Tofael Ahamed   *Editor*

IoT and AI in Agriculture

Self- sufficiency in Food Production to
Achieve Society 5.0 and SDG's Globally

Springer

IoT and AI in Agriculture

Tofael Ahamed
Editor

IoT and AI in Agriculture

Self- sufficiency in Food Production to Achieve Society 5.0 and SDG's Globally

②_ Springer

Editor
Tofael Ahamed
Faculty of Life & Environmental Sciences
University of Tsukuba
Tsukuba, Ibaraki, Japan

Foreword

It is my pleasure to have been invited to write this foreword for our TGSW- and Tsukuba Conference (TC)-related book from Springer Nature with regard to the session **Agriculture × IoT × AI: Self-Sufficiency in Food Production to Achieve Society 5.0 and SDG's Globally**. I have been participating as President of the University of Tsukuba in this special session series for the last 4 years starting from 2018. Each year, I had the opportunity to join this session and seen the networking of this session from all over world. The invited guests were from the USA, Europe, Australia, Japan, Malaysia, Indonesia, Bangladesh, Thailand, Vietnam, China, Taiwan, Sri Lanka, the Philippines, and continuously growing from other countries. I am very happy that our TGSW/TC conferences and networking grow over time and have contributed to a book in one frame with our friends and distinguished guest speakers of this session. As the COVID-19 pandemic started in 2020, we have the difficult time to meet face to face. On the other hand, we have become more closer with IoT technologies that bring even closer. We have overcome the situation with the help of scientific innovation. Our TGSW theme also changed with the New Normal better and more inclusive to bring our stakeholders together.

This session highlights the new challenges in food production, food traceability, and food safety globally. These challenges were magnified by the pandemic and are expected to be solved by science and technology. This book focuses on digital science and technology including IoT/ICT and AI that contribute to all phases of the food production system from preharvest to postharvest levels for crops, fruits, and vegetables. The book contents are divided into several sections, each of which is supplemented with a strategic short note from our TGSW/TC invited speakers that provides insightful thoughts on AI/IoT-based applications covering indoor and outdoor field practices to increase crop production and to achieve food security. The challenges clarified that worldwide and integrated efforts are required for its control, because there are no borders of all current events such as pandemics, for example, between countries and areas, between organizations, and between societies.

The novel approaches through AI have the impact in the intervention of utilizing limited resources to predict climate changes. The large data analytical platform brings an immense potential to serve decision support analysis in agriculture for smart applications of agricultural inputs such as water, seeds, fertilizers, herbicides, and pesticides as minimum as possible for maximum productivity, which is known as precision agriculture technology. For example, machinery companies in Japan started to provide smart assist systems from land preparation to food production for ensuring the traceability of agricultural inputs. To achieve the expected feature, interdisciplinary collaborations not only among life science, environmental science, medical science, and engineering technologies, but also including social science and humanity are needed to implement scientific achievements into society. Digital science and cooperation among different societies will bring a new decision matrix for sustainable food production.

The sustainability of global food production faces challenges over the years due to climate change, which causes drought, flash floods, tidal surges, salinization, and an uneven increase in the number of populations. Distant differences and population growth rates are dispersed in developed and developing countries. The labor force shifting from agriculture to industry threatens agricultural productivity and the constant supply of food production. Innovation toward modern agriculture, that is, an AI-based system, creates new opportunities and solutions to predict climate change hazards and minimize labor requirements. In this regard, this book solely discusses AI and IoT systems and their potential outlined in the 23 chapters related to smart agriculture.

The various smart farm technologies through different agricultural operations, including agricultural input management, plant monitoring, and farm machinery system, are highlighted with the effective approach in terms of resource utilization. Water management within irrigation systems is one of keys for sustainable and smart agriculture. IoT-based precision soil and irrigation management have become substantial, considering population growth and rising water shortages. In this regard, we are giving our best efforts for increasing water use efficiency, maintaining soil fertility, and boosting agricultural crop production in both protected and open field cultivation systems.

I am also happy that this book addresses automation in orchards, which is very challenging due to the complex canopy structure and fruits production. The book also further discusses the shortcomings of the present orchard automation approaches in the context of the shortage of seasonal labor and rising labor cost. The innovations and implementation of advanced machinery can be stratified based on regional demand, population engagement in agriculture, and socioeconomic aspects. The transformation of the present state of machinery and its automation levels is also described elusively. Furthermore, the poultry and livestock also require significant attention to increase production and reduce postprocess losses. This book also attempts to bring some important application of AI technologies and its integration in the poultry industry.

To conclude, I felt the application of digital science in agriculture becomes the leading area integrated with IoT and AI. The advancement of science, technology,

and innovation contributes to achieving Society 5.0, and SDGs, and is one of the driving forces of sustainable development with our partners from all over the world in these transboundary trends. In this regard, the University of Tsukuba is celebrating its 150-year anniversary from its inception this year and its 50-year anniversary of relocating to Tsukuba Science City next year. We are committed to "Design the Future, Together" with partner universities, research institutes, and industries. Therefore, this book has outstanding contributions relating to our TGSW and Tsukuba Conference series, will create certainly more signature impacts in smart agricultural application to increase productivity, and strengthen our partnerships among the collaborators.

University of Tsukuba, Tsukuba, Japan   Kyosuke Nagata

Preface

The sustainability of global food production faces challenges over the years due to climate change, which causes drought, flash floods, tidal surges, salinization, and an uneven increase in the number of populations. Distant differences and population growth rates are dispersed in developed and developing countries. The labor force shifting from agriculture to industry threatens agricultural productivity and the constant supply of food production. To address these two main problems of climate change and labor forces in agriculture, innovation in agriculture and its application are highly needed. Innovation toward modern agriculture, such as artificial intelligence (AI)-based systems, creates new opportunities and solutions to predict climate change hazards and minimize labor requirements. The Internet of Things (IoT) opened a new window with low bandwidth information sharing. In this regard, this book solely discusses AI and IoT systems and their potential outlined in the 23 chapters related to smart agriculture.

Agricultural inputs are the main criteria used to increase production. In this regard, Chap. 1 discusses the introductory notes related to the various smart farm technologies through different agricultural operations, including input management, plant monitoring, and farm machinery, to highlight the effective approach in terms of resource utilization. In addition, how to minimize postharvest losses and prevent agricultural losses and farm automation in its potential to establish a digital and big data analytics platform for the innovation of agriculture and food production systems have also been discussed throughout this chapter.

Chapter 2 is a strategic short note that highlights the concept of circular bioeconomy as applied to food and agriculture. In-depth explanation regarding the new transformation processes for controlled environment plant production system (CEPPS) is also presented. Chapter 3 discusses artificial lighting systems for plants grown in indoor farming, which aims to enlighten readers regarding lighting requirements for plants generally with some examples. Three main principles of light, light quality, light quantity, and light duration, and their effects on plant growth and development are reviewed in this chapter. Furthermore, different types and some examples of artificial lights have also been reviewed in this chapter to

provide insights into the light sources that are commonly used to enhance the growth and yield of plants cultivated in indoor farming. Light intensity and its control through smart application and IoT-based application layers are discussed to utilize through the maximum utility of artificial lighting for indoor plant growth.

Water management within irrigation systems is the key to sustainable and smart agriculture. In this regard, Chap. 4 discusses IoT-based precision irrigation management that has become substantial, considering population growth and rising water shortages, in increasing water use efficiency and boosting agricultural crop production in both protected and open field cultivation systems. Initially, extensive reviews were performed to provide insights into recent developments in IoT-based precision irrigation management, including monitoring and controlling protocols, and finally referred to some points that must be addressed in future studies. Chapter 5 continues to describe as a short note to focus on IoT and AI in urban water management to achieve sustainable hybrid water systems for domestic and agricultural uses.

Chapter 6 discusses the possibility of using IoT for monitoring of water management systems, especially focusing on solar water distillation systems from urban to agriculture to meet the water demands. While reusing, water quality and other parameters must be monitored regularly, and available sensors have been discussed along with the possibility of using IoT technology. Further resources are discussed to improve solar water distillation, which can be compiled and combined with a few systems that have been experimentally investigated. The results showed the potential to increase the scale of water reuse systems and continuous monitoring systems.

Chapter 7 introduces the application of the long-range wide area network (LoRaWAN) communication protocol in managing soil EC and pH in oil palm nurseries. This protocol consisted of four layers, including a sensor node, gateway, network server, and application server. LoRaWAN technology is a robust tool for outlying regions without cellular network coverage or for establishing private networks covering long distances with minimum power consumption and maintenance.

Chapter 8 reviews shortly the future of smart machine vision technology in agriculture, forestry, fisheries and animal husbandry to develop an automatic solutions to cater the labor shortages and replace human power. Chapter 9 discusses artificial intelligence (AI) in a general overview to open the discussion regarding this trending topic that has been changing our daily lives and society as well. This chapter unleashed the meaning and basic applications of AI, machine learning, and deep learning step by step in a time series of events, from their origins until the present. Furthermore, some of the applications of these new trends are explored for agricultural production to achieve the goals of sustainability and Society 5.0.

Chapter 10 illustrates the potential of thermal images in navigation systems by introducing and comparing three deep learning methods. As RGB cameras are easily affected by light, this study used thermal images for tree trunk detection. By training Faster R-CNN, YOLO, and CenterNet with the same dataset, based on the mAP and comparison of the test images between the three methods, the most stable deep learning method was obtained. The potential of using thermal images for automatic navigation in low light conditions was also demonstrated in this chapter.

Chapter 11 provides a systematic and pragmatic methodology for choosing the most suitable model for a desired application in agricultural sciences. It further

discusses in detail a comprehensive study conducted aiming to produce a robust real-time pear fruit counter for mobile applications using only RGB data, the variants of the state-of-the-art object detection model YOLOv4, and the multiple object tracking algorithm called Deep SORT.

Chapter 12 compares the evaluation of the recognition quality of different visual recognition techniques with the same dataset. Mask R-CNN and Faster R-CNN, which are two-stage algorithms, were compared and observed. Both methods had similar accuracy in the case of detecting individual pears, but unlike Faster R-CNN, Mask-RCNN generated masks made it more effective in the case of aggregated pears. Furthermore, the performance of the one-stage algorithm (YOLACT) and the two-stage algorithm (Mask R-CNN) were also compared. The results showed that YOLACT with 35 fps+ as the one-stage algorithm had lower accuracy than Mask R-CNN in generating the bounding box and mask. However, it was significantly better in real-time processing compared to Mask R-CNN with 5 fps+.

In addition to crops, livestock also require significant attention to increase production and reduce postprocess losses. Specifically, the poultry production sector is one of the most challenging sectors of the agriculture chain, as early embryo detection and sexual segregation at early stages represent bottlenecks to reach sustainability in this sector. In this regard, Chapter 13 presents a novel application of thermal micro cameras associated with deep learning algorithms to identify quail embryos at early stages during the incubation period, addressing high-throughput application at automatic and efficient incubator machines. The novel practice proposed in this chapter may overcome these issues, especially AI-based systems and sensor technologies. The application of AI-based system is further discusses briefly in Chap. 14 to explore its prospects for orchard picking harvester robot specifically for kiwifruit.

Chapter 15 discusses the application levels of mechanization to support labor shortages and increase productivity in developed and developing countries. The innovations and implementation of advanced machinery can be stratified based on regional demand, population engagement in agriculture, and socioeconomic aspects. This chapter solely describes the current trend in agricultural machinery adoption that can be used to recommend the levels of mechanization for appropriateness globally. The transformation of the present state of machinery to advanced levels is classified from Level 0 to Level 5. Each level is discussed in detail along with the transformation process from Level 0 to the automation level.

In continuation of automation and advanced development, Chap. 16 designed a human-controlled navigation system consisted of a leader vehicle and a follower vehicle, in which the follower vehicle follows the leader vehicle automatically. The system consisted of a monocular vision sensing system and rectangular markers, used least squares to reduce noise in data acquisition, and introduced a PID controller to maintain the tracking performance of the two-vehicle system. Higher accuracies were obtained for the straight line, turn, Z paths, and parallel trajectory tracking. The system can be used for grain harvesting and other agricultural developments, and the driver can perform the task effectively in agricultural operations.

Furthermore, automation in orchards, which is very challenging due to the complex canopy structure, must be addressed. In this regard, Chap. 17 provides an

account of opportunities and challenges in orchard automation with special reference to pre- and postharvest management practices. It further discusses the shortcomings of the present orchard automation approaches in the context of the shortage of seasonal labor and rising labor cost. Finally, both technically and commercially viable solutions are included in this chapter to overcome such drawbacks that hinder the development of the orchard sector.

Chapter 18 outlines the novel approach on detecting soil moisture for oil palm cultivation based on PALSAR-2 sensor data which is benefited to prepare for climate-related droughts. Chapter 19 on the other hand reports a system for the precision application of pesticides using unmanned aerial vehicles (UAVs) based on machine learning. The system used the mutual subspace method to classify the cropland and orchard images into spray and nonspray areas. In the field test, the classifier accuracies in offline cropland and orchard were observed with good accuracy. Similarly, in the online recognition, the average cropland accuracy was satisfactory. The average computational time of the recognition system indicated the possibility of implementing this method in real time. This recognition system is proven to be fast and accurate for the real-time recognition of spray and nonspray areas.

Chapter 20 discusses the feasibility of using thermal imaging techniques to distinguish healthy trees and basal stem rot (BSR)-infected trees. In particular, WEKA standard machine learning algorithms (ML) including naïve Bayes (NB), multilayer perceptron (MLP), and random forest (RF) were used to classify the trees into two categories: healthy and infected trees, with emphasis on imbalanced or nonuniform data approaches such as random undersampling (RUS), random oversampling (ROS), and synthetic minority oversampling (SMOTE).

Chapter 21 provides an in-depth explanation of the applicability of hyperspectral data and machine learning in the early detection of plant diseases, especially basal stem rot (BSR), which is one of the most destructive diseases caused by fungi in oil palm plantations. The results indicated that hyperspectral data and machine learning techniques have great potential with very high accuracy to classify and detect disease in the early stage.

Chapter 22 highlights the effort in development of an automated speed sprayer for precise application in orchard apple in Japan by utilizing GNSS, optical markers, image processing and LIDAR. The last chapter, Chap. 23, discusses the conclusion of this book referring to automation toward Society 5.0. A discussion is presented based on the supervised and remote autonomy of AI application in agriculture and machinery automation. Some of the basic standards, transformation from Level 0 to Level 5, are focused on how to adopt supervised and remote autonomy to achieve further AI-based application of machinery.

The book contents are divided into in depths chapters and supplemented with strategic short notes that provide insightful thoughts on AI-IoT-based applications covering indoor and outdoor field practices to increase crop production to achieve global food security.

Tsukuba, Ibaraki, Japan   Tofael Ahamed

Acknowledgments

It is my great privilege to have the Tsukuba Global Science Week (TGSW) and Tsukuba Conference (TC) organized yearly at the University of Tsukuba. I was very inspired to organize the session "Agriculture × AI × IoT for Global Food Production" from TGSW and TC over the past 5 years. During these years, we had opportunities to invite many experts from different universities globally to share their experiences and expertise, particularly in integrating agriculture with transboundary approaches along with AI and IoT-based big data schemes for food security. The knowledge exchange through all these conference series has inspired me to extend and share the outputs by curating this book.

I would like to express my great appreciation to the TGSW/TC Organizing Committee from the University of Tsukuba for contributing the foreword to this book. It has been a great honor for me to receive such invaluable and scholarly comments about the book despite their busy schedule enduring university affairs. I would also like to thank Professor Kyosuke Nagata, President of the University of Tsukuba, for creating time to meet and greet us during our session in TGSW/TC.

Furthermore, I extend my sincere gratitude to Professor K.C. Ting, University of Illinois at Urbana-Champaign, United States, Dr Martin Anda, Murdoch University, Australia, Professor Yan-Fu Kuo, National Taiwan University, Taiwan, Professor Ryozo Noguchi, Kyoto University, Japan, Professor Longsheng Fu, Northwest A&F University, China and Professor Abdul Rashid Mohamed Shariff, Universiti Putra Malaysia, Malaysia, for their contribution of strategic short notes to introduce the concept of each section of the book and interdisciplinary research ideas to readers. They have supported me each year in the TGSW/TC meetings as our distinguished speakers, and having them as part of this book is truly a great compliment.

Along with our lab contribution from the University of Tsukuba and our collaborators' contribution globally, the book is outlined in 23 chapters. It was such a tremendous effort to bring all the concepts into one frame with all the support from them. In this regard, I would like to extend my thanks and appreciation to the contributors from Universiti Putra Malaysia, Dr Siti Khairunniza Bejo, Dr Samsuzana Abd Aziz, and Professor Abdul Rashid Mohamed Shariff for their time spent preparing and reviewing their chapter at the different stages.

Contents

Chapter 1
IoT × AI: Introducing Agricultural Innovation for Global Food Production

Munirah Hayati Hamidon, Mohammad Hussain Seyar, P. D. Kahandage, Victor Massaki Nakaguchi, Arkar Minn, Ailian Jiang, R. M. Rasika D. Abeyrathna, and Tofael Ahamed

Abstract In many regions of the world, key inputs for food production, such as land, healthy soils, and water are becoming increasingly scarce, making it crucial to utilize and manage them sustainably. The advancement of various technologies in the agricultural sector allows stable and sustainable agricultural and food production. Innovation of artificial lighting and environmental control for indoor farming uses far less land but allows the intensification of food production. The Internet of Things (IoTs) technology serves as an alternative to better manage the main agricultural resources and water through smart irrigation. Additionally, the use of artificial intelligence (AI) and machine learning (ML) provides a new opportunity to improve the overall agriculture operations and production by involving real-time analysis and

M. H. Hamidon · V. M. Nakaguchi · A. Jiang
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

M. H. Seyar
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Soil Science and Irrigation, Kabul University, Kabul, Afghanistan

P. D. Kahandage
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering and Soil Science, Faculty of Agriculture, Rajarata University of Sri Lanka, Mihintale, Sri Lanka

A. Minn
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering, Yezin Agricultural University, Nay Pyi Taw, Myanmar

R. M. R. D. Abeyrathna
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering, Faculty of Agriculture, University of Peradeniya, Peradeniya, Sri Lanka

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

machinery automation. Therefore, the main purpose of this chapter is to highlight the future trend of agricultural innovation for realizing sustainable global food production.

Keywords Sustainable agriculture · Indoor farming · Artificial lighting · Internet of things (IoTs) · Smart irrigation · Artificial intelligence (AI) · Machine learning (ML)

1.1 Introduction

Over time, technology has evolved farming systems and has had a wide range of effects on the agriculture industry. In many countries around the world, agriculture is the main source of income. According to the United Nations projections, the rising population from 7 to 9 billion in 2050 will put more pressure on resources such as land and water to produce more food to feed the increasing population. Moreover, shortages in farm labor continue to be an issue due to the relatively low income in the agricultural sector and the fact that the majority of people have relocated to urban areas for better job expectations and more global competition.

Hence, in 2015, the Sustainable development goals (SDGs) were structured by the United Nations General Assembly (UN-GA) as a "blueprint to achieve a better and more sustainable future for all" and are intended to be accomplished by 2030. Among the 17 SDGs, SDG2 states that "end hunger, achieve food security and improve nutrition and promote sustainable agriculture" is a goal that needs to be achieved by agriculture and food systems, including both production and consumption, to eradicate hunger within the next generation (United Nations, 2017).

However, reliance on the conventional agriculture system will not be able to address these challenges. Urgent action is needed to revolve agriculture and food production systems in a smarter way to become more productive and less wasteful. In recent years, emerging cutting-edge technologies such as controlled environment agriculture (CEA), the Internet of Things (IoT), machine learning (ML), artificial intelligence (AI), deep learning (DL), unmanned aerial vehicles (UAV), and global positioning systems (GPS) have attracted much interest from both farmers and researchers to fulfill the rising demand for agricultural products and food. The adoption of these advanced technologies for remote and unmanned monitoring in agriculture fields, also by implementing solutions to create the most conducive environment for crop growth, has been proven to improve input management, reduce yield losses, and support growers and interveners in decision-making.

The combination of hardware and software technologies has optimized agricultural operations to improve production. Currently, there are many portable, low-cost, and power-efficient hardware and sensors with wireless connections that are widely implemented across both indoor and outdoor agriculture. The utilization of hardware and sensor networks to continuously monitor agricultural growth parameters such as temperature, relative humidity, and soil moisture provides farmers with essential information to allow better input management and plant monitoring and enhance quality and crop yield. Additionally, sophisticated hardware such as graphical

processing units (GPUs) can process an enormous volume of data collected by these modules, as prompted by AI framework-based software.

The current rise in AI and big data technologies support software systems, allowing the management of massive quantities of data collection by hardware modules. The use of these data as input to cutting-edge, AI-based analyzers may provide better-informed judgments and insights to farmers or decision-makers. Such automation and intelligent control of agricultural operations offer farmers the opportunity to efficiently manage their resources, such as irrigation water, fertilizer, pesticides, and herbicides, in the right place, time, and amount. Furthermore, advancements in AI and machine vision techniques to detect and classify objects have sparked interest in deployment in agriculture fields. Using this AI-machine vision approach, plant disease can be detected early and accurately replace manual assessment, which is time- and labor-consuming, especially for larger farms. AI technology, including deep learning, can also be utilized for the automation of farm machinery and orchard operations, such as navigation systems, transplanting, spraying, and harvesting, to combat agricultural labor shortages.

