elements is likely to be desired for realistic simulation applications in agricultural and field robotics with deformable tires. In the future, however, more interest will develop around the area of soil engaging tools as more autonomous capability will open the possibility of precision tillage and cultivation among other applications. Robotic mechanical weeding is a good example of how automation technologies will open up opportunities for weed control using precision cultivation (Steward et al. 2019). In these cases, soil engaging tools will result in draft forces that will act on the robot body in addition to the traction forces that must be characterized to develop a physically faithful simulation system for vehicle and tool attachment. There are analytical models for predicting draft force soil engaging tools such as tines (Godwin and Spoor 1977; Wheeler and Godwin 1996). These models are dependent on soil properties and tool geometry, and there are opportunities for more advanced modeling in particular to the DEM approaches described above.

12.12.2 Modeling and Simulating Perception Systems

The perception systems of agricultural and field robots use sensors that sense the environment around the robot. Typically, the location of the robot is important, which can be obtained using a global navigation satellite system (GNSS) sensor

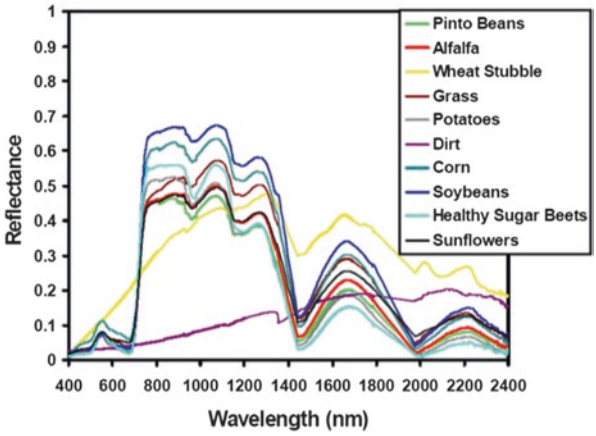


Fig. 12.13 Spectral signatures of different plants and soil. (Kyllo 2003)

(e.g., GPS, see Chap. 3, Sect. 3.3 for more details) or with some other local sensors such as cameras measuring reflected light from the crop and background which is used to find row structure (Fig. 12.13; Xue and Xu 2010). In addition, optical depth sensors are used to determine the distance from the vehicle to crop plants or the soil surface. Typical depth sensors include stereo vision, time-of-flight (TOF) cameras, and LIDAR (Mousazadeh 2013). Detailed discussion on various types of 3D sensing techniques and sensors can be found in Chap. 3. While there are many different types of sensors, typically each of them will have some perception algorithms converting the signals from the sensors into meaningful information of use to the robot. For example, images or video from cameras can be converted into robot pose information based on crop row structure (Higuti et al. 2018; Imperoli et al. 2018; Pinto et al. 2000). To test the perception systems and their interaction with robot control and behavior systems, simulation of the sensors interacting with the physical world is necessary.

In the context of an agricultural field or other off-road surfaces, many robotics sensors are interacting with plants and soil of the field scene. Plants offer a more complex scene than what many robot sensor simulators have been designed to use. Complexity can be found in the following aspects of crop field scenes:

- Changing lighting conditions. Sunlight varies with sky conditions such that when the skies are clear, the sunlight is direct and intense. When the skies are cloudy, then the light is diffuse and less intense. These conditions can change rapidly with weather condition.
- The interaction of the light with the leaf will result in different proportions of the light being reflected with specularity or diffusivity.
- Reflection will be from multiple leaves and stems making up the plant canopies.
- Light reflectance is a function of the wavelength resulting in a characteristic reflectance spectrum across the visible and near-infrared regions (Fig. 12.12). Though following a typical characteristic reflective curve, this spectrum is

dependent on plant species and conditions and is thus temporally and spatially varying (more details on spectral sensing techniques can be found in Chap. 4).

These types of interactions between light and plant canopy have been well studied from a remote sensing perspective (Li et al. 2014; Walter-Shea and Norman 1991). However, for agricultural and field robotic applications, the interest is often in proximal sensing, more at the single plant morphology or plant component level. This is an area where there is an opportunity for more work to develop the means to simulate sensors.

Specifically for agricultural robots, the morphological characteristics of plants, such as the shapes of plant canopy and leaves in the projected two-dimensional (2D) image plane or in three-dimensional (3D) space, can be used in perception systems to discriminate plants of different species (Dyrmann et al. 2018; Tang and Tian 2008; Wu et al. 2007). Additionally, crop row pattern are also used for localization and automated guidance through crop row (Åstrand et al. 2002; Higuti et al. 2018; Liu et al. 2016; Xue and Xu 2010). Techniques such as machine learning are applied to the extracted features to classify species of plants and identify crop rows (Pinto et al. 2000; Rehman et al. 2019; Tang et al. 2003). Since the shapes of plant canopies and leaves are complex and varied, the most challenging task is the investigation of effective and robust descriptors to differentiate different crop species in images.

As discussed in Chap. 2, cameras are commonly used as the primary source to extract the morphology characteristics. However, light reflectance sensors that acquire spectral information and no spatial information to discriminate between plant species have been investigated (Vrindts et al. 2002). This approach can be challenging unless the light is controlled and the plants are sparse. Since most spectral cameras are passive receivers of reflected light, they are dependent on the quality of the reflected light received. The spectral reflectance similarity of vegetation pixels can lead to difficulties in separating leaves or plants with occlusions, and shadow effects or saturation effects caused by uncontrolled illumination may affect the segmentation and feature extraction performance (Smith et al. 2018).

There is also loss of structural information with camera sensors because the 3D scene is projected onto a two-dimensional image plane. Thus depth sensors providing three-dimensional (3D) plant shape features extracted from 3D point clouds, as discussed in Chap. 3, have been found promising in addressing some of the problems in plant identification associated with reflectance-based sensors alone. These features are more robust to external illuminance changes and shadow effects than those extracted from color images (Gai et al. 2019; Li and Tang 2018; Vázquez-Arellano et al. 2016).

For many agricultural and field applications, the modeling and simulation of sensors needs to have a fidelity that includes the spectral reflectance characteristics of the plants and soil background in the field of view. Additionally, the plants grow with morphologies that are substantially more complex than for which many robotic simulations are capable as they are often designed to simulate robot in indoor environments with objects that can be represented with regular geometric shapes.

For current applications of robotic simulations, it is important to first identify scale for which the perception system is working with in the plant context, and what plant model fidelity is sufficient for specific applications. For example, if the perception system is using corn plant rows for determining robot pose, it may be possible to use vertically oriented cylinders to represent the corn stalks to evaluate perception algorithms. In addition, many solid models are available for integration into virtual worlds.

In summary, depending on the fidelity required, the sensor models and simulation for the agricultural and field robotic simulation may be very complex to represent the interaction with different light spectra and complex soil, plant canopy, and environmental structures.

12.12.3 Modeling and Simulating Machine Operators

Depending on the amount of interaction between the robot and a human operator, the human operator can be an important element in the closed loop of the simulation system. Specifically, where there is a strong human-in-the-loop aspect of the human operator, then complete vehicle system simulation must have models of the operator, which is typical with off-road vehicles (Filla et al. 2005). Existing operator modeling approaches for off-highway vehicles fit into two categories: (1) task-oriented operations in which the operator controls the machine through a repeated sequence of tasks to accomplish high-level goals (Filla 2005; Elezaby 2011) and (2) reference-oriented operations in which the operator is guiding the machinery along a particular path to accomplish some types of operation (Norris et al. 2003). More recently, virtual operator models were developed for the excavator trenching operation with adaptability to the work site environment (Du et al. 2018) and also learning the characteristics of the operation resulting in optimized performance (Du et al. 2019). In the case of agricultural and field robotics, likely human operators will play a supervisor role into the future and their response is an important aspect to be modeled and included in the simulation. From another perspective, operator behavior has been modeled for particular operations and used as the strategies for automating those operations (Bradley and Seward 1998; Wu 2003; Enes 2010).

12.13 Case Study: A Phenotyping Robot

12.13.1 Introduction

The Iowa State University (ISU) Phenobot Project is a good example of how modeling and simulation was used during the design process of a navigational control system and mission planning of an agricultural robot over soil surfaces. The third

Fig. 12.14 Phenobot 3.0 is articulated with a front and back section, and a vertical sensor mast for carrying sensors between corn plants is mounted at the center

generation of the ISU Phenobot—Phenobot 3.0—was designed to traverse corn rows with a conventional spacing of 0.71 m and carry a sensor package mounted on a mast so that phenotypical data of corn plants could be acquired across fields throughout the growing season. The width of the robot was 0.51 m to facilitate vehicle guidance through the crop rows (Fig. 12.14), and the sensor mast height was adjustable ranging from 2.1 to 3.7 m. Additionally, an anti-rolling mechanism was designed to correct the rolling angle of the sensor mast via a closed-loop control system.

The goal of the navigation system is to guide the Phenobot between the corn plant rows and keep it centered between two crop rows. Sensors for this task include a GPS antenna mounted at the top of the sensor mast, which is used for global localization of the vehicle along the path between the crop rows. A forward facing camera is continuously capturing images in front of the vehicle so that the robot can be aware of its location relative to the nearby corn rows. Motor encoders and an IMU (inertial measurement unit) sensor were used for vehicle odometry. During the navigation control development process, ROS was used as a middleware to link different functional processes in the control system implemented on an on-board PC. Gazebo was used to simulate the Phenobot for debugging and fine-tuning the system. As described above, the simulation required a definition of the robot model, a world model, and sensor models.

12.13.2 Robot, World, and Sensor Models

The Phenobot 3.0 robot model consisted of eight bodies (or links in URDF nomenclature), which are the front and rear bodies, the front and rear wheels, two bodies for the connection between the front and rear, and the telescoping sensor mast which has a mast base and an upper mast (Fig. 12.15). Connecting these bodies together are seven joints: two continuous joints for the wheels, two revolute joints for steering and mast roll control, one fixed joint connecting the center tube link to the rear, and one prismatic joint for the height control of the mast. Four power trains (URDF transmission elements) were defined, connecting actuators to actuated joints, which are the front and rear wheels, the articulated steering joint, and the mast roll joint. The front and rear wheels are controlled in terms of velocity, while the articulated steering joint and the mast roll joint are controlled in their angular position. Three sensors are included in the robot model. First, the inertial measurement unit (IMU) was mounted to the front section of the vehicle. The GPS unit was mounted to the top of the sensor mast, and the TOF camera (Kinect II, Microsoft, Redmond, WA) was mounted at the front face of the vehicle (Fig. 12.15).

The robot world for the application consisted of realistic corn plant models that were purchased from CGTrader (Vilnius, Lithuania) and then were replicated into rows. The ground surface is made up of an uneven surface which increased the fidelity of the soil surface condition, and enabled the simulation of the mast roll as the vehicle moves through the field. The robot world was developed using the world editor application within Gazebo (Fig. 12.16).

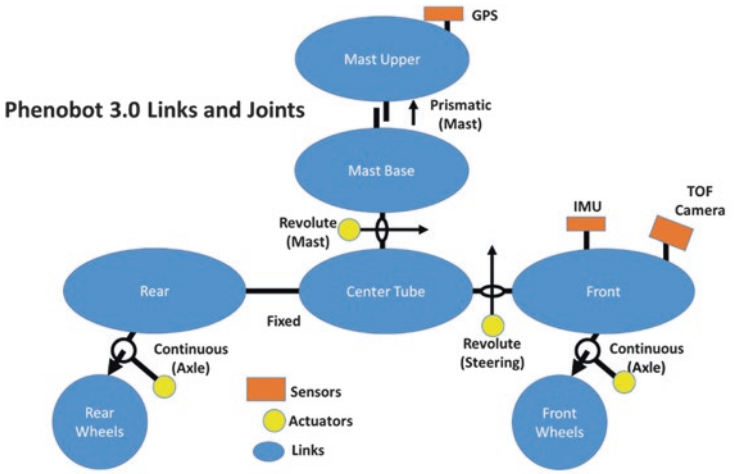


Fig. 12.15 Phenobot Robot model consisted of seven links or bodies and six joints connecting the links together

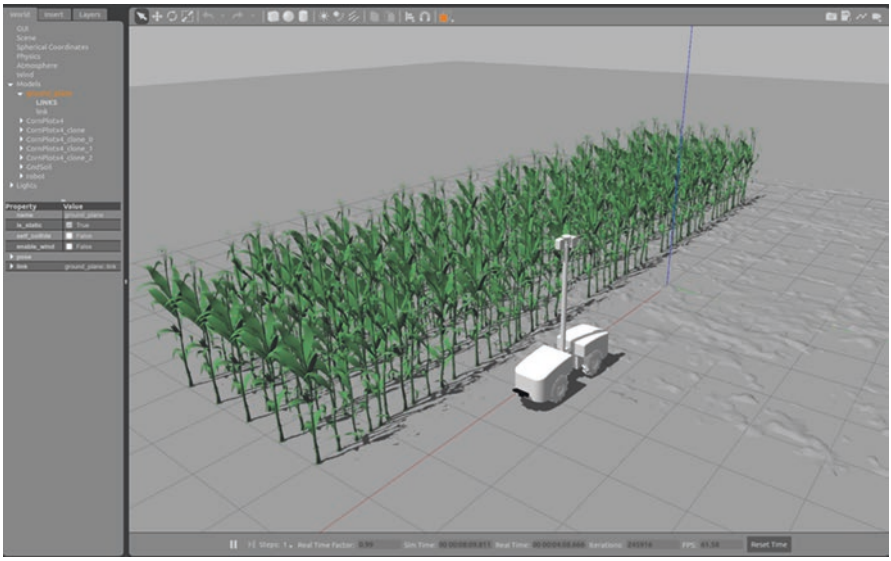


Fig. 12.16 The Phenobot 3.0 and the virtual world simulated by Gazebo. The virtual world was a field scene with rows of modeled corn plants and a soil surface with modeled roughness

12.13.3 ROS Node Network

The control program was implemented in ROS and has several modules. The modules, from lower hardware level to higher robot behavioral level, included hardware control, robot localization, robot navigation, and mission planning. The hardware control module listens to the robot movement commands and controls individual joints. The robot localization module listens to the sensor outputs (the encoders, IMU, GPS, and the front navigation camera) and calculates the current pose in both the vehicle local coordinate system and the Universal Transverse Mercator (UTM) coordinate system. Based on the pose of the robot and local circumstance, the robot navigation module plans and follows paths to reach specific targets during the mission. The mission planning module creates plans based on the user-specified tasks. With the modules above, a "graph" of ROS nodes for robot control was established (Fig. 12.17).

12.13.4 Robot Simulation with Gazebo

In this project, the robot was simulated in Gazebo to test the ROS program developed for robot control. At first, the robot and the sensor models in Gazebo were verified by using ROS visualization tools. Here in this project, RViz was used to inspect the robot model and the sensor output. Then, the developed ROS programs were

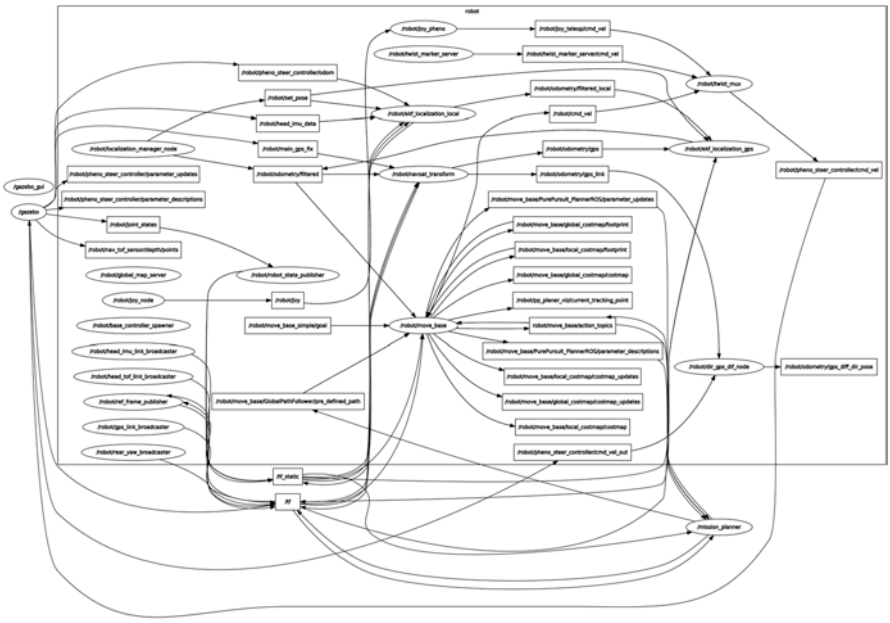


Fig. 12.17 The ROS node network of the robot control system (in the big rectangle named "robot") and their interaction with Gazebo (the leftmost ellipse). The ellipses are ROS nodes and the rectangles are the ROS message topics for data interchanges between nodes

attached to the Gazebo simulation environment and tested from the lower level hardware control module to the higher level mission planning module. The robot hardware control module was tested by inspecting the robot movement with some user-specified speed command. When testing the localization module, the parameters in the module were tuned by comparing the calculated localization result and the robot actual pose in the simulation world. The navigation module was developed and validated by observing the robot navigation behaviors, such as path planning, path following, and obstacle avoidance, when providing a target pose or a reference path. The mission planning module was finally developed and attached to the ROS network, which enabled the robot to traverse through the crop rows automatically (Fig. 12.18).

12.14 Summary and Concluding Thoughts

The simulation of agricultural robots is possible with the current set of tools that are available from either the dynamical systems and controls discipline or the robots/gaming discipline. Each family of tools has strengths and weakness that should be considered at the outset of a product development process. Before choosing a

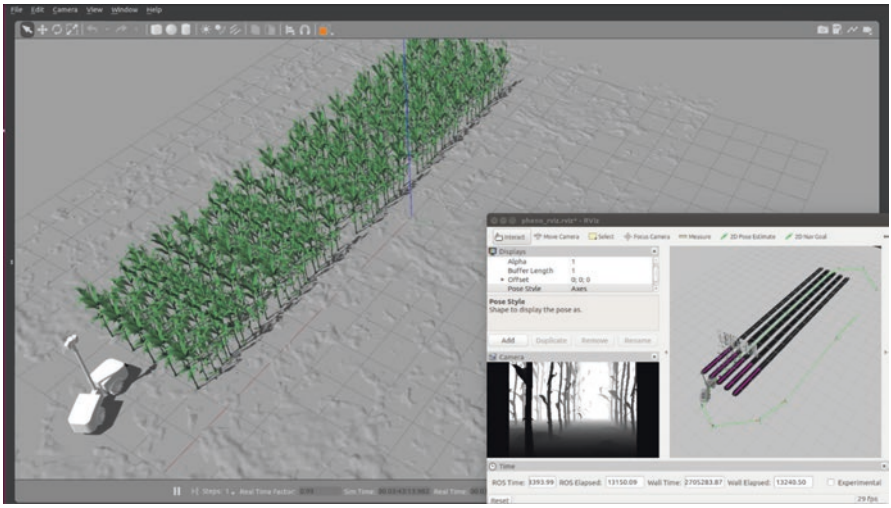


Fig. 12.18 Phenobot in Gazebo-simulated virtual world. The robot model, the world, and the sensors were simulated in the simulated environment. The robot pose, the map, the planned trajectory, and the sensor output were displayed on the Rviz-based control panel on the bottom right

simulation tool framework, it is important to decide on the modeling fidelity and scope required for the project.

Robot simulation can be divided into two subsystems: the algorithms needed to do all of the perception, control, and behavior for the robot, and then models of the physical hardware and environment in which the robot will be operating. For agricultural and field robots, much of the complexity in faithfully representing reality occurs at the interface between the robot and the environment. Specifically, the soil-machine interface is important for robot mobility across deformable soils and robotic control of soil engaging tools. Sensors are required for robot perception and in the agricultural/field context; those sensors interact with complex field scenes, making high fidelity simulation a challenge.

Simulation technology for autonomous systems is advancing at a very rapid pace. It is likely that the two different families of tools described in this chapter will converge drawing from the strength of each type of tool. Additional physics-based approaches for modeling and simulating agricultural and field robot and environment interactions will increase with higher-fidelity models of machine-environment interactions and improved processes for calibrating and validating models.

References

ABAQUS, Version 6.4. (2013) ABAQUS theory manual. Providence (RI, USA): ABAQUS, Inc.

ASABE (2018) Soil cone penetrometer. ASAE Standard S313.3 FEB1999 (R2018)

Åstrand B, Baerveldt AJ, Astrand B, Baerveldt AJ (2002) An agricultural mobile robot with vision-based perception for mechanical weed control. Auton Robot 13(1):21–35

Autodesk (2012) DXF Reference. Available at: images.autodesk.com/adsk/files/autocad_2012_pdf_dxf-reference_enu.pdf
Bipin K (2018) Robot operating system cookbook. Packt Publishing, Birmingham
Box GEP, Draper NR (1987) Empirical model-building and response surfaces. Wiley
Bradley D, Seward D (1998) The development, control and operation of an autonomous robotic excavator. J Intell Robot Syst 21:73–97
Cellier FE (1991) Continuous system modeling. Springer, New York
Chen WF, Mizuno E (1990) Non-linear analysis in soil mechanics: theory and implementation. Dev Geotech Eng 53
Chiroux RC, Foster WA Jr, Johnson CE, Shoop SA, Raper RL (2005) Three-dimensional finite element analysis of soil interaction with rigid wheel. Appl Math Comput 2005(162):707–722
Choi S, Seo KM, Kim T (2017) Accelerated simulation of discrete event dynamic systems via a multi-fidelity modeling framework. Appl Sci 7(10):1056
Coumans, E. (2015). Bullet physics simulation: introduction to rigid body dynamics and collision detection
Cundall PA, Strack ODL (1979) A discrete numerical model for granular assemblies. Geotechnique 29(1):47–65
DeBord M (2018) Waymo just crossed 10 million self-driving miles—but the company has a secret weapon that gives it even more of an edge. Business Insider. Accessed 6 June 2019
Du Y, Dorneich MC, Steward BL (2018) Modeling expertise and adaptability in virtual operator models. Autom Constr 90:223–234
Du Y, Dorneich MC, Steward BL (2019) Development of a learning capability in virtual operator models. SAE Int J Commer Veh 12(2)
Dymola (2018) Dymola user manual volume 1, March 2018, Version 2019, p 609
Dyrmann M, Christiansen P, Midtiby HS (2018) Estimation of plant species by classifying plants and leaves in combination. J Field Robot 35(2):202–212
EDEM (2011) EDEM theory reference guide. DEM Solutions, Edinburgh
Elezaby AA (2011) Virtual autonomous operator model for construction equipment applications. Dissertation, University of Illinois – Chicago, Illinois
Enes AR (2010) Shared control of hydraulic manipulators to decrease cycle time. Ph.D. thesis, Georgia Tech
Erez T, Tassa Y, Todorov E (2015, May) Simulation tools for model-based robotics: Comparison of bullet, havok, mujoco, ode and physx. In 2015 IEEE international conference on robotics and automation (ICRA). IEEE, pp 4397–4404
Filla R (2005) Operator and machine models for dynamic simulation of construction machinery. Thesis. Linkoping University, Linkoping, Sweden
Filla R, Ericsson A, Palmberg JO (2005) Dynamic simulation of construction machinery: towards an operator model. International fluid power exhibition 2005 technical conference, Las Vegas, (NV), USA, pp 429–438
FMI (2014) Functional mock-up interface for model exchange and co-simulation. Version 2.0. July 25, 2014. Accessed at fmi-standard.org/docs/2.0.1-develop/
Foley, J. D., van Dam, A., Feiner, S. K., Hughes, J. F., Hughes, J. & Angel, E. (1996). Computer graphics: principles and practice in C (2nd). Addison-Wesley Professional Upper Saddle River
Fritzson P (2015) Principles of object-oriented modeling and simulation with Modelica 3.3: a cyber-physical approach. IEEE Press/Wiley, Piscataway
Gai, J., Tang, L., & Steward, B. L. (2019). Automated crop plant detection based on the fusion of color and depth images for robotic weed control J Field Robot 21897
Gazebo (2019) Tutorial: using a URDF in gazebo. From gazebosimorg/tutorials/?tut=ros_urdf. Accessed on 2 Aug 2019
Gazebo (2020) Gazebo tutorials: sensors. From http://gazebosim.org/tutorials?cat=sensors. Accessed on 23 Apr 2020
Gill WR, Vanden Berg GE (1968) Soil dynamics in tillage and traction. Agriculture handbook no. 316. USDA-Agricultural Research Service, Washington, DC

Godwin RJ, Spoor G (1977) Soil failure with narrow tines. J Agric Eng Res 22(3):213–228
Grimm (2004) User's guide to rapid prototyping. Society of Manufacturing Engineers, Dearborn
Han SF, Steward BL, Tang L (2015) Intelligent agricultural machinery and field robots. In: Zhang Q (ed) Precision agriculture for crop farming. CRC Press, Boca Raton, pp 133–176
Higuti VAH, Velasquez AEB, Magalhaes DV, Becker M, Chowdhary G (2018) Under canopy light detection and ranging-based autonomous navigation. J Field Robot
Imperoli M, Potena C, Nardi D, Grisetti G, Pretto A (2018) An effective multi-cue positioning system for agricultural robotics. IEEE Robot Autom Lett 3(4):3685–3692
ISO (2016) ISO standard ISO 10303-21:2016. Industrial automation systems and integration—product data representation and exchange—part 21: Implementation methods: clear text encoding of the exchange structure
Janosi Z (1962) Theoretical analysis of the performance of tracks and wheels operating on deformable soil. Trans ASABE 5(64):133–134
Karkee M, Steward BL (2010) Study of the open and closed loop characteristics of a tractor and a single axle towed implement system. J Terrramech 47(6):379–393
Kavan L (2003) Rigid body collision response. Vectors 1000:2
Koolen AJ, Kuipers H (1983) Agricultural soil mechanics. Advanced series in agricultural sciences 13. Springer, Berlin/Heidelberg
Kyllo KP (2003) NASA funded research on agricultural remote sensing, Department of Space Studies, University of North Dakota
Lee J, Grey M, Ha S, Kunz T, Jain S, Ye Y et al (2018) Dart: dynamic animation and robotics toolkit. J Open Source Softw 3(22):500
Li J, Tang L (2018) Crop recognition under weedy conditions based on 3D imaging for robotic weed control. J Field Robot 35(4):596–611
Li L, Zhang Q, Huang D (2014) A review of imaging techniques for plant phenotyping. Sensors (Switzerland). MDPI AG
Liu L, Mei T, Niu R, Wang J, Liu Y, Chu S (2016) RBF-based monocular vision navigation for small vehicles in narrow space below maize canopy. Appl Sci 6(6):182
Mathworks (2019a) Basic principles of modeling physical networks. Accessed 10 June 2019 at https://www.mathworks.com/help/physmod/simscape/ug/basic-principles-of-modeling-physical-networks.html
Mathworks (2019b) Lane-following control with monocular camera perception. Accessed 8 Aug 2019 at https://www.mathworks.com/help/mpc/ug/lane-following-control-with-monocular-camera-perception.html
Mathworks (2020a) Simulation of a bouncing ball. Accessed 23 Apr 2020 at https://www.mathworks.com/help/simulink/slref/simulation-of-a-bouncing-ball.html
Mathworks (2020b) UAV competition example. Accessed 23 Apr 2020 at https://www.mathworks.com/help/sl3d/examples/uav-competition-example.html
Modelica (2019) Modelica tools. Accessed www.modelica.org/tools on 13 Sept 2019
Mondesire SC, Stevens DBMJ, Zielinski S, Martin GA (2016) Physics engine benchmarking in three-dimensional virtual world simulation. MODSIM World:5–8
Moore M, Wilhelms J (1988) Collision detection and response for computer animation. ACM Siggraph Comput Graph 22(4):289–298
Mousazadeh H (2013, June 1) A technical review on navigation systems of agricultural autonomous off-road vehicles. J Terramech. Elsevier Ltd
Nelson R (2018) Simulation and test drive vehicle success. EE-Evaluat Eng 57(10):6–13
Norris WR, Zhang Q, Sreenivas R, Lopez-Dominguez JC (2003) A design tool for operator-adaptive steering controllers. Trans Am Soc Agric Eng 46(3):883–892
ODE (2019) ODE manual. Accessed at http://ode.org/wiki/index.php?title=Manual#Collision_handling on 15 July
Pinto FAC, Reid JF, Zhang Q, Noguchi N (2000) Vehicle guidance parameter determination from crop row images using principal component analysis. J Agric Eng Res

Pratt MJ (2001) Introduction to ISO 10303—the STEP standard for product data exchange. J Comput Inf Sci Eng 1(1):102–103
Raper RL, Johnson CE, Bailey AC, Burt EC, Block WA (1995) Prediction of soil stresses beneath a rigid wheel. J Agric Eng Res 61:57–62
Rehman TU, Mahmud MS, Chang YK, Jin J, Shin J (2019, January 1) Current and future applications of statistical machine learning algorithms for agricultural machine vision systems. Computers and electronics in agriculture. Elsevier B.V.
ROS (2019) ROS/Introduction – ROS Wiki. Retrieved September 8, 2019, from http://wiki.ros.org/ROS/Introduction
Rosique F, Navarro PJ, Fernández C, Padilla A (2019) A systematic review of perception system and simulators for autonomous vehicles research. Sensors 19(3):648
Sakaguchi E, Ozaki E, Igarashi T (1993) Plugging of the flow of granular materials during the discharge from a silo. Int J Mod Phys B 7:1949–1963
Shabana AA (2020) Dynamics of multibody systems, 5th edn. Cambridge University Press, New York
Shamshiri R, Hameed IA, Pitonakova L, Weltzien C, Balasundram SK, Yule IJ, Grift TE, Chowdhary G (2018) Simulation software and virtual environments for acceleration of agricultural robotics: features highlights and performance comparison. Int J Agric Biol Eng 11(4):15–31
Shen J, Kushwaha RL (1998) Soil–machine interaction a finite element perspective. Marcel Dekker, Inc., New York
Sherman MA, Seth A, Delp SL (2011) Simbody: multibody dynamics for biomedical research. Procedia Iutam 2:241–261
Smith LN, Zhang W, Hansen MF, Hales IJ, Smith ML (2018) Innovative 3D and 2D machine vision methods for analysis of plants and crops in the field. Comput Ind 97:122–131
Steward BL, Gai J, Tang L (2019) The use of agricultural robots in weed management and control. In Robotics and Automation for Improving Agriculture. ed. J. Billingsley. Burleigh Dodds Science Publishing, Cambridge, UK
Taheri S, Sandu C, Taheri S, Pinto E, Gorsich D (2015) A technical survey on Terramechanics models for tire–terrain interaction used in modeling and simulation of wheeled vehicles. J Terrramech 57:1–22
Tang L, Tian LF (2008) Plant identification in mosaicked crop row images for automatic emerged corn plant spacing measurement. Trans ASABE 51(6):2181
Tang L, Tian LF, Steward BL (2003) Classification of broadleaf and grass weeds using Gabor wavelets and an artificial neural network. Trans ASAE 46(4):1247
Tekeste MZ, Tollner EW, Raper RL, Way TR, Johnson CE (2009) Non-linear finite element analysis of cone penetration in layered sandy loam soil – considering precompression stress state. J Terrramech 46:229–239
Tiller MM (2001) Introduction to physical modeling with Modelica. Kluwer Academic Publishers, Boston
Tsuji Y, Tanaka T, Ishida T (1992) Lagrangian numerical simulation of plug flow ofcohesionless particles in a horizontal pipe. Powder Technol 71:239–250
Upadhyaya SK (2009) Traction prediction equations. In: Upadhyaya SK, Chancellor WJ, Perumpral JV, Wulfsohn D, Way TRW (eds) Advances in soil dynamics, vol 3. ASAE, St. Joseph, pp 117–153
Upadhyaya SK, Rosa UA, Wulfsohn D (2002) Application of the finite element method in agricultural soil mechanics. In: Upadhyaya SK, Chancellor WJ, Perumpral JV, Schafer RL, Gill WR, VandenBerg GE (eds) Advances in soil dynamics, vol 2. ASAE, St. Joseph, pp 117–153. [Chapter 2]
Varghese A, Turner JL, Way RT, Johnson CE, Dorfi HR (2013) Traction prediction of a smooth rigid wheel using coupled Eulerian-Largrangian analysis. Proceedings of 2012 SIMULIA Community Conference, Providence RI, USA
Vázquez-Arellano M, Griepentrog HW, Reiser D, Paraforos DS (2016) 3-D imaging systems for agricultural applications—a review. Sensors 16(5)

Vrindts E, De Baerdemaeker J, Ramon H (2002) Weed detection using canopy reflection. Precis Agric 3(1):63–80

Walter-Shea EA, Norman JM (1991) Leaf optical properties. In Photon-vegetation interactions, pp 229–251

Wheeler PN, Godwin RJ (1996) Soil dynamics of single and multiple tines at speeds up to 20 km/h. J Agric Eng Res 63(3):243–249

Williams MA, Alleyne AG (2014, January) Variable fidelity modeling in closed loop dynamical systems. In ASME 2014 Dynamic Systems and Control Conference, DSCC 2014

Wong JY (2010) Terramechanics and off-road vehicle engineering: terrain behavior, off-road vehicle performance and design, 2nd edn. The Butterworth-Heinemann, Oxford

Wood MD (1990) Soil behavior and critical state soil mechanics. Cambridge University Press, Cambridge

Wu L (2003) A study on automatic control of wheel loaders in rock/soil loading. Ph.D. thesis, University of Arizona

Wu, S. G., Bao, F. S., Xu, E. Y., Wang YX, Chang YF, Xiang QL (2007) A leaf recognition algorithm for plant classification using probabilistic neural network. ISSPIT 2007 – 2007 IEEE international symposium on signal processing and information technology, pp 11–16

Xue J, Xu L (2010) Autonomous agricultural robot and its row guidance. In 2010 international conference on measuring technology and mechatronics automation, ICMTMA 2010, 1, pp 725–729

Part III
Emerging Topics in Agricultural and Field Robotics

Chapter 13
Advanced Learning and Classification Techniques for Agricultural and Field Robotics

Abhisesh Silwal, Tanvir Prahar, and Harjatin Baweja

13.1 Introduction

It is estimated that more than 80% of data type in big data are in pixel format, i.e., images and videos. As the daily volume of data generated globally rises exponentially, the task of extracting valuable information has already become a cumbersome task. One such technique to extract information from images and videos is classification, which involves classifying or clustering contextual information within images. In today's world, to process such high volume of information with high accuracy, both research and industry experts are depending on Artificial Intelligence (AI).

AI is a field of study that aims to develop machines that behave as if they were intelligent. A definite goal of research in AI is to understand intelligence and build intelligent systems that come closer to human performance of specific tasks at hand (Wolfgang 2011). In this era of bigdata, driven by the combination of improved and cost-effective sensors, increased availability of data, and most importantly high computational power, AI technologies are becoming the backbone of the new technological renaissance. These innovative technologies include search engines, natural language processing, machine learning, robotics, and computer vision, to mention a few.

In today's world, we can feel the presence of AI everywhere from targeted advertisement seen in social media feeds, and tackling climate change, to searching for relevant literatures to include in this book chapter. It is the predictive ability of AI and machine learning (a branch of AI) methods that has enabled it to gain such momentum. AI-powered analytics have given enterprises unparalleled edge to extract valuable information from bigdata to improve sales, boost marketing, and to predict the trends in the financial market in real time (Oliver 2018). Similar to

A. Silwal (✉) · T. Prahar · H. Baweja
Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: asilwal@andrew.cmu.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_13

manufacturing, financial, and service industries, in recent history, agricultural industries have also started to adopt and apply AI-based solutions for various cumbersome tasks.

As discussed in Chap. 1 and other chapters in this book, agricultural industry is one of the most important business entities in the world. It plays a critical role in global economy, and its sustainability (or their lack of it) has the potential to threaten global food security (Karkee et al. 2017). Through traditional automation, agricultural industries have increased productivity several folds while reducing the number of manual workers and other types of inputs thus decreasing production costs, increasing quality and reducing environmental impacts (Kapach et al. 2012). However, the current rate of production might not be able to meet the future demands of the population growth predicted by commonly accepted population growth models (Karkee et al. 2017). Therefore, improving currently existing agricultural technologies to sustain the future agricultural demand is vital (Karkee et al. 2017).

To increase productivity, the concept of "precision agriculture" was originated in the 1990s. In simple terms, it refers to the site-specific farm management concept of observing, measuring, and responding to variability in crops (McBratney et al. 2005). With ease in the availability of advanced technology such as remote sensing, internet of things (IoT) and robotic platforms, the concept of precision agriculture has evolved to "smart farming" (Santos et al. 2019; Walter et al. 2017). To increase productivity, smart agricultural systems deploy suites of sensors to constantly monitor different variables that affect crops while generating large quantity of data that quickly becomes impractical for manual analysis. To process such data in real time and accurately, agricultural industry is also leaning toward AI-based approach, which includes classical machine learning tools such as support vector machine (SVM), artificial neural network (ANN) to the state-of-the-art deep learning models to classify contextual information from images and other non-imaging sensors.