In this context, this chapter discusses the various smart farm technologies through different agriculture operations, including input management, plant monitoring, and farm machinery to highlight the effective approach in terms of resource usage, preventing agricultural losses and farm automation in its potential to establish a platform of big data analytics for agricultural innovation and food production systems (Fig. 1.1).

1.2 Key Factors for Plant Growth and Agricultural Production

Plant growth is greatly affected by the growing environment. The primary factors driving plant growth include light, water, nutrients, humidity, and temperature. These factors determine plant development and their growth hormones, which indicate that they grow either slowly or rapidly. Providing an ideal growing environment by properly controlling all these elements will make the plants thrive. Environmental stresses, in contrast, will affect normal plant development, resulting in stunted or slow growth. Plants may potentially die under excessive stress circumstances.

Plants need energy from light to complete the process of photosynthesis, which allows them to manufacture their food by converting water and carbon dioxide into carbohydrates and oxygen. The pace of this process is strongly reliant on the amount of light available. As our population grows and puts pressure on agricultural production, significant reforms in sustainable agriculture are needed. Sustainable agriculture can ensure a sufficient supply of safe and affordable food. Utilizing artificial lights in indoor farms with less use of water and land than traditional
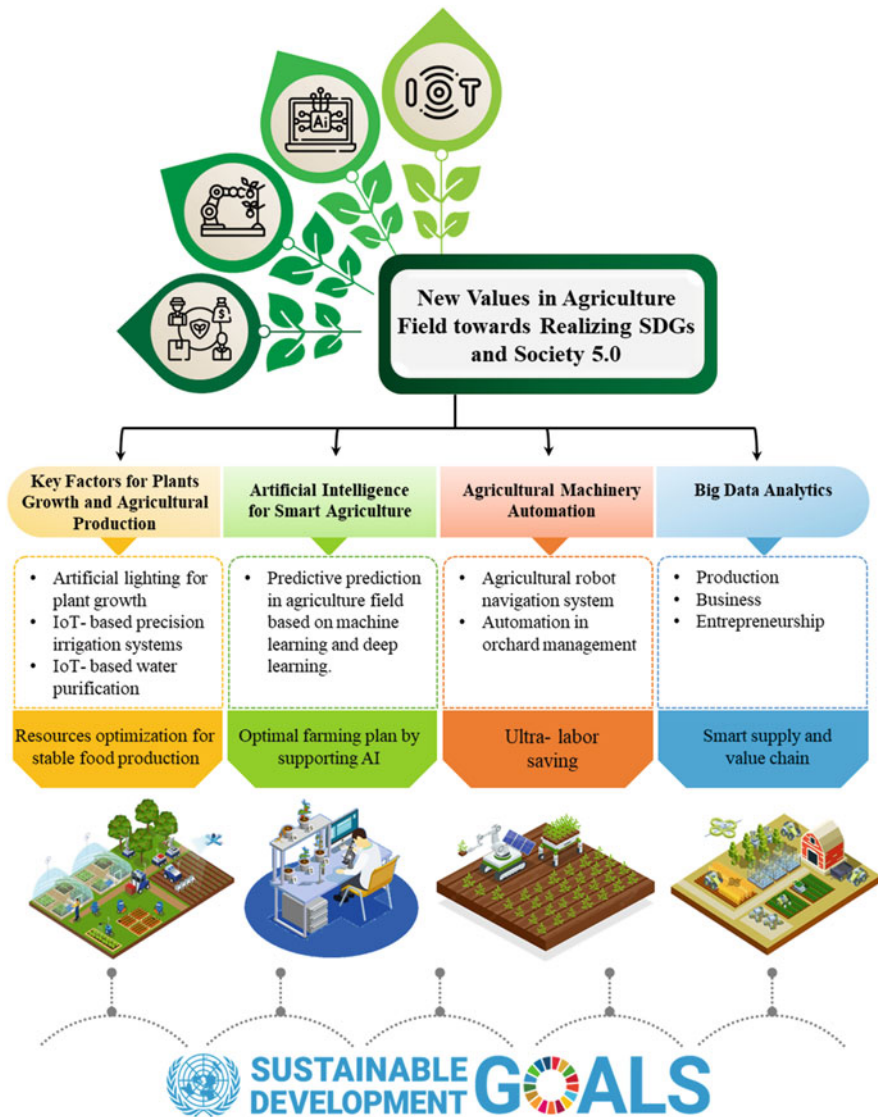
Fig. 1.1 Advanced agricultural innovation for global food production

agriculture practices may play an important role as a sustainable alternative to our future food system.

On the other hand, clean water is also a crucial resource for agriculture and is an asset in producing food. Land degradation and significant population growth have had a tremendous impact on agriculture's water resources. The implementation of intelligent irrigation techniques employing automated and Internet of Things (IoT)

technology can be one of the alternatives to overcome water issues in agricultural fields.

1.2.1 Source of Light for Indoor Farming Systems

The potential of indoor farming as one of the sustainable food production systems is promising when considering more needs to produce to meet the food demand of the growing population. Crops cultivated in controlled environments are anticipated to become much trendier, especially in urban areas, concurrent with the emergence of agriculture technologies to cater the issues related to plant-based diets, food safety, and labor shortages. Climate changes and frequent weather extremes, such as droughts, floods, storms, and severe freezes, have significantly impacted the conventional farms that produce fresh fruits and vegetables. Hence, indoor farming's controlled environment approach permits year-round food production in any climate, allowing farmers to prioritize food quality and safety.

Sustainability is a major reason for the effort to produce more food in indoor farming operations. Practicing indoor farming under proper conditions can reduce the environmental impact of growing more food with less land resource degradation. Additionally, consumers currently prefer to consume food that is produced more sustainably. Consumers are increasingly appreciating the benefits of receiving food as soon as it is harvested. Moreover, they also want pesticide-free products, as pesticides can be harmful to long-term human health. Thus, the trend in consumers' preferences for consuming more fresh plant-based foods and foods prepared in a more environmentally friendly manner is growing.

Labor shortages have always been one of the greatest challenges in the agriculture sector, with problems of an aging agricultural labor force and a lack of a new young workforce. However, indoor farms have various automated controls and duties and are less impacted by labor market volatility and stressors. In addition, the global COVID-19 pandemic has also accelerated the growth of indoor farming. When some merchants faced supply chain challenges and could not obtain fresh items, they resorted to new partners and vendors who operated indoor farms with more controlled conditions and inventory. Indoor farms are typically faster to construct and have more adaptable operations that allow for transitions in the case of unforeseen circumstances.

Several key factors, such as obtaining the right climate, lighting, and spacing, need to be considered when starting an indoor farm. Among those factors, lighting is one of the crucial factors that can determine the success or failure of indoor farming. After all, the reliance on artificial lighting distinguishes an indoor farm from other plant growing methods. The most important light principles influencing plant growth are light intensity, photoperiod, spectrum, and uniformity. Each plant needs specific light requirements based on its principles, as the yield for indoor farming is highly dependent on the light. Light quantity refers to the concentration or intensity of the light source. Light quality describes the color or wavelength that reaches the plants.

Fig. 1.2 Plants grow under different spectra of light

Table 1.1 List of important parameters to understand artificial light use in indoor farms

Parameters	Definition
Photosynthetically Active Radiation (PAR)	Wavelengths of light ranging from 400 to 700 nm. This is the light spectrum portion used by the plants for photosynthesis.
Photosynthetic Photo Flux (PPF)	The number of photons emitted by a light source.
Photosynthetic Photo Flux Density (PPFD)	The quantity of PAR delivered to the plant (measured as PPF).
Daily Light Integral (DLI)	Number of photosynthetically active photons that accumulate on a surface over a 24-h period.

The length of time a plant must be exposed to a light source is referred to as the light duration or photoperiod. Choosing an appropriate light spectrum is difficult in regard to lighting systems for indoor farming, as under the incorrect spectrum, plants will not thrive. Additionally, this spectrum and wavelength vary depending on the plant development stage. For instance, a blue spectrum is ideal during the vegetative stage, while during the flowering stage, adding a red spectrum enhances the blooming process (Fig. 1.2).

In addition to understanding light principles, it is also important to understand light parameters for determining the most suitable light for plant growth and development. The parameters provided in Table 1.1 were created to better regulate and measure light exposure and perceived plant growth in a controlled indoor farming environment. The parameters were developed from the perspective of how light is absorbed and utilized by plants. Hence, combining both light principles and parameters helps to qualify the proper artificial lighting system for plant growth in indoor farming (Fig. 1.3).
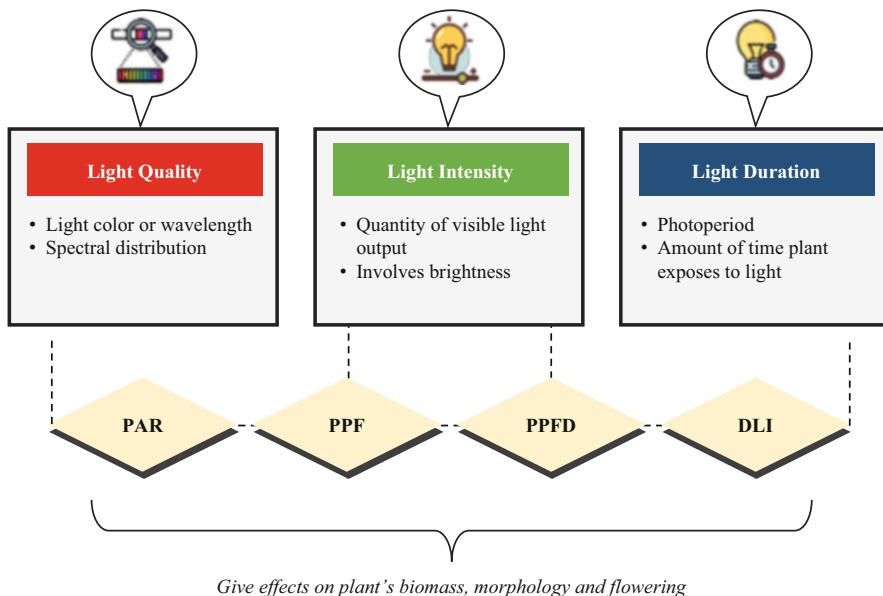
Fig. 1.3 Important parameters related to artificial lighting for indoor farming

1.2.2 IoT-Based Precision Irrigation Systems

Agricultural production is facing numerous challenges, including staggering population growth, decreasing land due to rural buildings and growing water scarcity exacerbated by practicing traditional irrigation systems, which are usually associated with the waste of water and resources. With all the aforementioned global problems, water shortages are more critical, especially in arid and semiarid regions. Thus, seeking approaches to boost water use efficiency within agricultural systems is the key to saving water and feeding the global population. In other words, agriculture in these areas is not possible without irrigation because of limited and erratic rainfall that makes them unavailable for plant consumption. Moreover, agriculture alone constitutes over 70% of global freshwater, which makes it the largest water consumer. Therefore, employing methods to help use limited water efficiently while maintaining a sustainable production level is the key to feeding the global population over the coming years.

In recent decades, various studies have been carried out to find approaches that can contribute to water use efficiency in agriculture or farming systems. There have been a couple of methods suggested by researchers worldwide to deal with water scarcity while increasing production levels. Employing deficit irrigation, which refers to applying water lower than the plant water requirement is an effective strategy to increase water use efficiency (Ali et al., 2007; Bekele & Tilahun, 2007; Chai et al., 2016; Comas et al., 2019). However, this approach can be further
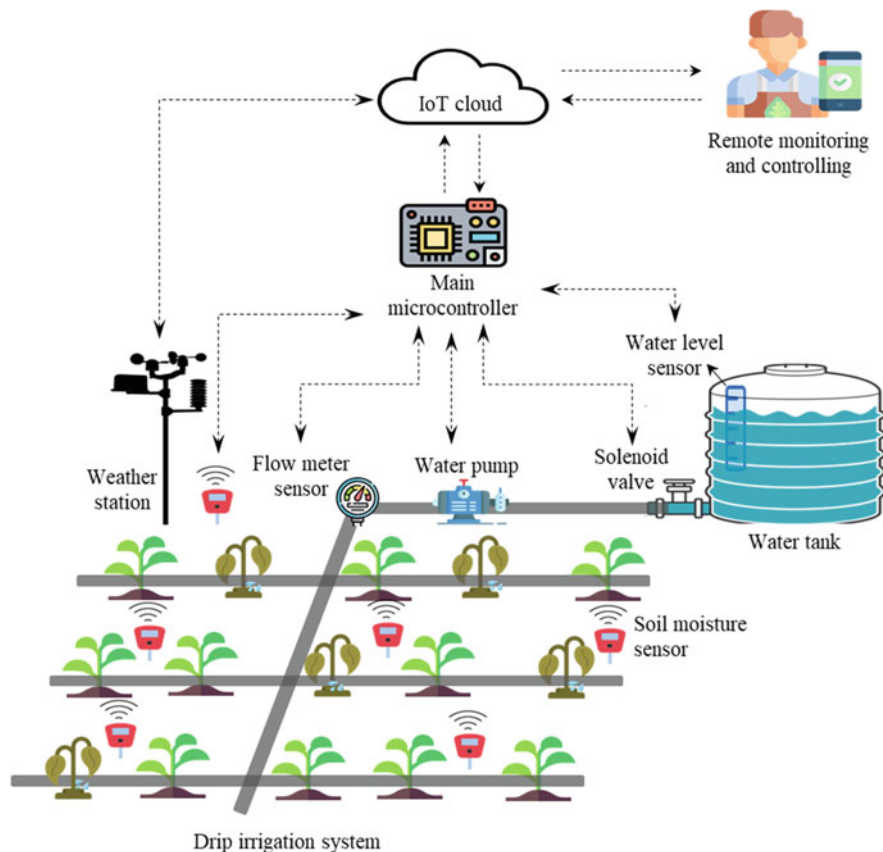
Fig. 1.4 The concept of precision agriculture based on IoT

improved through IoT-based technology applications for monitoring and controlling watering or irrigation (Fig. 1.4).

Practicing precision irrigation systems is another effective approach to water saving and enhancing production, but if these systems are integrated with IoT devices, including sensors, better results can be achieved since farmers can have real-time feedback on their fields and provide the plants with an optimum amount and on time water application. If the farmer is not in the field, he or she can still monitor and perform irrigation, which makes these systems robust and helpful. Several IoT technologies are employed in agriculture, and the major and widespread types are the long-range wide area networks (LoRaWANs), narrowband Internet of Things (NB-IoT), and Wi-Fi networks, where any of them can be employed under specific conditions. For example, for short distances, using Wi-Fi is recommended, while for long distances, using NB-IoT and LoRaWAN technologies are preferred. These systems usually run based on sensor data that are deployed in greenhouses or open fields. Data are retrieved by the sensors and uploaded to the cloud server, where

the users can access the data remotely and control the irrigation or watering operations over a long-distance range.

1.2.3 IoT-Based Water Purification

The Internet of Things (IoT) has become a very beneficial application in a vast range of areas, including home management, traffic management, autonomous vehicles, environmental monitoring, industrial, agricultural, and commercial administration. The application of the IoT in water management is also admirable and brings many rewards by increasing operational efficiency. Potable water is one of the basic needs of human beings for their survival and good health. Clean water is one of the major inputs for agriculture as well. Guaranteeing the availability and sustainable management of water is identified as one of the Sustainable development goals (SDGs) by the United Nations (United Nations, 2020). Even though 1.4 billion km^3 of water is available on Earth, only 2.5% of it is considered fresh, while approximately the remaining 97.5% is sea water (Modi & Modi, 2019). As many of the freshwater sources on Earth are inaccessible to humans or are polluted, only 0.4% of freshwater is usable and drinkable for 7.8 billion people worldwide, so most developing countries do not have enough resources to provide safe potable water to people (Misachi, 2018). Agriculture also acts as a source of water pollution, as heavy metal ions, including As, Cd, Cu, and Zn, from chemical fertilizers and other agrochemicals accumulate in surface water bodies and groundwater. Chronic kidney disease (CKD) is a common severe health issue in the North Central Province of Sri Lanka, and high levels of As and Cd in the water have been suspected as root causes of CKD (Jayasekara, 2017).

Therefore, water purification and management are crucial for achieving a sustainable environment for all living beings. Although the large-scale municipal water purification process consists of several steps, there are some simple purification methods possible at the domestic level. Boiling, chlorination, filtration, coagulation and sedimentation, ion exchange, solar treatments, distillation, and combined treatment systems are some of the small-scale purification methods in use at the domestic level (Agrawal & Bhalwar, 2009). The most important thing expected from a water purification system is maintaining water quality within the purest range. Therefore, continuous water quality monitoring is essential for confirming that the system is working properly. IoT technology facilitates this requirement with different water quality sensors, such as pH sensors, temperature sensors, turbidity sensors, dissolved oxygen sensors, conductivity sensors, salinity sensors, and total dissolved solids (TDS) sensors (Singh & Ahmed, 2020). In addition to determining water quality, determining water level and water flow can be achieved with ultrasonic sensors and flow sensors. The Arduino microcontroller board or Raspberry Pi minicomputer boards can be selected based on the program requirements, and the method of communication can be selected based on the required operational distance. The system can be controlled using a smartphone application.
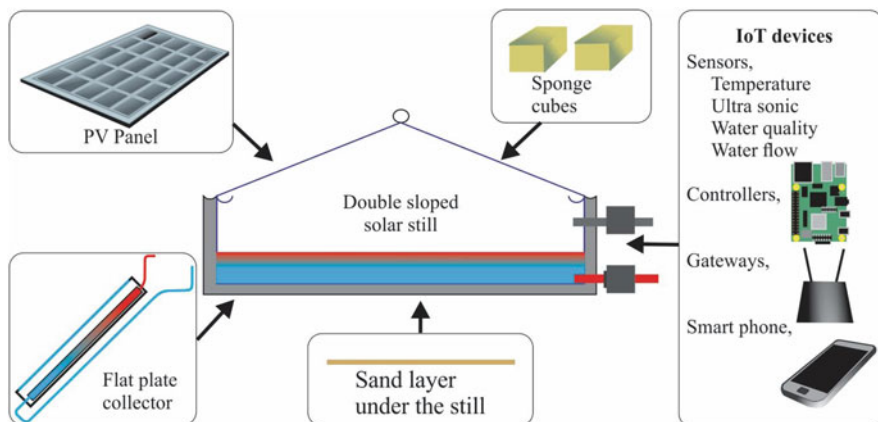
Fig. 1.5 Combination of strategies to improve solar still functionality

Solar water distillation is an economical and feasible method of purifying polluted water in many developing countries with sufficient solar power throughout the year. The flat plate collector of an active type solar water distillation system is a major component of a solar heating system as well. Therefore, switching the operation between water distillation and water heating can be achieved with some structural changes to the system. Therefore, the concept of designing and fabricating a solar system that can be used for two purposes, water distillation and hot water, and controlled by IoT technology with real-time data monitoring was introduced. The solar still of the system also acts as a storage tank by remotely insulating the glass top using IoT when the system must switch to a hot water system. Sensors are used to determine the water and air temperatures of the solar still, pH, and TDS of water and distilled output. Several innovative strategies, such as connecting a flat plate collector, placing sponge cubes in the still and placing a sand layer under the still, were selected to improve the evaporation capacity. Figure 1.5 shows the strategies used to improve the solar still.

The flat plate solar collector absorbs solar heat and transfers it to the water circulating through it by increasing the water temperature inside the solar still. Sponge cubes are placed in the solar still such that half of a cube is submerged, and the other half is above the water level. Then, sponge cubes absorb water and improve evaporation by reducing the water surface tension. The sand layer below the solar still can store sensible heat during the daytime and release nocturnal time. As solar panels provide photovoltaic power to operate the components that utilize electricity, the whole system is driven by renewable power. Figure 1.6 shows the constructed solar still with a flat plate collector.

Fig. 1.6 Constructed solar still

1.3   Artificial Intelligence for Smart Agriculture

Agriculture in the twenty-first century is no longer a human activity that everyone can perform with low or almost no skills as it used to be. Every day, more automated processes involving complex machinery and powerful computers are necessary inside all sectors of the productive chain. Innovation, which has always been present throughout agricultural history, despite being associated with technicians and mechanics, has shown enhancements along with the prefarming, infarming, and postfarming sectors. New advancements have been proposed for chemometrics, materials engineering, nanotechnology, low-cost microprocessors, bioinformatics, software, and machinery.

During the industrial revolution, agriculture was issued to supply the demand for raw materials and energy, thus working on the backstage of technological progress. Industry 4.0, however, has placed agriculture into the digital era, where IoT technology, big data, fast-network connections, and business intelligence are responsible for controlling and performing improvements in production, logistics, and markets across the globe. Mechanization and automation, especially robotics, can deliver
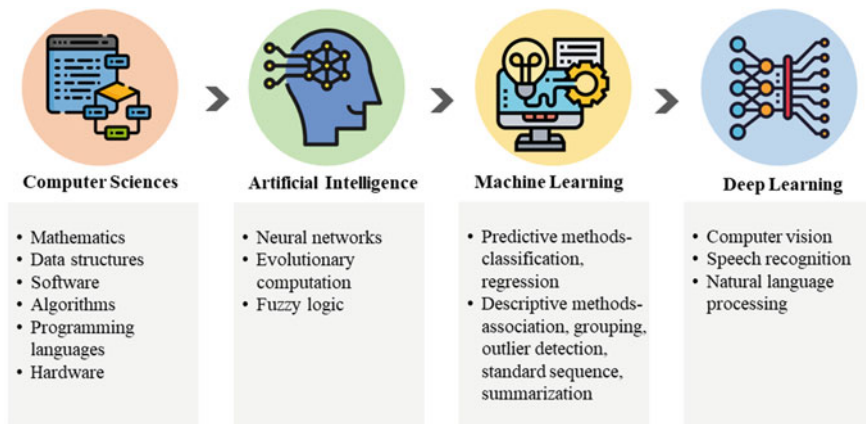
Fig. 1.7 Terminology chart to conceptualize the main terms regarding computer sciences, AI, machine learning, and deep learning

solutions to optimize the workforce in addition to supplying labor demand of operations/quality at the field level.

Global issues related to overpopulation, global warming, environmental pollution, water contamination, and desertification are the main pressure forces that have been pushing farmers, companies, and all industries to sustainable systems in agrifood industries. In this sense, artificial intelligence (AI) was set as one of the main technologies able to support decision-making toward big data and multicriteria analysis. Thus, it has been elected the fundamental stone to settle down Society 5.0 (CSTI, 2016).

In the field of computer sciences, AI is a hot topic that has recently attracted attention worldwide. Although it was introduced many decades ago, its enormous potential has recently been identified, associated with big data, efficient algorithms, and parallel processing using graphic processing units (GPUs). Machine learning (ML), which uses mathematical and statistical approaches to solve problems involving big data analysis, is the main tool in AI. ML uses predictive methods based on classification and regression techniques, and another technique is descriptive methods using association, grouping, outlier detection, standard sequences, and summarization methods. Deep learning (DL) can be classified under ML; this subject focuses on neural networks (NNs) to solve complex problems involving nonlinear and multidimensional data (Fig. 1.7), but it also uses ML algorithms embedded in some processes. Currently, DL is considered the state of the art of AI, where most of the applications have been developed thus far.

DL-based algorithms are providing new solutions for computer vision (CV) systems and are thus responsible for breakthrough solutions regarding complex problems involving real-time analysis and automation of machinery. Supported by heuristic algorithms for image recognition and classification, these systems are directly involved with recent approaches for phenomics and self-decision-making

Fig. 1.8 Computer vision system based on deep learning object detection algorithm

systems. Advancements in NN algorithms can deliver faster and more accurate models able to mimic humans' capacity to solve problems faster and in real time. Moreover, when associated with sensors such as optical and radiometric cameras (multispectral sensors), CV systems represent the state of the art for navigation of autonomous driving vehicles and robotic arms.

NNs are the workhorse for DL models. Generally, we divide NNs into three groups: recurrent neural networks (RNNs), convolutional neural networks (CNNs), and generative adversarial networks (GANs). Among these groups, countless types of architectures can deliver models that vary in terms of accuracy, speed, and portability. However, most efforts have been applied to CNNs, which have proven to be the best technique for image processing tasks. CNNs are NNs that can extract features from images and generalize classes of images by using a nonlinearity approach and extensive training datasets. By using CNNs, robust algorithms are frequently released, improving the speed of training, deployment and accuracy of classification and detections. Figure 1.8 shows an example of a DL model that uses a CNN on the backbone of the object detector algorithm.

For agrifood industries, image recognition and classification can support the automation of thousands of operations, especially those regarding fast sorting, tracking, and grading of products; moreover, it can help in the identification of disease symptoms in early stages and improve the quality and safety of operations. It can also improve the welfare conditions of livestock and supply labor shortages in the fields with onboard machinery and management systems.

Artificial intelligence in agriculture is a recent topic that has already extended content associated with IoT technologies and big data, and a global commitment to achieve Society 5.0 is becoming reality. For the next few years, we will see fantastic

applications for the automation of farms by incorporating AI-based management systems and they will be responsible for ensuring the sustainability of all operations.

1.4 Agricultural Machinery Automation

Currently, automated agricultural machinery is being developed because of climate change, labor shortages, increased food demands, etc. Each country changes its farm machinery operations with the aid of automation according to its available resources and technologies. The operations of agricultural machinery can be distinguished according to their operations. They are land preparation operations, planting, sowing and fertilizer application operations, weeding and intercultural operations, crop protection operations, and harvesting and threshing operations.

1.4.1 Farm Automation Technology

The main purpose of farm automation technology is to cover ordinary tasks more easily. Currently, the agriculture sector widely uses sensors. Sensors come with less weight and size, low cost, and high performance. Moreover, the combination technology of sensor data fusion was significantly developed so that farm automation is more precise and easier to perform in farming operations (Fig. 1.9).

The following technologies are commonly utilized in farm machinery operations.

An autosteering guidance system transforms an ordinary tractor's steering into semi or partially automated steering. Autosteering devices are available in the markets so that farmers can choose suitable autosteering systems.
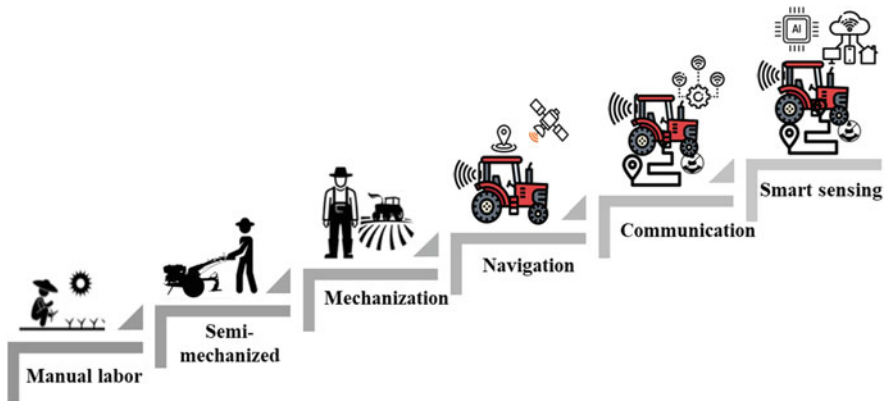
Fig. 1.9 Flowchart of farm automation

The navigation system achieves not only vehicle location but also path planning. A real-time tracking system has been developed in agriculture by using RTK-GNSS with centimeter accuracy. Furthermore, a new hybrid navigation system was launched that combines an inertial measurement unit (IMU) and GNSS so that vehicles can easily obtain precise locations and headings. Farmers can install navigation systems in high-precision and low-cost devices on their vehicles.

Sensing and recognition not only perform real-time sensing but also create driverless cab conditions. RGB cameras, 3D depth cameras, and LiDAR are widely used in this system. Variable-rate technology (VRT) is the most expected one to reduce agricultural inputs such as chemical fertilizers, pesticides, herbicides, irrigation water, and precision seed placement.

The implementation of a communication system enhances the combination of control on all attached implements. The operator can control all implements and equipment with a single control device. The ISOBUS system is widely used among the manufacturers of farm machinery.

The power transmission system provides easy access to power distribution on the whole vehicles and implements system. John Deere Co. established new technology in power transmission systems. This system changed conventional continuously variable transmissions (CVTs) to electromechanical power splits. An electric device is used as the main power transmission source.

Electrification can substitute for the prime power source: the engine. Electrification uses a battery as the power source. Batteries use electrical drives, such as electric motors, which have heavy torque at low speeds, are more powerful, more reliable, lightweight, compact, and have lower noise. Currently, electric drive vehicles are available in farm machinery markets.

Artificial intelligence (AI) is the technology that recognizes the difference between the desired work and environment so that individual tasks can be operated. Currently, this technology is used not only in farm machinery operations but also in postharvest and process engineering operations.

1.4.2 Agricultural Robot Navigation System

In agricultural applications, a large quantity of labors is required for planting, fertilizing, and spraying pesticides, as well as harvesting and transportation. With the aging of labor and labor migration, the labor shortage in agriculture is becoming increasingly serious, and the automation of agricultural production and management has become an inevitable trend for future development.

The ability of robots to move through the agricultural environment is a fundamental requirement to accomplish production tasks in all phases of production management. Navigation in agricultural environments remains difficult due to weather and light variations and differences in terrain and vegetation, which require the design of an efficient and stable sensing control system to solve. Stable operation in agricultural environments requires robots with environmental sensing, path
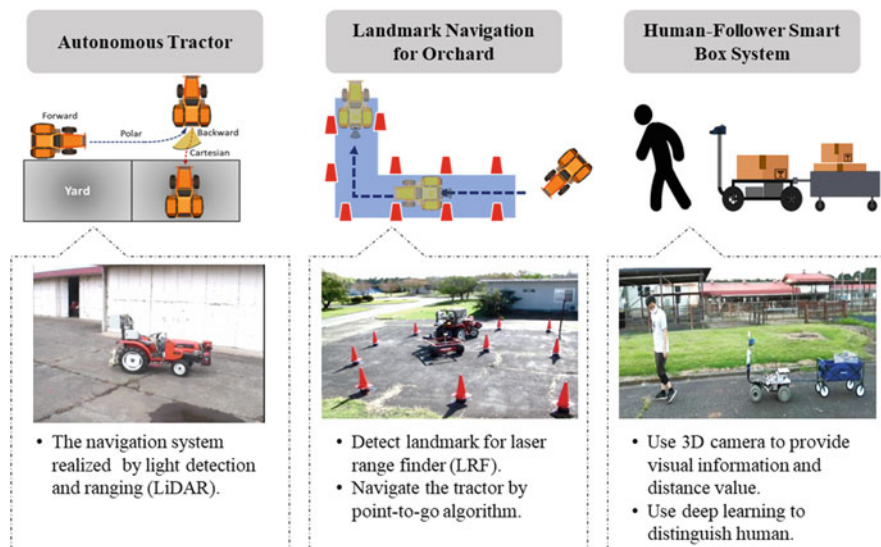
Fig. 1.10 Different agricultural navigation systems

planning, and obstacle avoidance capabilities and can operate on optimal routes in complex agricultural environments while ensuring that the robots do not crash with objects. They generally combine sensing, communication, detection, and control technologies, such as tractors and human-follower smart boxes (Fig. 1.10), which need to detect their surroundings while operating to plan their paths and perform specific functions. These robots can improve operational efficiency and reduce the use of manual labor. Before a robot can complete a task, it needs to select the appropriate sensors and control algorithms based on the robot's operating environment and task requirements. Figure 1.11 shows the sensors, algorithms, and control strategies commonly used in agricultural robot navigation (Shalal et al., 2013).

The agricultural environment is composed of various factors, including crops, trees, weeds, soil, and man-made landmarks. This complex environment makes it difficult for robots to operate, such as large land areas, which can lead to poor signal and interruption problems. Problems such as unevenness of the ground, rain, and sunlight can also affect the data obtained by the robot's sensors. The color and state of plants in different growth stages can also have an impact on the identification of target objects. However, there is also a certain simplicity in the agricultural environment. For example, crops of the same type are planted together, and for target detection, it is not necessary to prepare a large number of detection species. When planted, crops are generally arranged in straight lines at almost the same intervals, and with the development of automation, the crops tend to be planted in a way that makes the machinery easy to manage and crops easy to harvest. Man-made landmarks in the environment can also be used as a position reference when navigating and guiding the robot's operation.

Navigation Sensors

Primary sensors
- RTK-GNSS
- LiDAR/Laser
- Machine Vision
- Other sensors

Secondary sensors
- IMU
- Gyroscope

Computational Methods

- Kalman Flitter
- Hough Transform
- Image Segmentation

Control Strategies

- PID Controller
- Fuzzy Logic Controller
- Neural Network
- Genetic Algorithm

Autonomous Navigation

- Path Planning
- Path Following
- Obstacle Avoidance
- Mapping
- Localization

Fig. 1.11 Agricultural robot navigation system elements

Autonomous navigation is an important issue in robotics applications, where the purpose is to safely guide the robot in different environments. The navigation capability of the robot depends on the sensor system feedback and the control algorithm calculations. The robot first detects its surroundings, then analyzes the surrounding objects, and when it is moving, needs to consider following crops and landmarks and avoiding obstacles. In response, many researchers have proposed many different navigation methods, which can be divided into two groups. In one group, the robot runs according to a map or creates a map as it explores. If the environment map is perfect, the robot can easily determine its position and orientation at each time point. Localization and map learning are interdependent because accurate localization is necessary to build a good map, and having an accurate map is essential for good localization. Simultaneous localization and mapping (SLAM) technology was used to create a map in an unknown environment and use this map to locate the current position of the robot. Based on the current position and the map, the robot can find the optimal path from the current position to the target spot without hitting an object. The other group is path planning based on objects detected in real time. Since most of the crops are grown in straight lines, an algorithm can be developed to plan the detected crops as straight lines and obtain the planned path so that the robot runs accurately along the straight line. However, during the detection process, multiple uncertain factors need to be considered, such as the crop only on one side of the boundary, the possibility of missing crops during the operation, and the irregular shape and size of the crop. In addition, the weeds add noise to the path calculation, which can affect the planned path and the machine operation process.

A good autonomous navigation system requires the selection of appropriate sensors, algorithms, and control strategies depending on the environment and the task to make them work and interoperate with others to obtain the highest efficiency. As robotic automation is increasingly needed, more research needs to be realized to improve the technology and overcome the limitations of autonomous agricultural robots.

1.4.3   Automation in Orchard Management

The quality and quantity of high-value crops such as orchard fruits depend on the effective and efficient management of the orchard farm. Thus, conducting orchard operations in such a way that the resulting optimum outcome is of greater importance. However, the shortage of seasonal labor along with the rising labor cost has made achieving this task difficult. Since skilled labor is one of the key components of the production system, finding ways and means for overcoming this situation is at the forefront of the debate in the recent past. In this context, modern machinery has acquired a crucial role in orchard fruit production, and different orchard management robots have been introduced to simplify different tasks. Pruning, thinning, spraying chemicals and fertilizers, harvesting, and transportation are key areas in that automation takes place to address the labor shortage and aims for decent quality orchard outputs.

During the initial stages of orchard automation, robotic platforms were semiautomated structures that were not overly complex but could be adapted to conventional fruit orchards. The developments of high-tech sensors and actuators integrated with high-performing computers have paved the way for the development of complex robots that run based on machine learning and trained neural networks.

Therefore, identifying the potential of operating robotic arms as an attachment to existing tractors or simple robotic vehicles with the help of developed orchard architecture is important for the development of commercially and financially viable orchard automation options. As illustrated in Fig. 1.12, conventional orchard structure (Fig. 1.12a), the higher-degree robotic arm should avoid collisions and should penetrate inside the canopy to reach the target location, Fig. 1.12b shows easily harvesting an orange without collisions with the branches and other fruits. The challenge is maintaining a healthy canopy architecture, which can deal with various robotic platforms and machinery. As indicated in Fig. 1.13a, b, an autonomous tractor/autonomous crawler tractor can be used to attach robotic arms to take them near the orchard canopy. In regard to cost, maintenance, and safety, robots with lower degrees of freedom can be easily handled by farmers.
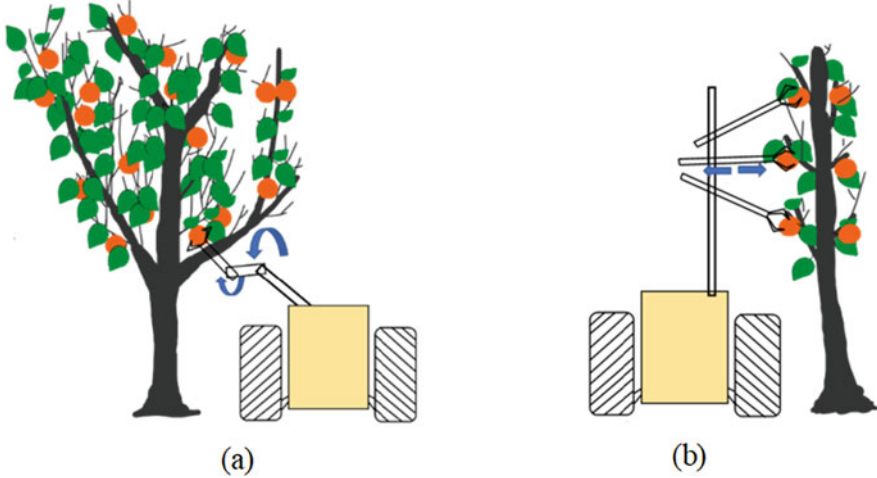
Fig. 1.12 (**a**) Orange harvesting under complex canopy structure (**b**) Under improved canopy structure

Fig. 1.13 (**a**) Autonomous 4-wheel tractor (**b**) Compact autonomous tractor

1.5   Conclusion

Agriculture worldwide is facing the same issues in terms of land degradation, climate change and a decrease in the number of skilled farm workers. It is very important to manage our agriculture resources wisely and intelligently to boost productivity and ensure safe and sustainable food production for present and future generations. As a method of addressing these issues, various advanced technologies, such as lighting systems for indoor farming, IoT, sensors, location systems, robots, and AI, have been developed to improve the agricultural system by establishing an integrated smart agricultural system that covers all aspects of farming. Smart farming based on the IoT increases production by making better use of natural resources and inputs, as well as better land and environmental management. Furthermore, artificial

intelligence enables the automation of traditional agriculture operations that must be done manually through the use of machine vision and deep learning. Image analysis and machine learning algorithms based on AI can be utilized to automatically detect and track fruits, pests and diseases, plant phenotyping and yield prediction. Adopting these cutting-edge technologies digitalizes and transforms the agricultural system to alleviate the pressure of producing more food with less for the growing world population.

References

Agrawal, V. K., & Bhalwar, R. (2009). Household water purification: Low-cost interventions. *Medical Journal, Armed Forces India, 65*(3), 260–263. https://doi.org/10.1016/S0377-1237 (09)80019-1

Ali, M. H., Hoque, M. R., Hassan, A. A., & Khair, A. (2007). Effects of deficit irrigation on yield, water productivity, and economic returns of wheat. *Agricultural Water Management, 92*(3), 151–161. https://doi.org/10.1016/j.agwat.2007.05.010

Bekele, S., & Tilahun, K. (2007). Regulated deficit irrigation scheduling of onion in a semiarid region of Ethiopia. *Agricultural Water Management, 89*(1–2), 148–152. https://doi.org/10. 1016/j.agwat.2007.01.002

Chai, Q., Gan, Y., Zhao, C., Xu, H. L., Waskom, R. M., Niu, Y., & Siddique, K. H. M. (2016). Regulated deficit irrigation for crop production under drought stress. A review. *Agronomy for Sustainable Development, 36*(1), 1–21. https://doi.org/10.1007/s13593-015-0338-6

Comas, L. H., Trout, T. J., DeJonge, K. C., Zhang, H., & Gleason, S. M. (2019). Water productivity under strategic growth stage-based deficit irrigation in maize. *Agricultural Water Management, 212*, 433–440. https://doi.org/10.1016/j.agwat.2018.07.015

Government of Japan. (2016). *Council of Science, Technology, and Innovation fifth science and technology basic plan*. Author.

Jayasekara, K. J. D. B. J. (2017). *Desalination needs and appropriate technology for Sri Lanka*. University of Ruhuna.

Misachi, J. (2018). What percentage of the earth's water is drinkable? *WorldAtlas*. Retrieved June 23, 2022, from https://www.worldatlas.com/articles/what-percentage-of-the-earth-s-water-is-drinkable.html

Modi, K. V., & Modi, J. G. (2019). Performance of single-slope double-basin solar stills with small pile of wick materials. *Applied Thermal Engineering, 149*, 723–730. https://doi.org/10.1016/j. applthermaleng.2018.12.071

Shalal, N., Low, T., Mccarthy, C., & Hancock, N. (2013). A review of autonomous navigation systems in agricultural environments. In *Innovative agricultural technologies for a sustainable future* (pp. 1–16). Society for Engineering in Agriculture (SEAg).

Singh, M., & Ahmed, S. (2020). IoT based smart water management systems: A systematic review. *Materials Today: Proceedings, 46*, 5211–5218. https://doi.org/10.1016/j.matpr.2020.08.588

United Nations. (2017). *Resolution adopted by the General Assembly on 6 July 2017, Work of the Statistical Commission pertaining to the Agenda 2030 for sustainable development*. Author.

United Nations. (2020). *The sustainable development goals report 2020*. United Nations Statics Division. Retrieved from https://unstats.un.org/sdgs

Chapter 2
Strategic Short Note: Transforming Controlled Environment Plant Production Toward Circular Bioeconomy Systems

K. C. Ting

Abstract The American Society of Agricultural and Biological Engineers (ASABE) embarked a Roundtable 2020 initiative on transforming food and agriculture toward circular systems (TFACS). A series of online meetings were conducted to explore challenges and opportunities in transformation processes for three constituent systems of (1) open-field plant production system, (2) controlled environment food and agriculture system including controlled environment plant production system (CEPPS), and (3) animal production system, respectively. In fall 2021, ASABE appointed a TFACS taskforce to continue the work of Roundtable 2020 and start promoting the transformation of food and agriculture toward circular bioeconomy systems (CBS). This strategic short communication note reports high potential opportunities for improving the circularity of CEPPS value chains and their interconnected stages. System of systems (SoS) approach and mission-oriented coordination/collaboration are needed in providing systems informatics and analytics to effectively move the transformation forward. Intelligence driven and empowered agricultural systems (IDEAS), that make things work well, better, together, smarter, and wiser are needed in advancing the goals of circular bioeconomy systems, including CEPPS.

Keywords Circular economy · Food and agriculture · Sustainability · System of systems · Intelligent agricultural systems

2.1 Introduction

In spring 2020, American Society of Agricultural and Biological Engineers (ASABE) formed a taskforce to move forward a Roundtable 2020 initiative on transforming food and agriculture toward circular systems (TFACS). The activities

K. C. Ting (✉)

Department of Agricultural and Biological Engineering, University of Illinois at Urbana-Champaign, Champaign, IL, USA
e-mail: kcting@illinois.edu

of the initiative included conducting a series of online meetings, preparing reports, and authoring articles for a special issue of Resource, an ASABE membership magazine. The online meetings were to discuss the concept of circular bioeconomy as applied to food and agriculture, set the missions and goals, and brainstorm challenges, opportunities, and required expertise in improving the circularity of food and agriculture systems (ASABE, 2021). ASABE appointed a TFACS taskforce in Fall 2021 to continue advancing the goals of circular bioeconomy systems (CBS).