This book chapter is organized in the following manner. The first section intends to provide background concepts of the fundamental building blocks of machine learning and deep learning, the context to advanced classification and learning in practice today. The second section in this chapter reviews literatures in the area of computer vision and robotics in agriculture highlighting the applications of machine/deep learning techniques described in the first section. Then it focuses on the significance and impacts made by the advanced techniques compared to the traditional approaches. This book chapter then closes with some concluding thoughts on current limitations and future challenges.

13.2 Machine Learning

Machine learning (ML) is an important subset of Artificial Intelligence where the central idea is to learn from data without explicit programming (Samuel 1959). From application perspective, machine learning can also be regarded as a computationally intensive applied statistics to estimate complicated functions but

with a decreased emphasis on proving confidence intervals of these functions (Goodfellow et al. 2016).

13.2.1 Learning Algorithms

A popular definition of learning algorithm is often written as "a computer program said to have learned from experience, *E* with respect to some class of tasks, *T* and performance measure, *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*" (Mitchell 1997). Machine learning algorithms are typically implemented in situations that are too cumbersome for manual programming tasks. It should be noted that the task "***T***" in machine learning is not itself to learn but to attain the ability to do the task (Goodfellow et al. 2016). The following text provides brief description of different types of ML tasks frequently seen in agricultural applications.

Classification Classification task involves classifying input data to a predetermined set of possible classes. In another term, the learning algorithm learns to map input(s) to a fixed possible set of categories. The output of a classifier is categorical or qualitative value.

Regression Regression, on the other hand, produces quantitative output. Here, the task is to predict a numerical value from a set of inputs.

Clustering Clustering task is to group or cluster data with similar characteristics but without any supervision. Unlike classification, the total class labels are not known, and the learning algorithm must exploit the similarities to group data points into specific clusters.

There are more types of ML tasks including transcription, anomaly detection, machine translation, structured output, synthesis and sampling, imputation of missing values, denoising, density estimation, and recommendation. However, most of these tasks are either application specific or have common occurrence on other field of studies. For example, machine translation is commonly applied in natural language processing to translate between different languages. A detailed description of several of these tasks can be found in Goodfellow et al. (2016).

The measurement "*P*" is to quantify the performance of the task T. For instance, in the task of classification, the performance of the classifier is often attributed in accuracy as a metric to measure how the classifier is performing. Finally, the experience "*E*" is described as the experience the ML algorithm is allowed to have during the learning process (Goodfellow et al. 2016). This experience could be induced in a supervised or unsupervised way, which is the two broad categories of ML algorithms. In supervised learning, labels and targets in the dataset are preset and the ML algorithm learns to map the labels to a specific target. ML tasks such as classification and regression fall under this category. For instance, a dataset could

contain multiple labels of fruits associated to the different types of fruit. A classifier trained in this dataset would learn to associate new input images to one of the preset classes of fruits in a supervised manner. Similarly, a different dataset could just have images of fruits without any labels or hint to the classifier. The task is to cluster just fruit pixels from the rest of the image pixels. This scenario falls under the unsupervised ML category. It should be noted that there are no formal definitions of supervised and unsupervised learning (Goodfellow et al. 2016). The above example and distinctions are based on a commonly accepted descriptions and concepts that can provide a framework for appropriate application of these techniques to solve practical problems.

There are other variants of machine learning algorithms including semi-supervised and reinforcement learning. A semi-supervised ML, as the name suggests, falls between supervised (labeled data and defined class) and unsupervised (no labels and classes) ML categories. Thus, this class of ML algorithm experience both labeled dataset (often minimal) as well as unlabeled classes. Reinforcement learning, on the other hand, not only experiences fixed dataset but also interacts with the environment and learns through trial and error. In this chapter, we primarily focus on research on computer vision in agriculture within supervised and unsupervised ML categories.

13.2.2 Common Learning Models

Linear Regression Linear regression is the simplest learning model, and as the name implies, it solves linear regression between dependent variables (inputs) and independent variables (outputs). The equation governing this learning mode is as follows:

$$\hat{y} = w^T x \tag{13.1}$$

where $w \in \mathrm{R}^n$.

In the above equation, the input "x" is multiplied with the parameter "w" to the get models prediction $\hat{y}$.

Logistic Regression Logistic regression extends the linear behavior of linear regression with the inclusion of logistic function in its hypothesis representation. The equation governing logistic regression learning is as follows:

$$\hat{y} = \frac{1}{\left(1 + \mathrm{e}^{-\mathrm{w}^{\mathrm{T}}\mathrm{x}}\right)} \tag{13.2}$$

Perceptron Artificial neurons are mathematical abstractions and the basic building block of a neural network. Figure 13.3 depicts a block diagram of a perceptron. A perceptron is the most basic artificial neuron that takes several binary inputs and produces a single binary output (McCulloch and Pitts 1943). Here each input is multiplied by a weight, which can be any real number. It is these weights that are adjusted during the network training to produce desirable outputs. If the combined inputs are greater than a threshold, the output is high otherwise the output is low.

$$\text{Output} = \begin{cases} 0 if \sum_i w_i x_i \leq threshold \\ 1 if \sum_i w_i x_i > threshold \end{cases} \tag{13.1}$$

The perceptron can be a good choice if the decision manifold is linear. For example, let us consider a simple case where a farmer needs to decide to harvest or not to harvest in a given day. He might consider several variables to make this decision, but for brevity let us consider three variables viz. the possibility to rain that day (x_1), number of available workers (x_2), and amount of reserved fuel (x_3). In this example, let us further assume that some of these considerations are more crucial than others. For instance, for maintaining fruit quality, it should not be raining, so the input associated to the variable x_1 can have a higher weight. Similarly, availability of fuel (x_3) might not be that big of a deal, so that can have a lower weight. The threshold or the decision boundary can be set arbitrarily but usually set to 0.5, i.e., if the weighted sum of inputs is greater than 0.5, the algorithm favors the decision to harvest and vice versa.

In this example, the weights are $w_1 = 0.8$ for x_1, $w_2 = 0.5$ for x_2, and $w_3 = 0.1$ for x_3. If the weather is good ($x_1 = 1$), the labor is adequately available $x_2 = 1$ and the fuel is available $x_3 = 1$ then, the *output* = 1, so the farmer will continue to harvest. The weights and the thresholds can be varied as needed for representing different scenarios. Though a powerful tool, such simple linear cases do not always suffice for complex decision-making tasks. However, cascading multiple layer of these perceptrons, as shown in Fig. 13.1, can represent complex and nonlinear decision-making manifolds.

In this network, the first layer takes in the inputs multiplied by their respective weights. The output of each perceptron in the current layer is used as inputs for the next layer. Also, as a simplification of notation, the threshold can be replaced by a bias, and the sum of products of inputs and weights can be written as a dot product. So, the perceptron can now be represented as:

$$\text{Output} = \begin{cases} 0 \text{ if} w.x + b \leq 0 \\ 1 \text{ if} w.x + b > 0 \end{cases} \tag{13.2}$$

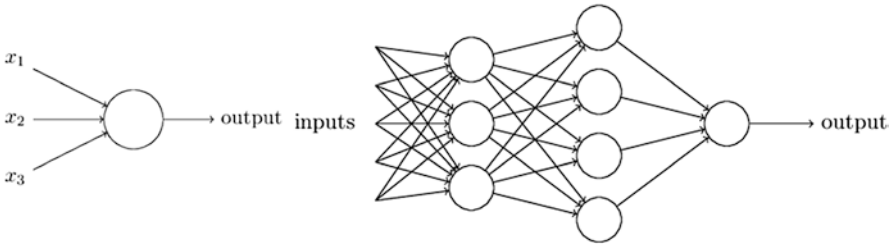


Fig. 13.1 A single perceptron with three inputs and single output (left). A multi-perceptron network (right)

Sigmoid Neurons A combination of these perceptrons makes up what is known as an artificial neural network (or just neural network). A crucial aspect of neural network (NN) is tuning the weights associated to each neuron, so that it produces the right output for the given input data. This process involves providing the network with an input, observing the output, and computing the loss (or error) on that output. The loss is computed by a special function called the loss function. The loss function returns a high value if the prediction made by the network is incorrect and a low value otherwise. These loss values are a measure of how much the weights should change in order to produce the correct output for the given input. In practice, gradient-based methods are generally used for calculating the updates of the weights. This requires the neural network to be made from continuous and differentiable units. Thus, the issue with vanilla perceptron is the discontinuity of the output (i.e., binary output). So, a modification to the perceptron is introduced in the form of a sigmoid neuron (Fig. 13.2 left). The sigmoid activation function adds a continuous nonlinear function output at the back of the perceptron unit, and the output can take any value between 0 and 1. The sigmoid function is represented as:

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{13.4}$$

Now the output of the neuron with sigmoid activation becomes:

$$\text{output} = \frac{1}{1+\exp(-z)} = \frac{1}{1+\exp(-w.x-b)} \tag{13.5}$$

The sigmoid neurons are in effect like perceptrons. If the value of z is very large, the neuron outputs a value close to 1, and if z it is very small, the neuron outputs a value close to 0. The ability to generate all real values between 0 and 1 also makes it continuous and differentiable. Fig. 13.2 (right) shows the output curve of the sigmoid neuron.

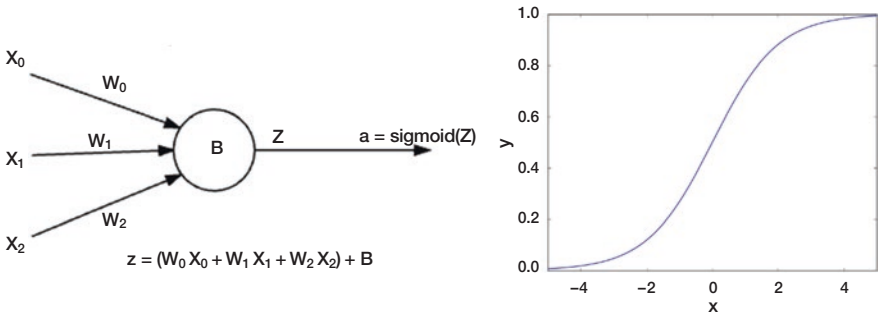


Fig. 13.2 A neuron with sigmoid activation layer (left). Output of the sigmoid activation function (right)

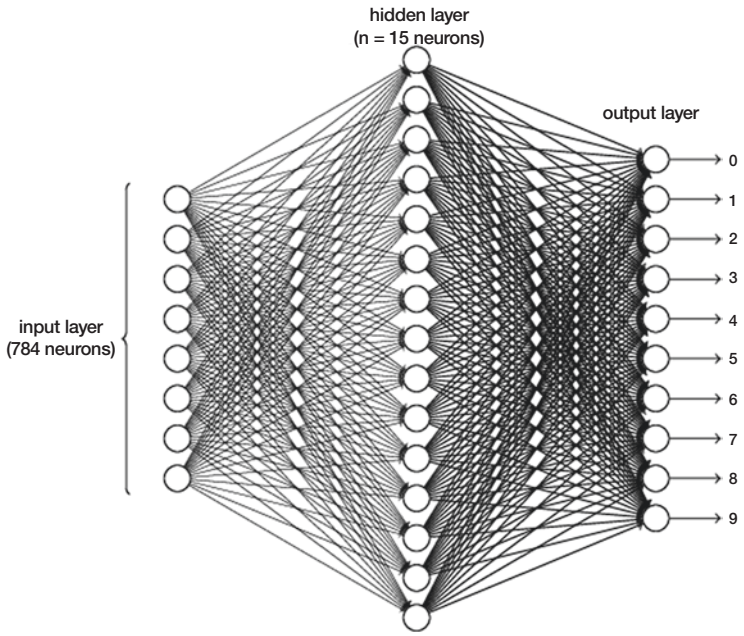


Fig. 13.3 A multi-perceptron network for hand-digit classification

13.2.3 Learning Weights with Gradient Descent

The real value of NN-based learning algorithms lies in the fact that the weights of neurons can be automatically altered to fit to the pattern or relationship inherently represented by the available input-output data. The most prevalent algorithm for doing so is called Gradient Descent. This is a gradient-based method, where the weights are altered such that the value generated by the loss function reduces over

time. To discuss about gradient descent, let us consider a simple task and a simple neural-network architecture shown below.

1. *Neural Network Architecture*: Suppose the task is to classify images of handwritten digits each of size 28x28 pixels in size. Here, the input to each neuron in the first layer (input layer) will be a 28x28 = 784x1 dimensional vector. Since, the output can take 1 out of 10 possible values (0–9), the output will be a 10x1 dimensional vector. In neural network architectures, varying numbers of neurons can be added in the input layer and the layer in between the input and the output layers (these are called hidden layers). These (including number of layers and number of neurons in each layer) are network hyperparameters and are generally decided by empirical data analysis. Figure 13.3 shows the network architecture developed for modeling this character recognition problem.

2. *Loss Function:* The neural network can now be considered as a function that transforms inputs (vectorized images) to outputs (one vector for the class the image belongs to). Here, the predicted class can be denoted by the function *y(x)*. As mentioned before, the loss function is a function that returns a higher value if the prediction is incorrect and a low value otherwise. A common loss function used in NN is the L2 loss as shown in eq. 13.4. This loss function is parameterized by NN parameters, as it depends on them.

$$L(w,b) = \left\| y(x) - a \right\|_2^2 \tag{13.6}$$

Here, w is the weights of the neural network and b are the biases, x is the input, $y(x)$ is the predicted output, and a is the actual output (or ground truth). Equation 13.6 shows the loss for a single output/ground-truth pair. The loss can also be calculated for batch updates, by just averaging the L2 loss over all the examples (input-output pairs available for training the network) in the batch, as shown in eq. 13.7

$$L_{\text{batch}}(w,b) = \frac{1}{n} \Sigma_{i=1}^{n} \left\| y(x_i) - a_i \right\|_2^2 \tag{13.7}$$

3. *Back Propagation:* Once the loss is calculated, it can be used to adjust the parameters of the neural network so that the network generates increasingly more accurate output. This is accomplished by iterating the process of computing the output, calculating the loss, and adjusting the parameters (weights and biases) to reduce the loss. The algorithm used to accomplish this process is called Gradient Descent by Back Propagation of Loss. To show the algorithm in action, let us assume a small network as shown in Fig. 13.4. This network has a four-dimensional input, two hidden layers, and a two-dimensional output. X_0 is the input vector, $x_1 = f(W_1 x_0 + b_1)$, $x_2 = f(W_2 x_1 + b_2)$ and the output $x_3 = f(W_3 x_2 + b_3)$. The tunable parameters in the network are: W_1(5x4), W_2(3x5), W_3(2x3), b_1(5x1), b_2(3x1), and b_3(2x1).

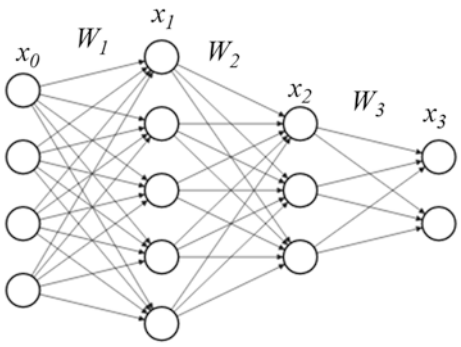


Fig. 13.4 An example neural network with four inputs, two hidden layer, and two outputs

The idea is to change theses tunable parameters such that the loss (eq. 13.6) reduces. Hence, the parameters need to be changed in the direction opposite to the gradient of the loss w.r.t. the parameters. Let us consider just the weights and not the biases for now. So, the weights need to be changed such that,

$$w := w - \alpha \frac{\partial L}{\partial w}, \text{for all weights w.} \tag{13.8}$$

The partial derivatives of Loss L w.r.t. all the network weights w are calculated through chain rule of derivatives. First the partial derivatives are derived for weights in the last layer

$$\frac{\partial L}{\partial W_3} = \frac{\partial L}{\partial x_3}\frac{\partial x_3}{\partial W_3}$$

$$\Rightarrow \frac{\partial L}{\partial W_3} = \frac{\partial}{\partial x_3}\frac{1}{2}\left\|x_3 - a\right\|_2^2 \frac{\partial x_3}{\partial W_{3.}}$$

$$\Rightarrow \frac{\partial L}{\partial W_3} = \left(x_3 - a\right)\frac{\partial x_3}{\partial W_3}$$

$$\Rightarrow \frac{\partial L}{\partial W_3} = \left(x_3 - a\right)\frac{\partial f\left(W_3 x_2\right)}{\partial W_3}$$

$$\Rightarrow \frac{\partial L}{\partial W_3} = \left(x_3 - a\right) \mathrm{o} f'\left(W_3 x_2\right)\frac{\partial W_3 x_2}{\partial W_3}$$

$$\Rightarrow \frac{\partial L}{\partial W_3} = \left[\left(x_3 - a\right) \mathrm{o} f'\left(W_3 x_2\right)\right] x_2^T$$

$$\Rightarrow \frac{\partial L}{\partial W_3} = \delta_3 x_2^T$$

Here, "o" is the Hadamard product. As a quick check, the dimensions of the LHS and RHS can be matched. So, $\partial L/\partial W_3$ must have the same dimensions as W_3. W_3 has a dimension of 2x3 and $(x_3 - a)$ has a dimension of 2x1,and $f_3'(W_3x_2)$ has a dimension of 2x1 as well. Variable x_2 has a dimension of xx_1, so $\delta_3 x_2^T$ is of dimension 2x3, which is the same as W_3.

Similarly, for W_2 we have

$$\frac{\partial E}{\partial W_2} = (x_3 - t)\frac{\partial x_3}{\partial W_2}$$

$$\Rightarrow \frac{\partial E}{\partial W_2} = \left[(x_3 - t) o f_3'(W_3 x_2)\right]\frac{\partial W_3 x_2}{\partial W_2}$$

$$\Rightarrow \frac{\partial E}{\partial W_2} = \delta_3 \frac{\partial W_3 x_2}{\partial W_2}$$

$$\Rightarrow \frac{\partial E}{\partial W_2} = W_3^T \delta_3 \frac{partial x_2}{\partial W_2}$$

$$\Rightarrow \frac{\partial E}{\partial W_2} = \left[W_3^T \delta_{3o} f_2'(W_2 x_1)\right]\frac{\partial W_2 x_1}{\partial W_2}$$

$$\Rightarrow \frac{\partial E}{\partial W_2} = \delta_2 x_1^T$$

And for W_1, the derivatives will be

$$\frac{\partial E}{\partial W_1} = \left[W_2^T \delta_{2o} f_1'(W_1 x_0)\right] x_0^T$$

$$\Rightarrow \frac{\partial E}{\partial W_1} = \delta_1 x_0^T$$

Thus, for every layer, there emerges a recursive relation for the gradients of the weights. Once the gradient equations are obtained, these can then be used to update the weights, such that the loss is minimized. The backpropagation algorithm consists of three parts,

Forward Pass All inputs are passed through the network to compute projected outputs, and the loss are computed in this pass.

$$x_i = f_i(W_i x_{i-1})$$

$$E = \|x_L - a\|_2^2$$

Backward Pass All the δs are symbolically computed in this pass

$$\delta_L = \left(x_N - a\right) \text{o} f_N^{'}\left(W_N x_{N-1}\right)$$

$$\delta_i = W_{i+1}^T \delta_{i+1} \text{o} f_i^{'}\left(W_i x_{i-1}\right)$$

Once the first layer is reached during the backward pass, input (x_0) is known. This is then plugged in, and all the numerical values for all the gradients are thus computed.

Weight Update The numerical values of the gradients are then used to update the weights

$$\frac{\partial E}{\partial W_i} = \delta_i x_{i-1}^T$$

$$W_i := W_i - \alpha \frac{\partial E}{\partial W_i}$$

The biases too can be updated in a similar way.

13.3 Deep Learning

Deep learning (DL) is currently one of the leading methods used in machine learning. The term "Deep" in DL refers to the increased number of layers between the input and output aka depth in DL architecture. Typically, in the artificial neural networks (ANNs) or neural networks (NNs) discussed in Sect. 13.2, the depth of the networks has remained fairly small. DL uses the same basic principles of the ANNs but then extends the depth of the network that provides enhanced capability for much more robust modeling and classification framework. Figure 13.5 shows the basic difference between classical machine learning and deep learning methods. As opposed to feature engineering in classical ML including ANN, DL methods offer end-to-end architecture that learns features from example dataset leading to simpler modeling/classification architecture and better generalization capability. Because of this very reasons, deep learning-based approach have out-performed classical machine learning and has become the go-to technique for most AI type problems (Seif 2018). Following several paragraphs provide brief description of some of the principle DL architectures found in the literature.

Deep Feedforward Networks As the name suggests, deep forward networks, also known as multilayer perceptrons (MLPs), are the type of neural network where information flows in just forward direction. These networks approximate some functions that map input to a certain category. These networks are often considered as the steppingstone to more complex networks discussed below (Goodfellow et al. 2016).

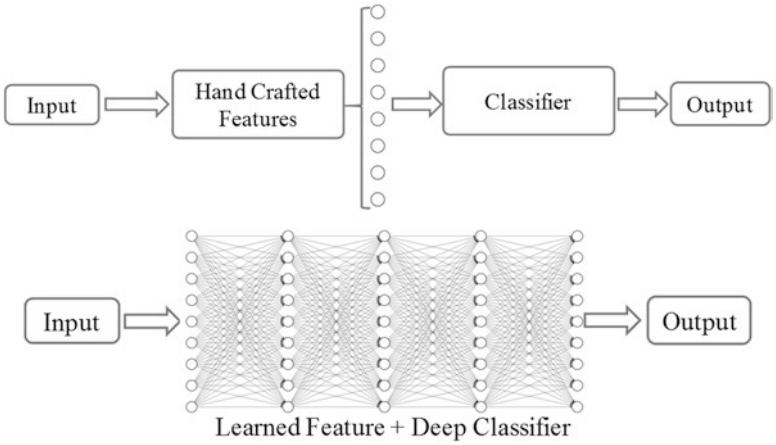


Fig. 13.5 General block diagram of classical machine learning algorithm (top) and deep learning architecture (bottom)

Convolutional Neural Networks Convolutional neural networks (CNNs) are a class of deep feed-forward networks described above. This is the most common architecture used in several research articles for feature extraction and classification. For this reason, CNNs are described in more details in this subsection. As seen earlier, a regular neural network requires a vectorized representation of data including input images if they are used to process images. This approach can perform well with images of small dimensions, but the network grows exponentially as the image size increases. For example, an image of size 200x200x3 would have 200 * 200 * 3 = 20,000 weights going to just one input neuron. Evidently, with several input and hidden layer neurons, the size and complexity of the model/network quickly gets unmanageable. Convolutional neural networks take advantage of the fact that the inputs to the neural network are images. So, the neurons are not fully connected to the preceding layer. Instead the neurons in a layer are only connected to a small region of the outputs of the layer before it. Just the neurons of the final output layers are connected fully to the preceding layer. Figure 13.6 shows a simple CNN architecture consisting of *N* convolution + ReLU + Max Pooling layers followed by a fully connected layer for *N* class classification.

1. Convolutional (CONV) Layer: The convolutional layer lies at the core of a Convolutional neural network (CNN). Each convolutional layer consists of learnable filters (weights). These filters have a small spatial span of a certain width and height but span through the whole depth of the input layer. For instance, the filter for the first CONV layer might be of size (5x5x3), which means that the width and height are both 5 pixels, and it is 3 pixel/channel deep for the 3 color channels of an RGB input image. These filters are slid over the input volume and the dot product between the entries of the filter and the input

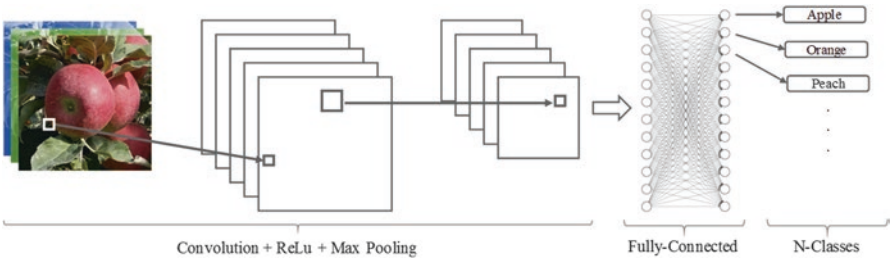


Fig. 13.6 A simple CNN architecture for image classification

array is computed. This process is called convolution. The output of each filter is thus a two-dimensional layer, called the activation map of that filter. Each layer in a CNN can have a number of these filters. Suppose if a layer has 9 filters, then it will have 9 activation maps. These activation maps are concatenated in depth dimension, and this concatenated volume is the output of a convolution layer.

The size of the output maps depends on the stride of the filters, padding of input volume, and the number of filters in a layer.

(i) The stride with which the filter slides over the input volume decides the size of the output. The stride is 1 pixel when the filter moves one pixel at a time. It can be any integer value. Higher the stride, lower the spatial size of the output volume.
(ii) It is convenient to add zeros at the edge of the input volumes, which is called zero padding. This process allows the convolutional operation to cover the entire input image and keep the spatial size of output the same as that of input volumes.
(iii) The number of filters dictates the depth of the output volume, as each filter produces two-dimensional activation map.

2. Non-Linearity: The outputs of the convolutional operations (CONV layer) are then sent through a nonlinearity function. This can be a sigmoid unit as defined above, a tan-h nonlinearity function or anything else that is differentiable. In most of the CNNs, a nonlinearity called the rectified linear unit (ReLU) is used which is defined as follows:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$

3. Pooling: It is a common practice to insert a pooling layer in between two consecutive convolution layers. The function of pooling layer is to reduce the spatial size and hence the number of tunable parameters. The pooling layer operates independently on each activation map, using the max-pooling operation. Most commonly, 2x2 filters are applied that down-sample every 4 pixels in activation map to 1 pixel, by taking the maximum of those 4 pixels. So, a pooling layer with

a size of PxP and stride S accepts a volume of size W1xH1xD1 and produces a volume with size W2xH2xD2, where W2 = (W1 − P)/S + 1, H2 = (H1 − P)/S+ 1, and D2 = D1. Though not a differentiable operation, it works with the back-propagation algorithm, because during backpropagation, the gradients can be routed to the input that had the highest value in the forward pass. For this process to work, there just needs to be a track of the indices of the max value at each layer.

4. Final Layer: Finally, depending on the task, the final layer can either be a number of sigmoid neurons (classification task) or linear neurons (regression task). The final layer is fully connected to the last activation map volume. If the size of the volume is *WxHxD,* then each neuron in the final output layer will have *W*H*D* weights.

Fully Convolutional Networks Fully connected networks (FCNs) are end-to-end learning DL networks based on CNN. This network has two sub-networks: convolution and deconvolution network. The convolution network down samples the input images using combination of filters, max pooling, and activation function as described in the above section. Whereas the deconvolution network up samples the features from the convolution end up to the full size of the input image to produce a semantic mask as output. The architecture of the FCN from the original paper (Long et al. 2015) is shown in Fig. 13.7.

Recurrent Neural Networks Unlike CNNs, which are designed to process gird values (such as images), a recurrent neural network (RNN) is a specialized neural network for processing a sequence of values. RNNs, also referred as multi-temporal networks, have access to output from previous moment and can use that information for the task in the current observation. Such functionality is achieved by sharing parameters across different parts of the model. Additional discussion on some of the more complex networks can be found in Chap. 16 (Sect. 16.3.3).

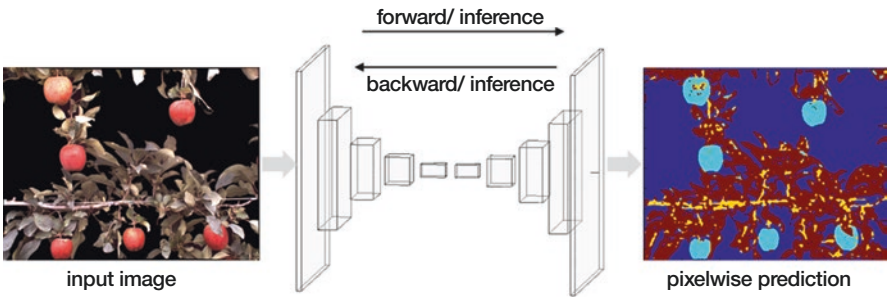


Fig. 13.7 Illustration of a FCN sematic segmentation framework

13.4 An Overview of Applications of Machine Learning in Agriculture

In the previous sections, basics on classical machine learning and deep learning were discussed, which provides a background on understanding the capabilities of various learning and classification techniques. This section summarizes relevant literatures that implement, improve, or propose custom machine/deep learning algorithms to solve computer vision problems relevant to agricultural and field robotics. The reviewed articles are categorized into two major subgroups: viz. object classification and clustering. As per the name, the classification subcategory includes algorithms based on supervised learning schemes that classify, localize, and count objects of interests such as fruit, weeds, and diseased canopy parts from images. It also includes articles for regression-based counting. On the other hand, the clustering subcategory includes studies that focus on implementing unsupervised learning methods.

Identifying objects such as fruits, vegetables, and diseased canopy parts directly from images and transforming that information into a layer in the decision-making process in the crop production cycle have the potential to revolutionize agricultural industries. Additionally, researchers often describe machine vision-based approach to agricultural automation as efficient, low-cost, nondestructive, and automated process scalable to large-scale implementations. The combination of above-mentioned advantages and demand from the industry has generated an enormous amount of interest in machine learning-based approaches to solving computer vision problems for agricultural automation and robotics. The plot in Fig. 13.8 shows the number of publications indexed by Google Scholar against the year of publication that include terms "Agriculture," and "Robotics," and phrase "Machine Learning/Deep Learning." This curve almost follows an exponential trend indicating the large attention to the application of learning-based approach in agriculture.

13.4.1 Classification

Crop/Fruit Detection Detecting fruits and vegetables in images is a fundamental requirement for any image-based automation task in agricultural fields (Silwal et al. 2014). It is often regarded as the critical factor for the success of autonomous systems in agriculture as it provides the ability to perceive and analyze sensory information to generate appropriate actions for tasks such as manipulating robot arms for harvesting or autonomous navigation in row crops (Silwal et al. 2017). As an example, a typical computer vision-based pipeline in fruit detection is to identify and localize individual fruit within the image. Although, this process looks simple and intuitive, getting consistent result in varying and uncertain field conditions has proven to be difficult. Numerous publications in the past three decades in vision-based agricultural robotics have regarded visual information processing as one of

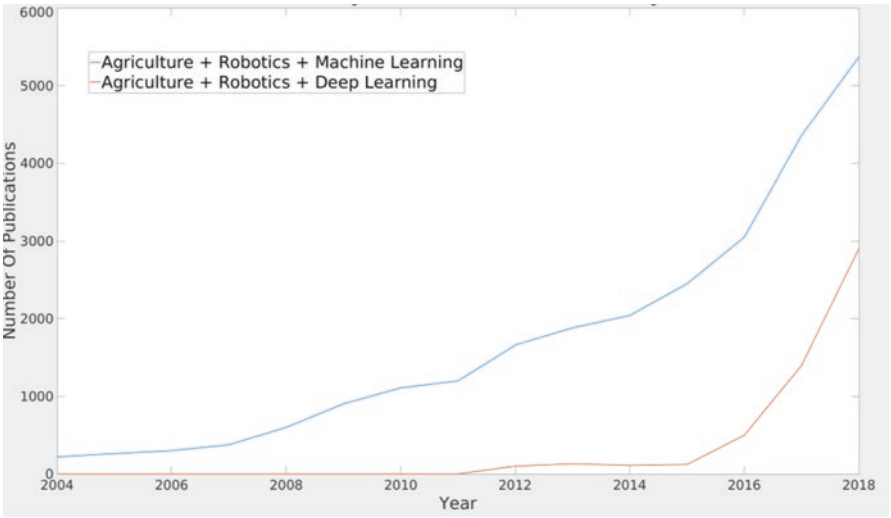


Fig. 13.8 The trends of using machine learning and deep learning in agricultural automation and robotics research. The plots were generated using advanced search feature on Google Scholar for terms: Agriculture, Robotics and "Machine Learning/Deep learning"

the major bottlenecks to reach commercial maturity (Kapach et al. 2012). To address the need for robust and more generalized fruit detection algorithm, this research problem has received much attention in the research community.

Support vector machine (SVM) is one of the most popular classical machine learning technique among researchers to detect fruits in images. Ji et al. (2012) and Wang et al. (2009) used SVM with color and shape features with three different SVM kernel functions (Poly, Radial basis function or RBF, and Sigmoid) to classify apples. With a dataset of 150 sample images, they reported apple detection accuracy of 93% for the RBF kernel. Similarly, Qiang et al. (2014) used SVM to classify objects into multiple class including citrus (as fruit), leaves, and branches. They also used RBF kernel, and detection accuracy was 92.4% in a dataset of 87 images. To detect green citrus, Sengupta and Lee (2014) trained SVM to classify textures on patches of circular objects detected using Circular Hough Transform (CHT). The dataset consisted of 100 images (38 training, 62 evaluation). Another frequently used classical learning tool for fruit detection is artificial neural network (ANN) as discussed above. Plebe and Grasso (2001) used hand crafted RGB color features to train ANN to identify oranges. The authors reported 87.0% oranges detection accuracy with 13.0% false positive and 5.0% false negative in a dataset of 800 images. Similarly, Regunathan and Lee (2005) used multilayer neural network for citrus detection based on manually crafted feature on hue, luminance, and saturation (HLS) color space. H and S values of every pixel in the test dataset were used to classify as citrus or background. Fruit was then counted by segmentation using morphological operation and watershed transformation. This study reported mean

percentage error of 39.6% on detecting fruit visible in the image. The dataset consisted of 74 images divided equally for training and testing.

Yield Estimation Yield prediction is one of the most critical yet complicated and inaccurate system that adds more uncertainty in the crop production cycle (Bargoti and Underwood 2017; Gongal et al. 2016). Currently, yield prediction in various crops such as apples and wine grapes is manually calculated as the product of total tree counts and the average number of fruits per tree/plant. As the average number of fruits in trees/plant could drastically vary, the final prediction could become highly inaccurate (Stein et al. 2016). Furthermore, in commercial orchards and vineyards, pre-/post-harvesting operations can cost up to a third of the total labor cost (Bac et al. 2014). Having accurate and early yield estimates could facilitate better management decisions and allow adequate time to allocate resources to such time-sensitive tasks to maximizing crop yield. Unsurprisingly, a large number of studies focused on early season yield prediction for various crops are evident from the published literature.

Stas et al. (2016) used Single DVI, Incremental NDVI, and Targeted NDVI for winter yield prediction of wheat in China. They compare Boosted Regression Trees and SVM for NDVI based yield prediction and concluded that Boosted Regression trees consistently outperform SVMs. Kaul et al. (2005) compared ANNs with multiple linear regression for yield estimation in corn and soybean farms. ANNs outperformed multiple linear regression for both the crops. Soil rating and weekly rainfall numbers were inputs to the learning algorithms. R^2 for corn prediction for ANN was 0.77 versus 0.42 for linear regression. In case of soybean, R^2 for NN-based approach was 0.81 as opposed to 0.46 for linear regression.

Morellos et al. (2016) used visible near infrared spectroscopy for total nitrogen, organic carbon, and moisture content of soil in Germany. Spectral wavelength from 305-2200 nm was used in this work to compare multivariate methods namely: principal component regression and partial least squares regression with ML approaches least squares support vector machines and Cubist (Kuhn and Johnson 2013). Least squares support vector machines performed the best for total nitrogen and organic carbon prediction with root mean squared prediction error (RMSPE) of 0.457% and 0.062%, respectively. Cubist method provided the best prediction results for total nitrogen with RMSPE = 0.071%. In order to exploit the full potential of several available spectral bands, You et al. (2017) use Deep Gaussian Process in conjunction with Long Short Term Memory (LSTM) to automatically extract relevant features while also incorporating spatiotemporal information for soybean yield prediction in the United States. The proposed method outperformed competing baseline approaches such as ridge regression, decision trees, and ANNs. González Sánchez et al. (2014) compare various ML approaches such as multiple linear regression, M5-Prime regression trees, multilayer neural networks, support vector regression, and k-nearest neighbor for field prediction for ten crop datasets. Attributes such as plantation area, irrigation depth, solar radiation, rainfall, and temperature range were used to train the models. On average M5-Prime achieved the largest number of crop yield models with the lowest errors. Nuske et al. (2011)

studied yield estimation over four growing seasons in various wine and table grape vineyards. They made use of active lighting stereo camera to get cues based on texture, color, and shape to train a random forest classifier to classify key-points between grapes and background. These detections are then fed into a calibration function that mapped dense visual measurements to predicted yield. This calibration function is generated using sparse in-season sampling or harvest from prior years. They achieved an average error between 3% and 11% from the actual total yield. More discussion on 3D sensing systems for yield estimation can be found in Chap. 3.