Ellen MacArthur Foundation (2019) defines the concept of circularity as "(1) design out waste and pollution, (2) keep productions and materials in use, and (3) regenerate natural systems." The TFACS taskforce added two more guidelines: increase use efficiencies and provide economic benefits. The Foundation also states three ambitions for achieving circular economy of urban food systems: "(1) source food grown regeneratively, and locally where appropriate, (2) design and market healthier food products, and (3) make the most of food." National Research Council (2015) provides a description of the links between the food supply chain (from farm input supply to consumer) and the larger biophysical and social/institution context (social organization, science and technology, biophysical environment, markets, and policies).

The current food and agriculture systems are mostly linear and open systems of systems (SoS), from resource inputs through the stages of production, harvest and storage, processing and packaging, sales, and consumption with waste processing and resource recovery and transportation and distribution as interconnections. These linear systems may be transformed to circular systems as depicted in Fig. 2.1.
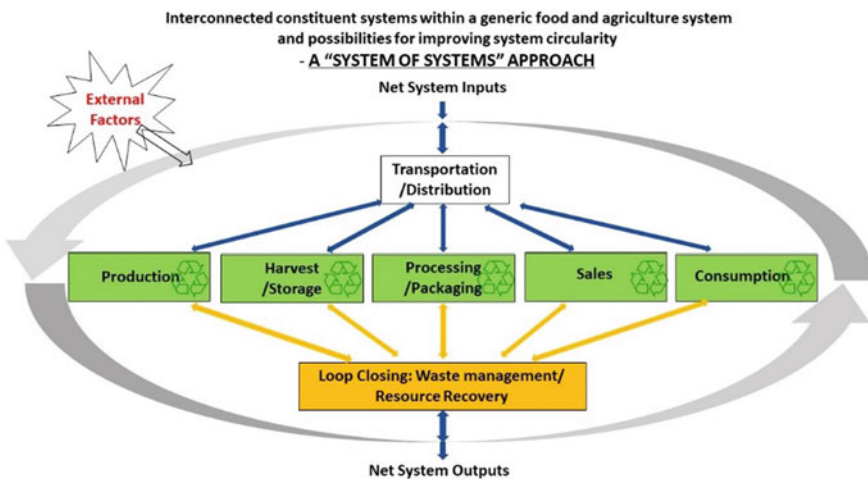
Fig. 2.1 Circular food and agriculture value chain as a system of systems

Fig. 2.2 Examples of high potential opportunities for transforming controlled environment plant production toward circular bioeconomy systems

2.2 Circular CEPPS

CEPPS has evolved from basic greenhouses to advanced greenhouses to vertical farms/plant factories. The unique features of vertical farms/plant factories are their high degree of closure and integration of sophisticated components/subsystems, like a well-structured production lines and warehouses seen in advanced manufacturing facilities. Some vertical farms/plant factories make use of both sun light and supplemental light powered by electricity. The latest versions have moved toward total "artificial" lighting (Kozai, 2018; Ting et al., 2022).

High potential opportunities for improving circularity of basic/advanced greenhouses and vertical farms/plant factories, at various stages of respective value chains and systems levels, are reported in the article by Ting et al. (2021). Examples of systems level opportunities are shown in Fig. 2.2. In addition, the concepts and methodologies of system of systems (SoS) and intelligence driven and empowered agricultural systems (IDEAS) are very useful.

2.3 Closing Remarks

CBS is both a means and a goal in making food and agriculture systems sustainable and profitable. Methodologies and tools for defining, analyzing, designing, implementing, managing, operating, and evaluating CBS are necessary in achieving its goals. Among them, the concepts and approaches of SoS and IDEAS are essential in ensuring that a CBS functions holistically and intelligently. SoS reveals the

behaviors of a system's individual constituent systems and their interactions. IDEAS may be enabled and implemented by human supervised cyber-physical systems. CEPPS, a high-tech form of bioeconomy system, has a high level of readiness for circularity improvement and may serve as a model for other types of circular food and agriculture systems.

References

ASABE. (2021). *Resource*, (March/April).

Ellen Macarthur Foundation. (2019). *Cities and circular economy for food*. Author.

Kozai, T. (2018). *Smart plant factory*. Springer.

National Research Council. (2015). *A framework for assessing effects of the food system*. The National Academies Press.

Ting, K. C., Scott, N., & Mohtar, R. (2021). Circular controlled-environment plant production systems. *Resource, 2021*(March/April), 25–29.

Ting, K. C., Timmons, M., & San Martin, R. (2022). Precision indoor farming and alternative protein food systems. *The Bridge, 2022*(Summer), 41–47.

Chapter 3
Artificial Lighting Systems for Plant Growth and Development in Indoor Farming

Munirah Hayati Hamidon and Tofael Ahamed

Abstract Due to the increasing demand for food from the growing global population, indoor farms have received attention from farmers because of their ability to produce more food as a supplement to the conventional farming system. In a highly controlled indoor environment, the application of artificial lighting is essential in replacing the function of natural sunlight. The primary purpose of artificial light sources is to promote plant growth. However, the operational requirements of indoor plants vary widely and uniquely, and thus artificial lighting systems become more complex. The light used for indoor farms must satisfactorily address a variety of performance to ensure the optimum plant growth and development. Therefore, this chapter discussed the key factors involved in the specific lighting parameters such as light quality, quantity, and duration for compliance to meet the growing requirements of indoor cultivation. Furthermore, several types of artificial lights also have been reviewed to understand which light type is reliable for indoor farms.

Keywords Indoor farming · Artificial lighting · Light quality · Light intensity · Light duration

3.1 Introduction

The land resources for agriculture have been decreasing as more areas are urbanized to accommodate other sectors, such as industry, manufacturing, and housing development. However, the global population is increasing rapidly each year and is projected to increase to 9.6 billion by 2050; thus, more food must be produced to feed this population. Hence, urban agriculture (UA) is one of the growing food security solutions to address the issues of global population and urbanization that are

M. H. Hamidon
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

rapidly increasing. UA is defined as the production, processing, and distribution of food produced in cities for local consumption. Vertical or indoor farming is one developing form of urban agriculture that is currently booming and has been explored specifically by developed countries.

Farmers are increasingly adopting indoor farming due to the possibility of regulating environmental conditions such as light and temperature in accordance with the plant types and environmental requirements. Indoor faming typically provides a much higher yield than traditional farming methods and has the potential to supplement food provided by existing agriculture practices. It allows farmers to produce their crops year-round without being affected by the increasing trend in global warming and extreme climate changes. Moreover, this advanced agricultural model has enabled crops to be grown without the need for clearing additional land.

Generally, several main factors must be considered when setting up and maintaining an indoor farm: location, type of cultivation system, crops, and technology used (Wong et al., 2020). Indoor farms range in size from small shipping containers to large vertical multi-layered buildings, enabling them to be built not only in rural areas but also in cities. Regarding the cultivation type, the soilless culture system is the most suitable cultivation practice for indoor farming, as it frees plant from soil-related issues. Leafy vegetables, small fruits such as tomatoes and strawberries, herbs, flowers, and microgreens are some of the common types of crops that are suitable to grow in indoor farms. Currently, green vegetables dominate the highest percentage of plants grown in indoor farming worldwide due to their high nutrition per calorie compared to other foods. Moreover, they are also rich in vitamins, minerals, fiber, and antioxidants (Wong et al., 2020).

The use of technology is frequently the determining factor in the success of an indoor farm. Technological advancements such as light emitting diode (LED) lighting, sensors, and automation have made indoor farming more efficient and customized for producing certain crops. The growth and development of plants grown indoors rely entirely on artificial lighting systems (Fig. 3.1). Understanding what type of light is required and how much it is required is the key to effectively and successfully growing plants indoors. Plants require an energy supply from light for photosynthesis. Photosynthesis is a process when energy from light sources converts carbon dioxide (CO_2) and water into carbohydrates and oxygen with the aid of chlorophyll pigments in the leaves. Different plants grow differentially when they receive different light qualities, intensities, and durations.

3.2   Light for Plant Growth

Light is composed of particles called photons, which are bundles of the electromagnetic field that contain a specific amount of energy. Electromagnetic radiation is one type of wave that is expressed by speed (c), wavelength (λ), and frequency (ν).
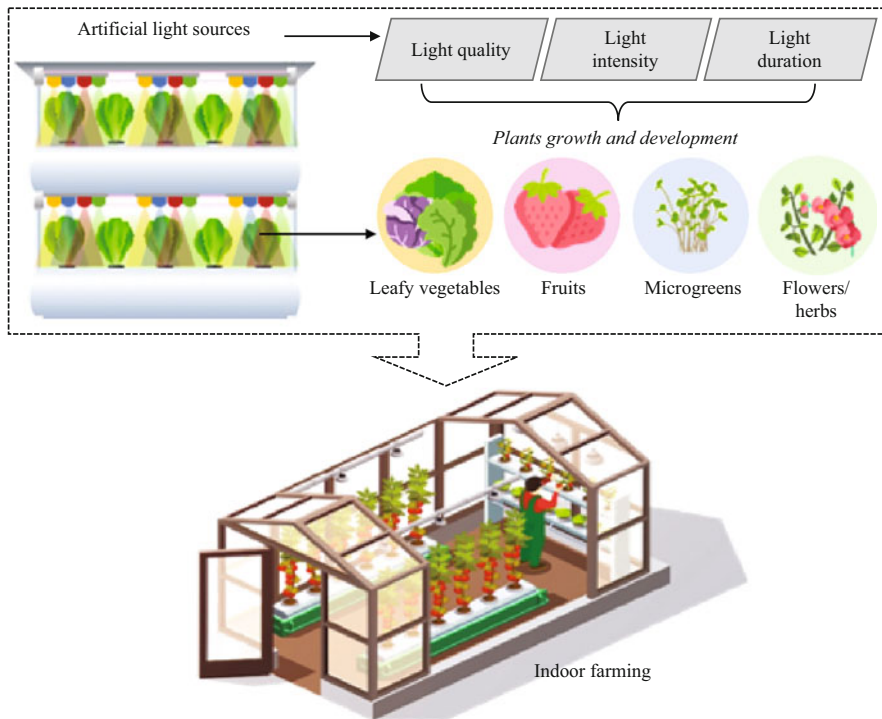
Fig. 3.1 Artificial lighting for indoor farming

$$\lambda = c/v \qquad (3.1)$$

Equation (3.1) shows that the light wavelength is inversely proportional to the frequency. An increase in frequency will decrease the light wavelength with a corresponding increase in the energy of the photons that produce the light. These photons produce light when a charged particle shifts from a higher energy level to a lower energy level (Xu et al., 2016) (Fig. 3.2). The energy of a photon is determined using the Planck–Einstein equation:

$$E = hv = hc/\lambda \qquad (3.2)$$

where h is the Planck constant.

Three key aspects of light influence how plants grow and develop: quantity or intensity, quality, and duration (Fig. 3.3) (Runkle, 2017). Light quantity refers to the intensity or concentration produced by a light source. Light quality, on the other hand, reflects the color or wavelength reaching the plant surface. Light duration, also known as the photoperiod, is defined as the amount of time a plant needs to be exposed to a light source.

Fig. 3.2 Photons produced from changes in the energy level to emit light

Change in energy levels, $\triangle E$ releases light energy

$$\triangle E = E_2 - E_1 = h.v$$

Electron at high energy level, E_2

LIGHT

Electron at low energy level, E_1

Nucleus

Light quantity (intensity)

Light quality (spectrum)

Light duration (photoperiod)

Plant growth

Plant development

Plant biomass

Morphology

Flowering/ fruiting
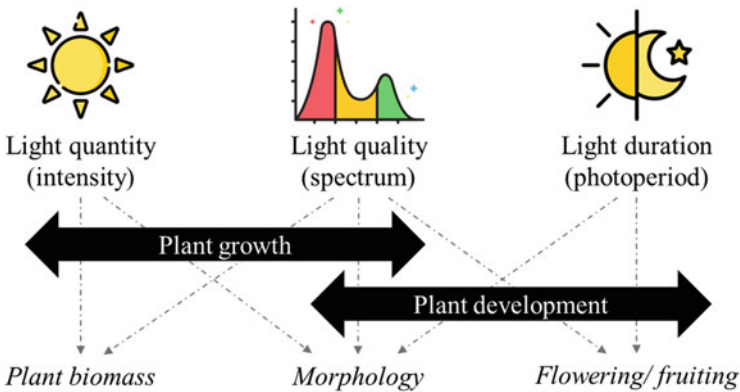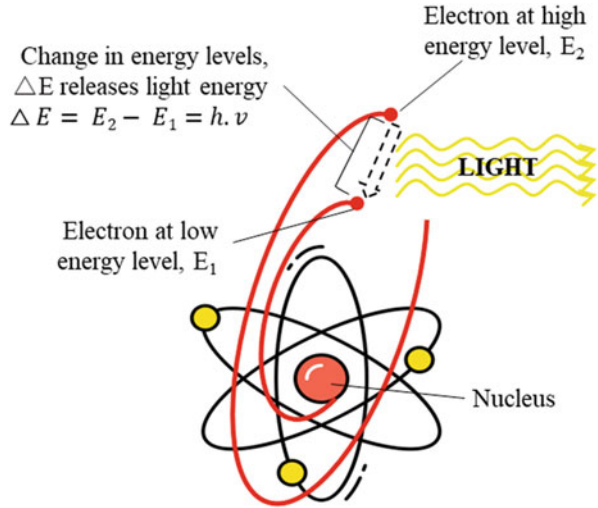
Fig. 3.3 The principles of light and its importance for plant growth and development

3.3 Light Quantity

The light quantity needed by each plant differs. Some plants may require a very minimal light intensity, but others may need a high light intensity to grow. Plants perform the photosynthesis process within the radiation spectrum between 400 and 700 nm, which is known as photosynthetically active radiation (PAR).

Numerous studies have been conducted and have shown that the light quantity is important for promoting plant growth and development. Under certain conditions and in different plant types, a higher light intensity leads to high yields. However, excess light may damage plants, and too little light will inhibit growth (Barber & Andersson, 1992). This difference is because light intensity is one of the most critical elements influencing the photosynthesis process in plants. Photosynthesis is defined
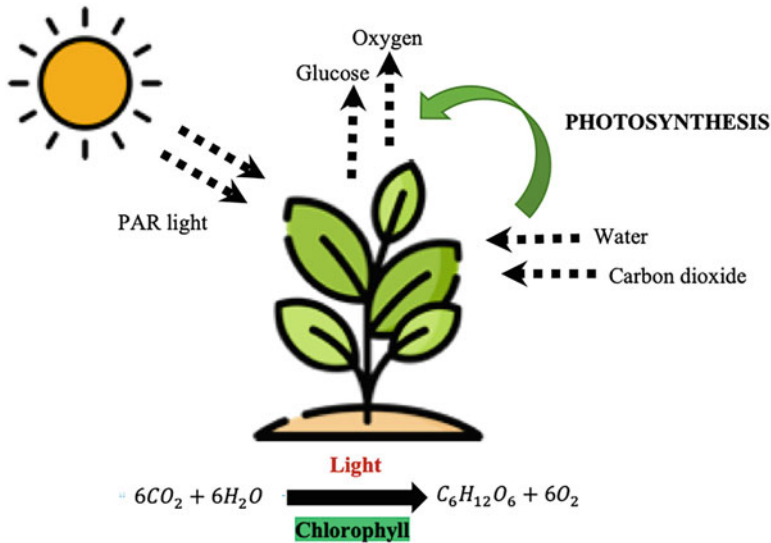
Fig. 3.4 The photosynthesis process

as the process by which green plants use light to synthesize nutrients from carbon dioxide and water to produce carbohydrates and oxygen (Fig. 3.4).

Plants store and use the carbohydrates produced during photosynthesis as their nutrient source. Every plant begins its photosynthesis process differently, depending on the energy levels of the light sources, which is known as the light compensation point. This threshold is reached when the amount of light energy available for the photosynthetic process is sufficient to create more oxygen than the plant needs for respiration. Similarly, the CO_2 released by the plants through respiration should not be greater than the amount of CO_2 consumed by the plant during photosynthesis, resulting in null or negative net photosynthesis.

Light falls to the ground as tiny particles known as photons. The photons encompass various energy quantities, depending on the light frequency. However, photosynthesis is mainly related to the number of photons rather than to their energy level. The light quantity reaching the plant varies according to the distance between the plants and the light source. The light quantity is determined by calculating the photosynthetic photon flux density (PPFD). The PPFD is measured by calculating the number of photons in the PAR region that reach a certain area at a constant distance (Fig. 3.5). It is typically reported in $\mu mol\, m^{-2} s^{-1}$, where μmol is a unit of counting based on Avogadro's number:

$$1\,\mu mol\, m^{-2} s^{-1} = 6.022 \times 10^{23} \times 10^{-6} \times \left(10^{9}\right)^{2} = 0.6022\, nm^{-2} s^{-1} \qquad (3.3)$$

By calculating the PPFD, we can understand the exact amount of light that is received by or reached the plants for photosynthesis.

Fig. 3.5 The measurement of light energy required for plants
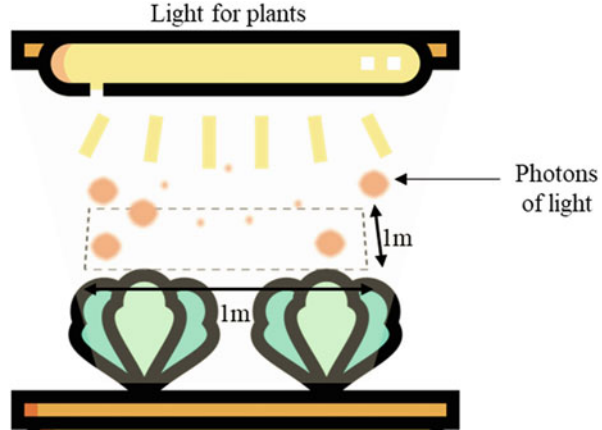
$$PPFD = \frac{\text{Photons of light}}{\text{area} \times \text{time}} \tag{3.4}$$

3.3.1 Plant Photosynthesis in Response to Light

In general, the photosynthesis rate is substantially influenced by the light intensity. Initially, the photosynthesis rate increases as the light intensity increases. However, after reaching the light saturation point, the rate eventually approaches its maximum value. The light saturation point is defined as the point when light intensity no longer affects the rate of photosynthesis. The photosynthesis rate curve or light response curve (LRC) becomes flat when it reaches this saturation point (Fig. 3.6).

The LRC model quantifies the relationship between photosynthesis and light intensity (Herrmann et al., 2020; Wan et al., 2018). The model is diversified and has been determined using several empirical models: (1) a rectangular hyperbola model, (2) a modified model of the rectangular hyperbola, (3) a nonrectangular hyperbola model, and (4) an exponential function model (Wan et al., 2018).

1. **Rectangular Hyperbola Model**

   The rectangular hyperbola model is calculated as follows:

$$P_n(I) = \frac{\alpha I P_{max}}{\alpha I + P_{max}} - R_d \tag{3.5}$$

where $P_n(I)$ is the net photosynthesis rate, I is the photosynthesis photon flux density, α is the initial gradient of the light response curve, P_{max} is the maximum net photosynthetic rate and R_d is the dark respiration rate. P_{max} is determined as:
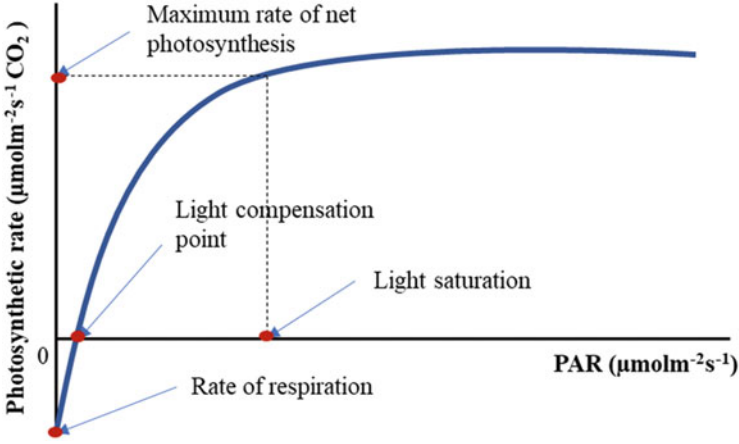
Fig. 3.6 Change of photosynthesis rate with the light intensity

$$P_{max} = \alpha \left(\frac{\sqrt{\beta + \gamma} - \sqrt{\beta}}{\gamma} \right)^2 - R_d \tag{3.6}$$

where β is the humidity coefficient and γ is the temperature coefficient.

2. **Modified Model of the Rectangular Hyperbola**

The mathematical expression of the modified model of the rectangular hyperbola is described as follows:

$$P_n(I) = \alpha \frac{1 - \beta I}{1 - \gamma I} I - R_d \tag{3.7}$$

where α is the initial gradient of the light response curve, β and γ are the humidity and temperature coefficients, respectively, I is the photosynthesis photon flux density, and R_d is the dark respiration.