Biotic/Abiotic Stress Detection Effective crop protection requires accurate and early detection of biotic and abiotic stresses in plants (Behmann et al. 2015). Biotic and abiotic stresses are the two major threats to higher yield (Behmann et al. 2015; Liakos et al. 2018;). Several machine learning approaches have been investigated over the last decade to identify these ailments as early as possible. Mokhtar et al. (2015) developed a computer vision pipeline to detect yellow leaf curl virus in tomatoes from RGB mages. Features extracted from a preprocessed and segmented image were passed through an SVM classifier with various kernel functions. The authors achieved a maximum accuracy of 92% with a quadratic kernel function. Hernández-Rabadán et al. (2014) used self-organizing maps in conjunction with a Bayesian classifier to segment Powdery Mildew in tomato plants. Fuentes et al. (2017) compared various deep learning architectures for detection, localization, and classification of diseases such as Gray mold, canker, leaf mold, plague, leaf miner, white fly, and nutritional deficiency in tomato cultivar. Region-based fully convolutional metwork (Dai et al. 2016) with ResNet-50 feature extractor performed the best overall with mean average precision (mAP) score of 0.8598.

Mohanty et al. (2016) used Google Net (Szegedy et al. 2015) to detect 26 diseases in 14 crop species. Some of the diseases classified were apple scab, apple black rot, apple cedar rust, cherry powdery mildew, corn gray leaf spot, corn common rust, corn northern leaf blight, and orange huanglongbing (citrus greening). The trained model achieved a near perfect accuracy of 99.4%. Ghosal et al. (2018) performed identification, classification, and quantification of ailments such as bacterial blight, bacterial pustule, frogeye leaf spot, Septoria brown spot, sudden death syndrome, and iron deficiency chlorosis in soybean crop using CNN and achieved an overall classification accuracy of 94.1%. The detection accuracy was negatively affected by confounding symptoms that are tough to classify even for expert plant pathologists. The authors used feature maps of CNN layers to generate interpretable results. More discussion on spectral sensing techniques for crop stress detection can be found in Chap. 4.

Plant Phenotyping As discussed in Chap. 6, plant phenotyping is the process of measuring physical plant traits. Due to maturity in camera technology and recent advancements in computer vision, a large number of phenotype measurement solutions have been investigated using images as input and ML techniques to process those images. Chen et al. (2017) used dual CNN architecture for fruit count estimation as a phenotypic parameter. The first CNN segmented potential blobs

inside an image that were likely to contain fruit and the second CNN estimated fruit count within each blob segmented by previous CNN. Baweja et al. (2018) used a combination of faster R-CNN (Ren et al. 2015) and fully connected networks (FCN; Long et al. 2015) to segment, count and size sorghum stalks, whereas Sodhi et al. (2017) used a combination of SVM and conditional random fields (CRFs) on reconstructed point clouds to achieve the same task.

Cudic et al. (2018) used machine learning to find new genetic markers in single nucleotide polymorphism (SNP) sequences that correlate well to known phenotype values. Drawing inspiration from genetics, they developed a novel locally connected autoencoder architecture to reduce dimensionality of SNP sequences. Activation values of some of the encoding neurons were bound to be correlated to phenotype values. Upon backpropagating error from the encoding neurons of interest (Dimopoulos et al. 1999), new genetic markers were found in the input SNP sequences. Given the images of plants, a CNN was able to predict the new markers found with an average accuracy of 0.58%.

Parhar et al. (2018) presented a deep learning-based high-throughput, online pipeline for in situ sorghum stalk detection and grasping. Instead of using standard CNN-based architectures for semantic segmentation (Badrinarayanan et al. 2017; Chen et al. 2014; Long et al. 2015; Zhao et al. 2017), Mirza and Osindero (2014) used Conditional Generative Adversarial Networks (CGAN) for semantically segmenting plant stalks. In a relatively homogeneous environment facilitated by active lighting, this approach not only required fewer training images but also generated a realistic looking segmentation mask. These segmentation masks were then fed to find optimal grasp locations in a 3D point cloud for plant phenotyping. More discussion on the application of vision system for high-throughput crop phenotyping can be found in Chap. 6.

13.4.2 Clustering

A typical application of unsupervised learning in agriculture and elsewhere is the clustering of image pixels into different groups. In this category of ML techniques, the task is to group similar looking inputs (or inputs with similar features) into a discrete set of clusters. K-means (Lloyd 1982) and its variants and hierarchical techniques (Johnson 1967) are some of the popular choices in unsupervised ML. However, comparatively few research studies have reported implementing the unsupervised method in addressing agricultural issues. To detect green apples, Wachs et al. (2010) used K-means to segment apples from thermal and color images. Morphological operations and CHT were then used to improve classification accuracy. This study reported fruit detection accuracy of 38.8% on color images, 50.6% on thermal, and 53.2% with the combination of two images. Similarly, Bulanon et al. (2004) used K-means to cluster pixels in tree canopies with red apple. The segmentation via clustering approach resulted in fruit detection accuracy of 80%.

13.5 Significance of Machine Learning in Agriculture

Table 13.1 summarizes nearly five dozen different articles implementing various machine learning-based (ML), deep learning-based (DL), and traditional non-learning-based (NL) algorithms that addressed wide variety of computer vision problems in agriculture, in general, and agricultural automation and robotics, in particular. For brevity, only studies that focused on fruit detection, yield estimation, and crop stress (e.g., disease) detection are included in the table although the reviews and discussion throughout the chapter included a much wider application range. Furthermore, only those studies reporting performance with similar metrics are included in this analysis. Additionally, some studies with multi-class accuracy were averaged across all classes. Table 13.1 also included the total size of the dataset used for training, testing, and validation of the algorithms.

The average accuracy across three different tasks (fruit detection, yield estimation, and stress (disease) detection for non-learning algorithm is 79%. For classical machine learning algorithms, the average accuracy across these tasks is 85.42%. Similarly, deep learning-based algorithm scored 96% accuracy across the same three different tasks. The average size of data used in NL application was 95 images, whereas ML and DL application relied on significantly larger datasets. In our limited comparison, DL-based approach outperformed NL and ML approached by 16% and 10% respectively. ML-based algorithm also outperformed NL approach by almost 6%. It should also be noted that the intension of this comparison is to see the average trend of the reported accuracy across different visual tasks in agriculture between the application of NL, ML, and DL algorithms. As there are no common dataset used among these studies, a direct comparison is not possible. Figure 13.9 shows the graphical representation of the accuracy and dataset size as seen in Table 13.1

As DL architectures are becoming more popular in agricultural applications (evident from Fig. 13.7 and Table 13.1), only few recent publications have compared performance of DL to classical ML in their study. dos et al. (2017), Farooq et al. (2018), Ma et al. (2018), Rançon et al. (2019), and Zhong et al. (2019) compared CNN-based DL model to several ML-based models for weed, fruit, and disease detection respectively. In these studies, DL model was based on CNN and ML models included variants of SVM, random forest, PCA, SIFT, etc. (Fig. 13.9). As expected, the CNN model outperformed ML models in all tasks. These studies showcase additional cases where DL generated better accuracy then classical ML using same datasets for training and testing. In some cases such as in Zhong et al. (2019) (Fig. 13.9), the performance of DL and ML are close; however it should be noted that the ML architectures are subjective to hand engineered features that require expert knowledge. Additional details on the advantages and disadvantages of deep learning algorithms over conventional machine learning approach are summarized below (Fig. 13.10).

Engineered Feature Versus Learned Feature Except the vast performance boost for classification tasks reported in the literature, the main advantage of DL over the rest is because of its feature learning aspect. Hand-engineered features often require expert knowledge as well as considerable amount of time and effort to engineer and

Table 13.1 Summary of algorithms for various tasks in agriculture

Non-learning algorithms				
Article(s)	Objective	Dataset Size (min–max)	Accuracy (min–max)	Average accuracy
Bulanon et al. (2002), Çakır et al. (2013), Cohen et al. (2010), Dobrusin et al. (1993), Edan et al. (2000), Kurtulmus et al. (2011), Linker et al. (2012), Okamoto and Lee (2009), Payne et al. (2014), Pla et al. (1993), Silwal et al. (2014) Stajnko et al. (2009), Wachs et al. (2010), Zhao et al. (2005)	Fruit detection	17–173	70–90	80
Zhou et al. (2012) Gongal et al. (2016)	Yield prediction	50–424	74–82	78
–	Stress & disease detection	–	–	–
Classical machine learning algorithms				
Bulanon et al. (2004), Chaivivatrakul and Dailey (2014), Chinchuluun et al. (2006), Ji et al. (2012), Kurtulmus et al. (2014), Plebe and Grasso (2001), Qiang et al. (2014), Rakun et al. (2011), Seng and Mirisaee (2009), Slaughter and Harrell (1989), Wang et al. (2009), Zhong et al. (2019)	Fruit detection	13–3600	75–93.3	84.15
Chinchuluun et al. (2006), Ramos et al. (2017), Amatya et al. (2016), Sengupta and Lee (2014), Ali et al. (2016), Pantazi et al. (2016a, 2016b), Senthilnath et al. (2016)	Yield prediction	100–4300	80.4–89.6	85
Pantazi et al. (2017a, 2017b), Ebrahimi et al. (2017), Chung et al. (2016), Moshou et al. (2014), Moshou et al. (2004), Moshou et al. (2005), Moshou et al. (2006), Binch and Fox (2017), Lu et al. (2017), Ma et al. (2018), Rançon et al. (2019), dos et al. (2017), Farooq et al. (2018)	Stress & disease detection	55–14,208	74.34–99.92	87.13
Deep learning algorithms				
Koirala et al. (2019), Liu et al. (2018), Zhong et al. (2019), Sa et al. (2016), Stein et al. (2016)	Fruit detection	1300–3600	94–98	96
Heinrich et al. (2019), Koirala et al. (2019), Bargoti and Underwood (2017)	Yield prediction	500–1730	99–99.6	99.3
dos et al. (2017), Farooq et al. (2018), Lammie et al. (2019), Fuentes et al. (2017), Ma et al. (2018), Rançon et al. (2019), Ferentinos (2018), Lu et al. (2017), Zhang et al. (2019)	Stress & disease detection	200–87,484	86–99.6	92.8

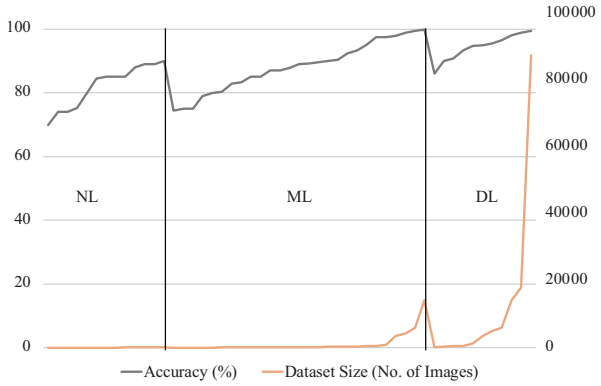


Fig. 13.9 Overall accuracy plot of NL, ML, and DL algorithm across three different tasks including crop/fruit detection, yield estimation, and biotic/abiotic stress detection

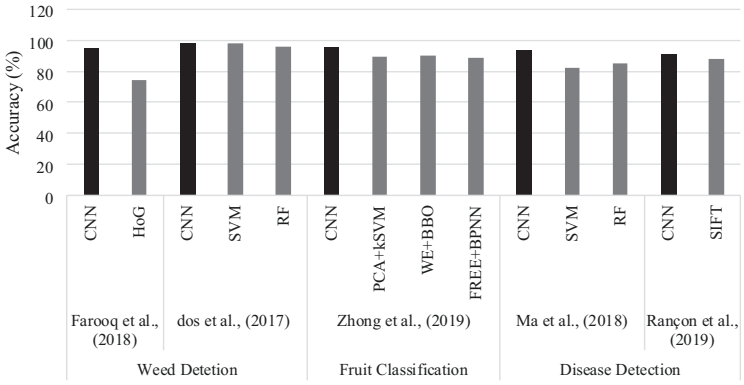


Fig. 13.10 Overall accuracy of ML and DL algorithm on same dataset. Darker color represents the model with highest accuracy as reported in the relevant paper

estimate (Kamilaris and Prenafeta-Boldú 2018). DL architecture inherits this feature as an integrated part of the pipeline and automatically learns to recognize the features. Furthermore, the hard-crafter features are often low-level features. With deep CNN layers, DL algorithms over training time learn much higher representation of the data further increasing robustness.

Generalization DL architectures offer end-to-end learning ability on the dataset and seem to generalize better. However, ML algorithm often offers just a part of the solution. As seen in the cited literatures, ML pipelines often had additional post-processing algorithm to further improve accuracy. As more algorithms are fused together, more parameters are added that could adversely affect generalization.

Data Requirement One often criticized aspect of DL algorithms is the amount of data required to train the network. This if often considered a major drawback for DL

architectures as not every researcher/research problem could have hundreds if not thousands of manicured datasets. Although, transfer learning, fine-tuning, and data augmentation (Goodfellow et al. 2016) offer some leverage over larger dataset requirement, a significantly larger dataset, compared to other traditional ML approaches, is always required for reliable performance. Data annotation is another significant requirement before the network can be trained to do anything. Often in field application of DL such as in agricultural sites, training on new experimental site often requires labeling and training on site. This can be a very cumbersome task.

Computing Resource and Training Time Training and deploying DL networks require computers with powerful graphical processing units (GPUs) whereas many ML algorithms can be deployed in less powerful and compact computing platforms. The training time for DL versus ML is also significantly different. Depending on the architectures and training scheme, DL network could take several hours to days. In contrary, ML algorithms are known to train and be ready to deploy in matter of minutes.

13.6 Summary and Concluding Thoughts

Variable lighting condition, fruit visibility, occlusion, and inconsistency in crop shape, size, and color are some of the most mentioned factors limiting the performance of computer vision algorithms in field environment published prior to the ML intervention (Gongal et al. 2015). With the intervention of classical machine learning in computer vision in agriculture and with the notion of learning from data, the solution to many agricultural tasks became relatively more consistent, but still the outcomes depended on experts to hand engineer reliable features to work with. With the advent of deep learning (DL), the availability of high computation power, and end-to-end learning scheme, the desire of having a general-purpose algorithm for classification problem in agriculture is getting closer to reality than it ever has been.

From the studies reviewed in this chapter, we can interpret that the focus of the computer vision research community in agriculture has been mainly to improve productivity, reduce manual work, and early detection of biotic/abiotic stresses (e.g., diseases) just from images. DL architectures are usually designed to work in two- or higher-dimensional data for usual tasks like classification. Throughout the reviewed studies, the application of DL in various agricultural problems showed improved accuracy compared to conventional approaches. In many cases, we could argue that the existing DL algorithms could provide out-of-box solutions to essential tasks in agriculture such as fruit detection and quality assessment. Public availability of such state-of-the-art tools could provide opportunities for new companies to commercially adopt these techniques in the near future.

The success and achievements of DL are usually attributed to the availability of high-power computation platforms and skilled programming. However, it can also be argued that the public sharing of data and open-source codes are equally contributing factors. Publicly shared large datasets such as Imagenet and COCO for

training DL algorithms provided a common resource that makes it convenient to evaluate competing algorithms. Such culture could also greatly benefit agricultural society as no such shared dataset exists.

In this chapter, we provided a brief description of the techniques and terminologies commonly used in machine learning. We summarized nearly five dozen papers applying machine learning in various tasks in agriculture. We also compared the performance of DL to various traditional approaches. To our findings, DL-based approaches generally outperformed in all applications they were applied to and were able to generalize to larger test datasets. Our aim in this chapter was to provide concise information of common components of ML and DL as well as the current status of ML and DL in agricultural applications. The application of DL in computer vision agricultural automation and robotics looks highly encouraging and could contribute toward more smart, automated, and sustainable farming.

References

Ali I, Cawkwell F, Dwyer E, Green S (2016) Modeling managed grassland biomass estimation by using multitemporal remote sensing data—a machine learning approach. IEEE J Select Topics Appl Earth Observ Remote Sens 10(7):3254–3264

Amatya S, Karkee M, Gongal A, Zhang Q, Whiting MD (2016) Detection of cherry tree branches with full foliage in planar architecture for automated sweet-cherry harvesting. Biosyst Eng 146:3–15

Bac CW, van Henten EJ, Hemming J, Edan Y (2014) Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. J Field Robot 31(6):888–911

Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell 39(12):2481–2495

Bargoti S, Underwood JP (2017) Image segmentation for fruit detection and yield estimation in apple orchards. J Field Robot 34(6):1039–1060

Baweja HS, Parhar T, Mirbod O, Nuske S (2018) Stalknet: a deep learning pipeline for high-throughput measurement of plant stalk count and stalk width. In: Hutter M, Siegwart R (eds) Field and service robotics. Springer, Cham, pp 271–284

Binch A, Fox CW (2017) Controlled comparison of machine vision algorithms for Rumex and Urtica detection in grassland. Comput Electron Agric 140:123–138

Behmann J, Mahlein AK, Rumpf T, Römer C, Plümer L (2015) A review of advanced machine learning methods for the detection of biotic stress in precision crop protection. Precision Agriculture 16(3):239–260

Bulanon DM, Kataoka T, Ota Y, Hiroma T (2002) AE—automation and emerging technologies: a segmentation algorithm for the automatic recognition of Fuji apples at harvest. Biosyst Eng 83(4):405–412

Bulanon DM, Kataoka T, Okamoto H, Hata SI (2004) Development of a real-time machine vision system for the apple harvesting robot. In: SICE 2004 annual conference, vol 1. IEEE, Sapporo, pp 595–598

Çakır Y, Kırcı M, Güneş EO, Üstündağ BB (2013) Detection of oranges in outdoor conditions. In: 2013 second international conference on agro-Geoinformatics (agro-Geoinformatics). IEEE, Fairfax, pp 500–503

Chaivivatrakul S, Dailey MN (2014) Texture-based fruit detection. Precis Agric 15(6):662–683

Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2014) Semantic image segmentation with deep convolutional nets and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell 40(4):834–848

Chen SW, Shivakumar SS, Dcunha S, Das J, Okon E, Qu C, Taylor CJ, Kumar V (2017) Counting apples and oranges with deep learning: a data-driven approach. IEEE Robot Automation Lett 2(2):781–788

Chinchuluun R, Lee WS, Burks TF (2006) Machine vision based Citrus yield mapping system. In Proceedings of the Florida State horticultural society (Vol. 119). Gainesville, pp 142–147

Chung CL, Huang KJ, Chen SY, Lai MH, Chen YC, Kuo YF (2016) Detecting Bakanae disease in rice seedlings by machine vision. Comput Electron Agric 121:404–411

Cohen O, Linker R, Naor A (2010) Estimation of the number of apples in color images recorded in orchards. In: International conference on computer and computing Technologies in Agriculture. Springer, Berlin/Heidelberg, pp 630–642

Cudic M, Baweja H, Parhar T, Nuske S (2018) Prediction of Sorghum bicolor genotype from in-situ images using autoencoder-identified SNPs. In: 2018 17th IEEE international conference on machine learning and applications (ICMLA). IEEE, Orlando, pp 23–31

Dai J, Li Y, He K, Sun J (2016) R-fcn: Object detection via region-based fully convolutional networks. In Advances in Neural Information Processing Systems. pp 379–387

Dimopoulos I, Chronopoulos J, Chronopoulou-Sereli A, Lek S (1999) Neural network models to study relationships between lead concentration in grasses and permanent urban descriptors in Athens city (Greece). Ecol Model 120(2–3):157–165

Dobrusin Y, Edan Y, Grinshpun J, Peiper UM, Wolf I, Hetzroni A (1993) Computer image analysis to locate targets for an agricultural robot. In: International conference on computer analysis of images and patterns. Springer, Berlin/Heidelberg, pp 775–779

dos Santos Ferreira A, Freitas DM, da Silva GG, Pistori H, Folhes MT (2017) Weed detection in soybean crops using ConvNets. Computers and Electronics in Agriculture 143:314–324

Ebrahimi MA, Khoshtaghaza MH, Minaei S, Jamshidi B (2017) Vision-based pest detection based on SVM classification method. Comput Electron Agric 137:52–58

Edan Y, Rogozin D, Flash T, Miles GE (2000) Robotic melon harvesting. IEEE Trans Robot Autom 16(6):831–835

Farooq A, Hu J, Jia X (2018) Analysis of spectral bands and spatial resolutions for weed classification via deep convolutional neural network. IEEE Geosci Remote Sens Lett 16(2):183–187

Ferentinos KP (2018) Deep learning models for plant disease detection and diagnosis. Comput Electron Agric 145:311–318

Fuentes A, Yoon S, Kim SC, Park DS (2017) A robust deep–learning–based detector for real–time tomato plant diseases and pests recognition. Sensors 17(9)

Ghosal S, Blystone D, Singh AK, Ganapathysubramanian B, Singh A, Sarkar S (2018) An explainable deep machine vision framework for plant stress phenotyping. Proc Natl Acad Sci 115(18):4613–4618

Gongal A, Amatya S, Karkee M, Zhang Q, Lewis K (2015) Sensors and systems for fruit detection and localization: a review. Comput Electron Agric 116:8–19

Gongal A, Silwal A, Amatya S, Karkee M, Zhang Q, Lewis K (2016) Apple crop–load estimation with over–the–row machine vision system. Comput Electron Agric 120:26–35

González Sánchez A, Frausto Solís J, Ojeda Bustamante W (2014) Predictive ability of machine learning methods for massive crop yield prediction. Spanish J Agric Res 2014 12(2):313–328

Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press, London

Heinrich K, Roth A, Breithaupt L, Möller B, Maresch J (2019) Yield prognosis for the agrarian management of vineyards using deep learning for object counting. In 14th international conference on Wirtschaftsinformatik. Siegan

Hernández-Rabadán DL, Ramos-Quintana F, Guerrero Juk J (2014) Integrating SOMs and a Bayesian classifier for segmenting diseased plants in uncontrolled environments. Sci World J 2014

Ji W, Zhao D, Cheng F, Xu B, Zhang Y, Wang J (2012) Automatic recognition vision system guided for apple harvesting robot. Comput Electrical Eng 38(5):1186–1195

Johnson SC (1967) Hierarchical clustering schemes. Psychometrika 32(3):241–254

Kamilaris A, Prenafeta-Boldú FX (2018) Deep learning in agriculture: a survey. Comput Electron Agric 147:70–90

Kapach K, Barnea E, Mairon R, Edan Y, Ben–Shahar, O. (2012) Computer vision for fruit harvesting robots—state of the art and challenges ahead. Int J Comput Vision Robot 3(1–2):4–34

Karkee M, Joseph AS, Davidson R (2017) Mechanical harvest and in-field handling of tree fruit crops. In: Zhang Q (ed) Automation in tree fruit production: principles and practice. CABI, Wallingford, p 179

Kaul M, Hill RL, Walthall C (2005) Artificial neural networks for corn and soybean yield prediction. Agric Syst 85(1):1–18

Koirala A, Walsh KB, Wang Z, McCarthy C (2019) Deep learning for real–time fruit detection and orchard fruit load estimation: benchmarking of 'MangoYOLO'. Precis Agric 20(6):1107–1135

Kuhn M, Johnson K (2013) Applied predictive modeling, vol 26. Springer, New York

Kurtulmus F, Lee WS, Vardar A (2011) Green citrus detection using 'eigenfruit', color and circular Gabor texture features under natural outdoor conditions. Comput Electron Agric 78(2):140–149

Kurtulmus F, Lee WS, Vardar A (2014) Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network. Precis Agric 15(1):57–79

Lammie C, Olsen A, Carrick T, Azghadi MR (2019) Low–power and high–speed deep FPGA inference Engines for Weed Classification at the edge. IEEE Access 7:51171–51184

Liakos KG, Busato P, Moshou D, Pearson S, Bochtis D (2018) Machine learning in agriculture: a review. Sensors 18(8):2674

Linker R, Cohen O, Naor A (2012) Determination of the number of green apples in RGB images recorded in orchards. Comput Electron Agric 81:45–57

Liu X, Chen SW, Aditya S, Sivakumar N, Dcunha S, Qu C et al (2018) Robust fruit counting: combining deep learning, tracking, and structure from motion. In: 2018 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, Madrid, pp 1045–1052

Lloyd S (1982) Least squares quantization in PCM. IEEE Trans Inf Theory 28(2):129–137

Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition. pp 3431–3440

Lu Y, Yi S, Zeng N, Liu Y, Zhang Y (2017) Identification of rice diseases using deep convolutional neural networks. Neurocomputing 267:378–384

Ma J, Du K, Zheng F, Zhang L, Gong Z, Sun Z (2018) A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. Comput Electron Agric 154:18–24

McBratney A, Whelan B, Ancev T, Bouma J (2005) Future directions of precision agriculture. Precis Agric 6(1):7–23

McCulloch WS, Pitts W (1943) A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 5(4):115–133

Mirza M, Osindero S (2014) Conditional generative adversarial nets. Retrieved from https://arxiv.org/abs/1411.1784

Mitchell TM (1997) Machine learning [pdf]. Retrieved from https://profs.info.uaic.ro/~ciortuz/SLIDES/2017s/ml0.pdf

Mohanty SP, Hughes DP, Salathé M (2016) Using deep learning for image-based plant disease detection. Front Plant Sci 7:1419

Mokhtar U, Ali MA, Hassanien AE, Hefny H (2015) Identifying two of tomatoes leaf viruses using support vector machine. In: Information systems design and intelligent applications. Springer, New Delhi, pp 771–782

Morellos A, Pantazi XE, Moshou D, Alexandridis T, Whetton R, Tziotzios G et al (2016) Machine learning based prediction of soil total nitrogen, organic carbon and moisture content by using VIS–NIR spectroscopy. Biosyst Eng 152:104–116

Moshou D, Bravo C, West J, Wahlen S, McCartney A, Ramon H (2004) Automatic detection of 'yellow rust' in wheat using reflectance measurements and neural networks. Comput Electron Agric 44(3):173–188

Moshou D, Bravo C, Oberti R, West J, Bodria L, McCartney A, Ramon H (2005) Plant disease detection based on data fusion of hyper-spectral and multi-spectral fluorescence imaging using Kohonen maps. Real-Time Imaging 11(2):75–83

Moshou D, Bravo C, Wahlen S, West J, McCartney A, De Baerdemaeker J, Ramon H (2006) Simultaneous identification of plant stresses and diseases in arable crops using proximal optical sensing and self-organising maps. Precis Agric 7(3):149–164

Moshou D, Pantazi XE, Kateris D, Gravalos I (2014) Water stress detection based on optical multisensor fusion with a least squares support vector machine classifier. Biosyst Eng 117:15–22
Nuske S, Achar S, Bates T, Narasimhan S, Singh S (2011) Yield estimation in vineyards by visual grape detection. In: 2011 IEEE/RSJ international conference on intelligent robots and systems. IEEE, Piscataway, pp 2352–2358
Okamoto H, Lee WS (2009) Green citrus detection using hyperspectral imaging. Comput Electron Agric 66(2):201–208
Oliver P (2018, June 6). What is predictive analytics. Retrieved April 20, 2020, from https://www.process.st/what-is-predictive-analytics/
Pantazi XE, Moshou D, Alexandridis T, Whetton RL, Mouazen AM (2016a) Wheat yield prediction using machine learning and advanced sensing techniques. Comput Electron Agric 121:57–65
Pantazi XE, Moshou D, Bravo C (2016b) Active learning system for weed species recognition based on hyperspectral sensing. Biosyst Eng 146:193–202
Pantazi XE, Tamouridou AA, Alexandridis TK, Lagopodi AL, Kontouris G, Moshou D (2017a) Detection of Silybum marianum infection with Microbotryum silybum using VNIR field spectroscopy. Comput Electron Agric 137:130–137
Pantazi XE, Tamouridou AA, Alexandridis TK, Lagopodi AL, Kashefi J, Moshou D (2017b) Evaluation of hierarchical self-organising maps for weed mapping using UAS multispectral imagery. Comput Electron Agric 139:224–230
Parhar T, Baweja H, Jenkins M, Kantor G (2018) A deep learning-based stalk grasping pipeline. In: 2018 IEEE international conference on robotics and automation (ICRA). IEEE, Piscataway, pp 1–5
Payne A, Walsh K, Subedi P, Jarvis D (2014) Estimating mango crop yield using image analysis using fruit at 'stone hardening' stage and night-time imaging. Comput Electron Agric 100:160–167
Pla F, Juste F, Ferri F (1993) Feature extraction of spherical objects in image analysis: an application to robotic citrus harvesting. Comput Electron Agric 8(1):57–72
Plebe A, Grasso G (2001) Localization of spherical fruits for robotic harvesting. Mach Vis Appl 13(2):70–79
Qiang L, Jianrong C, Bin L, Lie D, Yajing Z (2014) Identification of fruit and branch in natural scenes for citrus harvesting robot using machine vision and support vector machine. Int J Agric Biol Eng 7(2):115–121
Regunathan M, Lee W S (2005) Citrus fruit identification and size determination using machine vision and ultrasonic sensors. In 2005 ASABE Annual Meeting (p. 1). American Society of Agricultural and Biological Engineers. Regunathan, M., & Lee, W. S. (2005). Citrus fruit identification and size determination using machine vision and ultrasonic sensors. In 2005 ASABE Annual Meeting (p. 1). American Society of Agricultural and Biological Engineers.
Rakun J, Stajnko D, Zazula D (2011) Detecting fruits in natural scenes by using spatial frequency based texture analysis and multiview geometry. Comput Electron Agric 76(1):80–88
Ramos PJ, Prieto FA, Montoya EC, Oliveros CE (2017) Automatic fruit count on coffee branches using computer vision. Comput Electron Agric 137:9–22
Rançon F, Bombrun L, Keresztes B, Germain C (2019) Comparison of sift encoded and deep learning features for the classification and detection of esca disease in Bordeaux vineyards. Remote Sens 11(1):1
Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, Cambridge, MA, MIT Press, pp 91–99
Sa I, Ge Z, Dayoub F, Upcroft B, Perez T, McCool C (2016) Deepfruits: a fruit detection system using deep neural networks. Sensors 16(8):1222
Samuel AL (1959) Some studies in machine learning using the game of checkers. IBM J Res Dev 3(3):210–229
Santos L, Santos FN, Oliveira PM, Shinde P (2019) Deep learning applications in agriculture: a short review. In: Iberian robotics conference. Springer, Cham, pp 139–151
Seif G (2018) Deep learning vs classical machine learning. Retrieved April 4, 2020, from https://towardsdatascience.com/deep-learning-vs-classical-machine-learning–9a42c6d48aa

Seng WC, Mirisaee SH (2009) A new method for fruits recognition system. In: 2009 international conference on electrical engineering and informatics, vol 1. IEEE, Piscataway, pp 130–134
Sengupta S, Lee WS (2014) Identification and determination of the number of immature green citrus fruit in a canopy under different ambient light conditions. Biosyst Eng 117:51–61
Senthilnath J, Dokania A, Kandukuri M, Ramesh KN, Anand G, Omkar SN (2016) Detection of tomatoes using spectral-spatial methods in remotely sensed RGB images captured by UAV. Biosyst Eng 146:16–32
Silwal A, Gongal A, Karkee M (2014) Apple identification in field environment with over the row machine vision system. Agric Eng Int CIGR J 16(4):66–75
Silwal A, Davidson JR, Karkee M, Mo C, Zhang Q, Lewis K (2017) Design, integration, and field evaluation of a robotic apple harvester. J Field Robot 34(6):1140–1159
Slaughter DC, Harrell RC (1989) Discriminating fruit for robotic harvest using color in natural outdoor scenes. Trans ASABE 32(2):757–0763
Sodhi P, Vijayarangan S, Wettergreen D (2017) In-field segmentation and identification of plant structures using 3D imaging. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, Piscataway, pp 5180–5187
Stajnko D, Rakun J, Blanke M (2009) Modelling apple fruit yield using image analysis for fruit colour, shape and texture. Eur J Hortic Sci 74(6):260
Stas M, Van Orshoven J, Dong Q, Heremans S, Zhang B (2016) A comparison of machine learning algorithms for regional wheat yield prediction using NDVI time series of SPOT–VGT. In: 2016 fifth international conference on agro–Geoinformatics (agro–Geoinformatics). IEEE, Piscataway, pp 1–5
Stein M, Bargoti S, Underwood J (2016) Image based mango fruit detection, localisation and yield estimation using multiple view geometry. Sensors 16(11):1915
Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D et al (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, Piscataway, pp 1–9
Wachs JP, Stern HI, Burks T, Alchanatis V (2010) Low and high-level visual feature–based apple detection from multi–modal images. Precis Agric 11(6):717–735
Walter A, Finger R, Huber R, Buchmann N (2017) Opinion: smart farming is key to developing sustainable agriculture. Proc Natl Acad Sci 114(24):6148–6150
Wang JJ, Zhao DA, Ji W, Tu JJ, Zhang Y (2009, June) Application of support vector machine to apple recognition using in apple harvesting robot. In: 2009 international conference on information and automation. IEEE, Piscataway, pp 1110–1115
Wolfgang E (2011) Introduction to artificial intelligence (N. Black & F. Mast, Trans.). Springer-Verlag London Limited (Original work published 2017)
You J, Li X, Low M, Lobell D, Ermon S (2017) Deep gaussian process for crop yield prediction based on remote sensing data. In: Proceedings of the thirty-first AAAI conference on artificial intelligence. AAAI Press, Palo Alto
Zhang YD, Dong Z, Chen X, Jia W, Du S, Muhammad K, Wang SH (2019) Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. Multimed Tools Appl 78(3):3613–3632
Zhao J, Tow J, Katupitiya J (2005) On-tree fruit recognition using texture properties and color data. In: 2005 IEEE/RSJ international conference on intelligent robots and systems, Piscataway, pp 263, IEEE–268
Zhao H, Shi J, Qi X, Wang X, Jia J (2017) Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, Piscataway, pp 2881–2890
Zhong L, Hu L, Zhou H (2019) Deep learning based multi–temporal crop classification. Remote Sens Environ 221:430–443
Zhou R, Damerow L, Sun Y, Blanke MM (2012) Using colour features of cv. 'Gala' apple fruits in an orchard in image processing to predict yield. Precis Agric 13(5):568–580

Chapter 14
Digital Farming and Field Robotics: Internet of Things, Cloud Computing, and Big Data

Dimitrios S. Paraforos and Hans W. Griepentrog

14.1 Introduction

The demand for agricultural production is increasing, due to the 60–70% expected increase in food demand by 2050 (Porter et al. 2014). The increased production will have to face the challenge of sustainable farm management through optimized use of the available natural resources with a limited environmental impact, to meet the societal challenges of Sustainable Development Goals (United Nations 2015), such as food security and resource use efficiency. Advances in many disciplines of technology could provide their service toward giving solutions to this challenge.

Widely used precision agriculture (PA) techniques and tools, combined with technological developments in the Internet of Things (IoT), cloud computing, and big data analytics, are expected to bring the fourth revolution in farming and food production. This agricultural revolution, which in many cases is being referred to as "Agriculture 4.0" or "Digital Farming," follows the previous three revolutions. The first agricultural revolution took place around 10,000 BC and was characterized by the domestication of plants and animals. The second agricultural revolution was prompted by the industrial revolution during the nineteenth century when farming became mechanized and commercial with the development of new inventions and technologies. The third revolution or "Green" Revolution was a period when the productivity increased drastically as a result of new advances. During this period, high-yield crops were developed and introduced, and also new chemical fertilizers, synthetic herbicides, and pesticides were created.

The digital transformation of agriculture is being facilitated by the emergence of technologies focused on data acquisition and data management, which are expected to have a profound impact. The IoT is in the core (Fig. 14.1) as it is the connection

D. S. Paraforos (✉) · H. W. Griepentrog
University of Hohenheim, Stuttgart, Germany
e-mail: d.paraforos@uni-hohenheim.de

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_14

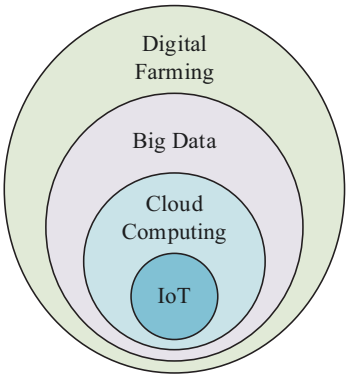


Fig. 14.1 General architecture of digital technologies in agriculture

with the physical world where various sensors and devices continuously monitor and control environmental variables related to atmospheric conditions, soil state, and biomass of biological organisms. As IoT devices are not always reachable in real time, due to energy-saving issues, the recorded timestamped information is transferred to the cloud through network gateways, in order for the data to be available for cloud-based IoT applications. Cloud computing technologies offer new possibilities in terms of storage and computation. Big data analytics are one level higher where data mining is being performed to detect trends, patterns, and associations, but also possible deviations. Digital farming, as was described above, requires this entire architecture in order to be able to support the farmer in everyday decision-making. An indispensable aspect of digital farming is field robotics as the latter act as the implementing component of the decision-making process. Agricultural robots, together with intelligent and highly automated agricultural implements, are responsible for the precise execution of agricultural operations (spraying, pruning, harvesting, etc.).

Since digital farming is being formulated as a highly cognitive system, it is necessary to have the layered architecture presented in Fig. 14.1, in order to acknowledge the increased complexity from one level to the other. The lower level of IoT can potentially operate as a stand-alone procedure in a reactive manner as closed loops can perform deterministic actions based on sensed data. Cloud computing techniques enhance manipulation of the acquired data, while big data technologies (including deep learning) introduce an associative formulation by linking sensory inputs with well-established patterns (Strube 1998). Digital farming being at the higher level of cognition requires knowledge, experience, and sensing and, thus, includes all previous technologies in order to achieve its mission.