3. **Nonrectangular Hyperbola**

The following equation is used to calculate the nonrectangular hyperbola model:

$$P_n(I) = \frac{\alpha I + P_{max} - \sqrt{(\alpha I + P_{max})^2 - 4\theta \alpha I P_{max}}}{2\theta} - R_d \tag{3.8}$$

where $P_n(I)$ is the net photosynthesis rate, I is the photosynthesis photon flux density, θ is the curve curvature, and α is the gradient of the plant photosynthesis to the light response curve when $I = 0$, namely, the initial gradient of the light response curve, which is also called the initial quantum efficiency. P_{max} is the maximum net photosynthetic rate, and R_d is the dark respiration rate.

4. **Exponential Function Model**

   The exponential function model that been used to determine the light response of plant photosynthesis is calculated as follows:

$$P_n(I) = P_{max} \left(1 - e^{-\alpha I / P_{max}} \right) - R_d \tag{3.9}$$

   In this equation, the definitions of all parameters are the same as those described in Eqs. (3.5) to (3.8).

3.4 Light Quality

Light quality indicates the light wavelength or spectrum. The wavelengths emitted by the sun range from 280 to 2800 nm, which is 97% of the total spectral distribution (Kozai et al., 2015). They are classified as ultraviolet, visible light, and infrared. The lowest wavelengths have the highest energy, indicating that ultraviolet radiation has higher energy than infrared radiation. The wavelengths visible to humans range from 380 to 770 nm, which are referred to as visible light. The visible light is split into violet (380–430 nm), blue (430–500 nm), green (500–570 nm), yellow (570–590 nm), orange (590–630 nm), and red (630–770 nm).

   Two prominent types of effective radiation for plants are known as (1) photosynthetically active radiation (PAR) and (2) physiologically active radiation. The physiologically active radiation is divided into five wavebands: Ultraviolet (UV) 300–400 nm, blue light (B) 400–500 nm, green light (G) 500–600 nm, red light (R) 600–700 nm, and far-red light (FR) 700–800 nm. Meanwhile, the waveband of PAR ranges from 400 to 700 nm (Fig. 3.7) (Tazawa, 1999; McCree, 1981).

   The green pigment in plants that absorbs PAR, chlorophyll, has two absorption peaks: red and blue light. Plants usually absorb only a small amount of green light and reflect it back, which is the reason most plants appear green to humans (Fig. 3.8).
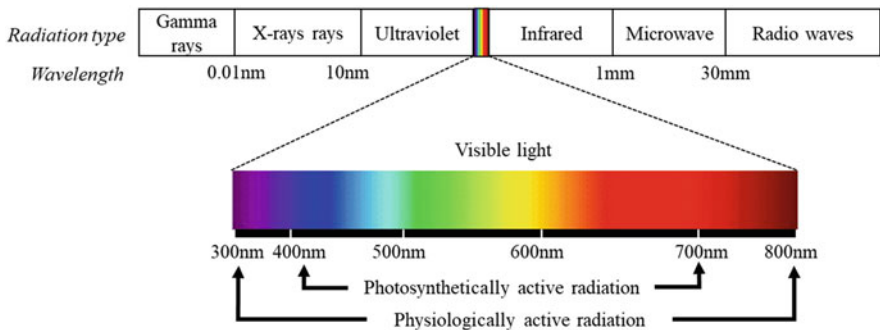
Fig. 3.7 The visible light spectrum distributions

Visible radiation
- 80% absorbed: mostly red and blue
- 10% reflected: mostly green
- 10% transmitted: others
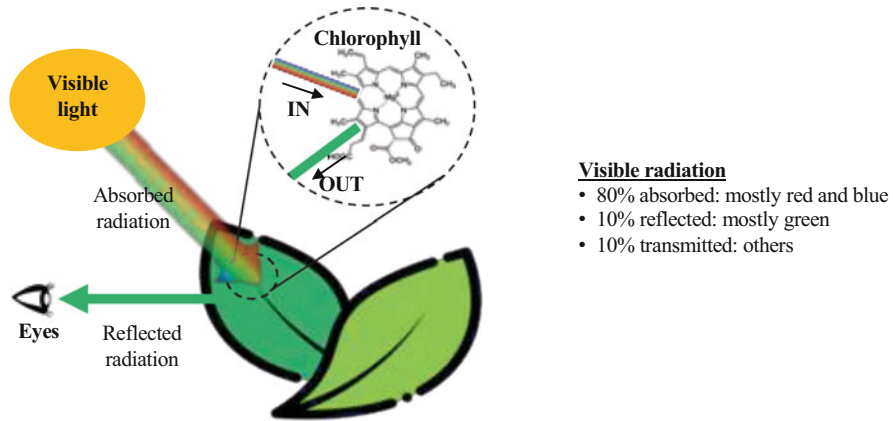
Fig. 3.8 Spectrum of plant absorption, reflection, and transmission

The functions of light for food production should be to promote rapid seedling growth, improve branching density, support the developing leaf size, and enhance the flowering/fruiting cycle irrespective of the season to maximize production quality and quantity. An understanding of the effects of different wavelengths of the light spectrum on plant growth is essential to provide a proper artificial light system for cultivating vegetables in indoor environments. Light quality not only affects the growth and morphology of plants but also regulates metabolite accumulation (phytochemicals) in plants (Fig. 3.9) (Paradiso & Proietti, 2021; Wong et al., 2020).

3.4.1 Light Energy Use Efficiency of Lamps (LUE_L) and Plant Community (LUE_P)

When estimating the photosynthesis rate or production of plants in indoor farms, all the essential resources for plant growth, such as water, light, CO_2, and temperature, are connected (Fig. 3.10) (Kozai, 2011). Among these resources, the usage of artificial light in indoor farming is the most costly. Hence, evaluating the light energy use efficiency of lamps (LUE_L) and plant communities (LUE_P) is the primary concern in plant production for indoor farms (Kozai, 2013).

The LUE_L and LUE_P are determined using the as following equations:

$$LUE_L = fD/PAR_L \tag{3.10}$$

$$LUE_P = fD/PAR_P \tag{3.11}$$

In these equations, f is the conversion factor from dry mass to chemical energy fixed in dry mass (approximately 20 MJ kg^{-1}), D is the increase in the dry mass of
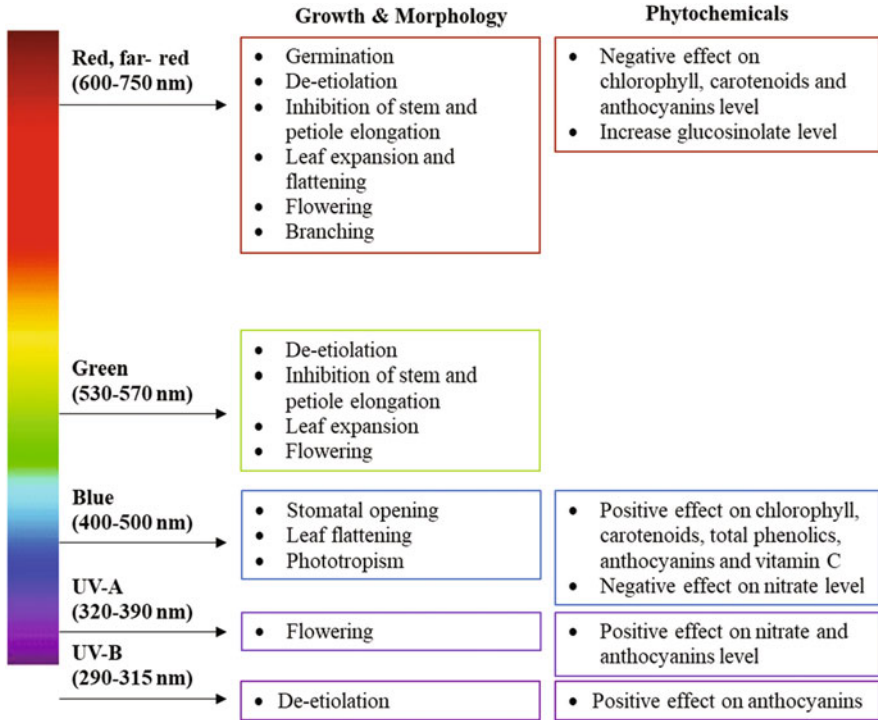
Fig. 3.9 The effect of the light spectrum on plant growth and development
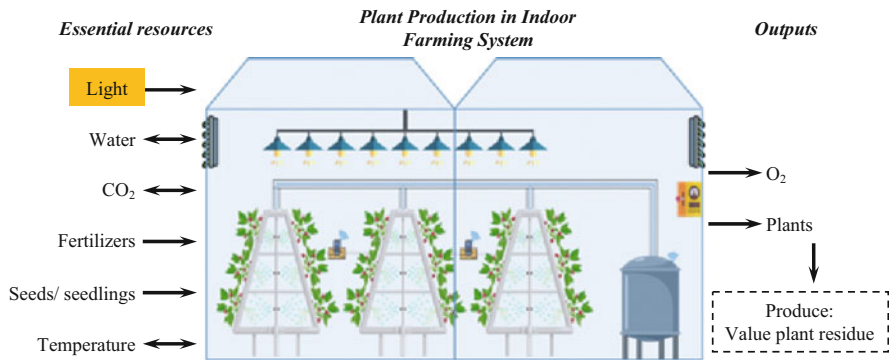
Fig. 3.10 Essential resources for plants grown in indoor farms

whole plants or salable parts of plants in the closed plant production system (CPPS) (kg m^{-2} h^{-1}), PAR$_L$ is the photosynthetically active radiation (PAR) emitted from lamps, and PAR$_P$ is PAR received at the plant community surface in the CPPS (MJ m^{-2} h^{-1}). Alternatively, LUE$_L$ and LUE$_P$ have also been calculated using the following equations:
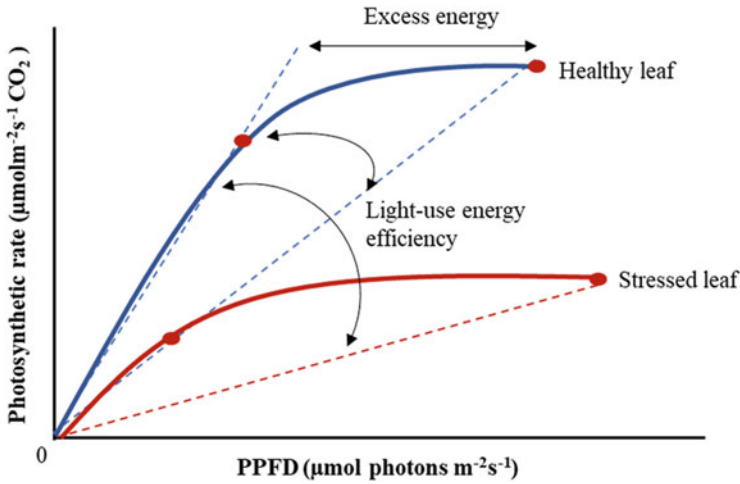
Fig. 3.11 Light use efficiency (LUE) concept for estimating the photosynthesis rate

$$LUE_L = bC_P/PAR_L \qquad (3.12)$$

$$LUE_P = bC_P/PAR_P \qquad (3.13)$$

where b is the minimum PAR energy to fix 1 mol of CO_2 in plants (0.475 MJ mol^{-1}) and C_P is the net photosynthetic rate of plants (mol m^{-2} h^{-1}). The ratio of PAR$_P$ to PAR$_L$ is known as the "utilization factor" in illumination engineering. The LUE models have become valuable tools for estimating vegetation productivity (Fig. 3.11) (Gamon, 2015).

3.5   Light Duration or Photoperiod

The quantity of daily daylight hours exerts a direct effect on the growth and flowering of many plant species. Therefore, the manipulation of daylight hours is important for scheduling the amount of light needed for certain plants growing indoors. A photoperiod is defined as the light duration within a specific time span, usually a 24-h period. Photoperiod control is essential for indoor farming, especially for the flowering process, as every plant responds differently to the relative length of the light and dark periods (Boyle, 1992). Plants are categorized into three classes according to the light duration needed to trigger blooming or flowering.

1. **Day-Neutral Plants (DNP)**
    The flowering time for day-neutral plants may be any duration of the day. Normally, they bloom after achieving specified developmental phases. Examples include tomatoes, cucumber, hydrangea, sunflower, and pepper.

2. **Short-Day Plants (SDP)**

These plants only require a short period of daylight (approximately 8–10 h). The flowering process only occurs when the photoperiod is less than a critical day length (Fig. 3.12). For SDP, flower blooms occur either in early spring or fall. When the day length surpasses a certain threshold, plants discontinue flowering and begin the vegetative phase. Examples include potato, zinnia, soybean, cosmos, poinsettia, and chrysanthemum.

3. **Long-Day Plants (LDP)**

LDP require a relatively longer time of daylight than night to bloom (usually 14–16 h). The plants only bloom when the photoperiod surpasses the critical day length (Fig. 3.12). Late spring to early summer is the usual flowering season for LDP. Examples include hibiscus, radish, cabbage, spinach, and lettuce.

By understanding how daylength affects plant development, we can further manipulate the photoperiod of indoor farming to promote different stages of
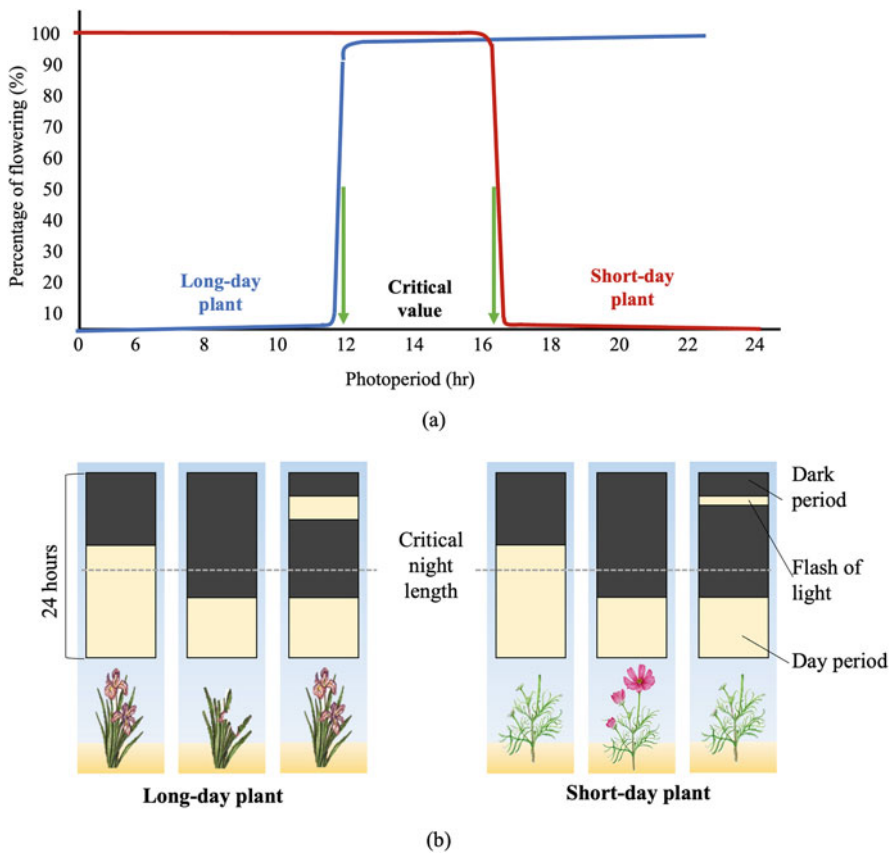
Fig. 3.12 (**a**) The photoperiodic response in long- and short-day plants. (**b**) The photoperiodic regulation of flowering in long- and short-day plants

development in photoperiod-responsive crops. Furthermore, flowering in plants is more responsive to dark periods than to light periods. Thus, manipulation of the photoperiod benefits growers in planning the flowering process for their plants and ensuring that they produce the flower even if it is not during their flowering period. If growers need the LDP to bloom, they can extend the light period into the night, which is known as night interruption control.

3.5.1 Daily Light Integral (DLI)

Often, the light duration or photoperiod of plant growth is related to the daily light integral (DLI). The DLI is the total number of photons of PAR accumulated in a particular area received by plants in 24 h. It is an important index that combines both the light intensity and photoperiod (Xu et al., 2021). DLI exerts significant effects not only on plant growth and development but also on plant yield and quality, as it ensures that the plants perceive the optimum amount of light according to their growth cycle (Xu et al., 2021).

The DLI is a simple metric to calculate for indoor farming compared to cultivation outdoors, since light intensity and photoperiod remain constant in each stage of cultivation. In outdoor cultivation, calculating the DLI becomes trickier due to the atmosphere and seasonal changes that occur during the 24-h period. The DLI is calculated as follows:

$$DLI\left(mol/m^2\,d\right) = \frac{PPFD\ \left(\mu mol/m^2\,s\right) \times Photoperiod(h/d) \times 3600(s/h)}{10^6} \quad (3.14)$$

By determining the DLI in indoor farming, growers can strategize their artificial lighting operation so that it meets the specific requirements corresponding to the plants' growth cycle and stage.

3.6 Artificial Lights for Plants Growth

A significant increase in the number of studies conducted on the use of lighting in the fields of plant and animal production has been observed during the last several decades. Different types of lighting, light wavelength ratios, and light intensities on various species of plants have been the research focus for improving plant growth, particularly in indoor horticulture fields. Studies have reported that even when equivalent light treatments are applied, the growth outcomes vary depending on the plant species.

Generally, artificial light sources are categorized into three main types according to the principle of light emission: incandescence, discharge light emission, and electroluminescence (Kozai et al., 2015; Tazawa, 1999) (Fig. 3.13).
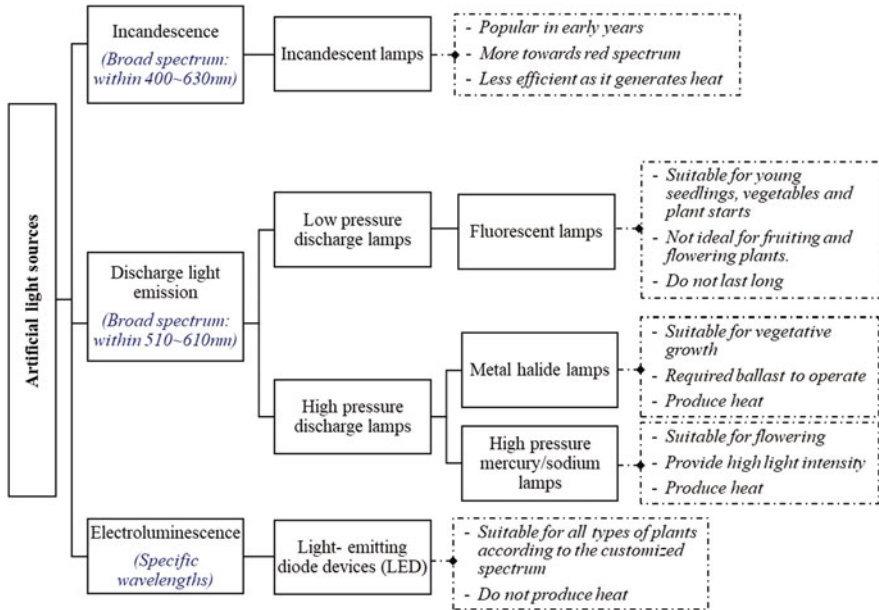
Fig. 3.13 Classification of artificial light sources

3.6.1 *Incandescent Lamps*

The incandescent lamp was one of the pioneering artificial light sources used for indoor farming before any other lights were developed. Incandescent light sources, as the name indicates, create light by incandescence, which is the emission of light due to heat. An incandescent light bulb is generated by placing a filament in a vacuum or inert-gas-filled container called a bulb. When an electric current is passed across the filament, it will heat up and glow with blackbody radiation and finally achieve incandescence. The filament must be placed in a bulb to prevent the metal from oxidizing and overheating. Instead of a vacuum, inert gas helps delay the evaporation of the heated filament.

In the early years, incandescent light bulbs were used as indoor lighting since they only need a power supply to operate without any supplementary equipment. The production costs are also relatively inexpensive and may operate on either alternating current (AC) or direct current (DC). However, they are very inefficient due to the intrinsic physics of blackbody radiation and are thus much less suitable for plant growth compared to other grow lights. A substantial amount of the input energy utilized to generate light is wasted as heat. When contemplating their usage to grow plants, the high heat generation of incandescent lights may be harmful to plants.

Incandescent lights are not recommended to be placed closer than 24 inch to a plant to avoid damage. However, this minimum operating distance established for safety reasons may prevent the plants from receiving proper light intensity and eventually affect their growth. Additionally, incandescent lights emit more toward
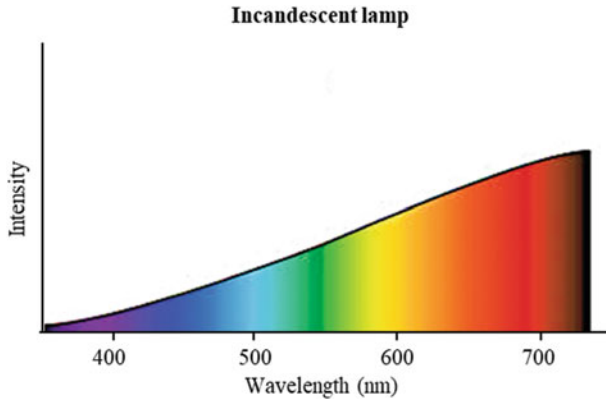
Incandescent lamp

Fig. 3.14 Spectrum of the incandescent lamp

the red spectrum, depriving plants of blue light (Fig. 3.14) (Trinklein, 2016; G2V Optics Inc., n.d.).

3.6.2 Fluorescent Lamps

Another trendy type of lamp that is used for indoor farming is fluorescent lamps. A fluorescent lamp generates light by transferring energy via a gas contained within a tube. It generates visible light using the physical principles of optical fluorescence via a low-pressure mercury-vapor gas discharge. Fluorescence occurs when electrons return from an excited singlet state to their ground state.

Fluorescent lights outperform incandescent bulbs in producing the light required for a variety of plants, particularly vegetables grown inside, since the lights are far more efficient and produce light mainly in the visible spectrum (Fig. 3.15) (Trinklein, 2016). Moreover, fluorescent lights come in a variety of colors and generate lower temperatures. Another advantage is that they typically have a long lifespan. Despite their advantages over incandescent bulbs, they require additional equipment, a ballast, which is a device that restricts the amount of current flowing through a circuit, for their operation. The ballast is needed to maintain the light at the proper voltage and current level. In addition, they also do not last very long, are delicate, bulky, and have limitations in providing a high lumen intensity.

3.6.3 High Intensity Discharged (HID) Lamps

HID lamps have been traditionally used to supplement indoor farm lighting that requires greater light intensities than fluorescent lighting. They are also known as

Fig. 3.15 Spectrum of the fluorescent lamp
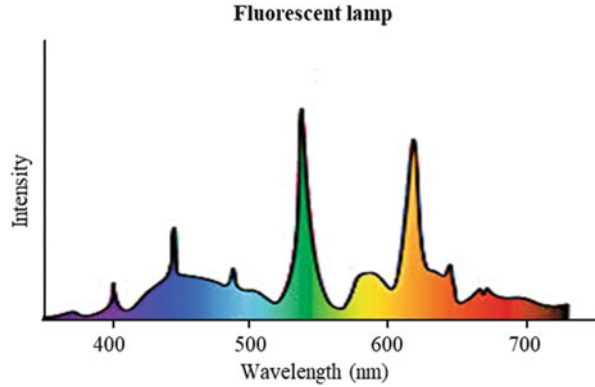
high-pressure discharge lamps, which operate at very high pressure and temperature (Dutta Gupta & Agarwal, 2017). A series of high voltage electrical pulses are transmitted between two electrodes at the end of the tube to generate the HID light. Similar to a fluorescent lamp, an HID lamp also requires ballast, as it provides a high voltage to generate light and sustains a sufficient electrical current flow to the bulb.

Two main popular types of HID lamps are used for indoor farming: high-pressure sodium (HPS) and ceramic metal halide (MH) lamps. Both HPS and MH lamps use a gas discharge lamp with a combination of various vaporized metals, each of which has a distinct color range. The HPS lamps mostly emit a fairly monochromatic yellow–orange light, while the MH fixtures tend to produce a blue–white color of light (Fig. 3.16) (U.S. Department of Energy, 2017).

3.6.4 Light Emitting Diodes (LED)

LEDs are more robust and have longer lifespans than other traditional light sources, such as incandescent and fluorescent lamps. The color of light emitted from the LED is determined by the semiconductor material composition of an LED, which defines the total wavelength of the photon light emissions. With the currently available cutting-edge technology, LEDs can be customized to emit a certain wavelength and generate either narrow- or wide-band light that matches a specific plant requirement for maximum growth (Fig. 3.17) (Xu et al., 2016).

For instance, red and blue LEDs are frequently used because they span the spectrum where plants most efficiently absorb light. Nevertheless, some studies have shown that certain plants also utilize some parts from green and yellow regions, suggesting that grow light producers must develop LEDs with broader radiation. Additionally, some wavelengths outside the 400–700 nm range may also exert beneficial effects on plant growth. As a result, some manufacturers have modified

High pressure sodium

Metal halide
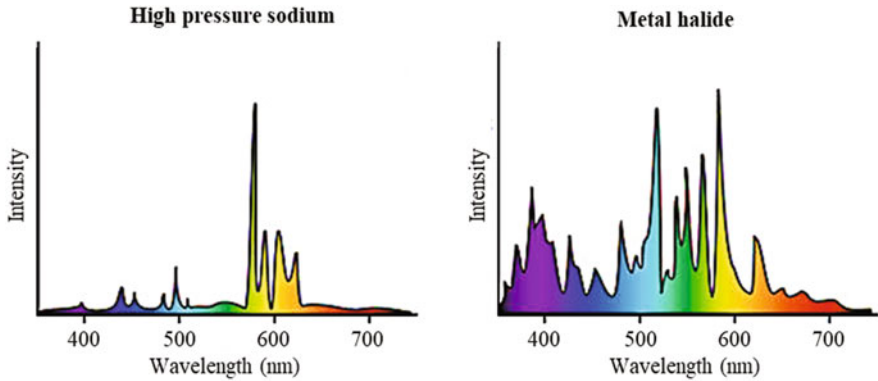
Fig. 3.16 Spectra of the HPS and MH lamps

Monochromatic LEDs

White LED (Blue LED with phosphor)
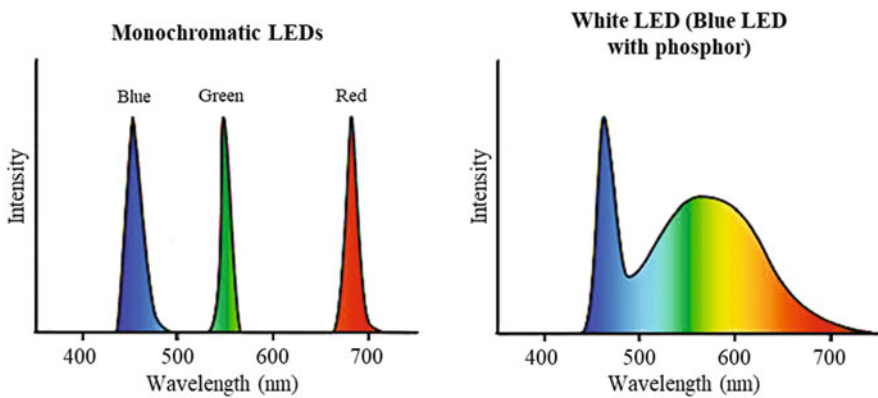
Fig. 3.17 Spectra of LEDs

LEDs such that they also deliver light that encompasses areas below 400 nm or above 700 nm.

As mentioned above, most of the grow light sources not only produce light but also produce heat, which may be harmful to the plants. Depending on the grow light types, plants may experience heat stress and localized low humidity if they are positioned too close to those lights. LED lights generate far less heat. Unlike HID, placing LEDs closer to the plants may help maximize the light intensity and spectrum received by the plants without overheating them. Additionally, LEDs are environmentally friendly, work without a ballast and have extended lamp lifetimes. Due to their effectiveness and affordability, LED grow lights are increasingly
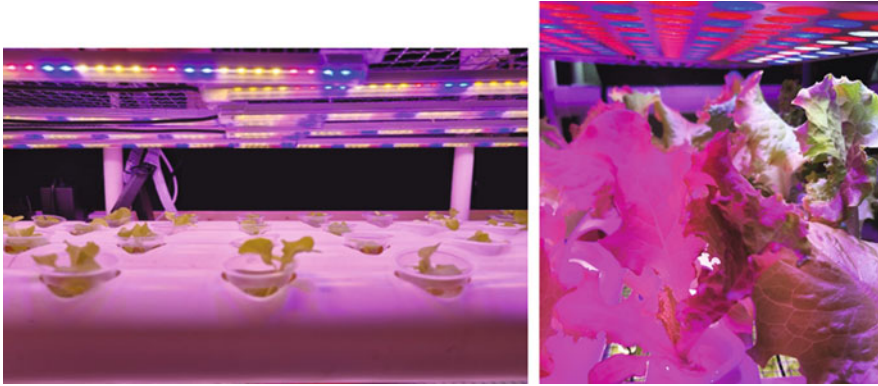
Fig. 3.18 Example of the usage of LEDs for growing lettuces indoors

replacing other grow lights, such as fluorescent lights and HIDs, in indoor farming operations (Fig. 3.18).

3.7 Effect of Using Artificial Light on Plants Grown Indoor

Types of plants	Plant	Light	Light duration	Effect on plants	References
Leafy vegetables	Chinese cabbage	Red + Blue + White LED (130 µmol m^{-2} s^{-1})	18 h day^{-1}	Optimal growth of leaf length and weight.	Duc et al. (2021)
		Fluorescent lamps	14 h day^{-1}	High glucosinolate content.	
	Lettuce ("Cheongchima")	External electrode fluorescent lamps (150 µmol m^{-2} s^{-1})	20 h day^{-1}	Increased fresh weight, dry weight, leaf area, and daily biomass accumulation rate.	Cho et al. (2020)
	Sweet basil	Red (663 nm) + Blue (435 nm) LED	8 h day^{-1}	Increased the growth yield.	Rihan et al. (2020)
	Water spinach	Red LED (200 ± 15 µmol m^{-2} s^{-1})	14 h day^{-1}	Enhanced water spinach growth without causing intumescence.	Kitayama et al. (2019)
	Lettuce ("Frillice Crisp")	Far-red (700–800 nm) + Red (600–700 nm) + Blue (400–500 nm) LED	–	Promoted the growth and increased macro- and micro-nutrient content.	Pinho et al. (2017)
Fruit vegetable/fruits	Tomato ("Dometica")	HPS lamps + LED (30 mol m^{-2} day^{-1})	18 h day^{-1}	Increased fresh weigh of fruits.	Verheul et al. (2022)
	Tomato and red pepper	Cool white LED + supplemental Far-red LED (200 µmol m^{-2} s^{-1})	16 h day^{-1}	Increased hypocotyl length.	Hwang et al. (2020)
	Tomata ("Merlice")	HPS lamps (20 µmol m^{-2} s^{-1})	16 h day^{-1}	Increased leaf temperature, photosynthetic rate, transpiration rate,	Kim et al. (2019)

(continued)

Types of plants	Plant	Light	Light duration	Effect on plants	References
				and plant tissues water content	
	Tomata ("Komeett")	Far-red (730 nm) + Red (638 nm) + Blue (450 nm) LED	16 h day[1]	Increased plant growth, accelerated flowering, and increased fruit yield.	Kalaitzoglou et al. (2019)
	Strawberry ("Maehyang")	Mint-white LEDs (180 µmol m^{-2} s^{-1})	16 h day[1]	Increased leaves number and area, top/root dry weight ratio, and number of newly formed runners/propagule.	Lee et al. (2020)
	Everbearing strawberry ("Elan")	White LED (300 µmol m^{-2} s^{-1})	24 h day[1]	Increased yield, TSS, ascorbic acid, anthocyanin, and crop productivity.	Maeda and Ito (2020)
	Capsicum seedlings ("CAU-24")	White LED (with Red + Blue ratio) (250 µmol m^{-2} s^{-1})	12 h day[1]	Increased seedlings biomass, chlorophyll content, and photosynthetic rate.	Liu et al. (2019)
Microgreens	Kale, Cabbage, Arugula, and Mustard	Red + Blue LED light (100–600 µmol m^{-2} s^{-1})	16 h day[1]	Fresh and dry weight increased as light intensity increased, but decreased hypocotyl length and hue angle linearly.	Jones-Baumgardt et al. (2019)
	Soybean sprout	White fluorescence light (100 µmol m^{-2} s^{-1})	14 h day[1]	Reduced hypocotyl length but increased the diameter of hypocotyls.	Yuan et al. (2015)
	Common buckwheat sprouts	Blue LED (12.41 µmol m^{-2} s^{-1})	–	Increased hypocotyl length.	Lee et al. (2014)
Flowers	Peony ("Da Fugui")	Red + Blue LED light (200–220 µmol m^{-2} s^{-1})	14 h day[1]	Increased number of flowers, enhanced blooming rate, flower diameter, and florescence.	Wan et al. (2020)
	Digitalis purpurea	Red + Blue LED light (150 µmol m^{-2} s^{-1})	18 h day[1]	Increased micro- and macro-elements as well as cardenolides accumulation.	Verma et al. (2018)

3.8   Conclusion

In conclusion, this chapter has provided adequate information on principles of light (light intensity, quality, and photoperiod) and their importance as main indicators that must be considered when using artificial lighting in indoor farms. Higher irradiance is often associated with faster development in most plants because it produces more photons to drive photosynthesis and enhance biomass accumulation. However, irradiance greater than 300 μmol m^{-2} s^{-1} may have reached the light saturation limit. As a result, photosynthetic efficiency decreases and affects the light consumption efficiency.

Despite having many diverse types of light, LEDs are currently the preferred light source for indoor farming lighting. With appropriate lighting control, they provide sufficient flexibility in altering the spectrum range of artificial lights. This approach may appropriately influence the biological and physiological attributes of a plant's growth. Furthermore, intelligent dimming may also be used to offer a seamless transition between photoperiods while achieving the required plant irradiance, depending on the time of the day.

The photoperiod is an environmental indicator that regulates flowering and has been manipulated to adjust the process of flowering for indoor ornamental crops. Photoperiod requirements will be adjusted based on plant types, mostly for flowers of either long-day or short-day plants. However, research on the effect of the photoperiod on leafy vegetables cultivated indoors is relatively less prominent. Compared to the light quantity and quality, photoperiod exerts less of an effect on the growth of and phytonutrient accumulation in leafy vegetables. The photoperiods (14–18 h light) that are normally used for leafy vegetables are currently at the optimal level. Hence, any changes in the common photoperiod of the leafy vegetables will not provide any substantial advantages for their growth.

By knowing the light information, growers using indoor cultivation can select and choose the best light intensity and light quality to achieve the preferred growth characteristics of their plants. Furthermore, this knowledge will also enable light producers to invent more ideal lighting devices, depending on the market targeted for consumption and the morphological attributes desired by indoor farming growers.

References

Barber, J., & Andersson, B. (1992). Too much of a good thing: Light can be bad for photosynthesis. *Trends in Biochemical Sciences, 17*(2), 61–66.

Boyle, T. (1992). Photoperiod control systems for greenhouse crops. *Floral Notes, 4*(6), 2–4.

Cho, J. Y., Yoo, K. S., Kim, J., Choi, B. J., & Oh, W. (2020). Growth and bioactive compounds of lettuce as affected by light intensity and photoperiod in a plant factory using external electrode fluorescent lamps. *Horticultural Science and Technology, 38*, 645.

Duc, N. V., Chowdhury, M., Ali, M., Park, S. U., Kim, Y. J., & Chung, S. O. (2021). Effects of light conditions on growth and glucosinolate content of Chinese cabbage grown in a plant factory. In

K.-J. Bergstrand & M. T. Naznin (Eds.), *IX International Symposium on Light in Horticulture* (Vol. 1337, pp. 171–178). ISHS.

Dutta Gupta, S., & Agarwal, A. (2017). Artificial lighting system for plant growth and development: Chronological advancement, working principles, and comparative assessment. In S. D. Gupta (Ed.), *Light emitting diodes for agriculture* (pp. 1–25). Springer.

G2V Optics Inc. (n.d.). *Horticulture lighting for commercial growers*. Author. Retrieved from https://www.mtbrandao.com/files/products/Horticulture_Lighting_Article_by_G2V_Optics.pdf

Gamon, J. A. (2015). Optical sampling of the flux tower footprint. *Biogeosciences Discussions, 12*(6), 4973.

Herrmann, H. A., Schwartz, J. M., & Johnson, G. N. (2020). From empirical to theoretical models of light response curves-linking photosynthetic and metabolic acclimation. *Photosynthesis Research, 145*(1), 5–14.

Hwang, H., An, S., Lee, B., & Chun, C. (2020). Improvement of growth and morphology of vegetable seedlings with supplemental far-red enriched LED lights in a plant factory. *Horticulturae, 6*(4), 109.

Jones-Baumgardt, C., Llewellyn, D., Ying, Q., & Zheng, Y. (2019). Intensity of sole-source light-emitting diodes affects growth, yield, and quality of Brassicaceae microgreens. *HortScience, 54*(7), 1168–1174.

Kalaitzoglou, P., Van Ieperen, W., Harbinson, J., Van der Meer, M., Martinakos, S., Weerheim, K., Nicole, C., & Marcelis, L. F. (2019). Effects of continuous or end-of-day far-red light on tomato plant growth, morphology, light absorption, and fruit production. *Frontiers in Plant Science, 10*, 322.

Kim, H. J., Lin, M. Y., & Mitchell, C. A. (2019). Light spectral and thermal properties govern biomass allocation in tomato through morphological and physiological changes. *Environmental and Experimental Botany, 157*, 228–240.

Kitayama, M., Nguyen, D. T., Lu, N., & Takagaki, M. (2019). Effect of light quality on physiological disorder, growth, and secondary metabolite content of water spinach (Ipomoea aquatica forsk) cultivated in a closed-type plant production system. *Horticultural Science and Technology, 37*, 206.

Kozai, T. (2011). Improving light energy utilization efficiency for a sustainable plant factory with artificial light. *Proceedings of Green Lighting Shanghai Forum, 2011*, 375–383.

Kozai, T. (2013). Resource use efficiency of closed plant production system with artificial light: Concept, estimation and application to plant factory. *Proceedings of the Japan Academy, 89B*(10), 447–461.

Kozai, T., Niu, G., & Takagaki, M. (2015). *Plant factory: An indoor vertical farming system for efficient quality food*. Academic Press.

Lee, S. W., Seo, J. M., Lee, M. K., Chun, J. H., Antonisamy, P., Arasu, M. V., Suzuki, T., Al-Dhabi, N. A., & Kim, S. J. (2014). Influence of different LED lamps on the production of phenolic compounds in common and Tartary buckwheat sprouts. *Industrial Crops and Products, 54*, 320–326.

Lee, H., Park, S. W., Pham, M. D., Hwang, H., & Chun, C. (2020). Effect of the light spectrum of white LEDs on the productivity of strawberry transplants in a plant factory with artificial lighting. *Horticulture, Environment, and Biotechnology, 61*(6), 971–979.

Liu, N., Ji, F., Xu, L., & He, D. (2019). Effects of LED light quality on the growth of pepper seedling in plant factory. *International Journal of Agricultural and Biological Engineering, 12*(5), 44–50.

Maeda, K., & Ito, Y. (2020). Effect of different PPFDs and photoperiods on growth and yield of everbearing strawberry 'Elan' in plant factory with white LED lighting. *Environmental Control in Biology, 58*(4), 99–104.

McCree, K. J. (1981). Photosynthetically active radiation. In O. L. Lange et al. (Eds.), *Physiological plant ecology I* (pp. 41–55). Springer.

Paradiso, R., & Proietti, S. (2021). Light-quality manipulation to control plant growth and photo-morphogenesis in greenhouse horticulture: The state of the art and the opportunities of modern LED systems. *Journal of Plant Growth Regulation, 41*, 1–39.

Pinho, P., Jokinen, K., & Halonen, L. (2017). The influence of the LED light spectrum on the growth and nutrient uptake of hydroponically grown lettuce. *Lighting Research & Technology, 49*(7), 866–881.

Rihan, H. Z., Aldarkazali, M., Mohamed, S. J., McMulkin, N. B., Jbara, M. H., & Fuller, M. P. (2020). A novel new light recipe significantly increases the growth and yield of sweet basil (Ocimum basilicum) grown in a plant factory system. *Agronomy, 10*(7), 934.

Runkle, E. (2017). *Effects of blue light on plants*. Michigan State University Extension: Floriculture Team. Retrieved from http://www.flor.hrt.msu.edu/assets/Uploads/Blue-light.pdf

Tazawa, S. (1999). Effects of various radiant sources on plant growth (Part 1). *Japan Agricultural Research Quarterly, 33*, 163–176.

Trinklein, D. H. (2016). *Lighting indoor houseplants*. University of Missouri. Retrieved from https://extension.missouri.edu/media/wysiwyg/Extensiondata/Pub/pdf/agguides/hort/g06515.pdf

U.S Department of Energy. (2017). *Energy savings potential of SSL in horticultural applications*. Author. Retrieved from https://www.energy.gov/sites/prod/files/2017/12/f46/ssl_horticulture_dec2017.pdf

Verheul, M. J., Maessen, H. F., Paponov, M., Panosyan, A., Kechasov, D., Naseer, M., & Paponov, I. A. (2022). Artificial top-light is more efficient for tomato production than inter-light. *Scientia Horticulturae, 291*, 110537.

Verma, S. K., Gantait, S., Jeong, B. R., & Hwang, S. J. (2018). Enhanced growth and cardenolides production in Digitalis purpurea under the influence of different LED exposures in the plant factory. *Scientific Reports, 8*(1), 1–12.

Wan, L., Xing, Z., Chang, X., Liu, J., & Zhang, G. (2018). Research on light response curve fitting model of four Chamaenerion plants on the Serzilla Mountains. *American Journal of Plant Sciences, 9*(8), 1630–1645.

Wan, Y., Wu, Y., Zhang, M., Hong, A., & Liu, Y. (2020). Effects of photoperiod extension via red–blue light-emitting diodes and high-pressure sodium lamps on the growth and photosynthetic characteristics in Paeonia lactiflora. *Acta Physiologiae Plantarum, 42*(12), 1–9.

Wong, C. E., Teo, Z. W. N., Shen, L., & Yu, H. (2020). Seeing the lights for leafy greens in indoor vertical farming. *Trends in Food Science & Technology, 106*, 48–63.

Xu, Y., Chang, Y., Chen, G., & Lin, H. (2016). The research on LED supplementary lighting system for plants. *Optik, 127*(18), 7193–7201.

Xu, W., Lu, N., Kikuchi, M., & Takagaki, M. (2021). Continuous lighting and high daily light integral enhance yield and quality of mass-produced nasturtium (Tropaeolum majus L.) in plant factories. *Plants, 10*(6), 1203.