The aim of this chapter is to give an overview of the offered technologies related to IoT, cloud computing, and big data from an agriculture-related perspective. The basic principles of each technology but also the emerging trends that could potentially have a profound impact in the agricultural domain will be also presented. One specific objective is to define a conceptual architecture that integrates all mentioned technologies and how these will interconnect in the future toward making Agriculture 4.0 a reality.

14.2 Internet of Things (IoT)

It is difficult to provide a unified definition of IoT as many exist due to the rapid penetration of the related technologies into our everyday life. Irrespective of all ambiguous definitions, everyone agrees that the IoT is an emerging technological paradigm where smart devices (Things) are equipped with sensors but can also act in order to control the physical world. The devices are assigned unique identifiers and can connect to the Internet but also interact with each other (Borgia 2014) forming a wireless sensor network (WSN). Based on the latest technology advances, it is becoming clear that the IoT is moving gradually to the cloud. In order to save energy, the IoT devices are unreachable most of the time. Nevertheless, the information should be always available for the application. Thus, a mirrored entity is being created in the cloud which holds all the information acquired from the physical entity (Atzori et al. 2017). By forming this cyber-physical ecosystem, the real and the digital world are in continuous interaction offering the user the capability to have a deeper and detailed view into the physical world.

14.2.1 IoT Architecture

A four-stage IoT architecture (Fuller 2016) is presented in Fig. 14.2. Stage 1 consists of all the "Things," typically sensors that collect environmental data such as temperature, humidity, soil properties, etc. In many cases, the Things can also be actuators with the purpose to vary the physical conditions based on the decision that is taken in situ or at a higher cognition level. At Stage 2, the data are being aggregated and converted into digital streams. Data preprocessing before entering the cloud and the data center takes place at Stage 3 by edge processing systems. Finally, at Stage 4, the data are stored on back-end cloud systems, and a thorough in-depth analysis is being performed. At this stage, advanced cloud-based systems and sophisticated big data techniques are being utilized (Popović et al. 2017).

14.2.2 IoT Hardware and Platforms

The last years the number of vendors offering IoT hardware platforms is increasing rapidly (Ray 2016). In Table 14.1, five widely used platforms are presented and are classified according to key parameters such as processor, memory, communication, logic level voltage, I/O connectivity, flash memory, operating temperature, power supply, and dimensions. The cost varies among offered platforms and at the moment this comparison was performed was from $30 up to $200 based on the offered hardware capabilities and programming interfaces.

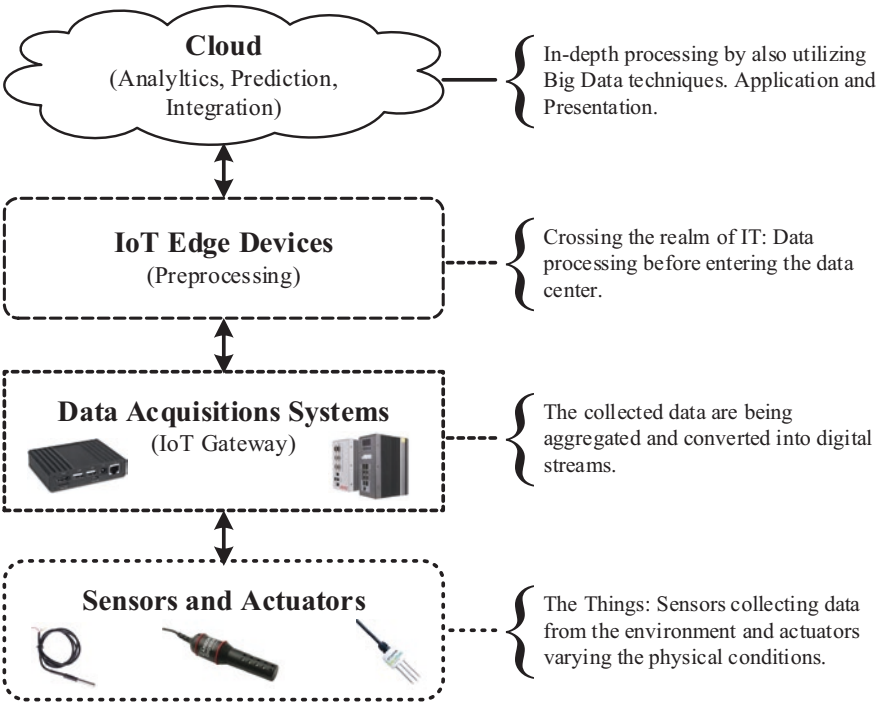


Fig. 14.2 A four-stage IoT architecture

14.2.3 Applications in Agriculture

The IoT has found wide implementation in agriculture with numerous applications. In order to categorize the various applications, three main domains have been identified: open-field farming, livestock farming, and protected agriculture. More details related to each category are presented in the following subsections.

14.2.3.1 Open-Field Farming

In open-field farming, the focus is on monitoring soil conditions such as temperature, moisture, pH, etc. Climate conditions and air monitoring are also examined by measuring temperature, humidity, and radiation (Talavera et al. 2017). Crop plant monitoring is being investigated by the detection of weeds, pests, and animal intrusion into the field, while crop growth is also being examined using IoT (Sreekantha and Kavya 2017). Irrigation control is of high importance as the proper amount and timing can minimize crop water stress and lead to water waste reduction. In this direction and toward developing an autonomous precision irrigation system through the integration of a center pivot irrigation system, an underground WSN was

Table 14.1 Basic characteristics of some of the widely used IoT hardware platforms

	Raspberry Pi 3 Model B+	BeagleBone Black	MinnowBoard Turbot Quad Core	Arduino Yún Rev 2	Qualcomm® DragonBoard™ 410c
Processor	Quad-Core Broadcom BCM2837B0, Cortex-A53 64-bit SoC at 1.4 GHz	AM335x 1 GHz ARM® Cortex-A8	Quad-Core Intel® Atom™ E3845 at 1.91 GHz	ATmega32u4 and Atheros AR9331	Quad-Core ARM® Cortex® A53 64-bit 1.2 GHz per core
Memory	1 GB LPDDR2 SDRAM	512 MB DDR3 RAM, 4 GB 8-bit eMMC on-board flash storage	2 GB DDR3L 1067MT/s DRAM	64 MB of DDR2 RAM	1 GB LPDDR3 533 MHz/8 GB eMMC 4.5/ SD 3.0 (UHS-I)
Communication	2.4 GHz and 5GHz IEEE, 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, Gigabit Ethernet over USB 2.0 (300Mbps), 4 × USB 2.0 ports	10/100 Ethernet, mini-USB 2.0 client port, USB 2.0 host port, 4 × UART, 8 × PWM/Timers, LCD, GPMC, MMC1, 2 × SPI, 2 × I2C, A/D converter, 2 × CAN bus (w/o PHY)	1 × USB 2.0, 1 × USB 3.0, microHDMI port, 1 × 1Gb Ethernet RJ45, SPI, I2C, I2S Audio, 2 × UARTs (TTL-level)	Ethernet: IEEE 802.3 10/100Mbit/s, Wi-Fi: IEEE 802.11b/g/n, 2.4GHz, 7 × PWM, BLE 4.0	Wi-Fi 802.11 b/g/n, 2.4 GHz, Bluetooth 4.1, Qualcomm® IZat™ location technology Gen8C, Wi-Fi, BT and GPS antenna, (expansion) UART, SPI, I2S, I2C x2
Logic level voltage	5 V	3.3 V	5 V	5 V	5 V
I/O connectivity	Extended 40-pin GPIO header	65 digital pins and 7 analog inputs	8 × GPIO (2 × supporting PWM)	20 digital I/O pins, 12 analog I/O pins	GPIO × 12
SD support/flash memory	MicroSD format for loading operating system and data storage	A single microSD (uSD) connector	1 × M.2 slot, with microSIM, 1x microSD	16 MB flash memory, microSD	MicroSD card slot, eMMC 4.5, 8 GB
Operating temperature	0–50 °C	0–60 °C (support for an industrial temperature range of −40 °C to +85 °C)	0–40 °C (wider range possible with a larger heat sink than provided with standard boards)	−40 to 85 °C	0–70 °C

(continued)

Table 14.1 (continued)

	Raspberry Pi 3 Model B+	BeagleBone Black	MinnowBoard Turbot Quad Core	Arduino Yún Rev 2	Qualcomm® DragonBoard™ 410c
Power supply	5 V/2.5 A DC via micro USB connector, 5 V DC via GPIO header, Power over Ethernet (PoE)-enabled (requires separate PoE HAT)	Mini-USB or 2.1 mm × 5.5 mm 5 V jack	5 VDC input via 2.5 mm center pin positive power jack	Micro-USB connection with 5 V, DC current per I/O pin: 40 mA on I/O pins; 50 mA on 3,3 pin	6.5 V to +18 V
Dimensions	85 × 56 mm	86 × 53 mm	99 × 74 mm	73 × 53 mm	54 × 85 mm

examined by Dong et al. (2013). The last years the tractor and implement communication data are also collected and analyzed. Paraforos et al. (2017b) connected an IoT device to the CAN bus diagnostics interface of the tractor, to inform the farmer regarding the performed agricultural operations.

14.2.3.2 Livestock Farming

An important contribution of IoT in livestock farming is related to animal tracking and behavioral analysis. Parameters such as animal health, proper insemination time, and reproductive health problems are monitored (Vannieuwenborg et al. 2017). Automated detection of lame animals, which produce less milk and have other problems, is being performed by collecting data from inertial measurement units. Analysis of these data reveals impaired movement or deviation from normal gait or posture (Haladjian et al. 2018). Other parameters include extreme climate condition detection that has an important impact on animal welfare, environmental conditions of a beehive, but also odor and hazardous gas monitoring.

14.2.3.3 Protected Agriculture

Greenhouses are highly intensive production systems that justify the use of advanced technologies such as IoT. High-precision monitoring and control systems are being implemented including micro-climate and crop sensing, valves and controllers for fertigation, and integrated pest management (Tzounis et al. 2017). The last years a novel farming system has emerged called "Vertical Farming" in which the crops are being cultivated inside buildings and in layers above one another, under a fully controlled and closed environment. In these systems, IoT devices are used to sense but also control various physical variables such as moisture, nutrients, light, and oxygen. Artificial lighting is offered by rows of LED grow lights, while plants are usually irrigated with recycled water by spraying the exposed hanging roots, suspended, from the crops.

14.2.3.4 Challenges in Applying IoT in Agriculture

One of the biggest challenges of applying IoT in agriculture is that the devices are exposed to harsh environmental conditions. Factors like extreme temperatures with a high variation, high humidity and intensive rainfall, strong wind, solar radiation, machine operation, and animal movement causing displacement and vibration are introducing problems to the proper functionality of the sensors and the electronic circuits. Since most of the IoT applications are based on wireless communication (e.g., WSN), the batteries that are used for power consumption offer a limited flexibility in functioning without surveillance for a longer period (Jawad et al. 2017). Except device-related problems, the harsh conditions are responsible for network

issues as well. A low wireless link quality affects the transceivers and the quality of the transmitted data (Villa-Henriksen et al. 2020), while the calculation of the signal strength in that case can be highly beneficial in assessing signal quality (Reiser et al. 2017). Another component has to do with security challenges. Authentication, confidentiality, and data privacy need to be secured against all possible external threats providing that only authorized users will have access to the collected data (Tzounis et al. 2017).

14.3 Cloud Computing

Cloud computing is on-demand computing services (e.g., databases, storage, networking, servers, software, analytics, etc.) over the Internet where the provider charges the user based on usage. The creation of virtual machines, by leveraging a distributed system consisting of a collection of interconnected and virtualized computers, enables the user to utilize elastic resources based on their current needs. The shift to cloud computing significantly reduces the cost of buying new hardware and software but also increases speed as IT resources can be provisioned in some minutes. Furthermore, factors as the high level of performance and reliability, especially when connected with IoT (Stergiou et al. 2018), contribute to the wide expansion of cloud computing technologies.

14.3.1 Cloud Services

The three main types of cloud computing service models are:

- Infrastructure as a service (IaaS): a vendor provides users access to preconfigured computing resources such as storage and networking, servers, and virtual machines. Popular IaaS offerings are Amazon EC2, IBM SoftLayer, Microsoft Azure VM, and Google Compute Engine (GCE).
- Platform as a service (PaaS): the cloud is used to deliver an on-demand environment to users for developing, testing, delivering, and managing software applications. Widely used PaaS products are Google App Engine, IBM Bluemix, and Apache Stratos.
- Software as a service (SaaS): it is a method for delivering software applications over the Internet, on-demand and typically on a subscription basis. SaaS offerings are the most widely visible of all the cloud computing service models. Two of the most popular SaaS applications include Microsoft Office 365 and Adobe Creative Cloud.

14.3.2 Emerging Architectures of Cloud Computing

New cloud computing technologies are emerging due to the development of large-scale applications. They move closer to a virtualized infrastructure and deal with factors such as scalability, flexibility, and privacy. In a review by Varghese and Buyya (2018), four new computing models are identified:

- Volunteer computing: Public participants share their idle computing resources to create an ad hoc cloud. It is expected to have an important implementation on projects with a societal or scientific focus.
- Fog and edge computing: Computational resources on edge nodes are being leveraged. This technology is strongly connected with the use of IoT as it was described in the previous section.
- Serverless computing: This doesn't mean there aren't any servers, but instead the applications are being executed only when it is necessary and not all the time. Function as a service (FaaS) is a form of serverless computing.
- Software-defined computing: Using virtualization technologies, the infrastructure can be broken up into resources that can be allocated on demand. This applies not only to networking but also to computation and storage.

14.3.3 Cloud Computing Implementation in Agriculture

One of the most important implementations of cloud computing in agriculture is the farm management systems (FMSs). The use of FMSs has widely expanded the last years which nowadays are regarded as important tools for managing the agricultural business and for implementing precision agriculture principles (Fountas et al. 2015). In order for a farmer to receive valuable information, all details related to the performed agricultural operations should be carefully recorded and imported into the FMS. This is why the combination of a cloud-based FMS with IoT sensor data is very promising for future systems. Furthermore, a cloud-based FMS can be interconnected with open agriculture-related databases (e.g., weather forecasting) or even an interface for receiving online agriculture consultation from advisors (Symeonaki et al. 2017).

As the level of communication of the infield autonomous systems with the cloud-based infrastructure is expected to increase in the next years, issues related to operational safety and resilience will need to be addressed. Challenging aspects, including unplanned cloud failures or hardware malfunctions that are highly possible to occur due to various reasons such as natural disasters or even targeted attacks, should be taken under consideration. Another important topic that could hinder farmers to adopt cloud computing tools is that reliable and broadband Internet access in rural areas, which are less densely populated, has not yet been achieved resulting in a digital divide compared to urban areas.

14.3.4 The Future Internet Ecosystem

An important example of innovative ICT tools that have emerged in the last years based on cloud computing is the Future Internet Public-Private Partnership Program (FI-PPP 2011), which was launched by the European Commission in 2011. The overarching aim of the FI-PPP is to create a library of software components that are called Generic Enablers (GEs). The GEs should be public and open-source and allow developers to create mash-up applications by implementing innovative FI functionalities such as IoT connectivity and big data analytics. All GEs are developed and described in detail as a set of application programming interfaces (APIs) in the FIWARE platform (FIWARE 2020). The FIWARE architectural chapters are provided in Fig. 14.3. The usability of FI technologies in the context of environmental applications has been thoroughly examined by Granell et al. (2016), while Paraforos et al. (2016) described an FMS architecture that utilizes advanced FI characteristics.

14.3.5 Fog and Edge Computing

One of the emerging technologies that were mentioned above, which is expected to have a substantial impact in agriculture, is fog and edge computing, where cloud services are extended to the edge of the network to decrease the latency and network congestion (Ferrández-Pastor et al. 2018; O'Grady et al. 2019). Both these technologies offer similar functionalities, and their main purpose is pushing both data and computational intelligence to platforms with data analytic capabilities that are located either on the IoT device or near to the source of origination of the data. The term fog computing was first used by Bonomi et al. (2012), while its main feature is to extend cloud computing services at the edge of the network (Moysiadis et al. 2018). The primary difference between fog and edge computing is the location where data processing occurs. For edge computing, the computation takes place

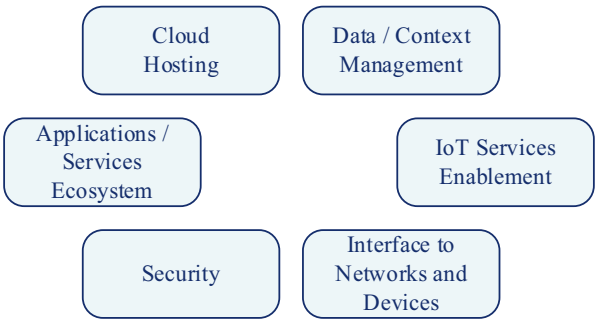


Fig. 14.3 Architectural chapters of the FIWARE platform

mainly on the devices with the IoT-connected sensors, while fog computing exists between the two layers (i.e., edge and cloud computing) and can be distributed in different locations serving a higher number of edge devices.

14.4 Big Data

Agriculture-related data are approaching the dimension of the 3 + 1 "Vs" that characterize big data: Volume, Velocity, Variety, and Veracity (Zhang et al. 2018) – Volume due to the produced amount of terabytes of data; Velocity for the increased pace that these data are becoming available to the user; Variety due to the heterogeneity of data sources, e.g., combination of structured and (semi-)structured data; and Veracity that deals with the quality and validity of the data (Lokers et al. 2016). Other studies add two more Vs in the definition: Visualization for facilitating the human interpretation of analyzed data and Visibility on efficiently processing geospatial data based on cloud computing technologies (Li et al. 2016). Big data analytics on the combination of machine data sets, with the sensor and unmanned aerial vehicle (UAV) data, could reveal hidden patterns, correlations, and other insights that are not detectable when using conventional methods for data analysis (Karmas et al. 2016).

14.4.1 Agricultural Geospatial Big Data

Geospatial data have the form of (a) raster data (images, 3D objects), (b) vector data (points, lines, polygons), and (c) graph data (nodes, edges paths). The term "geo" implies that the data correspond to a global coordinate reference system. Geospatial data have always been big data due to the vast amount of location-specific data that is being generated every day (Lee and Kang 2015). Sources of spatial data are airborne data coming from drones or satellites but also infield sensors such as IoT or mobile devices, cameras, etc. A necessary tool to store, integrate, analyze, and present geospatial data is a geographical information system (GIS).

One important source of agriculture-related data is the agricultural machines' subsystems. Different sensors and electronic control units (ECUs) that are installed on the tractors or agricultural implements produce data and communicate through control area network (CAN) bus by utilizing the ISO 11783 (commonly designated as ISOBUS) and J1939 communication protocols. Although the use of these data is intended for the correct operation as well as for the real-time inter-machine communication (Kortenbruck et al. 2017), the analysis of these data, when combined with positioning information from a global navigation satellite system (GNSS), could reveal valuable information about the performed agricultural operations but also the cultivated crop (Paraforos et al. 2017b). Details regarding the analytics of these data will be presented at one of the following sections.

14.4.2 Big Data Technologies

Conventional data analysis tools and techniques are not sophisticated enough to cope with big data and handle the vast and complex amount of data streaming from various sources. Thus, new technologies have been developed in the last years that are specialized in big data analytics by leveraging cloud-based resources. A characteristic example of an architecture that deals with geospatial big data is being developed in the frame of the BigGIS project (BigGIS 2020). This architecture (Fig. 14.4) includes most of the widely used toolboxes in big data analytics. The data sources vary and span from sensor data up to citizen sensing. In the specific architecture, StreamPipes works as an IoT platform which allows to integrate, process, and analyze big data streams by minimizing at the same time the programming effort as it uses graphical modeling. Although Apache Flink is specialized in analyzing IoT sensor data, Apache Spark featuring GeoTrellis could be incorporated when processing geographical data (e.g., raster or vector data). An important tool in the big data analysis layer is the concept of deep learning. The latter uses supervised and unsupervised techniques for data classification and data extrapolation (forecasting). Two common deep architectures of deep learning that are being implemented in big data analytics are the deep belief networks (DBF) and the convolutional neural networks (CNN) (Jan et al. 2017).

In order to handle the communication between the data processing elements, i.e., nodes, within the analytics pipelines, the Apache Kafka and the ActiveMQ could be utilized as message brokers. The Apache Hadoop software library and Exasol, CouchDB, and RDF4J are some of the offered storage back ends for the distributed processing of large data sets. The middleware is also responsible for the connection

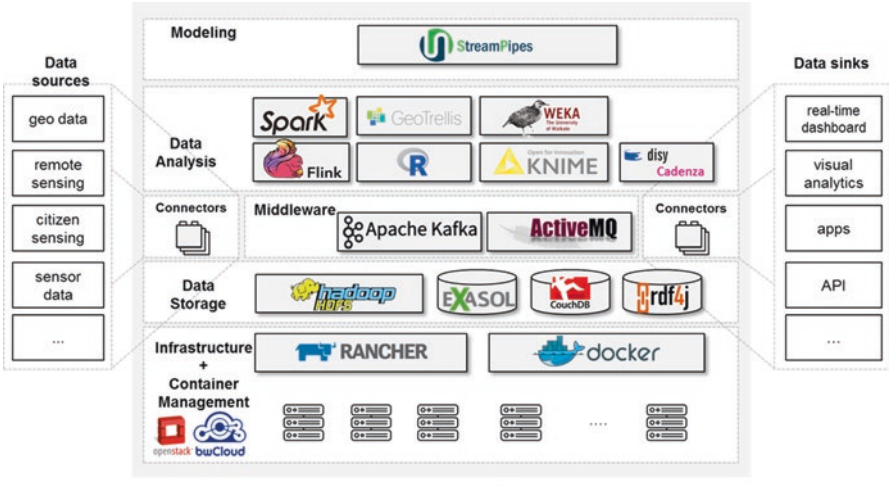


Fig. 14.4 Overview of offered big data technologies (Adapted from Abecker and Kutterer (2018), with permission)

with every possible data sink. The most important is to offer an API (application programming interface) to allow third-party software to communicate with the developed software ecosystem. The "Container Management" solutions like Rancher/Kubernetes support the whole software production chain, from software development up to testing and deployment in order to make the use of a container tool as efficient and effective as possible. Finally, the entire infrastructure is based on cloud computing technologies such as the Amazon EC2 (bwCloud in the specific example of Fig. 14.4).

14.4.3 Data Privacy and Ownership

Although the gradual shift to big data analytics and smart farming appears as an economic opportunity, it also raises important questions related to data privacy (Whitacre et al. 2014) and the balance that should exist between private and public open data as this is the core issue of the adoption decision for many hesitant farmers. Such questions are:

- Who owns the data generated on and around the farm?
- Who has control over the data?
- Who has access to the data?
- Who is entitled to the value of the data?

Mostly, "the farmer" is the answer to the first question. It is the farmer who decides (and thus gives permission) to share or sell his/her data. Still, it is not always so clear. What if the activity in which the data is generated is performed by an external person or third-party software? Does the application of particular software mean that a service provider becomes a data owner? There are actors who create data (farmers), those who collect (brokers) and those who analyze (analysts). Currently, the last ones shape the rules deciding how the data will be used and who is provided access, but farmers should be assured that their knowledge and decision-making capacities will not be replaced by algorithms, as they should always be the ones taking the last decision.

14.4.4 Open Agricultural Data

In the discussion regarding data related to agriculture and nutrition, open data concept plays an important role; that is data that anyone can access, use, or share to shape solutions by enabling more efficient and effective decision-making. According to Kunisch (2016), big data technologies offer new opportunities in giving answers to complex issues; thus, a public-private partnership framework, in terms of open source and open access, would be an appropriate platform for agricultural production and research data storage and exchange. This is also witnessed by open data

initiatives, such as the global open data for agriculture and nutrition (GODAN), which aims to make agricultural and nutritionally relevant data available, accessible, and usable for unrestricted use worldwide. By leveraging open agriculture-related data, the outcome of big data analytics, except the high importance for the farmer, by providing decision support on sustainable land management, would also be beneficial for farmer advisors, public authorities, and policy-makers (Carolan 2015) toward making decisions for giving solutions to societal challenges.

14.5 Digital Farming

Although the technical capabilities of precision agriculture are already well developed, the precision and the efficiency of the application could be further enhanced shifting toward smart agriculture adapted to the new digital era. The greatest potential lies in leveraging multisource data and all previously described technologies to enhance already existing agronomical algorithms. The overarching aim is to minimize yield gaps in order to allow farms to be more efficient but also more profitable, safe, and environmentally friendly.

14.5.1 Automated Robotic Farming

One of the main components of digital farming is the utilization of autonomous agricultural vehicles. This component is responsible for closing the loop that starts with infield sensing, continues with cloud-based data analysis, and ends up again in the field where the necessary actions need to be taken with the highest possible accuracy. Recent technologies that are discussed in this chapter, in combination with artificial intelligence progress, will lead to the new agricultural era (Saiz-Rubio and Rovira-Más 2020). Nowadays field robotics are dominating agriculture performing all kinds of agricultural operations with a proliferating number of commercial products (e.g., Naïo Technologies, Saga Robotics, Robotti Agrointelli) but also research efforts such as weed control (Wu et al. 2020), apple harvesting (Silwal et al. 2017), and robot-human collaboration (Vasconez et al. 2019), just to name a few. A future trend is to use collaborative and cooperative behavior in a fleet of robots that will offer the opportunity to spread tasks over multiple platforms. This will reduce the damage caused by heavy conventional agricultural platforms on the soil or existing crops (Duckett et al. 2018).

14.5.2 Increased Accuracy

As digital farming will be using in the future sophisticated technologies such as robots and autonomous vehicles for implementing precision agriculture principles, high infield position accuracy is at a high priority. Innovative instrumentation should be considered toward reaching this higher level of accuracy. A device offering an accuracy at the millimeter level is a total station (TS). Commonly, this device is used in the domain of civil engineering and provides a higher accuracy compared to satellite-based positioning systems. Paraforos et al. (2017a) utilized a highly accurate industrial robotic manipulator to examine if a robotic TS can offer millimeter accuracy for demanding agricultural operations. The obtained results validated that this device would play an important role in the future in agricultural operations related to individual plant treatment. Afterward, the TS was used to examine the seeding depth for each individual seed in cereal no-till sowing (Sharipov et al. 2017, 2018) but also to produce a three-dimensional reconstruction of maize crop plants (Vázquez-Arellano et al. 2018a, b).

14.5.3 Automated Farm Management Systems

A common problem in farm management is that the agricultural tasks are not recorded properly; additionally, a farmer often neglects to gather all necessary data and import them into an FMS (Paraforos et al. 2016). A solution that appears promising is to utilize agricultural machinery communication data (Paraforos et al. 2019). The connection of ISOBUS with an FMS has been described in detail in Part 10 of the standard (ISO 2015). Software architecture was developed by Paraforos et al. (2017c) to automate a commercial FMS, named ifarma (Agrostis 2017). The data flow of this architecture is presented in Fig. 14.5. Machine data from ISOBUS communication were collected, analyzed, and aggregated into agricultural tasks. A stand-alone application was developed using MATLAB and was installed at the remote cloud-based server. The ISOBUS service was calling the MATLAB App by passing the acquired ISOBUS data. Initially, in this App, the data were filtered using the information from the rear hitch positions (SPN 1873) and the ground-based machine speed (SPN 1859), to extract only the infield data from the complete data set. The ISOBUS service generated the performed tasks with all related data, and this information was forwarded to the ifarma service that was responsible for storing this information in the FMS database and for presenting it to the user using the App's graphical user interface.

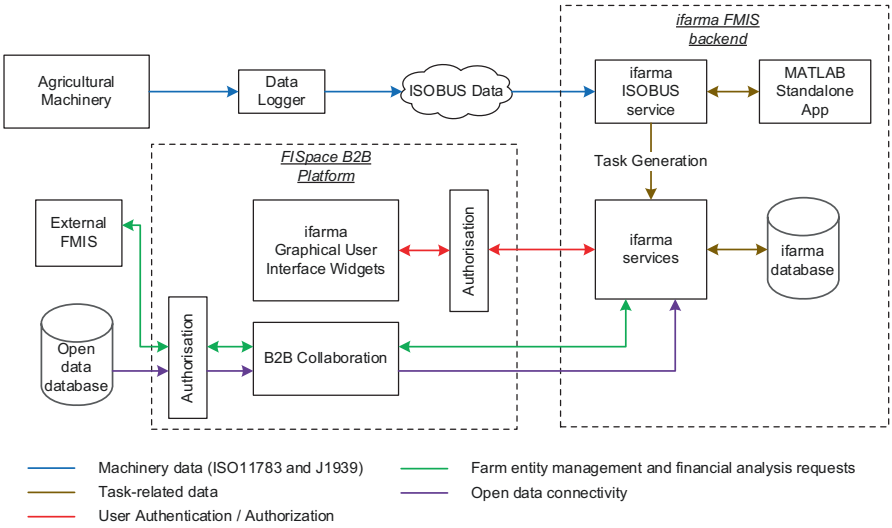


Fig. 14.5 Architecture of information flow in an automated FMS (Paraforos et al. 2017c)

14.5.4 Data-Driven Digital Agriculture

Figure 14.6 presents a scientific-technological approach and the software architecture for data-driven agricultural applications where all previously described technologies are utilized. The architecture is based on the fusion, management, and intelligent analysis of manifold kinds of agricultural data based on IoT technologies (and recently IoS – Internet of Services), especially (i) data from agricultural machinery, connected through the ISOBUS; (ii) open data as provided, for instance, by public administrations or other providers of open data for agriculture (like KTBL in Germany); (iii) sensor data originating from WSN in the field; and (iv) remote-sensing data generated by UAVs and satellites.

Such domain- and goal-specific analysis, reporting, and visualization functions will produce (a) precise inputs for innovative algorithms to support daily as well as strategic agro-technical and agro-economic decisions, as well as (b) the prerequisite for innovative agricultural data products and data services. Such data products can become the starting point for an agricultural data economy, for instance, helping agricultural advisors to improve their services or helping farm inputs suppliers to improve their products and to optimize their marketing and logistics processes. As a "side-product," the overall data infrastructure can also facilitate all reporting and data communication processes between farmers and public authorities or even from farmers to the general public ("Agricultural Open Data"), thus increasing farm management efficiency and supporting policy-making, nature protection, food production transparency, etc. This leads to a holistic FMS which, based on formerly underexploited data sources, data connectivity, and intelligent analyses, can deliver completely new levels of insights and decision support regarding farm operations

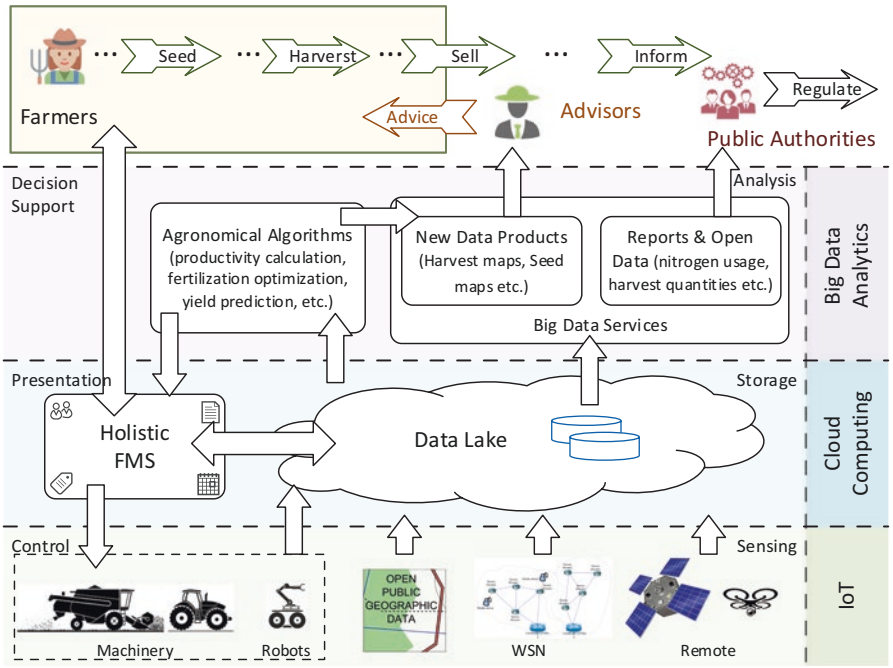


Fig. 14.6 Structural ecosystem for data-driven agricultural applications (Courtesy: Wassilios Kazakos, disy Informationssysteme GmbH, modified)

and management, especially with the ability of better spatially enabled and longer-term analyses.

14.6 Summary and Concluding Thoughts

It becomes clear that the farm of the future will be highly connected. Various sensors installed in the field but also agricultural machinery performing operations will constantly record, process, and transmit information to the cloud-based infrastructure for storage and in-depth analysis. Actuators and autonomous vehicles will be responsible for implementing the farmer's strategy, which will be supported by the results of the aforementioned analysis. The entire architecture will be based on a multilevel automation ecosystem, starting from simple closed-loop systems, i.e., irrigation, up to more complex systems with a higher level of cognition such as machine coordination.

A cutting-edge future technology that is expected to have a profound impact on agriculture in the next years is the concept of digital twins. The latter are already becoming available in other scientific disciplines, such as automotive and industrial informatics (Schluse et al. 2018). The digital twins are virtual, digital equivalents to

physical objects that provide a thorough representation of the object and the context that this object is working in. The main focus of the digital twins is to combine IoT sensor data with historical data and human expertise, by utilizing machine learning techniques to improve the outcome of prognostics. This technology could provide decision support to farmers and other stakeholders by enabling them to act immediately and efficiently in the presence of a predicted deviation.

The wide implementation of Digital Farming faces many challenges like security in digital transactions. Blockchain, which is the distributed ledger technology behind many cryptocurrencies, promises smallholder farmers' highly secure access to digital technologies. Other important challenging issues that were discussed related to operational safety of cloud computing technologies should be taken under consideration when designing a digital farming system. Consequently, resilience should be incorporated into the proposed data-driven agricultural ecosystem by developing decentralized systems that leverage multi-cloud and multi-region architectures (Varghese and Buyya 2018). Although this requires significant use of human and financial resources, it will decrease the overall vulnerability of the system.

Disclaimer

Mention of a commercial product is solely for the purpose of providing specific information and should not be construed as a product endorsement by the authors or the institution with which the authors are affiliated.

References

Abecker A, Kutterer J (2018) Geodaten für prädiktive und präskriptive Analysen—Ergebnisse aus dem Projekt BigGIS. In: Czarnecki C (ed) Workshops Der Informatik 2018, lecture notes in informatics (LNI). Gesellschaft für Informatik, Bonn

Agrostis (2017) Integrated farm management application – Agrostis. Retrieved from https://ifarma.agrostis.gr/index_en.php

Atzori L, Iera A, Morabito G (2017) Understanding the internet of things: definition, potentials, and societal role of a fast evolving paradigm. Ad Hoc Netw 56:122–140

BigGIS – Scalable GIS for predictive and prescriptive analytics. Retrieved April 21, 2020, from http://biggis-project.eu/biggis-docs/

Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the internet of things. In: MCC'12 – Proceedings of the 1st ACM Mobile Cloud Computing Workshop. ACM Press, New York, pp 13–15

Borgia E (2014) The internet of things vision: key features, applications and open issues. Comput Commun 54:1–31

Carolan M (2015) Publicising food: big data, precision agriculture, and co-experimental techniques of addition. Sociol Rural 57(2):135–154

Dong X, Vuran MC, Irmak S (2013) Autonomous precision agriculture through integration of wireless underground sensor networks with center pivot irrigation systems. Ad Hoc Netw 11(7):1975–1987

Duckett T, Pearson S, Blackmore S, Grieve B, Chen WH, Cielniak G, Cleaversmith J, Dai J, Davis S, Fox C, From P, Georgilas I, Gill R, Gould I, Hanheide M, Hunter A, Iida F, Mihalyova

L, Nefti-Meziani S, Yang GZ (2018) Agricultural robotics: the future of robotic agriculture. Retrieved from https://arxiv.org/abs/1806.06762

Ferrández-Pastor F, García-Chamizo J, Nieto-Hidalgo M, Mora-Martínez J (2018) Precision agriculture design method using a distributed computing architecture on internet of things context. Sensors 18:1731

FI-PPP (2011) Future Internet Public-Private Partnership. Retrieved April 20, 2020, from https://www.fi-ppp.eu/

Fountas S, Carli G, Sørensen CG, Tsiropoulos Z, Cavalaris C, Vatsanidou A, Liakos B, Canavari M, Wiebensohn J, Tisserye B (2015) Farm management information systems: current situation and future perspectives. Comput Electron Agric 115:40–50

Fuller JR (2016) How to design an IoT-ready infrastructure: the 4-stage architecture. Retrieved September 11, 2018, from https://techbeacon.com/4-stages-iot-architecture

Granell C, Havlik D, Schade S, Sabeur Z, Delaney C, Pielorz J, Usländer T, Mazzetti P, Schleidt K, Kobernus M, Havlik F, Bodsberg NR, Berre A, Mon JL (2016) Future internet technologies for environmental applications. Environ Model Softw 78:1–15

Haladjian J, Haug J, Nüske S, Bruegge B, Haladjian J, Haug J, Nüske S, Bruegge B (2018) A wearable sensor system for lameness detection in dairy cattle. Multimodal Technol Interaction 2(2):27

ISO (2015) ISO 11783-10:2015 Tractors and machinery for agriculture and forestry—Serial control and communications data network—Part 10: task controller and management information system data interchange. Retrieved from https://www.iso.org/standard/61581.html

Jan B, Farman H, Khan M, Imran M, Islam IU, Ahmad A, Ali S, Jeon G (2017) Deep learning in big data analytics: a comparative study. Comput Electr Eng 75:275–297

Jawad H, Nordin R, Gharghan S, Jawad A, Ismail M (2017) Energy-efficient wireless sensor networks for precision agriculture: a review. Sensors 17:1781

Karmas A, Tzotsos A, Karantzalos K (2016) Geospatial big data for environmental and agricultural applications. In: Yu S, Guo S (eds) Big data concepts, theories, and applications. Springer, Cham, pp 353–390

Kortenbruck D, Griepentrog HW, Paraforos DS (2017) Machine operation profiles generated from ISO 11783 communication data. Comput Electron Agric 140:227–236

Kunisch M (2016) Big data in agriculture—perspectives for a service organisation. Landtechnik 71(1):1–3

Lee J-G, Kang M (2015) Geospatial big data: challenges and opportunities. Big Data Res 2:74–81

Li S, Dragicevic S, Castro FA, Sester M, Winter S, Coltekin A, Pettit C, Jiang B, Haworth J, Stein A, Cheng T (2016) Geospatial big data handling theory and methods: a review and research challenges. ISPRS J Photogramm Remote Sens 115:119–133

Lokers R, Knapen R, Janssen S, van Randen Y, Jansen J (2016) Analysis of big data technologies for use in agro-environmental science. Environ Model Softw 84:494–504

Moysiadis V, Sarigiannidis P, Moscholios I (2018) Towards distributed data management in fog computing. Wirel Commun Mob Comput 2018:1–14

O'Grady MJ, Langton D, O'Hare GMP (2019) Edge computing: a tractable model for smart agriculture? Artif Intell Agric 3:42–51

Paraforos DS, Vassiliadis V, Kortenbruck D, Stamkopoulos K, Ziogas V, Sapounas AA, Griepentrog HW (2016) A farm management information system using future internet technologies. IFAC-PapersOnLine 49:324–329

Paraforos DS, Reutemann M, Sharipov G, Werner R, Griepentrog HW (2017a) Total station data assessment using an industrial robotic arm for dynamic 3D in-field positioning with subcentimetre accuracy. Comput Electron Agric 136:166–175

Paraforos DS, Vassiliadis V, Kortenbruck D, Stamkopoulos K, Ziogas V, Sapounas AA, Griepentrog HW (2017b) Automating the process of importing data into an FMIS using information from tractor's CAN-bus communication. Adv Anim Biosci 8:650–655

Paraforos DS, Vassiliadis V, Kortenbruck D, Stamkopoulos K, Ziogas V, Sapounas AA, Griepentrog HW (2017c) Multi-level automation of farm management information systems. Comput Electron Agric 142:504–514

Paraforos DS, Sharipov GM, Griepentrog HW (2019) ISO 11783-compatible industrial sensor and control systems and related research: a review. Comput Electron Agric 163:104863

Popović T, Latinović N, Pešić A, Zečević Ž, Krstajić B, Djukanović S (2017) Architecting an IoT-enabled platform for precision agriculture and ecological monitoring: a case study. Comput Electron Agric 140:255–265

Porter JR, Xie L, Challinor AJ, Cochrane K, Howden SM, Iqbal MM, Lobell DB, Travasso MI. 2014. Food security and food production systems. In: Field CB, Barros VR, Dokken DJ, Mach KJ, Mastrandrea MD, Bilir TE, Chatterjee M, Ebi KL, Estrada YO, Genova RC, Girma B, Kissel ES, Levy AN, MacCracken S, Mastrandrea PR, White LL, (Eds.). Climate Change 2014: Impacts, Adaptation, and Vulnerability. Part A: Global and Sectoral Aspects. Contribution of Working Group II to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, pp. 485–533

Ray PP (2016) A survey of IoT cloud platforms. Future Comput Inf J 1(1–2):35–46

Reiser D, Paraforos DS, Khan MT, Griepentrog HW, Vázquez-Arellano M (2017) Autonomous field navigation, data acquisition and node location in wireless sensor networks. Precis Agric 18:279–292

Saiz-Rubio V, Rovira-Más F (2020) From smart farming towards agriculture 5.0: a review on crop data management. Agronomy 10:207

Schluse M, Priggemeyer M, Atorf L, Rossmann J (2018) Experimentable digital twins—streamlining simulation-based systems engineering for industry 4.0. IEEE Trans Ind Inf 14:1722–1731

Sharipov G, Paraforos DS, Pulatov A, Griepentrog HW (2017) Dynamic performance of a no-till seeding assembly. Biosyst Eng 158:64–75

Sharipov GM, Paraforos DS, Griepentrog HW (2018) Implementation of a magnetorheological damper on a no-till seeding assembly for optimising seeding depth. Comput Electron Agric 150:465–475

Silwal A, Davidson JR, Karkee M, Mo C, Zhang Q, Lewis K (2017) Design, integration, and field evaluation of a robotic apple harvester. J Field Rob 34:1140–1159

Sreekantha DK, Kavya AM (2017) Agricultural crop monitoring using IOT – a study. In: 2017 11th International Conference on Intelligent Systems and Control (ISCO). IEEE, Coimbatore, pp 134–139

Stergiou C, Psannis KE, Kim B-G, Gupta B (2018) Secure integration of IoT and cloud computing. Futur Gener Comput Syst 78:964–975

Strube G (1998) Modelling motivation and action control in cognitive systems. In: Schmid U, Krems J, Wysocki F (eds) Mind modelling. Pabst, Berlin, pp 89–108

Symeonaki E, Arvanitis K, Piromalis D (2017) Review on the trends and challenges of cloud computing technology in climate—smart agriculture. CEUR Work Proc 2030:66–78

Talavera JM, Tobón LE, Gómez JA, Culman MA, Aranda JM, Parra DT, Quiroz LA, Hoyos A, Garreta LE (2017) Review of IoT applications in agro-industrial and environmental fields. Comput Electron Agric 142:283–297

The open source platform for our smart digital future – FIWARE. Retrieved April 20, 2020, from https://www.fiware.org/

Tzounis A, Katsoulas N, Bartzanas T, Kittas C (2017) Internet of things in agriculture, recent advances and future challenges. Biosyst Eng 164:31–48

United Nations (2015) Sustainable development knowledge platform. Retrieved April 20, 2020, from https://sustainabledevelopment.un.org/

Vannieuwenborg F, Verbrugge S, Colle D (2017) Designing and evaluating a smart cow monitoring system from a techno-economic perspective. In: 2017 Internet of Things Business Models, Users, and Networks, Copenhagen, Denmark, 2017, pp. 1-8, https://doi.org/10.1109/CTTE.2017.8260982

Varghese B, Buyya R (2018) Next generation cloud computing: new trends and research directions. Futur Gener Comput Syst 79:849–861

Vasconez JP, Kantor GA, Auat Cheein FA (2019) Human–robot interaction in agriculture: a survey and current challenges. Biosyst Eng 179:35–48

Vázquez-Arellano M, Paraforos DS, Reiser D, Garrido-Izard M, Griepentrog HW (2018a) Determination of stem position and height of reconstructed maize plants using a time-of-flight camera. Comput Electron Agric 154:276–288

Vázquez-Arellano M, Reiser D, Paraforos DS, Garrido-Izard M, Burce MEC, Griepentrog HW (2018b) 3-D reconstruction of maize plants using a time-of-flight camera. Comput Electron Agric 145:235–247

Villa-Henriksen A, Edwards GTC, Pesonen LA, Green O, Sørensen CAG (2020) Internet of things in arable farming: implementation, applications, challenges and potential. Biosyst Eng 191:60–84

Whitacre BE, Mark TB, Griffin TW (2014) How connected are our farms? Choices 29(3):1

Wu X, Aravecchia S, Lottes P, Stachniss C, Pradalier C (2020) Robotic weed control using automated weed and crop classification. J Field Rob 37:322–340

Zhang Q, Yang LT, Chen Z, Li P (2018) A survey on deep learning for big data. Inf Fusion 42:146–157

Chapter 15
Human-Machine Interactions

Danny Mann

15.1 Introduction

Why focus on ergonomics in a book devoted to robotics (or automation) in agriculture? The question can be answered in two ways. First of all, automation is likely to occur in stages. We have already seen this to be the case. Minor modifications are made to a machine that automates a specific function. A combine header might be equipped with sensors that enable the raising and lowering of the header to be automated. A tractor can be equipped with auto-steer technology so that the operator is not required to steer during parallel passes across the field. In both of these examples, the operator still retains responsibility for numerous other tasks. Therefore, the automated subsystems must be integrated with the remaining manual tasks. The second answer to the question is that ergonomics will continue to play an important role even when the entire machine becomes autonomous. When automation was introduced in the manufacturing sector, it was first envisioned that humans would be displaced from the manufacturing process. This did not occur. Although the assembly jobs may have been displaced, different roles were created as human operators were required to supervise the autonomous machines. Effective supervision requires a thorough understanding of the processes being supervised – an understanding that can only be achieved by ongoing and timely access to information from the autonomous machine. Thus, the effective functioning of a system involving a human supervisor and an autonomous machine will rely on a well-designed automation interface.

This chapter will cover four distinct topics. In the first section, you can expect to gain an understanding of human-machine interaction associated with agricultural machines. This will be followed by a discussion of the tools for assessing

D. Mann (✉)
University of Manitoba, Winnipeg, MB, Canada
e-mail: Danny.Mann@umanitoba.ca

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_15

human-machine interaction. The third section will discuss the progression of technologies that have been used to support the operator of an agricultural machine up to and including fully autonomous agricultural machines. In the final section, future challenges associated with remote supervision of autonomous agricultural machines will be discussed.

15.2 Human-Machine Interaction for Agricultural Machines

15.2.1 Ergonomics Defined

Meister (1971) defined ergonomics as "the application of behavioral principles and data to engineering design to: (i) maximize an individual's contribution to the effectiveness of the system of which he/she is a part and (ii) reduce the impact of that system on the individual." When we hear the word ergonomics, we might immediately think of concepts such as comfort (i.e., an "ergonomic" office chair is one that supports proper posture and keeps us comfortable throughout the workday). This aspect of ergonomics is certainly covered by Meister's definition through the second objective to "reduce the impact of that system on the individual." However, we must not forget to consider the first objective. There are at least three important facts associated with the objective to "maximize an individual's contribution to the effectiveness of the system of which he/she is a part." First, the individual and the machine together comprise a system; neither can function alone. Second, the effectiveness of the system will be influenced by the operator's interaction with the machine. Third, the design engineer's task is to design the machine such that the overall effectiveness of the human-machine system can be maximized. The engineer should always consider ergonomics when designing any machine or system; failure to adequately consider ergonomic principles will undoubtedly result in a less-than-optimal solution.

To explain the concept of ergonomics to engineering students, I employ a basic diagram that depicts two-way flow of information between a machine and its human operator (Fig. 15.1). The human operator uses "controls" to communicate instructions to a machine, and the machine has been designed with "displays" that

Fig. 15.1 The study of ergonomics is all about the transmission of information that must occur between a machine and its user. Input information is provided to the machine using "controls." Feedback from the machine is presented to the user through "displays"

communicate status information back to the human operator. Ergonomics is all about maximizing the efficiency of these two channels of communication through the physical design of the controls and displays and through understanding how the external environment influences the transmission of information. For the past several decades, farmers have interacted (or communicated) with their agricultural machines from an operator station mounted on the machine. Using an analogy from the world of entertainment, this has provided the operator with a front-row seat to everything that is happening with the machine. Engineers pay careful attention to the ergonomics of the control panel, and displays are designed to direct the operator to the most pertinent information related to the function of the machine (or in some cases to capture the operator's attention when a problem has been detected). There is little doubt that modern agricultural machines are better than their predecessors from previous decades in terms of the human-machine interactions that are facilitated. Dooley (2012) provided a thorough overview of ergonomics applied to agricultural vehicles. Design engineers should be proud of these achievements.

15.2.2 Designing to Support Flow of Information to the Machine

15.2.2.1 General Principles Guiding Control Panel Design

A primary challenge of the engineer is to design a control panel that enables the operator to effectively communicate with the machine through various control actions. Design of controls is a typical topic in many ergonomics textbooks, and it is not my intention to attempt a summary of the numerous concepts that are explained in those books. Rather, I will provide some specific examples that relate to agricultural machines to demonstrate the application of some of these important ergonomic design principles.

Complex agricultural machines have many controls which must be activated at various times according to the function being performed. Often they must be activated according to a particular sequence. In any given arrangement of controls, some will be used more often than others; some will be more important than others. Thus, designers can rely on four principles when arranging controls:

(i) Importance Principle: important controls should be placed in convenient locations.
(ii) Frequency-of-Use Principle: frequently used controls should be placed in convenient locations.
(iii) Functional Principle: controls should be grouped together according to their function.
(iv) Sequence-of-Use Principle: controls should be arranged to take advantage of the sequences of operation that take place during a task.

To effectively use these four design principles, it is necessary to have a good understanding of the tasks being controlled so that the characteristics of *importance*, *frequency of use*, *function*, and *sequence of use* can be correctly identified. It is relatively easy to describe the function of each control based on the components of the machine. Thus, the designer can readily group controls according to similar function (assuming that there are two or more controls having similar function). Understanding characteristics such as importance of individual controls, frequency of use of controls, and sequence of use of controls requires a thorough understanding of how the machine is used by a typical worker in a typical work environment. Task analysis, which will be introduced in an upcoming section, is an appropriate tool to gain an understanding of how a machine is used by a worker for the purpose of classifying controls according to importance, frequency of use, function, and sequence of use.

Sanders and McCormick (1993) have described the concept of a "focus attention task" to define the aspects that are *important* for a design. Driving a tractor can be described as a focus attention task where one must keep focus on one or a few select channels of information while not being distracted by other channels. In the case of a system consisting of a tractor and an air seeder, the channels of critical information are likely to change based on the position of the machine in the field. There are two distinct periods experienced by the operator. The first can be described as monitoring the air seeder attached behind the tractor during the parallel passes across the field. During this period, the operator's focus should be the monitoring of the air seeder to optimize air seeder efficiency. There are numerous parameters to monitor and control, although the timeliness of control adjustments is usually not critical (i.e., optimum efficiency may not be achieved, but there is no imminent danger to the machine). The second distinct period occurs during the headland turn at the end of each pass. During this short period, the operator's focus is solely on turning the machine for the next pass down the field. There is a critical sequence of actions that must occur within a short period of time to ensure that the machine is turned safely. From this discussion, it can be concluded that controls used to adjust the various functions of the air seeder are likely to be the most important controls during the periods when the air seeder is travelling on the parallel passes across the field. Similarly, the controls needed to enable the headland turn will be the most important controls during the periods when headland turns are being completed. This example emphasizes the fact that the most important controls might change based on the scenario being considered. Furthermore, it can be hypothesized that sequence of use of controls may be more critical during headland turns than during the actual seeding periods. No research has been found to support this hypothesis regarding sequence of use of controls.

A review of the literature identified one study in which the frequency of use of controls in agricultural machines was addressed. Drakopoulos and Mann (2006) described results of a survey of 10 experienced tractor operators in Greece. These individuals were given five categories of controls and asked to rank them in order from most frequently used controls to least frequently used controls. As expected, the steering wheel was identified as the most frequently used control on a tractor.

The next most frequently used controls were those used to control the functioning of the implement hitched to the tractor. Then, in decreasing frequency of use were the controls used to control the tractor itself divided into three groups: (i) controls related to the motion of the tractor, (ii) controls related to the internal environment, and (iii) controls related to the external environment. Although these results seem intuitive, they are based solely on the opinions of a relatively small number of experienced farmers. Ideally, these results should be confirmed through observation. Mastorakos (2012) completed ride-alongs with experienced air seeder operators. There were eight types of controls used by the experienced farmers during seeding operations (not including the steering wheel). Four of the controls can be categorized as relating to the motion of the tractor (i.e., throttle, transmission, auto-steer, and differential lock). The remaining four controls can be categorized as relating to the functioning of the implement hitched behind the tractor (i.e., air seeder lift, air seeder monitor, air seeder engage, and air seeder fan). The "air seeder lift" control was the most frequently activated control, averaging approximately 40 activations in the 1-h duration of the ride-alongs. Collectively, the controls related to the functioning of the air seeder were activated an average of 74.2 times during the 1-h duration of the ride-along compared with 21 activations for the controls associated with the motion of the tractor. This amounts to approximately 78% of all control activations. These results confirm the survey results of Drakopoulos and Mann (2006). In terms of frequency of use, controls used to adjust the machine hitched to the tractor are used more frequently than controls used to adjust the motion of the tractor during field operations.

15.2.2.2 Mathematical Models for Quantifying a Control Panel

Efficient operation of any machine depends upon the design of its control panel. Agricultural machines are no exception. The need for consideration of ergonomic principles in the design of control panels for agricultural machines is becoming increasingly important as the complexity of agricultural machines continues to increase. There are many factors that should be considered in the design of controls. Drakopoulos and Mann (2007) provided a review of the controls present in a tractor workstation and reported critical design dimensions for rotary switches, toggle switches, rocker switches, knobs, push buttons, hand levers, and steering wheels. For the most part, the ergonomic knowledge summarized by Drakopoulos and Mann (2007) relates only to physical characteristics of individual controls (i.e., length, width, height, diameter, separation distance, activation force).

Despite the importance of understanding how to ergonomically design individual controls, it is perhaps of greater importance to understand how the various pieces fit together. In other words, it is not sufficient that individual controls be ergonomically correct – the entire assembly of controls needs to be organized and grouped properly to enable the control panel to be used efficiently. Banks and Boone (1981) presented an "index of accessibility" that can be used to compare the layout of control panels based on two characteristics related to accessibility: (i) the distance

to the control and (ii) the ranked frequency of use of that control. In essence, the "index of accessibility" yields a high score when frequently used controls are placed closer to the operator than when those same controls are placed further from the operator. The "index of accessibility" is a useful tool for the design engineer when considering various options for the layout of a control panel, but it must be remembered that this model only accounts for frequency of use and proximity to the operator.

Drakopoulos and Mann (2008) proposed an alternate mathematical equation for quantifying control functionality in agricultural tractors. The model accounts for physical characteristics of controls, the relative frequency of use of controls, and four attributes of workstation design (i.e., placement of controls, suitability of controls, functional reach, and labeling of controls). Drakopoulos and Mann (2008) tested the model by comparing six tractors with manufacturing dates between 2003 and 2005 against six tractors with manufacturing dates between 1975 and 1981; the average score was 0.63 for the six modern tractors (with a value of 1.0 considered to be an optimal control panel arrangement based on functionality) compared with an average score of 0.19 for the late 1970s tractors. The so-called index of functionality proposed by Drakopoulos and Mann (2008) was capable of detecting ergonomic improvements that have occurred in agricultural tractors over the past three decades.

15.2.3 Designing to Support Flow of Information to the Operator

The other important challenge for the engineer is to design appropriate displays to enable the flow of information from the machine to the operator. The first task, of course, is to decide what information will be needed by the operator. After deciding what information to display, the next challenge is deciding how it should be displayed for the operator. Design of displays is a typical topic in most ergonomics textbooks; therefore, it is not my intention to provide a thorough review of the information that is typically presented in these books. Rather, I will highlight several specific examples that relate to agricultural machines.

There are many unique ways to design pictorials for a display on an agricultural machine. Three example displays are shown below (Fig. 15.2), all designed to provide information on the functional components of an air seeder. How do we know which pictorials are the most appropriate to use?

Rakhra (2018) completed a doctoral dissertation in which he investigated the design of a user interface for an air seeder. The dissertation included an in-depth review of various human factors principles that have been proposed for design of displays. Informed by these various principles, several pictorials were designed for each air seeder parameter (Rakhra and Mann 2018). Evaluation of the pictorials was

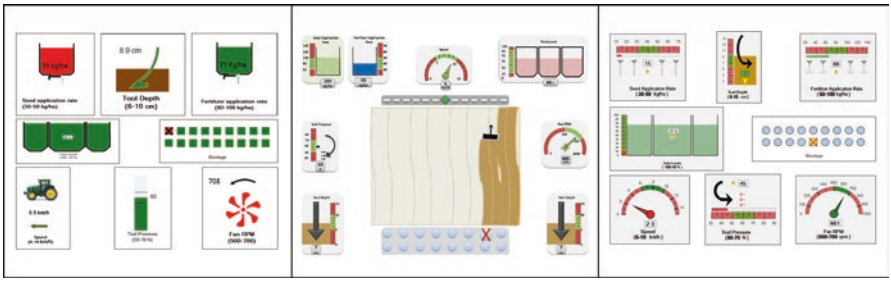


Fig. 15.2 Three unique designs of an interface to display information important to the monitoring and control of an air seeder. (Adapted from Rakhra and Mann 2018)

completed using the metrics of situation awareness and mental workload. Rakhra and Mann (2018) reported statistically significant improvements in operator situation awareness, and significantly lower mental workload, for most of the pictorials designed from a user-centered perspective. It is important to consider the user when designing displays for an agricultural machine – it is particularly useful to incorporate elements that support an appropriate level of situation awareness in the operator. Situation awareness will be discussed in depth in an upcoming section.

15.3 Tools for Assessing Human-Machine Interaction

15.3.1 Measuring the Unmeasurable

Engineers are typically very good at measuring things. If we want to know the engine temperature on a tractor, we install an appropriate sensor that can be calibrated to give us the appropriate temperature measurement to whatever precision is required. The data that are generated can be used to assess the performance of some particular aspect of a design. Now let's consider the control panel or an information console of an agricultural machine. How do we determine whether the control panel or information console that we have designed enables the operator to effectively interact with the machine? The answer is not so obvious. Because we are focused on the flow of information between the operator and the machine, there are no obvious physical parameters that can be measured. This section of the chapter will describe several tools that can be used by the engineer to assess the adequacy of the human-machine interaction from an ergonomic perspective. The intent is not to provide a thorough academic analysis of these tools. Rather, an overview will be provided that discusses the potential benefits of each tool, how they can be used with reference to understanding human-machine interactions for agricultural machines, and problems that have been encountered when attempting to use these tools.

15.3.2 Task Analysis

Task analysis is a fundamental tool for any engineer concerned with the interaction between an operator and a machine. Stated in simplest terms, task analysis is any process that is used to decompose a task into subtasks. Kirwan and Ainsworth (1992) wrote an entire book that describes specific task analysis techniques; however, any engineer can conduct an effective task analysis without being aware of any of the formal techniques. Common sense with a bit of creativity will suffice. As a general principle, remember that task analysis is about determining the step-by-step procedures followed by the human in the system. Any methodology that will help you obtain that information is useful. That may include talking with the operator, observing the operator, asking the operator to complete a written questionnaire, or reviewing existing documentation (i.e., procedures manuals). The engineer is familiar with the process of decomposition. We decompose a big problem into smaller problems to get to the point where we can fully understand all of the factors. This is the essence of task analysis.

The tangible output of a task analysis may be a table or a block diagram that clearly depicts the subtasks (or steps) in the order they must be completed to achieve a particular goal. Generation of this tangible output may be achieved by a knowledgeable engineer without ever leaving his/her desk; however, experience suggests that some form of observation may be extremely useful. I recall the days from my childhood when I rode along with my dad in the cab of the tractor. I spent hours watching his actions. When I was old enough to safely operate the machine on my own, I was already trained. At the time, I was not aware that I had essentially conducted a task analysis through observation of the operator, and I certainly never took the time to document the task using a flow chart or table, but I did learn the step-by-step procedures that were essential to operating each specific agricultural machine. The design engineer can do the same thing by riding along with an operator or by recording the operator's actions for subsequent analysis. Over the past 20 years, my research team has employed ride-alongs on several occasions to gain an understanding of how agricultural machines are operated. The following examples are provided to demonstrate the task analysis tools employed and the lessons learned in the process.

Dey and Mann (2009) applied task analysis concepts to gain an understanding of the workload associated with operating an agricultural sprayer equipped with a navigation device. Their task analysis consisted of a written survey and subsequent field observation of experienced sprayer operators. A digital video recorder was installed inside the cab of the sprayer with a clear view of the operator's face (particularly the eyes). During operation of the sprayer, video footage was recorded. In subsequent analysis, the proportion of time spent viewing four distinct visual sectors was determined. This eye-glance behavior was used to determine the amount of time spent looking at the GPS guidance display, the external view of the field, and the spraying apparatus. In addition to video observation of the subject, the researcher rode along with the sprayer operator and manually recorded other observations. The

study by Dey and Mann (2009) targeted the visual tasks of the sprayer operator (i.e., where is the operator looking to obtain the required information to guide the sprayer – the navigation device or the external field cues). It is interesting to note that contradictory results were reported between the written surveys (which are a form of subjective opinion) and the field observations (which are a form of objective data). The experienced sprayer operators reported that the lightbar navigation device was their most important source of guidance information on the survey; however, the observed eye-glance data suggested that the external field cues were more important than the navigation device because more time was devoted to viewing the external field cues. It is impossible to know whether these results are contradictory or whether the experienced sprayer operators were simply able to gain the information required from brief glances to the navigation device. It should be evident, however, that a deeper understanding of a task is enabled by triangulation of data from multiple sources.

Karimi et al. (2012) also used a form of task analysis to gain an understanding of the use of auto-steer systems on the eye-glance behavior and posture of 13 experienced air seeder operators. Similar to the work by Dey and Mann (2009), Karimi et al. employed ride-alongs and video recordings of the operator to reach their conclusions. Although not discussed in the publication (i.e., Karimi et al. 2012), it is important to note that the researcher who was present in the tractor cabs during the ride-alongs used the opportunity to observe the air seeder parameters most frequently monitored during field operation. These seven parameters (i.e., fan rpm, tank levels, application rates, blockage, forward speed, tool pressure, and tool depth) were used by Karimi et al. (2011) to enable a comparison of different formats for information presentation on air seeder displays. Thus, direct observation can be an effective task analysis technique for identification of the important parameters that must be monitored and controlled during operation of a machine.

Although my research team has employed a number of task analysis techniques, I would probably conclude that a simple video camera to record the behavior of the operator is the most effective technique. Experienced operators may have strong opinions about operational tendencies, but their opinions are not always supported by the observations of the video camera. Done with thoughtful preparation, task analysis is an important tool for the design engineer when seeking to understand the human-machine interaction for any agricultural machine.

15.3.3 Mental Workload

Mental workload is a concept that is easy to understand, but can be challenging to explain scientifically. Any task that we undertake invokes some level of mental workload – we must exert some mental energy to complete the task. Similarly, it is easy to understand that some tasks invoke a greater level of mental workload than others. Calculus invokes more mental workload than simple arithmetic. If the operation of an agricultural machine imposes a high level of mental workload, it is

reasonable to conclude that either (i) there will be poor machine performance or (ii) the operator will become mentally exhausted after only a short duration of operation. Excessive levels of mental workload must be avoided. At the other end of the spectrum, inadequate levels of mental workload will leave the operator understimulated (or bored). In such situations, machine performance is likely to suffer because the operator's normal level of mental capacity has been temporarily reduced. The challenge, therefore, is to find the right balance between inadequate and excessive mental workload. But, how do we measure mental workload?

The literature suggests that driver mental workload can be assessed indirectly by measuring fatigue because prolonged exposure to a heavy mental workload causes fatigue (Bartlett 1943). Although the measurement of driver fatigue is important in transportation research (Hartley 1998; Lee 1941), Mann (2000) showed that it may not be appropriate in research involving agricultural machines because tractor drivers must regularly leave the cab to perform various tasks. Each excursion from the cab corresponded with a short-term decrease in the driver's level of fatigue (i.e., the operator was somewhat refreshed by getting up out of the operator's seat). Consequently, it is challenging to correlate level of fatigue with the level of mental workload being imposed by the system.

Driver mental workload can also be measured in a more direct sense. Several behavioral and physiological responses have been correlated with the level of mental workload experienced by a driver. Methods to assess driver mental workload can be grouped into the following categories: subjective (i.e., self-report) measures, performance measures, and physiological measures (O'Donnell and Eggemeier 1986). Subjective measures assess the operator's opinion of the workload experienced. The operator fills out a scale, usually after the task is completed, rating his or her level of workload. It is generally believed that the operator's performance in a related task will decline if the mental workload is too high. A primary task of the machinery operator is to guide the agricultural machine along parallel passes to minimize lateral error (i.e., skipping or overlapping). Therefore, a performance measure such as lateral error can be correlated with mental workload. Physiological measures such as heart rate (HR) and heart rate variability (HRV) are popular for measuring workload with vehicle drivers (de Waard 1996). HR is an established measure, and there is a consensus that it increases with increases in physical workload (Lee and Park 1990). HRV, however, is not as well established. The consensus appears to be that HRV decreases with increases in both physical and mental workload (Lee and Park 1990). Each method has its own advantages and disadvantages, and not all methods are sensitive to workload in the same area of performance. It is recommended that a battery of tests be used to measure operator performance, assess physiological parameters, and gather self-report ratings simultaneously (Meijman and O'Hanlon 1984). The test battery should be selected based upon the specific research question to be answered (de Waard 1996).

My research team has experience with the measurement of mental workload. Of particular importance is a study by Dey and Mann (2011). In this paper, we conducted an in-depth study of the mental workload associated with operating an agricultural sprayer. We used a battery of mental workload measures including two

performance measures (i.e., lateral root mean square error of guidance deviation and reaction time), three physiological measures (i.e., heart rate variability, central nervous activity (EEG), and eye-glance behavior), and two subjective measures (i.e., NASA-TLX and SSWAT). Please refer to Dey and Mann (2011) for further information on these specific measures. The most important finding from this study was that the various measures did not all lead to the same conclusions. Most of the time, the physiological measures failed to agree with the performance and subjective measures. As researchers, we were faced with the dilemma of deciding which data to trust. I must confess that my confidence in the usefulness of mental workload as an assessment tool was shaken and my research team has ceased to use the metric of mental workload as a primary means of assessing human-machine interaction.

15.3.4 Situation Awareness

Typically, engineers are fascinated with technology. Technology can be used to design all kinds of gadgets to solve all kinds of problems. When the focus of the design engineer is on the technology, we can call this *technology-centered design*. The danger associated with technology-centered design is that the design engineer gets carried away with all of the "bells and whistles" that can be added based on the latest technology. Little or no attention is paid to the human operator who will attempt to operate the complex gadget. Consideration of the human operator during the design process can be referred to as *user-centered design*. Endsley et al. (2003) identified three principles that must be considered to achieve user-centered design:

(i) Organize the technology around the user's goals, tasks, and abilities.
(ii) Technology should be organized around the way users process information and make decisions.
(iii) Technology must keep the user in control and aware of the state of the system.

The key concept that emerges from these principles is called "situation awareness." Endsley (1988) defined situation awareness as "the perception of the elements of the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future." From this definition, we can see that situation awareness consists of perception, comprehension, and projection. These are referred to as the three levels of situation awareness. An operator who notices a flashing red light on his/her display panel is said to have level one situation awareness because the change in the environment (i.e., the red light flashing) has been perceived. If the operator understands that the flashing red light means that the engine oil pressure is low, it is said that the operator has level two situation awareness. To achieve level three situation awareness, the operator must realize that continuing to operate the engine without correcting the low oil pressure problem will result in damage to the engine. This is a projected consequence. As can be seen from this example, situation awareness depends upon both the training (or

experience) of the operator and the communication of information from the system to the human.

Techniques for measuring situation awareness belong to one of five categories: recall, anticipation, critical events, subjective ratings, and physiological indicators (Tenney and Pew 2006). The published literature supports the use of any of these techniques, although recall techniques seem to have been studied most thoroughly (Endsley et al. 2003). A recall technique requires a user to answer questions about a scenario while the scenario is occurring. The *situation awareness global assessment technique* (SAGAT) is the best-known objective metric, and it has been applied to driving scenarios. This technique is particularly suited to simulator environments because it works best if the scenario is halted (and all displays blanked); the operator must then answer questions about what was happening prior to the scenario being halted. If one does not want to halt the scenario, the *situation present assessment method* (SPAM) can be used. Questions are posed to the operator in real time with the displays in full view; the operator's response time is an indication of the operator's situation awareness. Sirkin et al. (2017) described a new system that has been developed to measure situation awareness in autonomous vehicles. Their "Daze" system was designed to function in both simulation and on-road driving scenarios without halting the simulation or stopping the driving activity. The driver is able to answer queries through a graphical interface that is simple to use and unobtrusive. Although the Daze system was designed with the on-road vehicle in mind, it seems reasonable that a similar concept could be utilized for applications involving off-road vehicles.

Given concerns associated with assessment of mental workload (as described in the previous section), my research team now utilizes the metric of situation awareness for assessing human-machine interaction. Bashiri and Mann (2014) used the situational awareness rating technique (SART) to investigate the impact of automation support on the situation awareness of the operator. The results showed that situation awareness increased as the level of automation support increased up to the point where the machine was operating completely autonomously. In this highest level of automation, the operator's situation awareness declined. Rakhra and Mann (2018) used the situation awareness global assessment technique (SAGAT) to evaluate various elements being considered for the design of an interface for an agricultural machine. We were able to differentiate display elements designed using a user-centered design approach from those elements not designed according to this approach using the metric of situation awareness. Situation awareness is an effective tool for assessing human-machine interaction.

It is worth noting that Rakhra and Mann (2018) used a subjective mental workload measure, the integrated workload scale (IWS) (Golightly et al. 2012; Pickup et al. 2005), in conjunction with the situation awareness technique (SAGAT). Both the IWS and SAGAT yielded the same conclusions suggesting that it might be appropriate to use both mental workload and situation awareness techniques simultaneously to increase confidence in experimental results.

15.3.5 Simulators

Drawing upon experience from other industries, it is often wise to conduct preliminary testing using some type of simulator. Simulators occupy a middle position on the continuum between pure laboratory tests and pure field tests. For human factors testing, field studies are invaluable because they provide directly valid data, and they allow the experimenter to determine the most important variables affecting performance. Due to variability of field conditions, weather, and other factors, however, field studies are often expensive and ineffective for determining functional relationships among different factors of interest (Duncan and Wegscheid 1982). Laboratory experiments are the opposite. They are very good for determining functional relationships, but the results are often relevant only to a laboratory situation that is much different than field conditions. Simulators lie somewhere in the middle. A simulator provides field-like tasks within a laboratory setting. As a result, functional relationships for field situations can be tested without the problems of field testing.

Distinctions between different simulators are often based on fidelity – how well a simulator mimics actual conditions in the physical environment. People often rely on fidelity as an indication of validity, assuming that people will perform more like a real situation the more lifelike the simulator. The problem is that high-fidelity simulators are very expensive, making them impractical for many applications. Low-fidelity computer-based simulators, on the other hand, are much less expensive and are suitable for many research applications (Gruening et al. 1998). For most applications, in fact, there is no consensus on how much fidelity is required in a simulation (Van Cott and Kinkade 1972). The best approach may be to use the least expensive simulator that suits the needs of the test, based on a task analysis.

With respect to operation of agricultural machines, it is appropriate to consider simulation of the driving task. A driving simulator is a system that provides an intelligent environment in which a human driver can perceive and control the operation of a virtual vehicle. If the observations in a driving simulator are to be generalized to real-world driving, the driving simulator must draw, from the drivers, the same driving behavior that they would exhibit in real-world driving. This means that the simulated vehicle and environment should have the same appearance and dynamics, provide the same information (i.e., feedback) to the driver, and provide the same means for the driver to input the necessary control commands. Although some driving simulators provide only visual feedback, most high-fidelity driving simulators allow the driver to interact with the vehicle and the environment in a multisensory fashion by providing motion, haptic, and auditory feedback (Kemeny and Panerai 2003). Not only do these nonvisual cues increase the realism of the simulation, but extensive research has shown that, depending on the driving task being simulated, some of these cues may be necessary (Siegler et al. 2001; Steele and Gillespie 2001).

Experimental control is the greatest advantage offered by driving simulators when they are used for research purposes. Many extraneous variables that cannot be controlled in real driving can be tightly controlled in driving simulators. Independent

variables can be separately controlled as desired, and several experiments can be run under identical experimental conditions. Conducting experiments in a driving simulator requires much less planning and is much less costly compared to experiments in an instrumented car in a real environment. Another important advantage of driving simulators is the safety of the test subject. This issue is particularly significant when studying issues such as driver fatigue or driving during low-visibility conditions. Measuring driving performance variables (and other parameters such as physiological and psychological response of the driver) is much easier to carry out in a driving simulator than in a real vehicle (Horiguchi and Suetomi 1995).