Yuan, M., Jia, X., Ding, C., Zeng, H., Du, L., Yuan, S., Zhang, Z., Wu, Q., Hu, C., & Liu, J. (2015). Effect of fluorescence light on phenolic compounds and antioxidant activities of soybeans (Glycine max L. Merrill) during germination. *Food Science and Biotechnology, 24*(5), 1859–1865.

Chapter 4
An IoT-Based Precision Irrigation System to Optimize Plant Water Requirements for Indoor and Outdoor Farming Systems

Mohammad Hussain Seyar, P. D. Kahandage, and Tofael Ahamed

Abstract Irrigated agriculture plays a crucial role in fulfilling global food demands since it is reliable in some parts of the world, particularly in arid and semiarid regions, for meeting plant water requirements. Considering the increasing water shortage due to the effects of climate change, water management in agriculture is the key to securing water for water consumers, including agriculture, municipal, industrial, and daily utilizations. Thus, efficient and robust irrigation management and controlling technologies should be adopted. Integrating precision irrigation into drip and sprinkler irrigation systems in greenhouses and open fields may represent a promising approach for water saving systems. The irrigation operation is carried out based on soil, plant, and atmospheric monitoring. The advances in technology have provided growers with better opportunities for irrigation management and control. Therefore, the aim of this chapter is to explore a precision water management system that optimizes plant water requirements through sensing and perceptions from indoor to outdoor environments. In the case of the outdoor environment, multiple factors, such as the field capacity and evapotranspiration, are considered while noting the best practices for saving water. In this regard, a concise review is presented to determine the scopes of IoT-based systems for precision application. Through this review, we have found that in implementing IoT-based irrigation systems, better water use efficiency can be achieved. Furthermore, if these systems are incorporated with computer algorithms such as neural networks, fuzzy logic systems and hybrid systems, water loss can be minimized, while the production level has the opportunity

M. H. Seyar
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Soil Science and Irrigation, Kabul University, Kabul, Afghanistan

P. D. Kahandage
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering and Soil Science, Faculty of Agriculture, Rajarata University of Sri Lanka, Mihintale, Sri Lanka

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

for a significant increase in the production of crops and vegetables. Thus, precision irrigation systems are outlined with a number of networks and integrated with intelligent IoT-based drip and sprinkler irrigation systems to save water and promote the implementation of reuse systems in irrigation projects.

Keywords Precision irrigation · IoT · Irrigation scheduling · Water shortage · Indoor and outdoor cultivation systems · Sensors · Artificial intelligence

4.1 Introduction

Considering the rapid population growth, which is projected to reach ten billion by 2050 (de Pascale et al., 2019), and water scarcity, an immediate requirement for producers to develop innovative and efficient methods of production to feed the increasing global population while using the limited available water has been noted. The problem of water scarcity, which is a global concern and has the chance to increase further due to climate change and increasing global food demands, requires serious attention over the coming years. In addressing this problem, water consumers should be identified. Generally, four categories of water consumption have been noted, including agricultural, industrial, municipality, and daily utilization. Of these, agriculture constitutes a large proportion of water use and accounts for over 70% of global freshwater usage. This trend is expected to increase due to the effects of climate change. Furthermore, less than 20% of the land worldwide is allocated to irrigated agriculture (Adeyemi et al., 2017), but it provides 40% of the global food supply, which highlights the importance of irrigation in global food security. Given the increasing water shortage, enhancing water use efficiency in agricultural systems such as irrigation is the key in securing water for agricultural, industrial, municipal, and daily utilization purposes (Buttaro et al., 2015). Therefore, practicing precision irrigation might not only help to save a significant amount of water but also increase productivity.

In the case of meeting global food demands, the implementation of precision irrigation systems in the protected environment where the condition is fully controlled and plants will be provided with ideal conditions of growth is essential. Protected agriculture has become popular over the last 30 years (Elaydi, 2017) and is rapidly increasing worldwide (Hemming et al., 2019). In this type of agriculture, almost all parameters are controlled, and producers can provide their plants the optimal environment throughout the growing season. In addition to the advantages of serving as a promising approach in terms of both quality production and addressing water shortages, it may also protect plants from unfavorable conditions such as strong wind, pests, intense rainfall, bird damage, and low temperature. Aside from other parameters, irrigation management is vital for successful production. Over-or under-irrigation is usually associated with adverse effects. Therefore, applying an optimal amount of irrigation is essential for reasonable production. Protected cultivation has been practiced in many forms, including greenhouses, indoor farms and under screens or screenhouses, which can save a significant proportion of water,

up to 37% in screenhouses (Pirkner et al., 2014) and 20–40% in greenhouses (Nikolaou et al., 2019). Mindful of the water shortage, it is a reliable system for the intensive production of vegetables, fruits, and flowers. Notably, the efficiency of protected agriculture in terms of water savings mainly depends on the type of irrigation systems used. For example, employing microirrigation systems such as drip and sprinkler irrigation, which are associated with high water application efficiency and low water loss, usually result in better water use efficiency. These systems are mostly practiced in greenhouses due to good water productivity. Incorporation of automation into these systems adds further benefits to the production system by which irrigators achieve full control over the greenhouse plants' growth factors, such as irrigation. In precision irrigation systems, the proper amount of water is managed and applied at the correct time, which leads to better water productivity and environmental sustainability.

Precision irrigation management has been carried out using Internet of Things (IoT) (Kodali et al., 2017), machine learning (Sidhu et al., 2020), a combination of IoT and machine learning (Goap et al., 2018), digital solutions (Mohamed et al., 2021), and remote sensing (Belaqziz et al., 2013). IoT-based precision irrigation is usually practiced by employing sensors due to its low cost, accuracy and overall profitability (Van Iersel et al., 2013). Precision irrigation combined with IoT technology further facilitates real-time monitoring, processing, and applies irrigation on demand. The IoT has been widely used in different sectors, including agriculture (irrigation), where sensors are deployed in the field to sense the data from the field and send it to the microcontroller. Two types of technology are commonly used to monitor the water status in crops, soil, and the environment: wireless and wired technologies. Each is deployed under specific conditions. Recently, wireless technology utilization has increased due to the ease of installation, economic considerations, and reliable performance. Wireless-based technology has been employed in different areas of agriculture, including precision agriculture, horticulture, cattle, and crop monitoring, as well as environmental monitoring, which are the main classes of agricultural IoT and wireless technology-based applications (Khriji et al., 2021).

IoT-based precision irrigation systems have been investigated and reviewed in several articles. However, a comprehensive review of IoT-based precision irrigation along with artificial intelligent systems in both indoor and outdoor cultivation environments has not been conducted. Therefore, we have performed an extensive review of the recent developments in the application of IoT- and intelligent-based irrigation, monitoring, and controlling. The main objective of this article is to provide a comprehensive literature review on different precision irrigation management practices in both greenhouse and open field cultivation systems along with their advantages and drawbacks.

4.2 Precision Irrigation Management

The irrigation operation is carried out to meet crop water requirements while substantially increasing crop yields. In other words, the overall goal of irrigation is to maintain the soil water balanced between irrigation/rainfall and evapotranspiration (Fig. 4.6). In irrigation management, two questions should be answered: when should irrigation be applied and how much water should be applied. However, answering these questions is not an easy task and requires careful consideration because several factors exert direct and indirect effects on plant water requirements and they vary based on plant type, environment, and type of soil. Furthermore, over-irrigation may lead to water loss, fungal diseases, CO_2 emissions, N_2O emissions, soil nutrient leaching, soil salinization, energy wastage, and low crop yields, while under-irrigation will result in water stress to plants and low crop yields.

Precision irrigation is defined as the precise management of applying the appropriate amount of water at the correct time and minimizing adverse environmental impacts. This irrigation system, in addition to enhancing water use efficiency, also improves both crop quality and quantity and reduces energy costs and adverse environmental impacts (Swarup et al., 2013). The time or frequency and volume of water to be applied depend on the types of plants and soil, environmental conditions, water availability and economic considerations. For instance, in sandy soil, frequent irrigation is usually applied, while in clayey soil, the irrigation interval might be longer. Two methods for measuring how much water the plants need or how much water to apply have been developed: direct and indirect methods. The direct method measures the plant and soil water contents, while the indirect method measures environmental parameters, such as the light intensity, air temperature, relative humidity, wind speed, and rainfall.

With the advances in technology and the emergence of new and cheap sensors, sensors are currently quite easy to employ to monitor the moisture content in the soil, plants, and atmosphere by implementing precision irrigation systems. Precision irrigation systems enable irrigators to obtain real-time field-monitoring data and perform irrigation on demand.

4.2.1 IoT Technologies for Precision Irrigation Systems

Different technologies have been developed in recent years either in terms of communication or controlling systems, each of which has been employed in various sectors, including agriculture and precision irrigation. However, these technologies have their own specifications based on the data transmission rate, frequency band, power consumption, security considerations, and bandwidth. This section introduces the recent IoT-based advanced technologies that have potential for widespread utilization.

4.2.1.1   IoT Networking Backbone

The Internet of Things has been used in numerous fields, including agriculture and irrigation. Generally, IoT-based irrigation systems consist of five layers: sensor node, network, gateway, cloud, and application layers (Fig. 4.1). The first layer is the sensor node, which consists of sensors and actuators. The sensors monitor the water status in the soil and plant and the evapotranspiration rate, while actuators turn the solenoid valve/water pump on/off. The second layer is called the communication or network layer, where various networks, such as radio-frequency identification (RFID), ultrawideband (UWB), Bluetooth, Wi-Fi, ZigBee, Ethernet, 2G/3G/4G/5G, LoRaWAN, Sigfox, and NB-IoT, are employed to connect fields with cloud and application layers. The application of each network differs based on the range of communication, data transmission rate, security consideration, power consumption, and frequency of communication. For instance, for short-distance purposes, RFID, UWB, and Bluetooth are used; for medium-distance purposes, Wi-Fi and ZigBee are used; and for long-distance purposes, 2G/3G/4G/5G/LoRaWAN and NB-IoT have been employed. Wireless technology plays a pivotal role in IoT-based systems. The use of low-cost wireless sensors enhances the efficiency of precision irrigation systems. Wireless sensor networks have been employed successfully and have achieved an 80% cost reduction compared to hardwired approaches (Khriji et al., 2021). Thus, the use of the aforementioned networks based on our purpose makes monitoring, analysis, and control much more robust and efficient.

Another layer is the gateway layer, which functions as a bridge to connect the sensor node with the application layer. The gateway layer helps form and makes the long range and wide area network possible. Cloud servers store the data for a long time, where the irrigators can access the collected data from any part of the world using a smartphone or computer if they have an internet connection. The final layer is the application layer, which runs the overall irrigation operations. In fact, it is the main layer of the IoT-based irrigation system that performs the analysis of collected data from the field and implements actions accordingly.
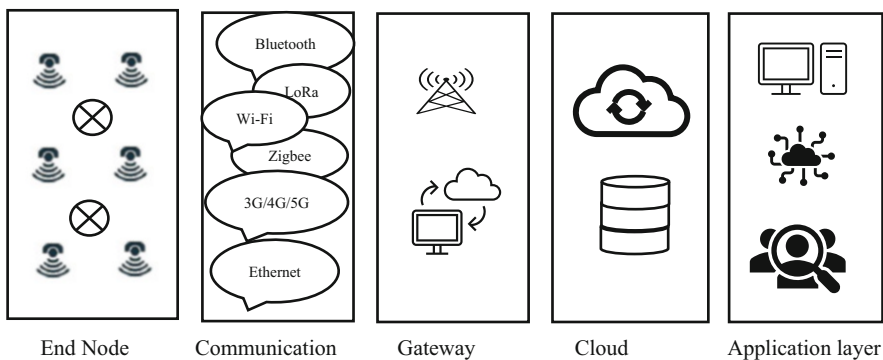
Fig. 4.1 Scheme of five layers of IoT-based precision irrigation

4.2.1.2 IoT: Irrigation Control System

Irrigation systems are controlled using either conventional methods or intelligent systems. In the conventional approach, the majority of the work is performed by humans, while in intelligent systems, most of the work is performed by machines and robots. Over time, conventional systems have been replaced by automatic and intelligent technologies. In some parts of the world, timer-based irrigation systems are practiced, which facilitate watering tasks. In developed countries, mostly intelligent irrigation control systems are employed due to the ease of control and precise water application. Adopting advanced irrigation control systems helps growers apply a proper amount of water at the right time, enhance water use efficiency, increase yield, reduce environmental impacts, and save labor (Owen et al., 2016).

The development of computer science has led to extensive application of programming in training and predicting, as well as controlling various machines. By training the machine and applying algorithms, the machine characterizes human thinking and executes specific actions. The machine has the potential to analyze complex problems and perform several tasks simultaneously, making the machine smarter. We have focused more on the fuzzy logic, artificial neural network, and hybrid systems that are widely implemented and recommended by researchers.

4.2.1.2.1 Fuzzy Logic-Based Control System

The fuzzy logic system is the traditional Boolean logic extension that has the potential to express logical values between false and true and explain problems in the world that are nonlinear and uncertain (Croock Al-Janabi et al., 2018). In irrigation operations, several factors are usually involved, which make the system complex and require a comprehensive model to consider many factors at the same time. In this case, the application of a mathematical model is impractical, and employing an advanced model such as a fuzzy logic-based model provides a substantial benefit. Fuzzy logic-based systems run based on inputs and produce one or several outputs (Krishnan et al., 2020). This system has been proven to be effective in precision irrigation management due to the high accuracy of the water requirement calculation, resulting in increased water use efficiency. However, this accuracy mainly depends on the individual's knowledge and skills.

A fuzzy logic system is practiced both in indoors and outdoors. Pacco (2022) used a fuzzy logic-based system to simulate temperature and predict irrigation timing in the greenhouse. In another study, Azaza et al. (2016) investigated the performance of a fuzzy logic system in controlling the microclimate inside a greenhouse and found that a system integrated with a wireless data monitoring platform resulted in significant energy and water savings. Similarly, Selmani et al. (2019) developed a solar-powered irrigation system based on fuzzy logic in a greenhouse, where the model consisted of two devices. The first device used environmental parameters to calculate the watering duration, while the second device used the actual resource

level to examine the pump potential. The authors concluded that the use of this system was effective in scheduling watering.

4.2.1.2.2 Artificial Neural Network-Based Control System

An artificial neural network is an algorithm that processes information similar to how information is processed by the human brain (Abioye et al., 2020). It has been applied in controlling precision irrigation because of its learning potential and adaptability based on different conditions. Generally, it consists of three or more layers (Fig. 4.1). As shown in Fig. 4.1, the first layer is called the input layer, where several parameters serve as inputs and then undergo a long process; ultimately, a single output or several outputs will be achieved. In the case of precision irrigation, the typical inputs include temperature, relative humidity, light intensity, rainfall, soil moisture content (Gu et al., 2021), wind speed, and soil physical properties such as texture and the crop coefficient. Artificial neural networks have been investigated in several studies and have been suggested to be a reliable tool for precision irrigation control. Trajkovic et al. (2022) investigated the radial basis function network for forecasting reference evapotranspiration, where they used environmental parameters such as temperature, sunshine, wind speed, and relative humidity as inputs and found that the neural network is useful as a robust tool for forecasting evapotranspiration. Similarly, Trajkovic (2005) studied temperature as a single input in the radial basis function network and compared the results with other standard methods used worldwide, including reduced Penman–Monteith, Hargreaves, and Thronthwaite methods. They also concluded that the neural network predicts evapotranspiration accurately. The benefit of employing neural networks for predicting the water requirements of plants is that they enable growers to predict the amount of water needed for the plants for the upcoming hours, days or even weeks. The system may be further improved by combining it with other intelligent models or systems. The typical input and output data of AI-based control system is presented in Fig. 4.2.

4.2.1.2.3 Hybrid Control System

In hybrid systems, two or more machine learning algorithms are usually involved. For example, in an intelligent system, the combination of an artificial neural network along with a fuzzy logic system or other algorithms is considered a hybrid system. In this system, IoT is used for collecting information and monitoring the field, while artificial intelligence is used for comprehensive and precise control. Several studies have highlighted the importance of hybrid systems compared to single control systems. The hybrid system has been investigated in reservoir operation (Deka & Chandramouli, 2009), pan daily evaporation modeling (Kişi, 2006), soil moisture prediction (Tseng et al., 2018), and other irrigation planning and operation procedures. Moghaddamnia et al. (2009) studied the estimation of evapotranspiration using an artificial neural network and adaptive fuzzy logic system and found that this
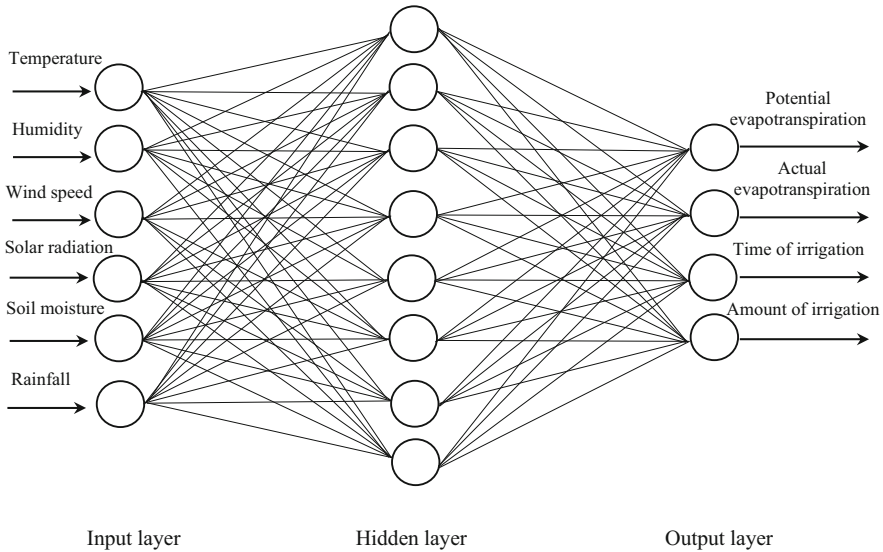
Fig. 4.2 Schematic representation of artificial neural network

system produced much better results than empirical formulas. Similarly, Falamarzi et al. (2014) estimated evapotranspiration using temperature and wind speed data as inputs using an artificial neural network and wavelength neural network, and the results showed that both algorithms predict evapotranspiration at an acceptable accuracy level. This system is employed both in indoor and outdoor cultivation systems. Wang et al. (2017) estimated the leaf area index inside a greenhouse with a combined backpropagation neural network, genetic algorithm and autoregressive exogenous model, and concluded that the estimation of transpiration under indoor conditions provides better results than those under outdoor conditions.

4.2.2 Indoor Precision Watering Management

Indoor agriculture has been widely practiced due to the advantages of off-season production and water use efficiency, as well as the possibility of providing plants with ideal growth conditions. These advantages are because microclimate parameters, including temperature, humidity, light intensity, carbon dioxide, and soil moisture contents, which exert tremendous effects on plant growth and yields, can be controlled under indoor conditions (de Pascale et al., 2019), making indoor farming very important. In indoor farming, high-value crops such as vegetables are usually produced since they will exhibit maximum yields and outcomes in these environments.

Proper watering management is one of the key factors for optimal production and increasing water use efficiency since these environments have no rainfall, and irrigation is the only water source. Excessive application of water results in NO_3 leaching (Incrocci et al., 2020), while underwatering leads to decreased yields. Considering the shortage of water, growers should employ approaches where water loss is minimized while the production level is maximized. However, growers usually apply water based on their past experiences. To this end, the adoption of science-based watering scheduling methods that increase water productivity is pivotal. Furthermore, employing these methods enables growers to obtain real-time data on the moisture content and plant water status and perform the watering operations accordingly. The timing/frequency and quantity of water to be applied are determined through the watering scheduling process. This process affects plant water productivity, plant yields, and quantity. Over the last few years, different methods for watering scheduling have been developed and practiced around the world. Of which, watering based on the soil moisture content and plant water status has been widely practiced and highlighted in several studies. The integration of IoT with the aforementioned methods further facilitates the watering operation in which monitoring the soil moisture content or plant water status and applying water are conducted remotely. In this type of system, the water application and moisture content are monitored and controlled using sensors.

4.2.2.1 Soil Moisture-Based Scheduling

In this method, the soil moisture content or soil matric potential are monitored. The properties of soil are monitored using sensors. Sensors that measure the volumetric water content or dielectric properties of soil are widely employed because of their low cost and ease of operation.

The advent of new and inexpensive sensors further facilitated the monitoring of the soil moisture content and deployment of more sensors in agriculture. The soil moisture sensor senses the volumetric water content in the soil, while the other type is called the matric potential sensor, which senses the matric potential of the soil. Both types of sensors are used worldwide (Van Iersel et al., 2013). The volumetric soil moisture sensors measure the water content based on volume, while the soil matric potential sensors measure how easily the water is extracted by the plant. Volumetric soil moisture sensors have the potential to provide growers with information on how much water and when to apply it to plants. The soil matric potential sensors measure the availability of water in the soil but do not provide information on how much water should be applied; they are also reliable sensors (Thompson et al., 2007).

The soil matric potential may differ for different crops and in various regions due to soil texture and crop characteristics. For instance, in clay soil, matric potentials of -30 and -40 kPa were used as thresholds and compared with the -10 kPa threshold, resulting in 35% and 45% water savings, respectively, especially in tomatoes and cucumbers (Nikolaou et al., 2019). IoT-based systems facilitate

watering operations since most of the monitoring and control is performed using soil moisture sensors.