However, driving simulators also have certain shortcomings. No driving simulator can perfectly reproduce the real driving experience. Models of vehicle dynamics and environmental disturbances can be made increasingly accurate, but can never be perfect. It is extremely difficult, if not impossible, to provide visual feedback that has the same field of view, resolution, and depth cues as those of a real visual scene. Certain motion cues are not possible to render even in the most advanced driving simulators because no driving simulator has unlimited motion range. Direct rendering of simple vehicle maneuvers requires large motion systems that are unrealistic. Engineers have developed special techniques such as motion washout filtering, tilt coordination, and motion scaling that can render most vehicle motions, but these techniques do not completely resolve the existing problems. Transport delays are another major issue; there is always a delay between the subjects' action and the simulator's response. This is due to the time required for the acquisition of the subject's commands, computation of the appropriate response, and the delay in the visual and motion subsystems. Not only should these delays be small, but all simulator subsystems should be synchronized, a requirement that is difficult to achieve (Horiguchi and Suetomi 1995; Kemeny and Panerai 2003).

I have effectively used driving simulators throughout my research career. The initial simulator (Young 2003) was designed to mimic the tasks associated with operating an agricultural sprayer. Steering and controlling the booms were the main subtasks involved. The information necessary for steering was obtained from both a lightbar and an aiming point ahead of the operator; the information necessary for controlling the booms was obtained from displays behind the operator. Therefore, the simulator required subjects to scan the same locations and perform similar control actions as a high-clearance sprayer operator (Fig. 15.3). Two PhD students contributed to improving the fidelity of the simulator. Displays and controls were added in response to an in-field task analysis of agricultural sprayer operators (Dey and Mann 2009). Further improvements (i.e., simulator yaw motion, steering torque feedback, auditory feedback, and a video projection system) were made by D. Karimi who researched the role of sensory cues on the physical and behavioral validity of the simulator (Karimi and Mann 2008a, b, c; Karimi et al. 2008a, b).

More recently, a simulator has been developed to simulate a tractor-air seeder system. A block diagram representation of the current simulator is presented in Fig. 15.4. A complete description of the simulator is provided by Mann et al. (2014). The simulator has been used to conduct research related to the automation of agricultural machines.

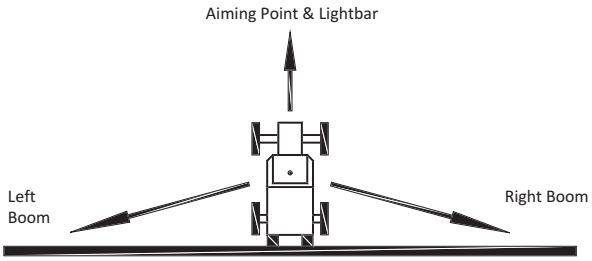


Fig. 15.3 Model representing the three visual sectors viewed by a sprayer operator. An aiming point is located ahead of the vehicle; sprayer booms are located to each side, slightly behind the vehicle

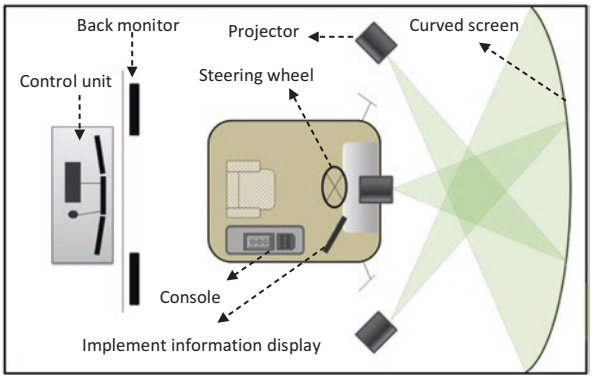


Fig. 15.4 Block diagram representation of a simulator used to simulate a tractor-air seeder system

Because a simulator only mimics the real field situation, results must be used cautiously. First, the tasks that are performed in a simulator must be designed carefully, with reference to a task analysis. Second, the relevance of results to the real world must not be overstated. Ultimately, it would be ideal if results are validated against real-world data to establish how well the simulation reflects field situations.

15.4 Use of Technology to Support the Operator

15.4.1 Guidance Aids

Beginning around the year 2000, manufacturers began to market different types of guidance aids that were intended to support the operator. While the quest for an automated guidance system was continuing, there was a realization that technology could be used to help the operator to reduce the lateral error typically associated with skipping and/or overlapping on parallel passes across a field.

Guidance aids might best be defined as "devices which provide guidance information to the driver, but do not attempt to replace the driver." A common characteristic of most of these devices is that the operator is required to visually monitor some type of interface to obtain guidance information.

Introduction of another interface into the cab of any vehicle may cause one of two results: (i) the task of the operator may become easier or (ii) the task of the operator may inadvertently become more difficult or more dangerous (Mollenhauer et al. 1997). Kaminaka et al. (1981) observed that steering performance decreased when operators were required to share their visual attention between the steering and rear-monitoring tasks. An explanation is offered by de Waard (1996): two competing visual channels cannot be watched simultaneously. The driver overcomes this problem by scanning one or both of the visual channels – a process that increases the driver's mental workload. Farmers have traditionally guided their machines by viewing the surrounding environment through the cab's windows. This behavior is not expected to change with the addition of a guidance aid. The driver would continue to obtain a general orientation within the field by looking out the windows, but the guidance aid's interface would be viewed for precise guidance information. The inevitable result is scanning of the two competing visual channels.

There are two types of guidance aids that were previously used by operators of agricultural machines: camera-based guidance aids and GPS-based guidance aids. A camera-based guidance aid is a system comprised of a camera that is mounted on the agricultural machine in a forward-facing direction and a display that is mounted near the operator's seat. What the camera "sees" is displayed on the display effectively giving the operator a close-up view of the region directly ahead of the camera. Tang and Mann (2003) investigated the factors contributing to guidance performance when using a camera-based guidance aid. Specifically, they focused on a parameter they described as "image velocity" which is the rate at which the image scrolled across the display as the machine moved forward across the field. This image velocity was influenced by the placement of the camera (height and angle of tilt). Operator performance declined as the image velocity increased. Given current technological advances, it is unlikely that camera-based systems are to be used as guidance aids. However, if camera-based systems were to be used for other monitoring purposes, attention should be devoted to placement of the camera to minimize the image velocity with the intent of maximizing the operator's ability to correctly interpret the visual information provided by the camera.

A GPS-based guidance aid is comprised of an antenna placed on the roof of the agricultural machine (used to obtain a GPS signal) and a lightbar (i.e., an array of light-emitting diodes (LEDs) that provided an indication of lateral error) that would be placed either in the cab or on the hood of the agricultural machine within the operator's line of sight. The lightbar displays the lateral error calculated based on the GPS signal. Discussions with dealers of this equipment revealed that the design of the lightbar was a major factor in how well operators adapted to the system. In particular, there was a demand for larger lightbars (Young 2003). Lightbars available commercially were not salient enough to be interpreted outside central vision. Because visual acuity degrades with eccentricity (Anstis 1974), a larger or more

salient lightbar could allow operators to acquire information from the lightbar further into their visual periphery. This could reduce scanning between the lightbar and the aiming point and allow information from the lightbar to be acquired more easily after scanning areas to the rear and sides. Possible ways of increasing salience include increasing size, increasing luminance, using flashing lights on the lightbar, or optimizing colors for peripheral vision (Ancman 1991; Christensen et al. 1986).

There are a number of factors that my research team considered in the design of an effective lightbar display. Young (2003) compared two lightbars that differed in size (by a factor of 10) and luminance (by a factor of 6). The larger, brighter lightbar enabled an 11% reduction in guidance error. Blue light is more easily perceived in the peripheral field of view than red light (Moreland and Cruz 1959); this is a likely explanation for the results reported by Ima and Mann (2003) who demonstrated that lightbars comprised of blue LEDs enabled the best tracking performance (i.e., lowest steering error) and monitoring performance (i.e., lowest reaction time). Ima and Mann (2003) also reported improved tracking and monitoring performance with enlarged lightbars, confirming the finding of Young (2003). Ima and Mann (2003) investigated the influence of auxiliary indicators – an additional cluster of LEDs – mounted on the side of the operator station where it would be detectable in the periphery of the operator. These auxiliary indicators yielded improved tracking performance, but at the expense of decreased monitoring performance. Overall, experimental results have confirmed that the operator's performance is influenced by the design of an interface such as a lightbar that is intended to provide supplemental information to the operator of an agricultural machine.

Although technology has advanced beyond these guidance aids, it is important to remember the lessons learned from the previous research. For both types of guidance aids, their ultimate effectiveness was based on the ability of the operator to attain visual information from the display. Thus, human-machine interaction must not be neglected.

15.4.2 Partial Automation

In past decades, the machinery operator was solely responsible for completion of all tasks. In the case of semi-autonomous agricultural machines, the responsibility for operating the machine is shared between the human operator and any microprocessors (computers) associated with automated systems or components. Deciding how to allocate responsibility (i.e., function) between the human and the automation is a nontrivial problem faced by the design engineer.

There has been much discussion of function allocation in the literature. One approach is to employ the "left-over principle" (Bye et al. 1999); this occurs when the human operator is assigned to any tasks that cannot be automated based on current technology (or are too expensive to automate) with essentially no consideration of ergonomic implications. Fitts (1951) proposed the "compensatory principle" where tasks are allocated to human or machine based on basic characteristics of

both humans and machines. Humans are perceived to have an advantage at *detecting* small visual and auditory signals, *perceiving* patterns of light and sound, and *improvising* and *exercising judgment*. Machines are able to respond to control signals more quickly and more precisely. Sheridan (2002) commented that "the human should be left to deal with the big picture while the computer copes with the details." Although the compensatory principle seems to be an improvement over the left-over principle, the "complementarity principle" was proposed by Jordan (1963). He argued that the system designer must "think about how we complement men by machines and vice versa to get a task done." Functions should be allocated in such a way that the operator remains in control of the situation and does not suffer any degradation in skills (Bye et al. 1999).

The first step in deciding the appropriate allocation of function between a human and a machine is to determine the goals to be achieved and the tasks necessary to achieve those goals (Hollnagel and Bye 2000). Task analysis techniques can be used to identify the goals and tasks associated with operating an agricultural machine. The nature of each task is compared to the attributes of both humans and computers to determine the most appropriate match. The outcome is a model depicting the allocation of function for a semi-autonomous human-machine system. Panfilov et al. (2016) provided three theoretical descriptions of a semi-autonomous system composed of a tractor and air seeder based on three distinct function allocation models (i.e., left-over principle, complementary principle, and compensatory principle) (Fig. 15.5). Their analysis concluded that the complementarity principle of function allocation is the most appropriate choice because it allows flexibility in allocation of function, it considers the human operator as an active participant in the functioning of the system, and it allocates functions according to the need to ensure stability of system performance. From this discussion, it is evident that the design of any autonomous or semi-autonomous agricultural vehicle should be based on allocation of function.

Bye et al. (1999) concluded their paper by saying "the allocation of functions between man and machine is an important part of design of automation systems." Parasuraman et al. (2000) proposed a framework for automation design (i.e., deciding which functions to automate and to what extent), consistent with the complementarity principle, that consists of four steps: (1) identifying types of automation (which requires identification of the relevant tasks), (2) determining appropriate levels of automation for tasks, (3) evaluating the system performance from the perspective of the human, and (4) evaluating the system performance from the perspective of the technology. To address the first step, Parasuraman et al. (2000) proposed a model with four classifications of functions: information acquisition, information analysis, decision and action selection, and action implementation. They contend that all tasks fit into one of these four classifications. For the system designer, this corresponds to four potential types of automation: acquisition, analysis, decision, and action. Each type of automation can have a level of automation that ranges from low (no automation or manual task) to high (fully automatic).

Bashiri (2015) completed a PhD thesis entitled "Effects of task automation on the mental workload and situation awareness of operators of agricultural

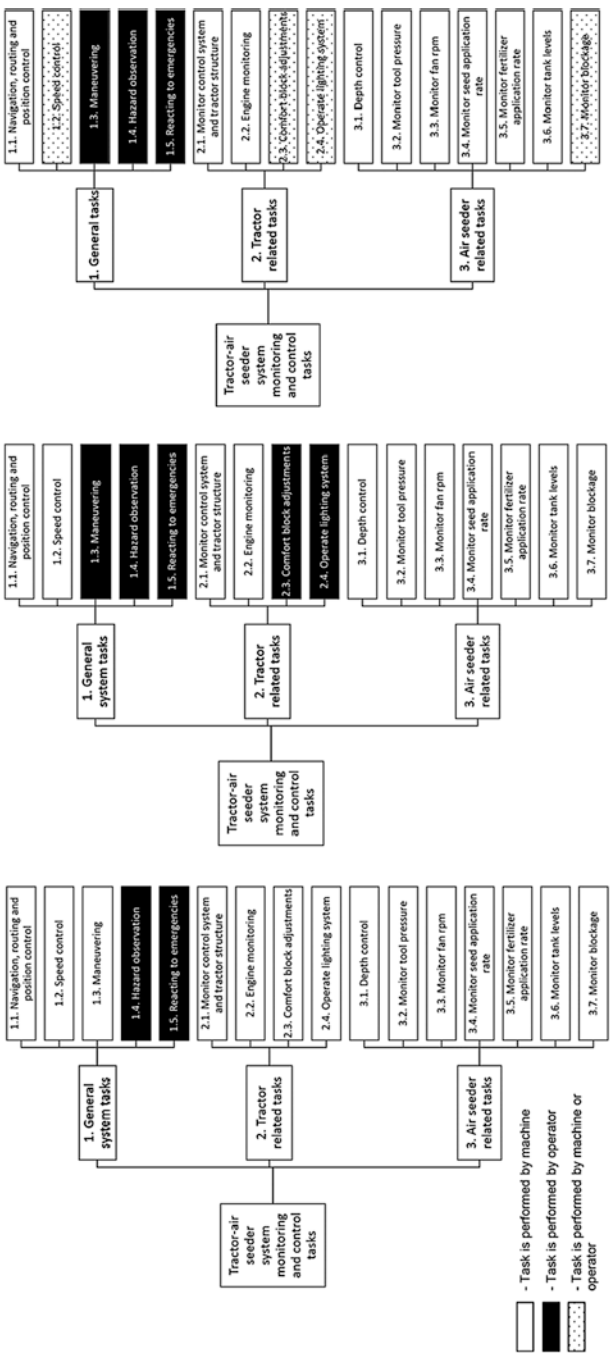


Fig. 15.5 Graphical comparison of function allocation during the design of a tractor-air seeder system according to (**a**) the left-over principle, (**b**) the compensatory principle, and (**c**) the complementarity principle. (Reproduced from Panfilov et al. 2016)

semi-autonomous vehicles." His research investigated the effects of vehicle steering task automation (VSTA) and implement control and monitoring task automation (ICMTA) on mental workload and situation awareness of subjects using a tractor-driving simulator. The most interesting results were observed for ICMTA which involved five levels of automation support. Reaction time and number of errors made by subjects both decreased as the automation level increased (Bashiri and Mann 2015). Situation awareness increased as the level of automation support increased except at the highest level of automation where the subjects were essentially eliminated from the task loop (Bashiri and Mann 2014). A similar result was obtained by Rakhra (2018) – the highest level of situation awareness was observed for moderate levels of automation (and situation awareness decreased for the fully autonomous condition). Based on these results, it is reasonable to conclude that caution must be exercised when contemplating a fully autonomous agricultural machine if it is envisioned that a human will be expected to monitor the performance of such a machine.

15.4.3 Fully Autonomous Agricultural Machine

The concept of the driverless tractor has appeared in the scientific literature over the past several decades. We have now arrived at the point in history where the technology exists to provide automated guidance systems for agricultural machines. We might be tempted to believe that our work is complete now that we have realized the agricultural machine that can drive itself across the field. We must resist this temptation because evidence suggests that automation cannot operate in isolation of humans.

Automated systems were first designed to relieve the human of repetitive or continuous manual tasks. It has been observed, however, that automation often redistributes workload rather than reducing it because the human is forced to assume a supervisory role (Sarter et al. 1997). According to Sheridan (1992), **supervisory control** occurs when there is at least one human operator that has the capability to both send instructions and receive status information from a system that itself functions autonomously. If a machine is designed to have **fully automatic control**, the human operator can observe status information through a display, but is unable to control the actions of the machine. It is impossible to predict with certainty whether future AAMs will be designed with fully automatic control or supervisory control; however, there is evidence to suggest that supervisory control may be the preferred option. Blackmore et al. (2002) proposed a system architecture to enable control of an autonomous tractor. Their view included a "coordinating process" which was to be handled by a human coordinator located remotely in a farm office. They proposed the phrase "tractor mimic display" for the computer interface that would be used by the coordinator to obtain tractor status information. Blackmore et al. (2002) suggested that the mimic display would also incorporate a "real-time video link to steerable on-board cameras" to allow the coordinator to have "a better

understanding of the tractor's environment." Several other research groups have described similar concepts where a remote supervisor has some level of control over one or more AAMs (Johnson et al. 2009; Moorehead et al. 2009; Moorehead et al. 2012; Stentz et al. 2002). Project Xaver is a new concept proposed by tractor manufacturer Fendt that is comprised of a "swarm" of small field robots that are directed via the Cloud from a "logistic unit." This logistic unit is responsible for transport of the field robots, battery charging (the field robots are electric vehicles), and supply of seeds (the model described is for precision seeding). The farmer manages the entire process through a tablet; the tablet enables task planning and live monitoring.

To summarize, it is most reasonable to conclude that autonomous agricultural machines will not operate in isolation. It is likely that multiple field robots will be required to communicate with each other, receiving overall instruction and support from some type of central unit. Furthermore, the farmer will have the ability to monitor operation and submit task planning instructions through some type of interface. In other words, the farmer will have an active supervisory role with an autonomous agricultural machine. Careful attention must be paid to the design of the automation interface to ensure that the human-autonomy system can operate at maximum efficiency.

15.5 Future Challenges Associated with Remote Supervision of Autonomous Agricultural Robots

15.5.1 Sensory Requirements for Remote Supervision of Autonomous Agricultural Robots

Historically, the human operator of an agricultural machine is seated in an operator station mounted on the machine. This means that the operator is situated in close proximity to the machine and can obtain sensory information (primarily visual and auditory) directly from the machine and the surrounding environment. I can remember my days as a teenager on the farm – specifically operating our IHC 503 combine. That late 1960s machine had very few sensors, and very limited information was available on the instrument console. Nevertheless, I became very proficient at operating that machine based on sensory input. I learned to detect when the machine was on the verge of being overloaded by the sound coming from the threshing chamber located beneath the cab. One might say that this lesson was learned the hard way as, on numerous occasions, I failed to adjust the speed of the machine in time resulting in the threshing cylinder becoming plugged. Of course, this meant slipping on gloves and wasting valuable time manually pulling stalks out of the machine. Over time, I learned to detect the early signs of overloading by the subtle change in the sound coming from the threshing chamber so that I could reduce the forward speed of the combine ultimately eliminating the manual task of unplugging the combine.

Admittedly, current machines employ more sensors than were present on our old IHC 503 combine. Nevertheless, it is reasonable to believe that machinery operators still obtain sensory information from the machine and the surrounding environment that contributes to successful operation of these machines. Projecting to the future, it is anticipated that the farm manager will supervise autonomous agricultural machines from a remote location and, therefore, will not have access to the direct sensory information. It is reasonable to speculate that the loss of this information may be detrimental to the ability of the farm manager to remotely supervise these machines. Thus, there is a need to determine whether sensory information is essential to the task of remote supervision of an agricultural machine.

Panfilov and Mann (2018) conducted a lab experiment in which research participants, who were acting as remote supervisors, were asked to detect machine malfunctions that were presented using either graphical indicators or with video footage. Their experimental results demonstrated, based on observed eye-glance behavior, that the live video footage was not particularly useful for detecting machine problems or malfunctions, but the supervisors felt more secure in their supervisory task when live video was present. Analysis of the gaze distribution showed that supervisors made long gazes at the screen with the graphical indicators and only brief gazes to the video footage. The recommendation from their research is that an interface for remote supervision should include real-time video of a general view of the agricultural machine with detailed information on machine status displayed using graphical indicators.

A follow-up study was conducted to determine the visual requirements for a remote supervisor of an autonomous sprayer (Edet and Mann 2020). Twenty-nine experienced sprayer operators participated in the project. During the experiment, a total of 11 different video clips showing various regions of a sprayer and the surrounding environment during spraying operation were presented sequentially to each participant; participants were asked to comment on each video clip. The results indicated that experienced sprayer operators considered views that are familiar and directly related to the spraying operation as “extremely important” while unfamiliar views were given a lower rank. Sprayer operators prefer (i) the view ahead of the sprayer (from the operator’s cab), (ii) the boom, and (iii) an aerial view of the sprayer in operation, as these regions enable them to assess the sprayer and field/crop conditions to make necessary decisions. The information that was perceived from these views included boom height, nozzle status (plugged or not), spray pattern, presence of obstacles, crop conditions, approximate travel speed, proximity of headland, weather conditions, and location of the sprayer within the context of the field. With this information, operators felt that they would be able to identify the need to make adjustments to the sprayer (i.e., change boom height, increase or decrease droplet size, increase or decrease travel velocity) or stop the sprayer until problems can be resolved (i.e., clean a plugged nozzle, weather conditions improve). There is a growing body of evidence to suggest that real-time video is essential to the task of remotely supervising an autonomous agricultural machine. Researchers have yet to investigate issues such as video transmission latency (delay) and the impact that this will have on the supervisor.

I began this section with an anecdotal story related to the contribution of auditory information to the operation of an agricultural machine. A recent graduate student in my research team investigated the role of auditory information to the task of remote supervision of an autonomous agricultural machine (Simundsson et al. 2019). Based on anecdotal information, it was recognized that machinery operators are often able to detect existing or impending problems from the changes in sound produced by the mechanical components of the machine. Karimi et al. (2008b) reported that the addition of auditory cues did not improve steering performance (in a simulated agricultural vehicle) perhaps because steering is a purely visual task; however, auditory cues did improve the monitoring task. Donmez et al. (2009) investigated the use of sonifications (continuous auditory alerts) during the control of unmanned aerial vehicles and found that visual information supported by sonifications yielded faster reaction times than visual information supported by discrete auditory signals. Simundsson et al. (2019) collected auditory information from combine harvesters in three distinct operating modes. A neural network was used to classify the auditory signals – it was able to correctly classify the three distinct operating modes. Based on the results of this proof-of-concept work completed by Simundsson, there is further opportunity to develop the technology to enable auditory information to be incorporated into the automation interface for remote supervision of an autonomous agricultural machine.

15.5.2 Shared Situation Awareness in Human-Autonomy Teams

In February 2017, Mica Endsley published an article entitled "From here to autonomy: lessons learned from human-automation research" (Endsley 2017). She argues that successful autonomous systems will be those that achieve "a successful approach to human-autonomy teaming." To guard against unexpected automation transitions "when the automation suddenly passes control to the human operator who may not be ready to take over" (Endsley 2017), it is critical that automation interfaces be designed carefully to support the situation awareness of the individual overseeing the automation. Grubb et al. (1994) demonstrated that passive monitoring of automation creates "a high-workload activity" for the individual overseeing the automation. The design of the automation interface is further complicated by the automation conundrum described by Endsley (2017) as "The more automation is added to a system, and the more reliable and robust that automation is, the less likely that human operators overseeing the automation will be aware of critical information and able to take over manual control when needed." To address this conundrum, Endsley proposed a "human-autonomy system oversight" model to assist system designers and researchers. It is recommended that automation interfaces be designed with transparency to enable the automation supervisor to successfully navigate mode transitions when human intervention is required. Essentially

this requires that both the human and the autonomous system have the same awareness of system status – a principle known as "shared situation awareness." The concept of shared situation awareness is not new, but it has traditionally been applied to situations involving two or more human teammates that contribute to the functioning of a system. There is a current void in the scientific literature with respect to shared situation awareness involving teams comprised of human supervisors and autonomous machines. Research is needed to fill this void, ultimately enabling the viability of remote supervision of AAMs. For a human-autonomy team, it is impossible to query the autonomous machine in the usual manner. Therefore, a novel technique must be developed to enable shared situation awareness of human-autonomy teams to be assessed.

15.6 Summary and Concluding Thoughts

There are numerous researchers and entrepreneurs working on the development of autonomous agricultural machines (i.e., robots). If designed appropriately, autonomous agricultural machines should be able to reduce the demand on the farm worker possibly enabling one farm worker to remotely supervise multiple machines from a central location. I believe that these machines will ultimately function as part of a larger system comprised of the autonomous agricultural machine and a human supervisor. The ultimate productivity of such a human-machine (or human-automation) system will depend on the ability of the farmer to efficiently and effectively obtain information from each of the autonomous agricultural machines through an interface. Great importance must be placed on the design of the interface so that human interaction with the autonomous agricultural machine can be optimized. Ultimately, the efficiency of this human-autonomy team will be dependent upon both members of the team having a common understanding of system status.

In many fields of research, there is interest in understanding the challenges of human interaction with automation. Although interest among those designing and researching agricultural machinery is not yet widespread, initial signs are beginning to appear. Lang et al. (2009) used the phrase "sustainable integration of man" in an article entitled "Analysis of human factors on agricultural machines." In 2010, Schmitz wrote an article entitled "Ergonomics and automation – safe manipulation of complex systems." Both articles share the message that as technology continues to advance, it is essential that ergonomics be considered in the design of modern agricultural machines.

References

Ancman E (1991) Peripherally located CRTs: color perception limitations. In: Proceedings of the IEEE National Aerospace and electronics conference. IEEE, New York, pp 960–965

Anstis SM (1974) A chart demonstrating variations in acuity with retinal position. Vis Res 14:589–592

Banks WW, Boone MP (1981) A method for quantifying control accessibility. Hum Factors 23(3):299–303

Bartlett FC (1943) Fatigue following highly skilled work. Proceedings of the Royal Society, pp 248–257

Bashiri B (2015) Effects of task automation on the mental workload and situation awareness of operators of agricultural semi-autonomous vehicles (unpublished doctoral dissertation). Department of Biosystems Engineering, University of Manitoba, Winnipeg

Bashiri B, Mann DD (2014) Automation and the situation awareness of drivers in agricultural semi-autonomous vehicles. Biosyst Eng 124:8–15

Bashiri B, Mann DD (2015) Impact of automation on drivers' performance in agricultural semi-autonomous vehicles. J Agric Safety Health 21(2):129–139

Blackmore BS, Have H, Fountas S (2002) A proposed system architecture to enable behavioural control of an autonomous tractor. In: Zhang Q (ed) Automation Technology for off-Road Equipment. ASAE, St. Joseph, pp 13–23

Bye A, Hollnagel E, Brendeford TS (1999) Human-machine function allocation: a functional modelling approach. Reliability Eng and System Safety 64:291–300

Christensen JM, O'Donnell RD, Shingledecker CA, Kraft CL, Williamson G (1986) Optimization of peripheral vision (USAFSAM-TR-85-96). USAF School of Aerospace Medicine, Brooks Air Force Base

de Waard D (1996) The Measurement of Drivers' mental workload (Unpublished doctoral dissertation). Traffic Research Centre, University of Groningen, Haren, Netherlands

Dey AK, Mann DD (2009) A complete task analysis to measure the workload associated with operating an agricultural sprayer equipped with a navigation device. Appl Ergon 41:146–149

Dey AK, Mann DD (2011) Mental workload associated with operating an agricultural sprayer: an empirical approach. J Agric Safety Health 17(2):91–110

Donmez B, Cummings ML, Graham HD (2009) Auditory decision aiding in supervisory control of multiple unmanned aerial vehicles. Hum Factors 51(5):718–729

Dooley WK (2012) Ergonomics and the development of agricultural vehicles, ASABE distinguished lecture series no. 36. ASABE, St. Joseph

Drakopoulos D, Mann DD (2006) Development of a method for ergonomic assessment of a control layout in tractors. In: Bust PD (ed) Contemporary ergonomics 2006. Taylor & Francis, Boca Raton, pp 611–615

Drakopoulos D, Mann DD (2007) An ergonomic analysis of the controls present in a tractor workstation. Can Biosyst Eng 49

Drakopoulos D, Mann DD (2008) A mathematical equation for quantifying control functionality in agricultural tractors. J Agric Saf Health 14(4):377–389

Duncan JR, Wegscheid EL (1982) Off-road vehicle simulation for human factors research. AGRIS: International Information System for the Agricultural Science and Technology

Edet U, Mann DD (2020) Visual information requirements for remotely supervised autonomous agricultural machines. Appl Sci 10(8):2794

Endsley MR (1988) Design and evaluation for situation awareness enhancement. In: Proceedings of the human factors society 32nd annual meeting. SAGE Publications, Santa Monica, pp 97–101

Endsley MR (2017) From here to autonomy: lessons learned from human-automation research. Hum Factors 59(1):5–27

Endsley MR, Bolté B, Jones DG (2003) Designing for situation awareness: an approach to user-centered design. Ergonomics 56(4):727–728

Fitts PM (1951) Human engineering for an effective air navigation and traffic control system. National Research Council, Washington, DC
Golightly D, Sharples S, Wilson J, Lowe E (2012) Developing a method for measuring situation awareness in rail signalling. In: Waard D d, Merat N, Jamson H, Barnard Y, Carsten O (eds) Human factors of systems and technology. Shaker, Maastricht, pp 15–26
Grubb PL, Miller LC, Nelson WT, Warm JS, Dember WN (1994) Cognitive failure and perceived workload in vigilance performance. In: Mouloua M, Parasuraman R (eds) Human performance in automated systems: current research and trends. Lawrence Erlbaum, Hillsdale, pp 115–121
Gruening J, Bernard J, Clover C, Hoffmeister K (1998) Driving simulation. In: 1998 SAE International Congress & Exposition. SAE, Detroit
Hartley L (1998) Managing fatigue in transportation. Pergamon, Dordrecht
Hollnagel E, Bye A (2000) Principles for modeling function allocation. Int J Human-Computer Stud 52:253–265
Horiguchi A, Suetomi T (1995) A Kansei engineering approach to a driver/vehicle system. Int J Ind Ergon 15:25–37
Ima CS, Mann DD (2003) Lightbar design: the effect of light colour, lightbar size and auxiliary indicators on tracking and monitoring performance. Agricultural Engineering International: The CIGR Ejournal
Johnson DA, Naffin DJ, Puhalla JS, Sanchez J, Wellington CK (2009) Development and implementation of a team of robotic tractors for autonomous peat moss harvesting. J Field Robot 26(6–7):549–571
Jordan N (1963) Allocation of functions between man and machines in automated systems. J Appl Psychol 47(3):161–165
Kaminaka MS, Rehkugler GE, Gunkel WW (1981) Visual monitoring in a simulated agricultural machinery operation. Hum Factors 23:165–173
Karimi D, Mann DD (2008a) Role of motion cues in straight-line driving of an agricultural vehicle. Biosyst Eng 101(3):283–292
Karimi D, Mann DD (2008b) Role of visual cues in driving an agricultural vehicle. Ergonomics Open J 1:54–61
Karimi D, Mann DD (2008c) Torque feedback on the steering wheel of agricultural vehicles. Comput Electron Agric 65:77–84
Karimi D, Mann DD, Ehsani R (2008a) Modeling of straight-line driving with a guidance aid for a tractor-driving simulator. Appl Eng Agric 24(4):403–408
Karimi D, Mondor T, Mann DD (2008b) Application of auditory signals to the operation of an agricultural vehicle. J Agric Saf Health 14(1):71–78
Karimi D, Mann DD, Yan J (2011) A comparison of textual, symbolic, and pictorial presentation of information on an air-seeder display. Aus J Agric Eng 2(4):90–95
Karimi D, Henry J, Mann DD (2012) Effect of using GPS autosteer guidance systems on the eye-glance behavior and posture of tractor operators. J Agric Safety Health 18(4):309–318
Kemeny A, Panerai F (2003) Evaluating perception in driving simulation experiments. Trends Cogn Sci 7(1):31–37
Kirwan B, Ainsworth LK (1992) A guide to task analysis. Taylor & Francis, Philadelphia
Lang T, Gores T, Junemann D, Vollrath M, Werneke J, Huemer AK (2009) Analysis of human factors on agricultural machines. Landtechnik 64(1):58–60
Lee R (1941) Fatigue and hours of service of interstate truck drivers, Public health bulletin no. 26. U.S. Public Health Service, Washington, DC
Lee DH, Park KS (1990) Multivariate analysis of mental and physical load components in sinus arrhythmia scores. Ergonomics 33(1):35–47
Mann DD (2000) Measurement of driver fatigue during operation of an agricultural machine. ASAE, St. Joseph
Mann D, Bashiri B, Rakhra A, Karimi D (2014) Development of a tractor driving simulator to research ergonomics of agricultural machines. In Proceedings of International Conference of Agricultural Engineering. Zurich, Switzerland

Mastorakos M (2012) Ergonomic control panel design for a remotely operated agricultural tractor (unpublished doctoral dissertation). Department of Biosystems Engineering, University of Manitoba, Winnipeg

Meijman TF, O'Hanlon JF (1984) Workload. An introduction to psych. Theories and measurement methods. In: Handbook of work and organizational psychology. Wiley, New York, pp 257–288

Meister D (1971) Human factors: theory and practice. Wiley, New York

Mollenhauer MA, Hulse MC, Dingus TA, Jahns SK, Carney C (1997) Design decision aids and human factors guidelines for ATIS displays. In: Ergonomics and safety of intelligent driver interfaces. Lawrence Erlbaum, Mahwah, pp 23–62

Moorehead S, Ackerman C, Smith D, Hoffman J, Wellington C (2009) Supervisory control of multiple tractors in an orchard environment. In 4th IFAC International Workshop on Bio-Robotics, Information Technology and Intelligent Control for Bioproduction Systems. pp 10–11

Moorehead SJ, Wellington CK, GilmoreBJ, Vallespi C (2012) Automating orchards: A system of autonomous tractors for orchard maintenance. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Workshop on Agricultural Robotics

Moreland J, Cruz A (1959) Color perception with the peripheral retina. Opt Acta 6:117–151

O'Donnell RD, Eggemeier FT (1986) Workload assessment methodology. In: Handbook of perception and human performance, vol 2. Wiley, New York, pp 42–49

Panfilov I, Mann DD (2018) The importance of real-time visual information for the remote supervision of an autonomous agricultural machine. Can Biosyst Eng 60(2):11–18

Panfilov I, Edet U, Mann DD (2016) Implications of function allocation on the design of semi-autonomous agricultural machines. CSBE/SCGAB 2016 Annual Conference

Parasuraman R, Wickens CD, Sheridan T (2000) A model for types and levels of human interaction with automation. IEEE Trans Syst Man Cybernet 30(2):86–297

Pickup L, Wilson JR, Norris BJ, Mitchell L, Morrisroe G (2005) The integrated workload scale (IWS): a new self-report tool to assess railway signaller workload. App Ergonomics 36(6):681–693

Rakhra A (2018) Design and evaluation of the user interface for tractor air-seeder systems (Unpublished doctoral dissertation). Department of Biosystems Engineering, University of Manitoba, Winnipeg, Manitoba, Canada

Rakhra AK, Mann DD (2018) Design and evaluation of the individual elements of the interface for an agricultural machine: impact on the situation awareness and mental workload of the operator. J Agric Safety Health 24(1):27–42

Sanders MS, McCormick J (1993) Human factors in engineering and design, 7th edn. McGraw-Hill, New York

Sarter NB, Woods DD, Billings CE (1997) Automation Surprises. In: Handbook of human factors and ergonomics. Wiley, New York

Schmitz B (2010) Ergonomics and automation—safe manipulation of complex systems. Landtechnik 65(3):167–169

Sheridan TB (1992) Telerobotics, automation, and human supervisory control. MIT Press, Cambridge, MA

Sheridan TB (2002) Humans and automation: system design and research issues. Wiley, Santa Monica

Siegler G, Reymond G, Kemeny A, Berthoz A (2001) Sensory motor integration in a driving simulator: contributions of motion cueing in elementary driving tasks. In: Proceedings 2001 driving simulator conference. Sophia Antipolis, pp 21–32

Simundsson A, Mann DD, Thomas G (2019) A neural network to classify auditory signals for use in autonomous harvester control systems. In CSBE/SCGAB 2019 Annual Conference. Vancouver, Canada

Sirkin D, Martelaro N, Johns M, Ju W (2017) Toward measurement of situation awareness in autonomous vehicles. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp 405–415

Steele M, Gillespie RB (2001) Shared control between human and machine. In Proceedings Human Factors and Ergonomics Society 45th Annual Meeting. Minneapolis

Stentz A, Dima C, Wellington C, Herman H, Stager D (2002) A system for semi-autonomous tractor operations. Auton Robot 13:87–104

Tang P, Mann DD (2003) Factors contributing to guidance performance when using a camera-based guidance aid. J Agric Saf Health 9(1):47–60

Tenney YJ, Pew RW (2006) Situation awareness catches on. In: Williges RC (ed) Reviews of human factors and ergonomics, vol 2. HFES, Santa Monica

Van Cott HP, Kinkade RG (1972) Human Engineering Guide to Equipment Design. US Department of Defense

Young S (2003) Design of an agricultural driving simulator for ergonomic evaluation of guidance displays (Unpublished master's thesis). Department of Biosystems Engineering, University of Manitoba, Winnipeg, Manitoba, Canada

Chapter 16
Machinery-Canopy Interactions in Tree Fruit Crops

Xin Zhang, Qin Zhang, Manoj Karkee, and Matthew D. Whiting

16.1 Introduction

The rapid development of the modern agricultural machinery has substantially advanced farming operations in recent years. Researchers and engineers, building on those successes, are continuing to work on developing intelligent solutions to solve various challenging problems in production agriculture. Among others, there has been a particular emphasis on developing automation and robotic solutions for tree fruit production (e.g., apple, cherry, citrus, pear, peach, and more) to address the challenge of labor shortage that has been increasingly impacting the industry all over the world. Despite these efforts, the progress in practically adopting mechanical or robotic solutions to these crops has been slow. One of the primary attributors causing the sluggish adoption is the large variation and complexity in tree structures. In addition to fulfilling the important expectation of crop yield and quality improvements, the creation and adoption of proper tree crop architectures make them more machine-friendly, which could be one of the critical ways to facilitate further advancement and adoption of mechanized or even robotic solutions in tree fruit production. In other words, a study on machine-canopy interaction will be a fundamental aspect of overall efforts on mechanization and automation in tree fruit crops.

The external structure of tree crops has been proved to be critically important in achieving effective mechanical or robotic operations in tree fruit orchards (Zhang et al. 2019a, b). Without proper intervention, most fruit trees will be forming a big and tall spherical structure. Moreover, such a canopy will grow randomly into an unstructured environment, which presents a high level of variability and uncertainty making it difficult for most mechanical or robotic operations in orchards (e.g.,

X. Zhang (✉) · Q. Zhang · M. Karkee · M. D. Whiting
Washington State University, Prosser, WA, USA
e-mail: xin.zhang4@wsu.edu; manoj.karkee@wsu.edu

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1_16

pruning, thinning, spraying, harvesting, and transporting) because of the need for adjusting machine positions appropriately for performing an operation with almost every branch.

Taking robotic harvesting in tree fruit as an example, Mehta and Burks (2014) used a programmed manipulator for robotic citrus harvesting. An analysis of results indicated that about half of the unsuccessful harvesting attempts were because of the difficulties caused by canopy occlusions (25%) and fruit clusters (23%). Such results revealed a strong dependence of a robot on crop canopy and environmental factors. Figure 16.1 visualizes the difference of canopy architectures between unstructured, naturally grown apple trees and structured, trellis-trained trees. In the naturally grown trees, the target apples might be distributed within a canopy space of 3 m in height and 2 m in width with heavy occlusions caused by leaves, branches, and other fruit (Fig. 16.1a). On the other hand, in the structured (trellis-trained), modern orchards, apples can mostly be located along the primary branches with minimum occlusions (Fig. 16.1b).

Lack of appropriate design and maintenance of crop/canopy structures can cause canopy occlusions and failure of robotic/mechanized activities, as depicted in Fig. 16.2 (Silwal et al. 2017). This study showed that the excessively long branches and offshoots often induced the failure of fruit removal (e.g., slipping out from the gripper or insufficient detaching distance) because of the limited working space for a robot. Another two studies on automated harvesting at Washington State University (Hohimer et al. 2019; Wang et al. 2018) also showed that clustered fruits caused

Fig. 16.1 Example of unstructured, conventional apple trees (**a**) and structured (trellis-trained), modern apple trees (**b**) in Washington State, USA (from Zhang 2020)

Fig. 16.2 An example of unsuccessful fruit detachment by a robot because of a long and thin offshoot bearing the fruit (from Silwal et al. 2017)

major problems for both the vision system and the manipulating arms during apple harvesting. The findings implied, as expected, that for a successful robotic system, unstructured crop architectures could present significant hurdles, as robots tend to perform well in a structured environment. Similarly, the harvesting performance (e.g., fruit detachment efficiency) of mechanical shake-and-catch systems could also be affected by its interactions with the canopy architectures (e.g., branch length and/or diameter). Partly because of these hurdles, the long effort in developing robotic/mechanized harvesting systems starting from the 1960s (Bac et al. 2014; Sistler 1987) has not yielded commercially successful solutions yet.

As the entire agricultural production is moving toward adopting more mechanized and automated technologies worldwide, this chapter will therefore provide a unique perspective on a critical aspect of automation and robotics in agricultural fields: interaction and integration of machinery-canopy systems into developing a systems solution for agricultural robotics. Specifically, the chapter will discuss the ways machine systems could interact with fruit tree architectures to achieve the highest possible work efficiency.

16.2 Orchard Mechanization and Role of Tree Fruit Crop Architectures

Previous studies have underscored the importance of crop architecture and canopy management practices on effectiveness of mechanized orchard operations. For example, in mass mechanical fruit harvesting, weak and pendant fruiting branches prevent shaking energy from being effectively transmitted to the target fruits. This effect is attributed to the higher energy dissipation on thin and long lateral branches (He et al. 2019).

As modern orchard systems have been proven to produce high-value fruits, there is a need to refine tree systems to develop simplified orchard systems that are productive, producing high-quality fruits, and equally important, amenable to facilitate the incorporation of mechanized and automated field operations in these crops.

Therefore, the incorporation of machinery in fruit production operations requires a transition from conventional low-density, complex orchard systems to modern planar architectures, which can be accomplished through appropriate orchard training, pruning, and thinning operations. Some preliminary results have shown the progresses on how machinery efficiency has been enhanced by certain horticultural practices. For example, Tombesi et al. (2017) found that mechanical harvesting efficiency of fruit removal could be enhanced over 12% by removing weak branches on free vase-trained olive trees. Peterson et al. (1999) studied the mechanical harvesting of apples in trees trained to a Y-trellis architecture. Their results also suggested that high fruit detachment efficiency could be achieved if proper crop/canopy pruning strategies were adopted. As another example, when a mass mechanical harvester was used to harvest blueberries planted in bushes, each fruiting shoots needed to be manually and individually vibrated during the harvesting process (Fig. 16.3; He 2017). Such naturally grown tree/bush architectures might negatively affect the harvesting efficiency in terms of fruit removal, fruit collection, fruit quality, and time taken. These findings indicated that complex crop conditions could be major hurdles for the success of robotic/mechanical operations in orchards.

Partially because of the lack of efforts in developing and adopting appropriate crop/canopy management systems, a long effort in developing robotic or mechanical systems has not yet led to commercially successful solutions for various orchard applications such as fresh market fruit harvesting and fruit tree pruning. Therefore, wider efforts are essential in keeping automation in the forefront while designing, developing, and adopting machine-friendly tree architectures, and the cultural practices should be optimized to provide a simpler and friendlier crop environment for the practical use of automated and robotic machines.

Fig. 16.3 An example of using mechanical harvester to vibrate blueberries in bushy architecture in a commercial orchard in Washington State, USA (Image Courtesy of Dr. Long He)

16.2.1 Conventional Fruit Tree Architectures

Fruit trees are perennial woody plants, which generally have a permanent trunk with branches/twigs growing from the trunks. The fruit tree canopies, therefore, could be described as a four-dimensional (4D) structure (i.e., a specific 3D trunk for individual trees plus variations between trees, and over time; Godin et al. 1999). The architecture will become more complex when it evolves over time through growth and development. Traditionally, trees are grown with a comparatively lower level of training and pruning and generally without a trellis system to support the structure, which can be described as "big-old-bushy" trees (Fig. 16.4). Many orchards are still planted in such manner lacking uniformity, which makes these sites unsuitable for using automated or autonomous orchard machinery.

Many different types of field operation are carried out in tree fruit production around the year including pruning, training, thinning, spraying, harvesting, and post-harvest transportation. The effectiveness and practical viability of utilizing orchard machines to carry out these operations have been limited, as discussed above, due to the complexity of the tree canopies in these conventional architectures. For example, selective robotic pruning will need to identify and precisely locate the target branches and associated cutting points. Such complicated structures pose increased level of difficulties for both the machine vision system to detect/localize the target branches and the robotic end-effector (hand) to approach and cut the branches at desired locations. Modern fruiting-wall tree architectures, such as some apple and sweet cherry orchards in Washington State, could potentially help in facilitating and simplifying the robotic tasks in tree fruit orchards.

Fig. 16.4 An example of an unstructured, conventional sweet cherry orchard in Washington State, USA

16.2.2 SNAP Concept for Improving Machine-Canopy Interactions

To minimize the complexity of crop canopies, modifications and improvements of tree architecture are being investigated that can facilitate machine operations in orchards (Tombesi et al. 2017). For example, one of the optimal tree architectures for effective automated/robotic operation would be a vertical fruiting-wall system in a medium- to high-density planting, which generally offers a uniform and consistent tree structure throughout an orchard. In such an architecture, shoots, flowers, and fruits would be primarily located on the canopy surface with minimal occlusions. In actual practical field conditions, the amount of completely exposed canopy objects would vary based on how well the orchards are managed. However, such an architecture provides insight into what would be a desirable canopy structure for an orchard machine to achieve and maintain satisfactory level of efficiency and productivity.

As a configuration of an ideal orchard for improving machine-canopy interaction in tree fruit crops, SNAP (i.e., simple, narrow, accessible, and productive) concept of tree architecture could be used to describe its main features (http://treefruit.wsu.edu/orchard-management/orchard-establishment/). The core of such a SNAP orchard production system is that the horticultural practices must be suited for using mechanized operations, including field conditions, tree population and spacing, and tree canopy shape and size. Establishing favorable field conditions for machinery systems should be considered even before orchard systems are designed. Standardization of tree sizes, featured by tree height, tree shape, canopy thickness, and tree spacing within and between rows, would allow orchard machinery to operate continuously without frequent adjusting and could substantially improve throughput of mechanized orchard operations and thus profitability. In addition, a systematic approach needs to be established to understand the relationship between machine and canopy components. For example, Case Study 2 presented later (see Sect. 16.3.2) will show how a machine learning technique was implemented to identify the canopy parameters with higher relevancy for shake-and-catch apple harvesting in terms of fruit removal efficiency.

Some preliminary evidences of how SNAP tree architecture facilitates orchard operations could be provided using manual tasks. For example, in a study of hand harvest in 11 commercial sweet cherry orchards, the highest mean harvest rates (0.94 kg min^{-1} and 0.78 kg min^{-1}) were recorded in "Cowiche" and "Tieton" orchards trained to the UFO system, respectively. Comparing to some conventional orchards, high harvest efficiency in these orchards was likely the result of the SNAP architecture and that most fruit were accessible from the ground (Whiting 2018). Similar study was conducted by Zhang (2020) in 2016 based on the infield data of tracking four to eight randomly selected pickers performing manual harvest of fresh market apples in three commercial orchards in Washington State. Table 16.1 listed the results that it took approximately 43–99 s longer to harvest apples in each picking cycle (started from the time once the ladder was completely set up until the

Table 16.1 Cycle time of worker picking fresh market apples (each cycle started from the time the ladder was completely set up until the ladder was moved to another location; from Zhang 2020)

Apple cultivar	Scifresh	Fuji	Pink Lady
Tree architecture	Vertical	V-trellis	Conventional
Harvest method	Pick	Pick + cut[a]	Pick + cut[a]
Recorded picker #	4	6	8
Recorded picking cycle #	27	23	56
Avg. time per cycle (s)	91	134	190
Standard deviation (s.d.) (s)	60	176	120

[a]Cutting the apple stem

ladder was moved to another location) in conventional trees ("Pink Lady") compared to formally trained trees ("Scifresh" in vertical and "Fuji" in V-trellis).

In addition to mass mechanical fruit harvesting (He et al. 2019; Fig. 16.5a) and selective, robotic harvesting (Hohimer et al. 2019; Fig. 16.5b), this SNAP architecture strategically facilitates the adoption of mechanization and automation for many other orchard tasks such as pruning (Fig. 16.5c), blossom/fruit thinning (Bhattarai et al. 2020; Wang et al. 2013; Fig. 16.5d), and pollination (Whiting 2017; Fig. 16.5e), improves labor efficiency, and simplifies cultural practices by allowing pruning of only the secondary fruiting shoots (Ampatzidis and Whiting 2013). For example, Fig. 16.5c depicted the machine-assisted manual pruning operations in a trellis-trained apple orchard, where SNAP tree architecture enabled the self-propelled platform to move continuously and smoothly inside the tree row while the labors/robots could complete the simplified tasks in an efficient manner. To gain a deeper understanding of such machinery-canopy interactions, Case Study 1 presented in Sect. 16.2.4 will show how a certain crop/canopy management influences the effectiveness of mass mechanical harvesting of apples for fresh market.

16.2.3 Example of SNAP Tree Architectures

This section introduces more features of SNAP concept using examples to explain a few specific architectures in apple and sweet cherry orchards commonly planted in Washington State. As an example of a modern orchard design that can facilitate emerging mechanized solutions, a formally trained architecture is introduced first. Formal training is one of the commonly used trellis systems for apples in Washington State. In this trellis-trained architecture, trees are planted in a medium- to high-density (3000–4500 trees per hectare) architecture that can offer increased productivity and profitability to growers. Main tree trunks are positioned vertically, and six to eight tiers of primary branches are trained horizontally along the trellis wires in both sides of the trunk using tapes. The trellis system can be designed to create vertical (Fig. 16.6a) or V-axis (Fig. 16.6b) canopies, creating a fruiting-wall (or 2D) tree canopy. This architecture has been adopted substantially in the US Pacific

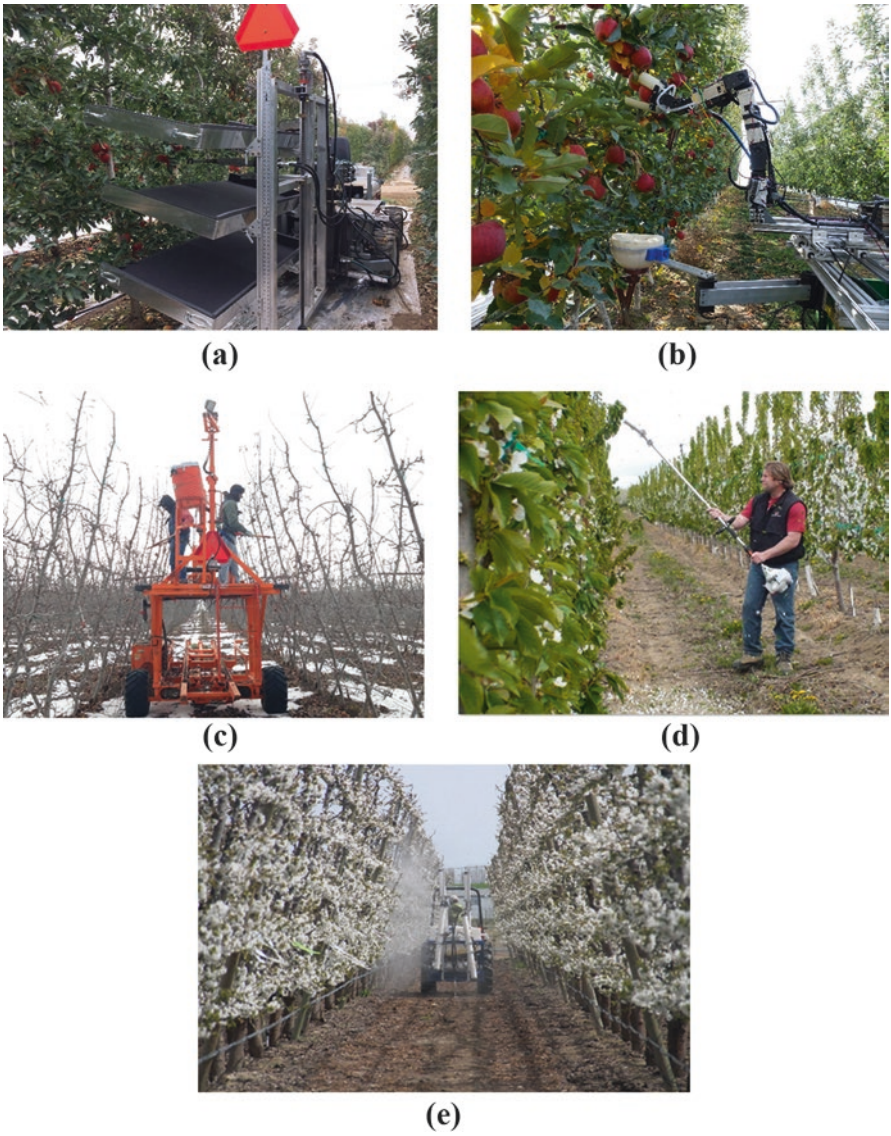


Fig. 16.5 Machine-assisted orchard operations of mass fruit harvesting (from He et al. 2019) (**a**), selective robotic fruit harvesting (from Hohimer et al. (2019)) (**b**), dormant canopy pruning (**c**), blossom thinning (from Wang et al. (2013)) (**d**), and pollen suspension spraying (from Whiting 2017) (**e**) in Washington State, USA

Northwest region because of various advantages including highly simplified, compact, and planar canopy structures that can facilitate canopy management by both labors and machines and good light penetration through the canopy with the potential for high yield and quality of fruits (Whiting 2018). Dormant and/or summer

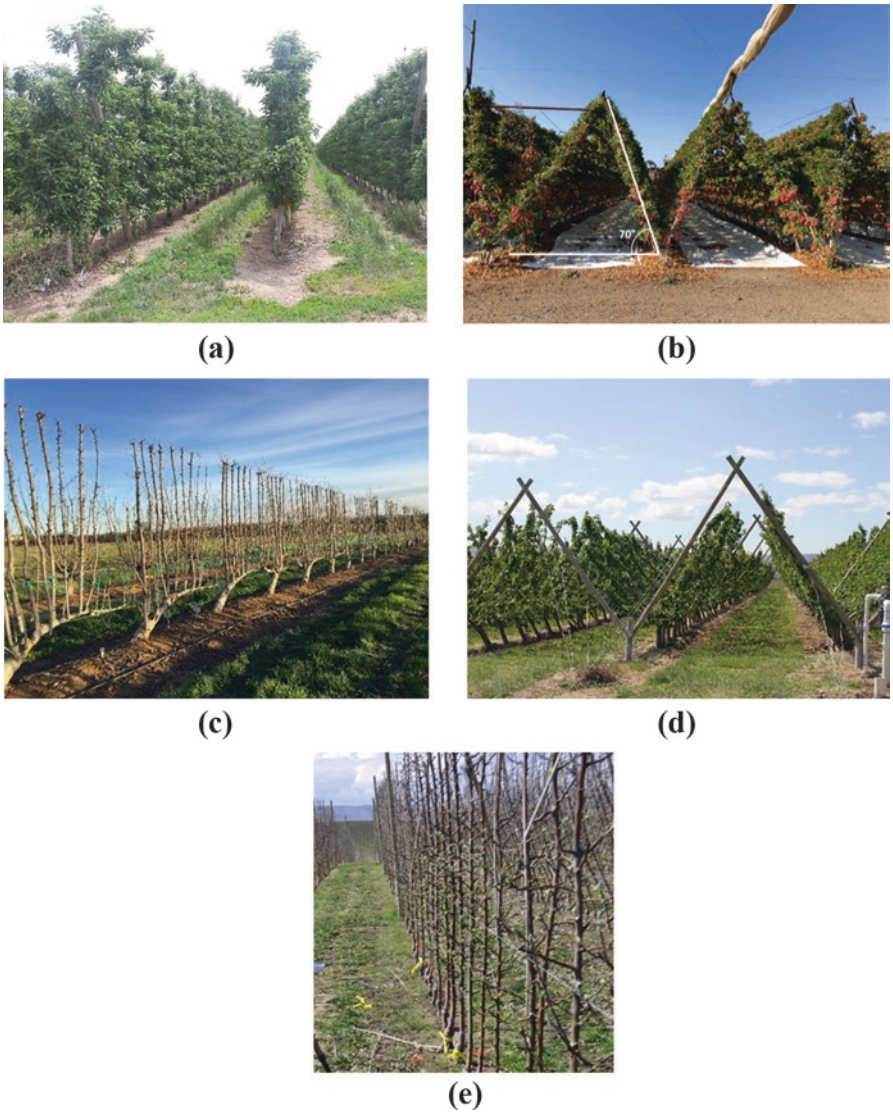


Fig. 16.6 Commercial apple orchards trained to formal vertical trellis (**a**), V-trellis (from Zhang et al. 2020a, b) (**b**), sweet cherry orchards trained to vertical Upright Fruiting Offshoot (UFO) (from Whiting 2018) (**c**), Y-trellised UFO (**d**), and apple orchard trained to tall or super spindle (from Karkee et al. 2014) (**e**) fruiting-wall architectures in Washington State, USA

pruning is normally required on the secondary fruiting shoots (the ones growing out of main, lateral branches) to maintain the compactness of the tree architecture.

Similar vertical or slightly inclined tree architectures could also be found in other tree fruit crops, such as cherries. As cherry is an exceptionally labor-intensive crop to harvest due to large tree size and a large number of smaller fruits per tree,

progress has been made toward mechanized harvesting (Larbi et al. 2015; Peterson et al. 2003). From a mass mechanical harvest research on trellised-trained "Bing" cherries in the USA, it was found that machine harvest cost was USD $0.04 per kilogram compared to $0.55 per kilogram for manual picking (Seavert and Whiting 2008). The study concluded that mechanically harvested sweet cherry orchards would be more profitable than conventional orchard systems. However, commercial adoption of the harvest system has yet to be realized because of various limitations including the ones related to orchard systems as discussed before. The mechanical harvester requires, to be efficient, the trellised tree architectures either in vertical Upright Fruiting Offshoot (UFO) (Fig. 16.6c) or Y-trellised (UFO or regular Y-axis) systems (Fig. 16.6d) that is not widespread in sweet cherry industry yet. As the name suggests, UFO architectures mainly have "upright offshoots" creating similar structure to the formally trained apples in the sense there will be one permanent wood (trunk) and several primary branches grown out of the permanent wood. The difference is that UFO would have a horizontal permanent structure and several vertical primary branches, whereas formal apple canopies would have vertical permanent structure. Mostly, UFO sweet cherry orchards in Washington State have been planted at a certain angle to the ground surface. However, special cases with vertical UFO canopies (fruiting surface angled at 90° to the ground; Fig. 16.6b) can also be commonly found. As they are similar, UFO architecture offers similar benefit for automated field operations as the ones offered by a formal apple tree architecture.

Further research with a similar shake-and-catch cherry harvesting system showed a potential for improving fruit removal rates with multiple harvest passes in a Y-trellised architecture (He et al. 2015). The study has shown that branch angle in Y-trellised fruiting walls is also important factor for efficient mechanical harvest and reducing fruit damage. Y-trellised sweet cherry orchards at Washington State University (WSU) research plots trained to 60° from horizontal were used as example systems to reduce the fruit drop height to the harvester catching surface placed underneath the canopies and to reduce the potential for fruit-branch impact during harvest. In this system, fruit removal and recovery rates were comparatively higher compared to steeper angle of fruiting walls, and fruit damage was comparable to that from hand picking (Ampatzidis and Whiting 2012; Peterson et al. 2003).

In addition to formal and UFO architectures described above, a random canopy architecture (Fig. 16.6e) is also commonly adopted in commercial orchards for apples, where branches are allowed to grow in all directions from the vertical trunk and are not trained to trellis wires in any particular fashion, though tree trunks are supported by the trellis system (e.g., a tall spindle or a super spindle apple orchard). To maintain a 2D fruiting-wall structure, branches are kept as short as possible through dormant pruning and occasional summer/fall pruning (Karkee et al. 2014).

Both formal and random architectures offer the same type of accessibility to automated or robotic systems in apple production when the canopies are sufficiently narrow, especially for harvesting. It is also important to note that picking a fruit standing freely and away from branches and trunks is always easier than a clustered fruit or a fruit close to a branch. A formal architecture provides more opportunities

compared to a random architecture to avoid fruiting in the vertical trunk/leader and to keep most of the fruit hanging down a fruiting branch and be located at the surface of the canopy (Hu et al. 2020). For a shake-and-catch mass harvesting system, a formal architecture provides an opportunity to shake only targeted fruiting branches and catch the fruit just under those branches, which has the potential to keep fruit quality at a desirable level for the fresh market (He 2018).

16.2.4 Case Study 1: Canopy-Machine Interaction in Mechanical Harvesting of Apples

Due to the extensive labor requirements for harvesting fresh market fruits, there is a burgeoning demand for mechanical harvesting solutions. Among all options, vibratory shake-and-catch mechanical harvesters have a huge potential to be adopted for a wide variety of tree fruit crops, such as apples and cherries. However, none of the harvesting technologies studied in the past have been fully adopted in commercial orchards for fresh market crops due to low harvest efficiency and/or high fruit damage, which may be primarily attributed to the complexity of canopy architectures (Zhang et al. 2018a, b). A transdisciplinary research was conducted at Washington State University (WSU) to test the hypothesis that strategic dormant pruning of apple fruiting branches (trained in formal vertical tree architecture; Fig. 16.6a) can enhance fruit removal efficiency (FRE) of vibratory mechanical harvesting systems. The primary goal is to study the influence of a dormant pruning strategy (i.e., pruning all lateral branches to a maximum length) on the performance of a vibratory mechanical harvesting system.

A vibratory mechanical harvesting system (Fig. 16.5a) composed of a hydraulically powered shake-and-catch platform was designed and fabricated by the WSU team in 2016 (He et al. 2017). This platform consisted of three major components: a four-wheel hydraulically driven self-propelled orchard platform (OPS, Blueline, Moxee, WA), a hydraulically driven vibratory shaker modified from a commercial handheld reciprocating saw (MGG20016-BA1B3, Parker Hannifin Corp., Mayfield Heights, Ohio, and SP200, Stihl Inc., Virginia Beach, VA), and an in-house designed and fabricated fruit catching and collection system with two three-layered supporting frames and six catching surfaces padded with cushioning foams. The vibratory shaker was installed on a sliding mechanism that could be moved in and out to reach targeted branches. Each catching surface was 2.50 m × 1.20 m with an adjustable elevation angle.

The proposed two pruning strategies applied to the branches were maximum 15 cm (guideline 1; G1) and maximum 23 cm pruning (guideline 2; G2), respectively. When branches were treated with G1, all fruiting shoots were pruned to be no longer than 15 cm, and when branches were treated with G2, all shoots were pruned to be no longer than 23 cm (Zhang et al. 2018a, b). In a commercial operation, G2 pruning is close to the commonly applied pruning length in Pacific Northwest apple

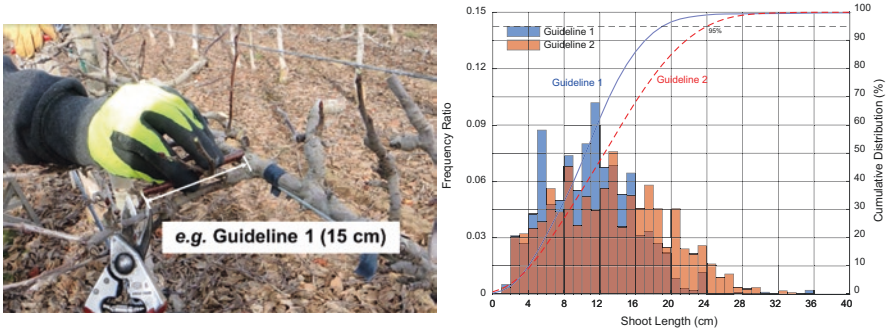


Fig. 16.7 An example of dormant pruning by skilled workers with specific guidelines (**a**) and resulted histograms and cumulative distributions for shoot length (**b**) (from Zhang et al. 2018a, b)

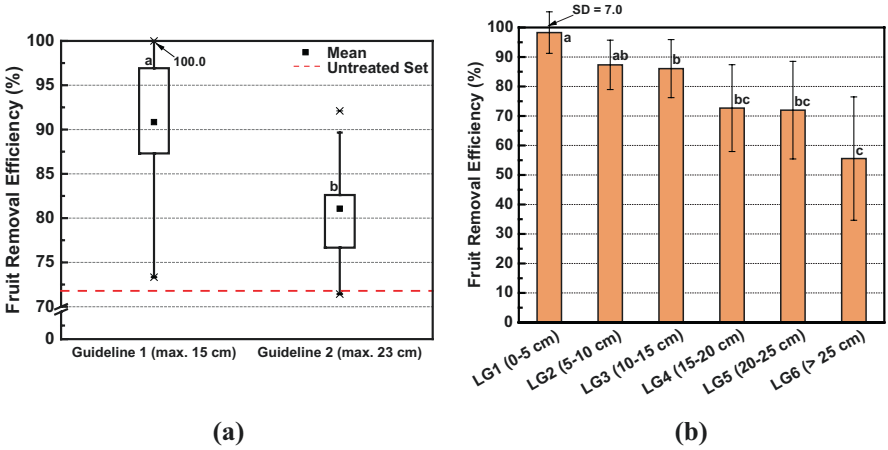


Fig. 16.8 Fruit removal efficiency (FRE) with specific pruning guidelines (**a**) and FRE with the increasing shoot length (**b**) (from Zhang et al. 2018a, b)

orchards (Fig. 16.7a). Pruning activity was performed in winter 2016 by a set of skilled orchard workers. The resulted shoot length distribution could be found in Fig. 16.7b.

The overall FRE of 91% was achieved with shoots pruned at G1, which was significantly higher than that of 81% with shoots pruned at G2 (Fig. 16.8a). With increased shoot length, FRE significantly and continuously decreased from about 98% to 56% (Fig. 16.8b). These findings verified the primary hypothesis that shorter shoots could improve the FRE without sacrificing the quality of harvested fruits in shake-and-catch harvesting of apples. Based on the results obtained in this study, an FRE of 85% or greater can be achieved if the pruning strategy of G1 is applied in vibratory mechanical harvesting. The results also showed that a minimum of 91% marketable fruit quality could be achieved for fresh market.

16.3 Technologies for Machine-Canopy Interaction Studies

Crop growth usually varies both spatially and temporally in a field, and farmers are trying to minimize farming inputs and maximize their profits by performing field operations only in the right place at the right time. To enable farmers to perform their time-sensitive, site-specific operations precisely, it requires various tools to support them in making effective decisions based on both spatial and temporal crop variabilities. As the world has witnessed rapid advancement in sensing technologies, artificial intelligence (including deep learning), computational infrastructure (including cloud computing), and robotic technologies in recent decades, various industries have been increasingly adopting smart and autonomous solutions. These technologies have also been and will continue to facilitate development and testing of smart, robotic solutions for farming including decision support tools.

Sensing is always an essential element in agricultural automation and robotics. In orchard automation, sensors are frequently used to measure the microclimate, quantify tree-absorbed sunlight, detect fruit location in tree canopies, and monitor fruit development, in addition to measuring soil properties and various crop stresses (Zhang 2018). For example, satellite-, aerial-, or ground vehicle-based crop monitoring or scouting technologies provide farmers with the capability of obtaining adequate spatial and temporal resolution of field data for various precision agriculture applications, such as plant biotic/abiotic stress monitor or disease control (more in Chaps. 5 and 6). Attributing to those agricultural industry accomplishments, the practice of investigating machinery-canopy interaction has been making some progresses.

16.3.1 Information Technologies for Decision-Making in Machine-Canopy Interaction

Information technology-based computational and sensing applications have become a trend in agriculture, including the utilization of global positioning systems (GPS; see Chap. 3), satellite and/or unmanned aerial systems (UASs), geographic information system (GIS), machine/computer vision (see Chaps. 2 and 3), artificial intelligence (AI) (including deep learning such as convolutional neural networks or CNNs; see Chap. 13), and infrared/multi-/hyper-spectral optical sensing systems (see Chap. 4), among others. Some of the technologies could be incorporated and used for multiple purposes. Figure 16.9a showed an example of using an UAS equipped with various sensing elements, such as thermal and/or multi-spectral cameras, scouting a commercial UFO sweet cherry orchard to monitor the water stress on the crop field. The area in the center of Fig. 16.9b has been identified as water stressed as shown by the higher temperature in those trees compared to others. Such stresses could be easier to be observed or estimated due to the uniform fruiting-wall structure of UFO tree architecture. If conventional cherry trees were used, the

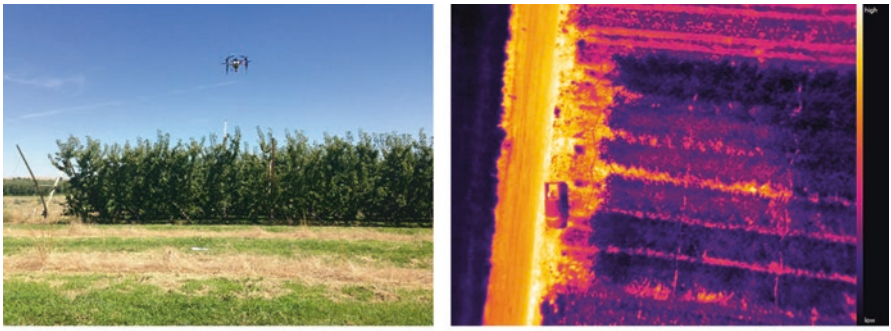


Fig. 16.9 Examples of using an unmanned aerial system (UAS) equipped with various sensors (e.g., thermal and multispectral cameras) scouting a commercial UFO sweet cherry orchard (**a**) and the corresponding thermal image for monitoring the water stress of the trees (i.e., lighter area in the center of the image) (**b**) in Washington State, USA

temperature at only the top of the canopy would be monitored in the orchards, which could heavily lower the accuracy and reliability of the monitoring process.

Information technology could always help the researchers to see the facts beyond the existing phenomenon between machineries and crops and to obtain more accurate and spatially consistent data for better understanding and/or decision-making in real time. The following two case studies will exemplify how information technologies could help with understanding and investigating machinery-canopy interactions in tree fruit harvesting.

16.3.2 Case Study 2: Machine Learning in Canopy Parameter Identification for Mechanical Apple Harvesting

As discussed in Case Study 1 (Sect. 16.2.4), a mass mechanical harvesting approach for apples offers an alternative and promising solution. In addition to harvester design elements, it is equally important to understand the key canopy parameters of apple trees as they are closely integrated and interact with each other during the harvesting process (Zhang et al. 2020a, b). The shaking process when such interactions occur was normally very short (i.e., a few seconds). The primary goal of this research was, therefore, to identify the most relevant canopy parameters affecting the fruit removal efficiency of mechanical harvesting of apples using supervised machine learning algorithm and principal component analysis (PCA). The study was conducted with vertically trained "Scifresh" (Fig. 16.6a) and V-trellised "Envy" (Fig. 16.6b) apple trees. Figure 16.10 showed a typical canopy structure in a modern commercial apple orchard during harvest season, where 11 physically measured canopy parameters were identified, which included 4 branch parameters, 4 fruit parameters, and 3 shoot parameters.

A complete definition of each parameter is provided as follows: (1) branch length, denoted as "BLength," refers to the full length of the branch from the base to the end; (2) branch basal diameter, denoted as "BBasalD," refers to the diameter of the base of the branch; (3) branch middle diameter, denoted as "BMiddleD," refers to the diameter of the middle of the branch; (4) branch end diameter, denoted as "BEndD," refers to the diameter of the end of the branch; (5) fruit load, denoted as "FLoad," refers to the fruit number per branch; (6) fruit density, denoted as "FDensity," refers to the fruit number per centimeter of the branch; (7) fruit location, denoted as "FLocation," refers to the distance from the fruit to the vibrating location of the branch; (8) fruit single mass, denoted as "FSingleMass," refers to the mass of a single fruit; (9) shoot length, denoted as "SLength," refers to the full length of the shoot from the base to the end; (10) shoot basal diameter, denoted as "SBasalD," refers to the diameter of the base of the shoot; (11) shoot index, denoted as "SIndex," refers to the ratio of a shoot basal diameter to its length (Zhang et al. 2020a, b).

Figure 16.11 shows four examples of the actual probability distributions of the manually measured canopy parameters in terms of "mechanically harvested" and "mechanically unharvested" apples when harvested with a mechanical shaking system. The distributions may indicate the possible relationship between certain canopy parameters and fruit removal results. For example, some parameters (e.g., "FLoad" – fruit load per branch) showed noticeable differences in actual distributions between "harvested" and "unharvested" apples as presented in Fig. 16.11b, indicating they might heavily influence the harvest result. While some other parameters (e.g., "FLocation" – the distance from fruit location to the branch base) were almost completely overlapped as can be seen in Fig. 16.11c, which suggested that these parameters may not substantially affect the harvest outcomes.

Supervised machine learning techniques such as support vector machines (SVM), decision trees, and *k*-nearest neighbors (*kNN*) classifiers are commonly used in data classification and regression studies for many other applications

Fig. 16.10 A typical canopy structure in a modern commercial apple orchard during harvest season, where 11 physically measured canopy parameters were identified, which include (1) 4 branch parameters, (2) 4 fruit parameters, and (3) 3 shoot parameters (from Zhang et al. 2020a, b)

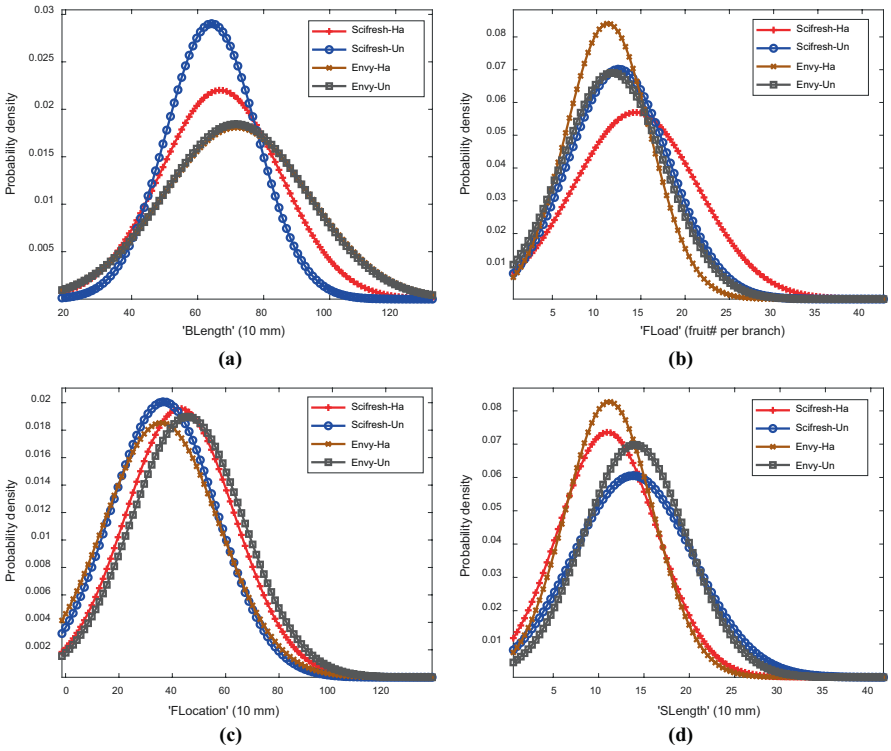


Fig. 16.11 Four examples of actual probability distributions of manually measured canopy parameters (i.e., a branch parameter (**a**), two fruit parameters (**b**–**c**), and a shoot parameter (**d**)) in terms of mechanically "harvested (-Ha)" and "unharvested (-Un)" apples in mass mechanical harvest (Adapted from Zhang et al. 2020a, b)

(Chlingaryan et al. 2018; Gongal et al. 2015). Unlike unsupervised machine learning (where only unidentified clusters of the dataset are created), supervised machine learning models use input-output dataset of known object classes or systems to "learn the pattern" from example responses (more details on Chap. 13). Application of supervised learning for identification of canopy parameters influencing fruit removal efficiency included three major steps of (i) data preparation, (ii) model training, and (iii) model testing.

As illustrated in Fig. 16.12, first, all ground-truth data points of canopy parameters were standardized into zero mean using the technique introduced by Breiman (2001) and then were used as system inputs for two apple cultivars used in this work. Principle component analysis (PCA) was applied to reduce the number of dimensions in dataset before the use of supervised machine learning. The preprocessed data were then used to train and test selected supervised learning technique; 85% of randomly selected data samples were used for model training, and the remaining 15% of the samples were used as new dataset for model testing. Binary classes were used as the known responses to evaluate the accuracy in predicting the

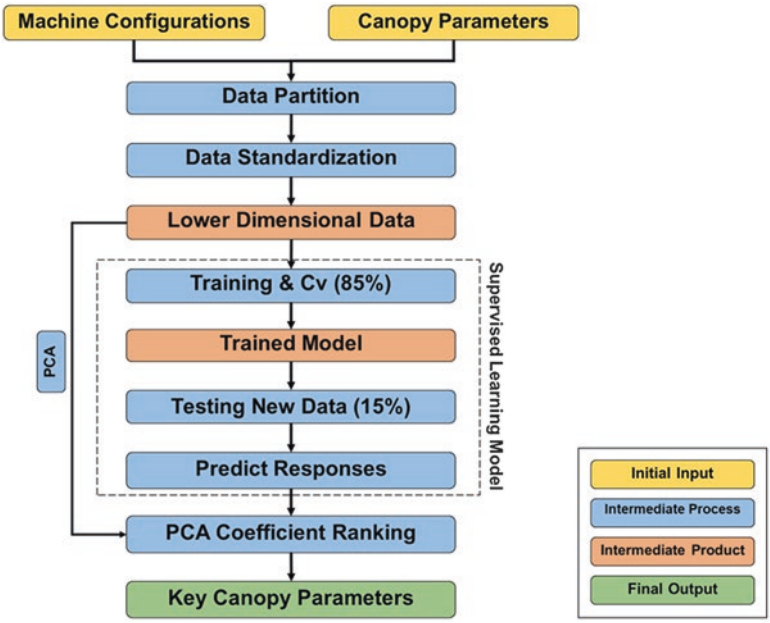


Fig. 16.12 Overall flowchart of identifying key canopy parameters using a supervised machine learning model and principal component analysis (PCA) (Adapted from Zhang et al. 2020a, b)

results. In addition, PCA was then used to examine the cumulative explained variances and coefficients of principal components (PCs), where cumulative explained variance represents the interpretation of the PCs against the entire dataset being explained. Once the model testing was completed, a few PCs of canopy dataset were identified by those PCs (Wold et al. 1987), which were used to determine key canopy parameters based on ranked coefficients of PCs.

In the canopy parameter dataset used in Case Study 2, it was assumed that if the target apples were physically alike (with similar geometric parameters) or were located at canopy areas with similar canopy parameters, the similar learning weights could be assigned to them as they would have similar likelihood of being removed (or not) during shake-and-catch harvesting. Based on this assumption, *kNN* learning algorithm was first considered due to its outstanding performance (Kurtulmus et al. 2014) in classifying the objects based on the classes their nearest neighbors belong to and its predictive assumption that the objects near each other share similar characteristics. After some trial and error, a weighted *kNN* (*w-kNN*) was finalized as the modeling framework for this study. Bayesian optimization algorithm was used to optimize the hyper-parameters of the *w-kNN* model in making skillful predictions (Snoek et al. 2012). The optimization procedure was completed (*EI(x,Q)*, Eq. 16.1) over 30 distance metrics of evaluations, such as "spearman," "hamming," "jaccard," and "cityblock,"

$$EI\left(x,Q\right)=E_Q\left[\max\left(0,\mu_Q\left(x_{\text{best}}\right)-f\left(x\right)\right)\right] \tag{16.1}$$

which evaluates the expected improvement in the objective function (f_{obj}) and ignores the values that could cause an increase in the function. x_{best} and $\mu_Q(x_{\text{best}})$ represent the location of the lowest posterior mean and the lowest value of the posterior mean, respectively.

The minimum observed and estimated values were compared by evaluating the objective function (f_{obj} = log(1 + cross validation loss)), where the function was expected to be minimized close to zero. Five distance metrics were then selected as the best evaluation measures (with minimum values of f_{obj} and shorter computational time). Finally, the most feasible distance metric was determined as "city-block" (Eq. 16.2). This distance metric was used to locate the nearest neighbors in *w-kNN* due to its minimum number of neighbors (k = 1) and estimated f_{obj} value (minimum errors = 0.187) with short computational time of 0.15 s

$$d_{st}=\sqrt[p]{\sum_{j=1}^{n}\left|x_{sj}-x_{tj}\right|^p} \tag{16.2}$$

where d_{st} represents the distance between two row vectors (sum of the absolute difference; each row is one feature vector forming one test sample) in Cartesian coordinates for a random row vector x_s and another random row vector x_t in a given m-by-n data matrix (s = 1, 2, …, m; and t = 1, 2, …, m; where s and t are different values), n = 11, and p = 1.

Using this technique, overall, 2678 ground-truth data points were classified into two classes of fruit removal status: mechanically "harvested" and "unharvested" apples, and test accuracy achieved was up to 91%. Assuming a parameter coefficient in PCs greater than 0.5 as being highly relevant (based on the empirical studies (Jolliffe 2011)), canopy parameters including "fruit load per branch," "branch basal diameter," and "shoot length" showed higher relevancy for shake-and-catch apple harvesting technology in terms of fruit removal in "Scifresh" apples (Table 16.2; Zhang et al. 2020a, b).

The results, in general, demonstrated that the development of mass mechanical harvesting technology should always be pursued in close interaction with the optimization of crop/canopy architecture, as canopy parameters play a critical role in the overall success of the harvesting technology. More specifically, results indicated that different canopy parameters respond differently to the shaking signal used in shake-and-catch harvesting (Zhang et al. 2020a, b). Results also suggested that the higher fruit load/density with larger basal diameter of branch and shorter fruiting offshoot could potentially result in a higher mechanical harvesting efficiency as observed from the probability density of samples collected for mechanically "harvested" and "unharvested" apples. The key canopy parameters identified in Case Study 2 could potentially be considered to create guidelines for crop/canopy management for improved harvesting efficiency.

Table 16.2 Coefficients of the first five principal components (PC1–PC5) for "Scifresh" with 11 canopy parameters

Scifresh	Parameters[a]	PC1 (29.9%)	PC2 (18.4%)	PC3 (14.4%)	PC4 (9.9%)	PC5 (7.6%)
1	**BLength**	0.398	−0.279	**−0.543**	0.073	−0.012
2	BBasalD	0.382	0.327	−0.125	−0.285	0.271
3	BMiddleD	0.360	0.491	−0.160	−0.201	0.168
4	**BEndD**	0.212	**0.593**	0.161	0.340	−0.321
5	**FLoad**	**0.542**	−0.340	0.205	0.000	0.004
6	**FDensity**	0.417	−0.227	**0.632**	−0.030	−0.049
7	FLocation	0.218	−0.177	−0.424	0.218	−0.441
8	FSingleMass	−0.013	0.132	0.014	0.389	−0.188
9	**SLength**	0.063	−0.018	0.020	**0.722**	**0.505**
10	SBasalD	0.035	0.078	0.123	0.096	−0.404
11	SIndex	−0.008	0.017	0.023	−0.166	−0.381

[a]A parameter with an absolute value of coefficient above 0.5 (in bold type) was deemed highly relevant

Adapted from Zhang et al. (2020a, b)

16.3.3 Case Study 3: Machine Vision in Mechanical Apple Harvesting

As discussed before, development and adoption of mechanical harvesting solutions (e.g., shake-and-catch systems) is essential for addressing the challenge of uncertain availability of seasonal semiskilled labor around the world. One of the limitations of shake-and-catch harvesting systems developed in the past has been the time taken to position the shaking end-effector and the catching device at appropriate locations within tree canopies. To address this challenge, some studies have been conducted in the past by a research team at Washington State University (Zhang et al. 2018a, b). This study provided a capability for automated branch detection using deep learning-based computer vision system; however, one of the drawbacks is that the study was conducted in the dormant season. In order to apply this method during the harvest season under the full foliage conditions, an additional positioning system (e.g., GPS or LiDAR) and the attitude sensor will be needed to relocate the movement path that was used during the dormant season. An alternative and more efficient way is to directly detect the target branches during harvest seasons (e.g., Amatya and Karkee 2016), which is more challenging due to the heavy occlusion from leaves and fruit clusters. To further address this challenge, the potential of using a computer vision system based on a deep learning technique will be discussed in Case Study 3 for automated identification and localization of desired geometric features of tree canopies during harvest season (Zhang 2020; Zhang et al. 2019a, b).

A large number of studies with the application of deep learning have been conducted in agriculture in recent years. Detailed discussion on this technique, as mentioned before, can be found in Chap. 13. Reported studies are focused around image

processing for agricultural applications due to its outstanding accuracy and robustness (e.g., ~41% higher overall classification accuracy compared to most of the conventional image processing algorithms; Kamilaris and Prenafeta-Boldú 2018). Convolutional neural networks (CNNs) are one of the most applied deep learning techniques due to their capabilities in processing high-resolution image data with reasonable computational time, which is made possible by network weight sharing among numerous convolutional layers. The primary goal of Case Study 3 is to precisely identify and locate the tree branches/trunks and to estimate suitable shaking locations in dense foliage tree canopies for automating mass mechanical harvesting systems for apples.

A Kinect imaging sensor (Kinect V2, Microsoft Inc., Redmond, WA) that consisted of red-green-blue (RGB), depth, and infrared channels was used in this study (Fig. 16.13a), which is both relatively stable in outdoor environment and economically affordable. The RGB camera images are helpful in object detection with color and other associated features. The depth camera, which operates in Time-of-Flight principle (see Chap. 3 for more details), recorded 3D information of the scene using infrared laser signals (Zhang et al. 2020a, b; Fig. 16.14a). The maximum effective pixel resolution of Kinect for RGB sensor was 1920 × 1080 and for depth sensor was 512 × 424. A customized platform mounted on an electric Toro Utility Vehicle was used for image acquisition. The camera was positioned orthogonal to the target canopies in both V-axis and vertical tree architecture systems. The distance from the camera to the center of the target canopies was maintained around 1.1–1.2 m to optimize the visualization of the tree trunks and branches (Fig. 16.13b) (Zhang 2020).

Some preprocessing steps have been applied to the acquired images before using them as the inputs to CNNs. For example, the image background was removed by applying a depth threshold (Fig. 16.13c). Images were then annotated to group pixels into one of the four different classes of interest: (i) tree branches, (ii) apples, (iii) tree trunks, and (iv) background (mostly leaves) (Fig. 16.14b). In total, a dataset of 674 "Fuji" images were collected, which were then divided into 70%, 15%, and 15%, respectively, for network training, validation, and testing.

In this study, pretrained deep learning networks were adopted and fine-tuned to the apple canopy images for semantic segmentation of individual pixels into specific classes (Fig. 16.13d). One of the most efficient CNNs was used in this study: DeepLab v3 + ResNet-18 (i.e., one of the typical directed acyclic graph (DAG) networks; originally with 72-layer of ResNet-18) (Chen et al. 2017, 2018). ResNet is a CNN developed by He et al. (2016) and was the winner of 2015 ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This model extensively utilizes the batch normalization layers (to accelerate the network training) but lacks fully connected layers (layers that have full connections to all activation channels in the previous layer) at the end of the architecture. Figure 16.15 visualizes the overall architecture of the modified DeepLab v3+ ResNet-18 (with 101-layer after DeepLab v3 was added; abbreviated as ResNet-18 in the following content) and the activation channels of the convolutional layers used in this work. The entire architecture can be divided into 16 processing blocks, from the initial steps of feeding RGB-D images to the network at B1, to the final steps in B16 where the last layer of the

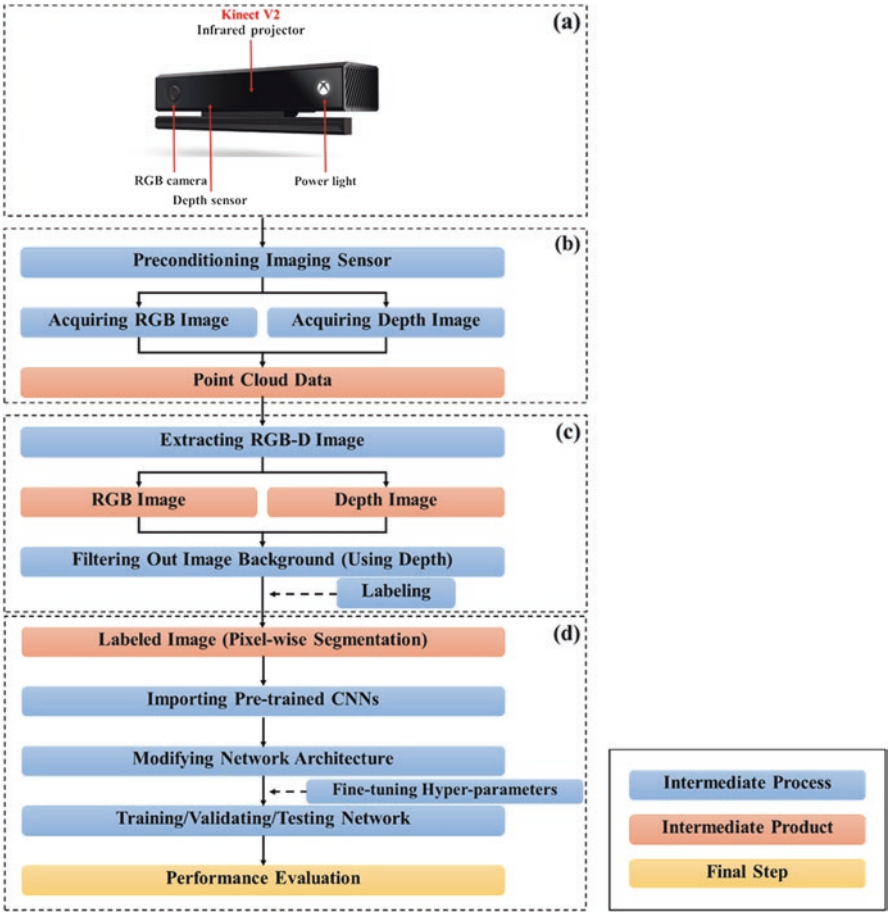


Fig. 16.13 Overall machine vision pipeline for branch detection including a Kinect V2 imaging sensor (**a**), image acquisition (**b**), preprocessing (**c**), and applications of the convolutional neural networks (CNNs) in processing the collected data (**d**) (from Zhang 2020)

(a) **(b)**

Fig. 16.14 An illustration of canopy point cloud data acquired from the orchard (**a**) and corresponding pixel-wise annotated (ground-truth) image (**b**) (from Zhang 2020)

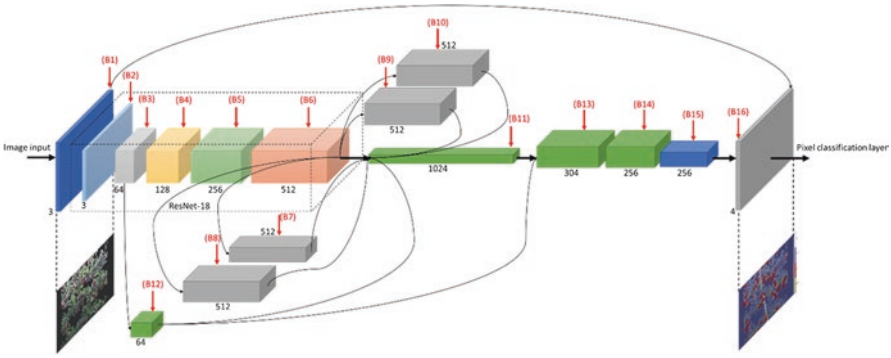


Fig. 16.15 Convolutional neural network (CNN) architecture used in Case Study 3 based on DeepLab v3+ ResNet-18 (Adapted from Zhang 2020)

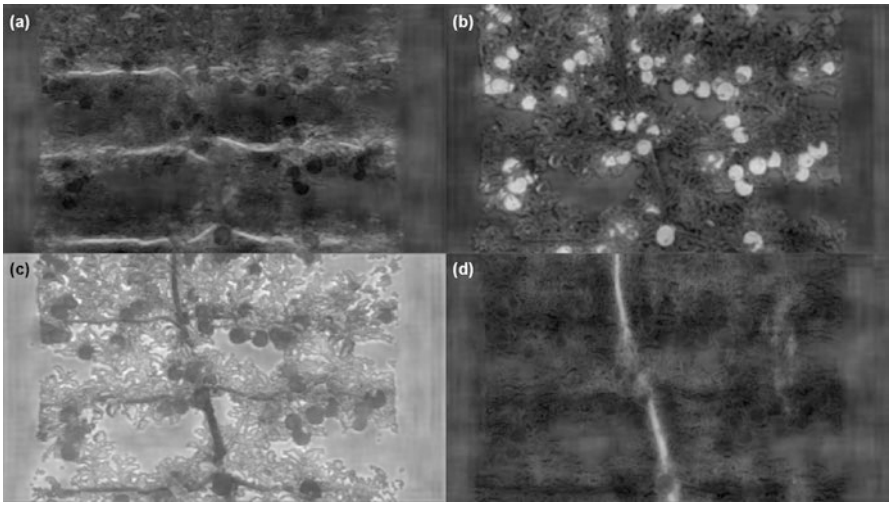


Fig. 16.16 Positive activation channels for four classes: (**a**) "branches," (**b**) "apples," (**c**) "leaves," and (**d**) "trunks" at "B15" in Fig. 16.15 of DeepLab v3+ ResNet-18 (from Zhang 2020)

network maps the non-normalized output to a probability distribution of the predicted output classes. Eventually, all positive activation channels for four classes of "branches" (Fig. 16.16a), "apples" (Fig. 16.16b), "leaves" (Fig. 16.16c), and "trunks" (Fig. 16.16d) were displayed, which confirmed that the modified ResNet-18 was working effectively to segment out all classes of interest by automatically learning their features.

The performance of ResNet-18 model was evaluated using per class accuracy (PcA), which measures the proportion of correctly classified pixels (Fig. 16.17a). In addition, intersection over union (IoU) (a measure of the overlaps between predicted classes and the ground-truth) and boundary-F1 score (BFScore; a measure of

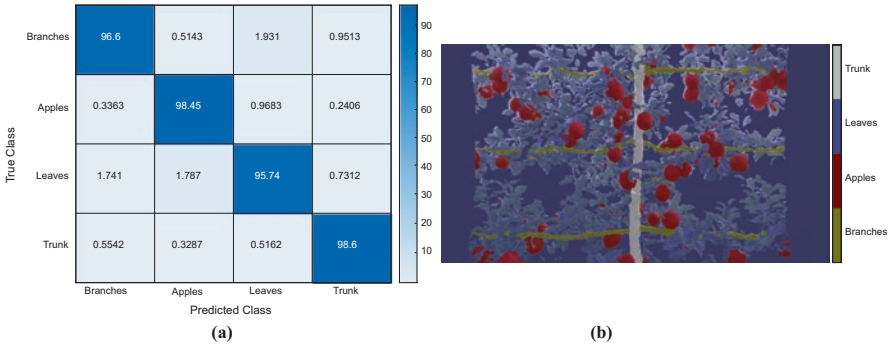


Fig. 16.17 Normalized confusion matrix comprising the pixels in the true class and the predicted class based on the segmentation results generated by modified DeepLab v3+ ResNet-18 model (**a**) and an example test result of segmentation model used (**b**) (from Zhang 2020)

how well the boundaries were preserved) were used. When the ResNet-18 model was tested with full resolution images, a mean PcA of 97%, a mean IoU of 0.69, and a mean BFScore of 0.89 per image basis were achieved for images collected in a V-trellised "Fuji" apple orchard. The network performance on per class basis was also good in segmenting "branches" and "trunks" out (Fig. 16.17b). For example, the IoUs for "branches" and "trunks" were 0.40 and 0.63, individually, and the same were 0.78 and 0.96 for "apples" and "leaves" (Zhang 2020). The results were considered satisfactory because they referred to a 57% and 77% overlap between predicted and ground-truth segments for branches and trunks, which meant that the actual trajectories of branches and trunks could be described within the tolerance of an end-effector of the shaking harvester.

16.4 Summary and Concluding Thoughts

Fruit crop production is a highly competitive agricultural industry worldwide with growers seeking more effective and efficient methods to precisely and automatically (when possible) manage their orchard operations to remain competitive in the international market. Also, there is little doubt that SNAP (simple, narrow, accessible, productive) fruit tree/canopy architectures will be favored in the next decades due to their excellent light interception, uniform distribution of fruit and other canopy parts, machine-friendlier configurations, and improved production efficiencies. However, development of any orchard machineries without understanding its interaction with canopies would lead to a suboptimal or even impractical solutions. At the same time, lack of understanding and consideration of desired canopy features for machine operations during new canopy development is going to lead to a canopy system that may be inefficient for mechanization, automation, and robotic operations. As we continue to develop and adopt advanced mechanization and

automation technologies and improved canopy architectures in fruit crop farming, it's more important than ever to understand various aspects and extents of machinery-canopy interactions and utilize such understanding in future research and development.

This chapter provided a unique perspective on discussing how the efficiency of overall systems could be maximized in tree fruit crops (e.g., apples and sweet cherries) by putting both biological/horticultural and engineering considerations together. The preliminary studies and experiences of the authors indicated a positive impact of various cropping system practices and operations on improving the effectiveness, applicability, and feasibility of orchard machinery technologies.

Traditionally, scientists and engineers primarily aimed at improving the efficiency of the orchard machineries without much consideration of the potential effects regarding the crop architectures and canopies. It is important to note that mechanical or robotic orchard machinery could only operate at maximum productivity and efficiency in high-density orchard blocks with organized, uniform, and accessible fruit tree architectures. In fact, the prospect of designing orchards to readily adopt significant automation, mechanization, and/or robotics technologies in tree fruit production is compelling – adoption to date has lagged behind other agricultural industries despite the high labor requirement for tree fruit production. It may be argued that one of the key factors inhibiting the utilization of automation technologies is the aged and complex and largely random orchard systems that are common in the majority of fruit-growing regions around the world.

For example, the concept of mass mechanical harvesting technology has been studied for decades since the early 1960s. However, no commercial success has been achieved yet for fresh market fruit harvesting. In recent years, development and adoption of formal tree architecture (Figs. 16.5a and 16.18; see Sect. 16.2.2 for more details) orchards provided a great opportunity for further developing such harvesting technology. In this architecture, most of the fruits would grow along the horizontally trained branches and be present at the surface of the canopy. As illustrated in Figs. 16.5a and 16.18, a multilayer harvesting approach could be used for mass mechanical harvesting that can be confined within the target branches. Such tree architectures offer an environment for achieving the improved fruit removal efficiency and fruit quality with the multilayer, targeted shake-and-catch harvesting machine by vibrating the individual branches with optimal frequency and amplitude and catching fruit right underneath, thus decreasing the likelihood of fruit damage.

Inspired by the potentials of crop canopies developed with due consideration of various aspects of machinery-canopy interactions in apples and sweet cherries as discussed above, other tree fruit growers (such as citrus growers) have also started planting and experimenting with trellised crop systems in California, Florida, and Israel. Undoubtedly, machine-canopy interaction is an emerging topic of research in agricultural mechanization and robotics, which lays the foundation for achieving effective mechanized/automated perennial crop production.

In future work, systematic research efforts should be directed toward closer and deeper understanding and consideration/implementation of interactions and integrations between orchard machinery and tree fruit crops to determine the most suitable

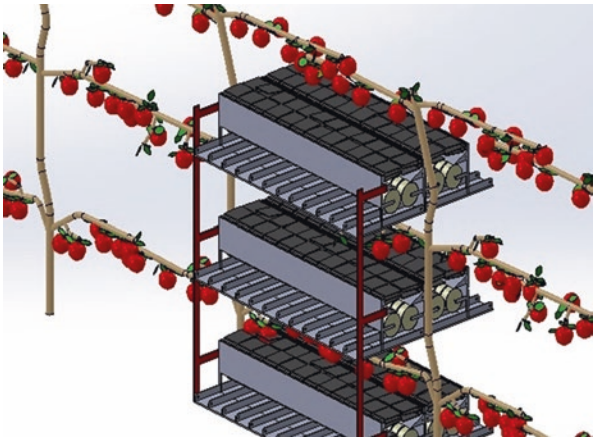


Fig. 16.18 An illustration of a trellis-trained, formal, fruiting-wall tree architecture, which is considered well-suited for multilayer, targeted shake-and-catch mechanical apple harvesting. More details on formal architecture can be found in Sect. 16.2.2. With this architecture, most of the fruits would grow along the horizontally trained branches and be present at the surface of the canopy (from Zhang 2020)

site-specific setups and horticultural practices for further development of orchard machinery/automation. During this process, the standardized criteria of orchard/canopy configuration could be established for different tree fruit crops in different fruit-growing regions worldwide. In addition, mechanized or automated solutions should also be developed to create and maintain desired crop and canopy management architecture (such as robotic tree training and pruning) so that issues related to unreliable seasonal human labor sources and rapidly increasing labor costs would not impact the capability in creating machine-friendly canopies (Zhang et al. 2019a, b).

Where to Look for Further Information

Currently, researchers at Washington State University (WSU), Center for Precision and Automated Agricultural Systems (CPAAS), have been working on the related topics of machinery-canopy interactions on tree fruit crops, specifically for apples and sweet cherries (http://cpaas.wsu.edu/). Some publicly accessible image datasets (e.g., relevant to Sect. 16.3.3) could be downloaded from WSU #AgRobotics Laboratory Research Exchange common portal (https://research.libraries.wsu.edu/xmlui/handle/2376/17718). The readers could refer to the following book chapters, PhD dissertations, and journal or conference publications to look for further information.

- Karkee, M., Silwal, A. & Davidson, J.R. (2017). Mechanical Harvest and In-field Handling of Tree Fruit Crops. In *Automation in tree fruit production: Principles and practice*.
- Whiting, M. D. (2018). Precision orchard systems. In *Automation in Tree Fruit Production: Principles and Practice* (pp. 75–93). CABI Wallingford, UK.

- Zhang, X. (2020). Study of canopy-machine interaction in mass mechanical harvest of fresh market apples. PhD Dissertation. Pullman, USA: Washington State University, Department of Biological Systems Engineering.
- Zhang, X., Fu, L., Karkee, M., Whiting, M. D., & Zhang, Q. (2019). Canopy Segmentation Using ResNet for Mechanical Harvesting of Apples. *IFAC-PapersOnLine*, *52*(30), 300–305.
- Zhang, X., He, L., Karkee, M., Whiting, M. D., & Zhang, Q. (2020). Field evaluation of targeted shake-and-catch harvesting technologies for fresh market apple. *Transactions of the ASABE, 63*(6), 1759–1771.
- Zhang, X., He, L., Majeed, Y., Karkee, M., Whiting, M. D., & Zhang, Q. (2018). A precision pruning strategy for improving efficiency of vibratory mechanical harvesting of apples. *Transactions of the ASABE*, 61(5), 1565–1576.
- Zhang, X., He, L., Zhang, J., Whiting, M. D., Karkee, M., & Zhang, Q. (2020). Determination of key canopy parameters for mass mechanical apple harvesting using supervised machine learning and principal component analysis (PCA). *Biosystems Engineering*, 193, 247–263.

References

Amatya S, Karkee M (2016) Integration of visible branch sections and cherry clusters for detecting cherry tree branches in dense foliage canopies. Biosyst Eng 149:72–81

Ampatzidis YG, Whiting M (2012) Comparing the efficiency of future harvest technologies for sweet cherry. In: I International Symposium on mechanical harvesting and handling systems of fruits and nuts, vol 965. ISHS, Leuven, pp 195–198

Ampatzidis YG, Whiting MD (2013) Training system affects sweet cherry harvest efficiency. HortScience 48(5):547–555

Bac CW, van Henten EJ, Hemming J, Edan Y (2014) Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. J Field Rob 31(6):888–911

Bhattarai U, Bhusal S, Majeed Y, Karkee M (2020) Automatic blossom detection in apple trees using deep learning. IFAC-PapersOnLine (in press)

Breiman L (2001) Random forests. Mach Learn 45(1):5–32

Chen LC, Papandreou G, Schroff F, Adam H (2017) Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587

Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV), pp 801–818

Chlingaryan A, Sukkarieh S, Whelan B (2018) Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: a review. Comput Electron Agric 151:61–69

Godin C, Costes E, Sinoquet H (1999) A method for describing plant architecture which integrates topology and geometry. Ann Bot 84(3):343–357

Gongal A, Amatya S, Karkee M, Zhang Q, Lewis K (2015) Sensors and systems for fruit detection and localization: a review. Comput Electron Agric 116:8–19

He L (2018) Evaluation of a localized shake-and-catch harvesting system for fresh market apples. Agric Eng Int CIGR J 19(4):36–44

He L (2017) Shake-and-catch harvesting for fresh market blueberries

He L, Zhou J, Zhang Q, Karkee M (2015) Evaluation of multipass mechanical harvesting on 'Skeena' sweet cherries trained to Y-trellis. HortScience 50(8):1178–1182
He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks. In: European conference on computer vision. Springer, Cham, pp 630–645
He L, Fu H, Sun D, Karkee M, Zhang Q (2017) Shake-and-catch harvesting for fresh market apples in trellis-trained trees. Trans ASABE 60(2):353–360
He L, Zhang X, Ye Y, Karkee M, Zhang Q (2019) Effect of shaking location and duration on mechanical harvesting of fresh market apples. Appl Eng Agric 35(2):175–183
Hohimer CJ, Wang H, Bhusal S, Miller J, Mo C, Karkee M (2019) Design and field evaluation of a robotic apple harvesting system with a 3D-printed soft-robotic end-effector. Trans ASABE 62(2):405–414
Hu G, Bu L, Chen J (2020) Simulation to determination of significant parameters on apple stress for combing harvesting in trellis trained trees. Sci Hortic. in press
Jolliffe I (2011) Principal component analysis. In: International encyclopedia of statistical science. Springer, Berlin, pp 1094–1096
Kamilaris A, Prenafeta-Boldú FX (2018) Deep learning in agriculture: a survey. Comput Electron Agric 147:70–90
Karkee M, Adhikari B, Amatya S, Zhang Q (2014) Identification of pruning branches in tall spindle apple trees for automated pruning. Comput Electron Agric 103:127–135
Kurtulmus F, Lee WS, Vardar A (2014) Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network. Precis Agric 15(1):57–79
Larbi PA, Karkee M, Amatya S, Zhang Q, Whiting MD (2015) Modification and field evaluation of an experimental mechanical sweet cherry harvester. Appl Eng Agric 31(3):387–397
Mehta SS, Burks TF (2014) Vision-based control of robotic manipulator for citrus harvesting. Comput Electron Agric 102:146–158
Peterson DL, Bennedsen BS, Anger WC, Wolford SD (1999) A systems approach to robotic bulk harvesting of apples. Trans ASAE 42(4):871
Peterson DL, Whiting MD, Wolford SD (2003) Fresh–market quality tree fruit harvester part I: sweet cherry. Appl Eng Agric 19(5):539
Seavert CF, Whiting MD (2008) Comparing the economics of mechanical and traditional sweet cherry harvest. In: IX International Symposium on integrating canopy, rootstock and environmental physiology in orchard systems, vol 903. ISHS, Leuven, pp 725–730
Silwal A, Davidson JR, Karkee M, Mo C, Zhang Q, Lewis K (2017) Design, integration, and field evaluation of a robotic apple harvester. J Field Rob 34(6):1140–1159
Sistler F (1987) Robotics and intelligent machines in agriculture. IEEE J Robot Autom 3(1):3–6
Snoek J, Larochelle H, Adams RP (2012) Practical Bayesian optimization of machine learning algorithms. In: Advances in neural information processing systems. NIPS, Lake Tahoe, pp 2951–2959
Tombesi S, Poni S, Palliotti A, Farinelli D (2017) Mechanical vibration transmission and harvesting effectiveness is affected by the presence of branch suckers in olive trees. Biosyst Eng 158:1–9
Wang M, Wang H, Zhang Q, Lewis KM, Scharf PA (2013) A hand-held mechanical blossom thinning device for fruit trees. Appl Eng Agric 29(2):155–160
Wang H, Hohimer CJ, Bhusal S, Karkee M, Mo C, Miller JH (2018) Simulation as a tool in designing and evaluating a robotic apple harvesting system. IFAC-PapersOnLine 51(17):135–140
Whiting MD (2018) Chapter 6: precision orchard systems. In: Zhang Q (ed) Automation in tree fruit production: principles and practice. CABI, Wallingford, pp 93–111
Whiting MD (2017) Advancing precision pollination systems to improve yield
Wold S, Esbensen K, Geladi P (1987) Principal component analysis. Chemom Intell Lab Syst 2(1–3):37–52
Zhang Q (2018) Chapter 1: tree fruit production automation. In: Zhang Q (ed) Automation in tree fruit production: principles and practice. CABI, Wallingford, pp 1–12

Zhang X (2020) Study of canopy-machine interaction in mass mechanical harvest of fresh market apples. PhD Dissertation. Washington State University, Department of Biological Systems Engineering, Pullman

Zhang J, He L, Karkee M, Zhang Q, Zhang X, Gao Z (2018a) Branch detection for apple trees trained in fruiting wall architecture using depth features and Regions-Convolutional Neural Network (R-CNN). Comput Electron Agric 155:386–393

Zhang X, He L, Majeed Y, Whiting MD, Karkee M, Zhang Q (2018b) A precision pruning strategy for improving efficiency of vibratory mechanical harvesting of apples. Trans ASABE 61(5):1565–1576

Zhang Q, Karkee M, Tabb A (2019a) The use of agricultural robots in orchard management. arXiv preprint arXiv:1907.13114

Zhang X, Fu L, Karkee M, Whiting MD, Zhang Q (2019b) Canopy segmentation using ResNet for mechanical harvesting of apples. IFAC-PapersOnLine 52(30):300–305

Zhang J, Karkee M, Zhang Q, Zhang X, Majeed Y, Fu L, Wang S (2020a) Multi-class object detection using faster R-CNN and estimation of shaking locations for automated shake-and-catch apple harvesting. Comput Electron Agric 173:105384

Zhang X, He L, Zhang J, Whiting MD, Karkee M, Zhang Q (2020b) Determination of key canopy parameters for mass mechanical apple harvesting using supervised machine learning and principal component analysis (PCA). Biosyst Eng 193:247–263

Index

M. Karkee, Q. Zhang (eds.), *Fundamentals of Agricultural and Field Robotics*,
Agriculture Automation and Control, https://doi.org/10.1007/978-3-030-70400-1