These sensors usually have high accuracy, and the reading error ranges from $\pm 1\%$ to $\pm 5\%$. Some researchers suggest that depending on sensors alone is not a wise method for watering scheduling. However, the sensor accuracy can be increased by performing in situ calibration and employing outlier removal techniques such as MZscor, GESD, Zscor, and Chauvenet. Using real soil moisture data, 90.4% water savings have been achieved, while 89.3% water savings have been achieved by applying MZscore and Zscore to the soil moisture data (Campos et al., 2020).

The application of an IoT-based system additionally enhances agricultural outcomes. Lichtenberg et al. (2013) investigated wireless sensors in greenhouse and nursery environments, where the time of production and crop loss decreased by half or more. As shown in Fig. 4.3, different wired or wireless sensors are deployed in the greenhouse, where they sense the moisture content in the soil or monitor the plant water status and send the information to the microcontroller, which is then forwarded to the cloud server (Fig. 4.4). For carrying the watering operation, a threshold should be defined, in which it triggers the system when to start watering and when to stop watering. Technologies such as LoRa, NB-IoT, Bluetooth, ZigBee, and Wi-Fi communication systems are employed in indoor cultivation. The general flowchart of this method is presented in Fig. 4.5.

4.2.2.2 Plant Water Status-Based Scheduling

Monitoring the plant water status is another key method of applying water in greenhouses or protected agriculture. Plant physiology or traits such as the stem diameter, amount of transpiration, leaf water potential, leaf thickness, canopy temperature, and crop reflectance have been widely used as indices for watering scheduling. Among these traits, canopy temperature has been researched and implemented mostly indoors and outdoors due to its low cost and easy installation. Various sensors are deployed to monitor each of these properties. For example, sap flow sensors for measuring crop transpiration, ZIM probes for measuring the turgidity of leaves, and infrared temperature sensors for monitoring canopy temperature have been used. In the case of canopy temperature measurements, the difference between the climate inside the canopy of the plant and the atmosphere is usually considered. Plants transpire water from stomata in the leaves, and this process influences the canopy temperature. Whenever plants do not experience water stress, the temperature inside the canopy will be lower, while the stomata will close under conditions in which plants experience water stress, and the temperature inside the canopy may increase. Generally, the plant water status-based method does not directly provide information on plant water requirements; therefore, it should be used in combination with other methods, such as the soil moisture-based method. Prenger et al. (2005) developed an irrigation system using plant canopy temperature as a method for turning the irrigation system on or off, while an ET model was developed for the purpose of determining how much water to apply. The IoT-based
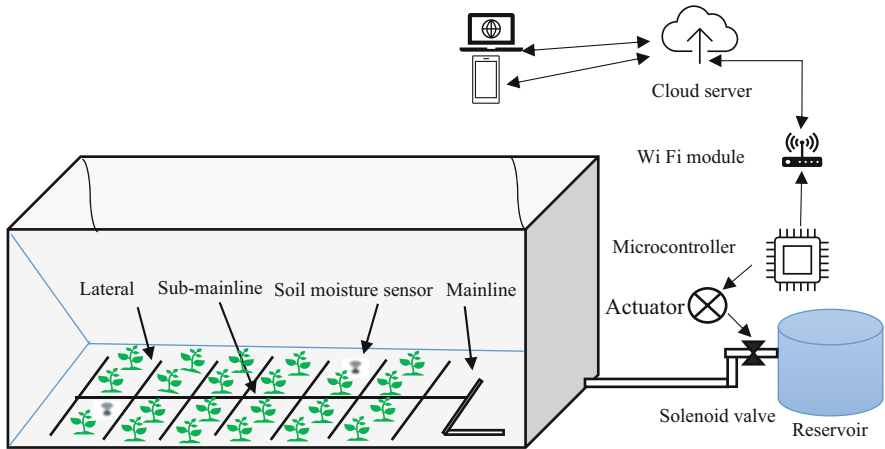
Fig. 4.3 Layout of a typical greenhouse IoT-based irrigation system
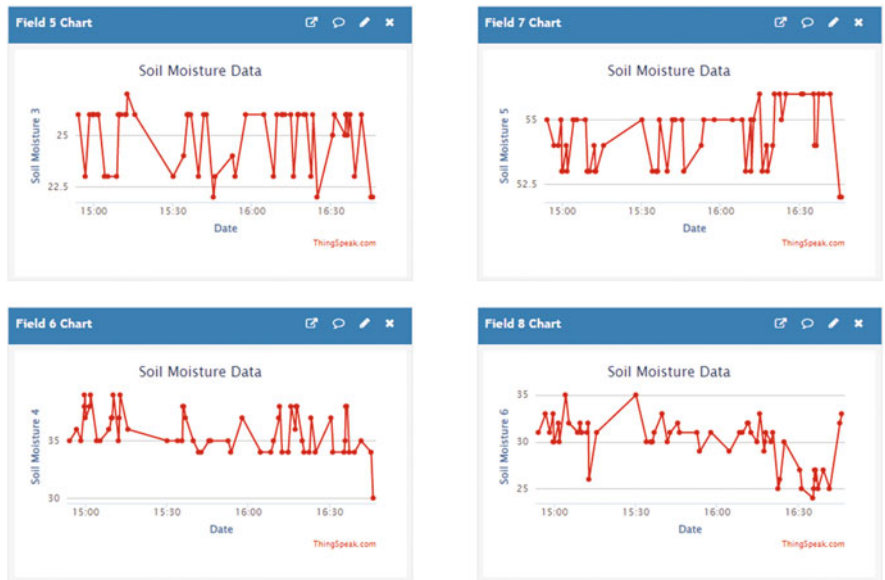
Fig. 4.4 Scheme of real-time data handling in the cloud server

or automation of this method has proven to be more efficient in terms of water savings. Seelig et al. (2012) used leaf thickness as a watering scheduling tool and saved 25–45% of water with an automated irrigation system.
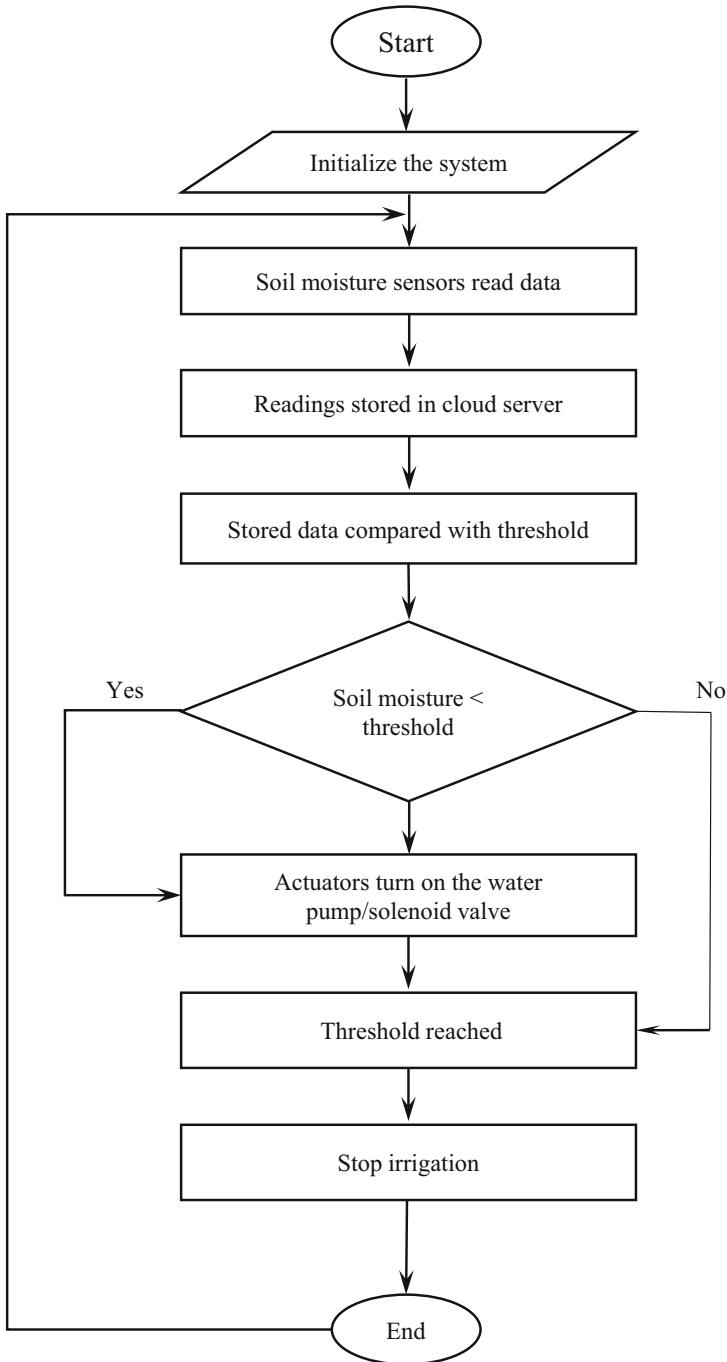
Fig. 4.5 IoT-based watering management flowchart
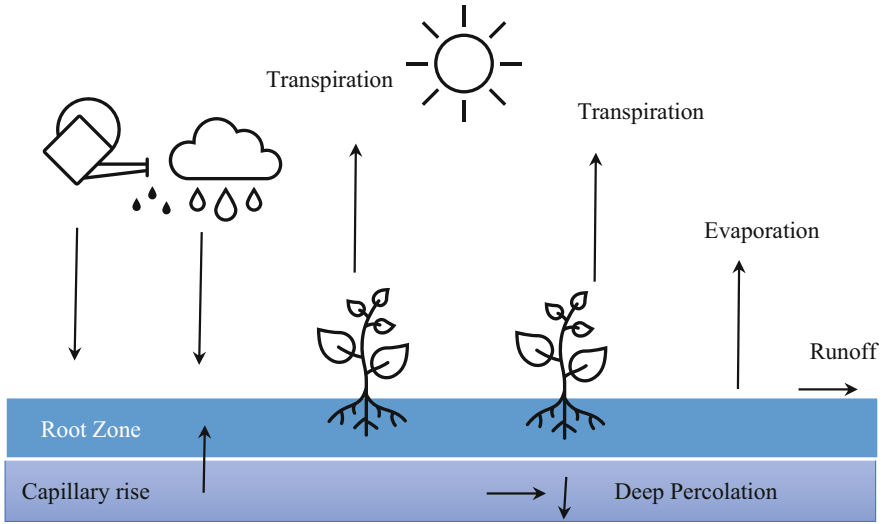
Fig. 4.6 Water cycle in the atmosphere

4.2.3   Outdoor Precision Irrigation Management

In outdoor environments, the source of water may be irrigation/rainfall or capillary rise, while in the greenhouse, irrigation is the main water source for meeting crop water requirements since there is no rainfall. Water is added to the soil through irrigation, rainfall, or capillary rise, which originates from groundwater, especially in areas where the water table is not deep (Fig. 4.6).

Some proportion of the added water will evaporate from the soil surface or transpire from the plant surface, while the remaining proportion will infiltrate into the soil, among which parts of the infiltrated water join the ground water through a deep percolation process and the remaining water will move into the horizontal direction (percolation) to be either absorbed by plant roots or retained in the soil pores. Runoff is also common in some irrigation schemes, where the water is lost from the soil surface in the form of running water and leaches soil nutrients, creating small ditches. Irrigation management, especially in arid and semiarid regions where rainfall does not replenish plant water requirement, is the key to agricultural production. Therefore, precise management of irrigation in agriculture by employing low-cost technologies by which water is managed as effectively as possible is essential for food security and meeting global food demands. Various approaches for saving water have been employed, including mulches, deficit irrigation and drip and sprinkler irrigation systems. However, these methods have been recognized to require extensive labor, a large volume of water and a significant amount of economic input. IoT-based irrigation systems have been used with minimum labor and water requirements to overcome the aforementioned problems.

4.2.3.1 IoT-Based Irrigation Scheduling

Irrigation scheduling is the approach by which irrigation managers understand how much and when to apply water to help save a significant proportion of water. Three methods for irrigation scheduling have been developed, plant water status-based, soil moisture-based, and weather-based, each of which can be employed at specific times and under specific conditions (Fig. 4.7).

The use of each method varies based on the climate condition, type of plant, data availability, skills of the researcher, accuracy, economic considerations, and applicability of the approaches. However, the combination of all these methods is usually recommended to carry out irrigation operations. In the IoT-based approach, sensors are deployed to monitor the water content of the soil, plants, and amount of water lost into atmosphere.

4.2.3.1.1 Soil Moisture-Based Irrigation Scheduling

In this method, irrigation is applied by monitoring the moisture content or water potential in the soil. As illustrated in Fig. 4.8, plants may undergo three stages of water conditions in the soil: saturation, field capacity, and permanent wilting point states. Under saturated conditions, plants wilt due to excess water that they cannot use, while under field capacity conditions, an ideal amount of moisture is present in the soil, and all of the moisture is available for plant consumption. At the permanent wilting point level, plants are no longer able to extract water from the soil, and they may subsequently wilt permanently.
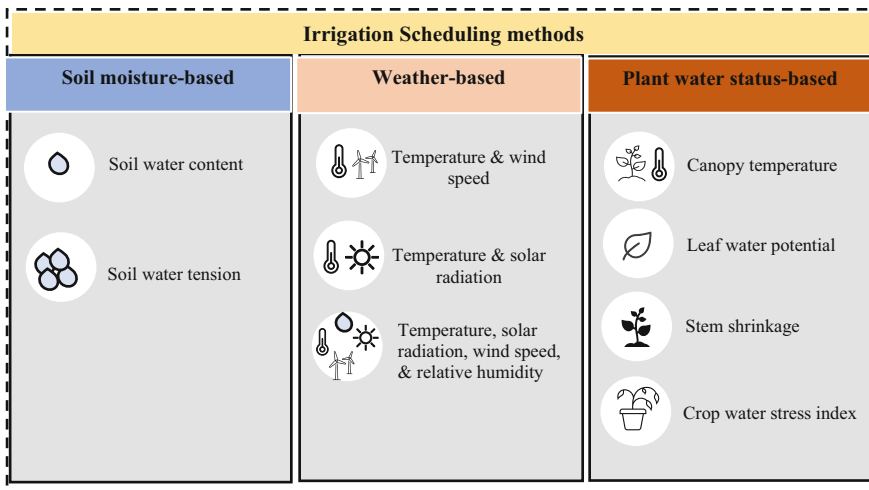
Fig. 4.7 IoT-based irrigation scheduling methods
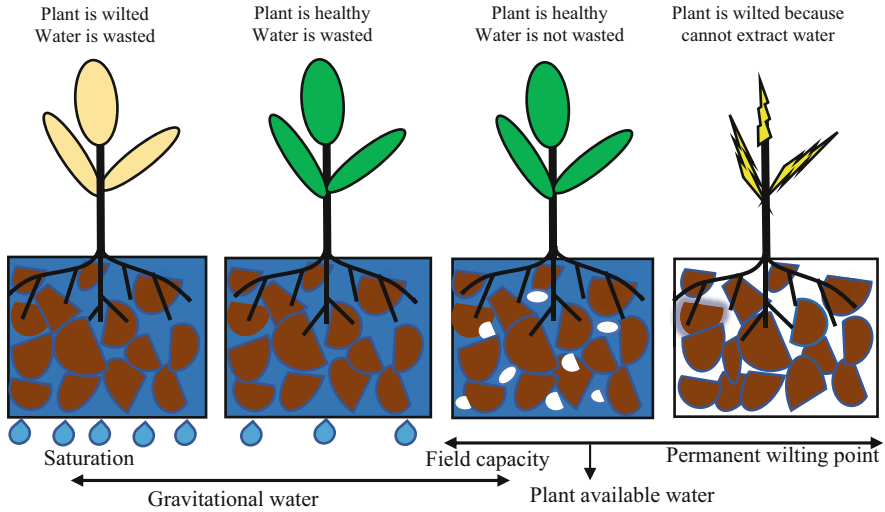
Fig. 4.8 Water availability and soil condition

The application of sensor technologies in irrigation paved the way for better water management in agriculture (Domínguez-Niño et al., 2020). This system follows a series of steps. For the first step, a threshold should be defined where the irrigation operation will be initiated by comparing data collected from the field with the predefined threshold. This is usually performed using the water balance equation by which irrigation deficit; which is the proportion of water which takes the soil back to field capacity, can be calculated.

$$\text{SWD}_t = \text{SWD}_{t-1} + \text{ET} - I_e - P_e \qquad (4.1)$$

where SWD_t is the soil water deficit for the day t, SWD_{t-1} is the soil water deficit for the previous day of t, ET is the evapotranspiration, I_e is the effective irrigation, and P_e is the effective rainfall.

After calculating the soil water deficit (SWD) for a given soil under specific period, then, critical soil water deficit (SWD_c) should be determined which is the threshold.

$$\text{SWD}_c = Z_r \times \text{PAW} \times \text{AD} \qquad (4.2)$$

where SWD_c is the critical water deficit, Z_r is the effective root depth, PAW is the plant available water, and AD is the allowable depletion. Several tools and devices for measuring the water content in the soil, such as gypsum blocks and granular matrix blocks, tensiometers, neutron probes and sensors. Soil moisture sensors are widely used due to their low cost and user friendliness. Soil capacitive moisture sensors measure the volumetric water content in the soil, while soil matric potential sensors measure how easy the water can be extracted by the plant. Research has
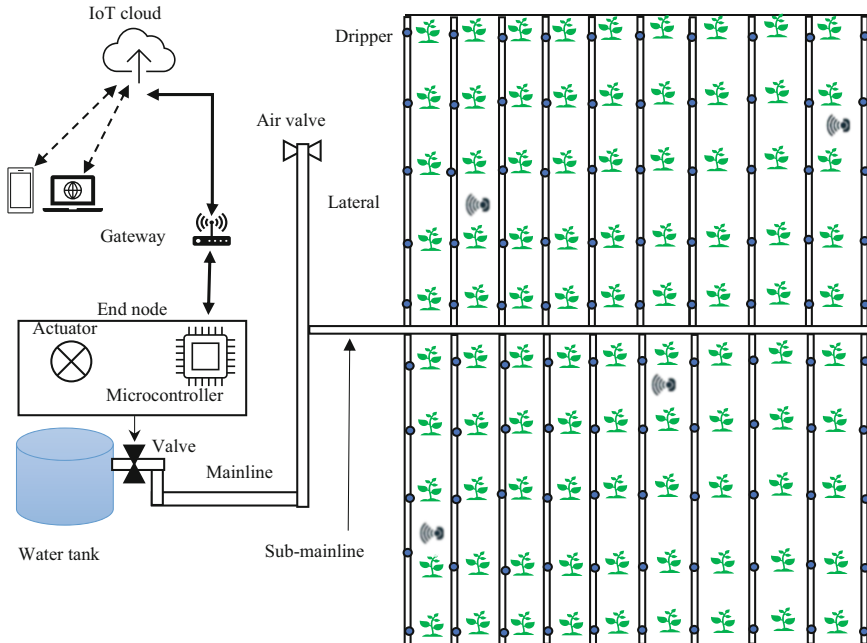
Fig. 4.9 Schematic representation of typical IoT-based irrigation layout

indicated that irrigation scheduling based on soil matric potential in clay soil where tomato and cucumber are being cultivated may save 35–46% of water (Nikolaou et al., 2019). In another study, an automatic irrigation system was developed based on soil moisture and plant temperature sensors and saved 90% of water compared to conventional irrigation systems (Gutiérrez et al., 2013). The use of the plant water stress index and soil water methods may produce similar results both in terms of water savings and yield (Cesari de Maria et al., 2017).

Buttaro et al. (2015) studied soil water potential as an irrigation threshold using a tensiometer, which resulted in 35–46% water savings. In China, Liao et al. (2021) developed an automatic irrigation system using soil moisture sensor for tomato plants inside the greenhouse and achieved an irrigation water use efficiency of 41.23 kg/m^3 compared to 31.58 kg/m^3 for the conventional irrigation system.

As illustrated in Fig. 4.9 sensors are deployed at different locations in the field where they measure the moisture content in the soil and send the data to the microcontroller. The microcontroller subsequently turns the water pump on/off based on a predefined threshold. Recently, a more common approach has been to send data to cloud servers that store them and allow anyone to access the data later from any location with an internet connection. Different methods are used to collect and send data to cloud servers, and the most widely used networks and methods are presented in Fig. 4.1.

4.2.3.1.2   Weather-Based Irrigation Scheduling

Weather-based methods have also been employed by many irrigation managers worldwide. They usually require environmental parameters, including relative air humidity, temperature, solar radiation, wind speed and rainfall, which exert direct and indirect effects on the water depletion rate from the plant–soil–atmosphere system. These data are collected by manual/automatic weather stations or metrological networks in which data are uploaded in a timely fashioned to cloud servers (Fig. 4.10).

   The use of these data helps growers determine evapotranspiration, which is the amount of water lost from soil and water, as well as plant surfaces, which is called reference or potential evapotranspiration. In other words, the approach is associated with a mathematical calculation to obtain the potential evapotranspiration of the reference plant or grass, which is then multiplied by the specific-crop coefficient to achieve crop evapotranspiration. The crop coefficient differs in terms of plant growth stages.

   Although the use of all environmental parameters to accurately achieve potential evapotranspiration is generally recommended, some researchers use a few environmental parameters to calculate reference evapotranspiration. Hargreaves et al. (1985) utilized only temperature and extraterrestrial radiation to calculate potential evapotranspiration (ET_o), while Jones and Richard (1990) used temperature and solar radiation to determine ET_o (Eqs. 4.2 and 4.3). The use of limited data will further
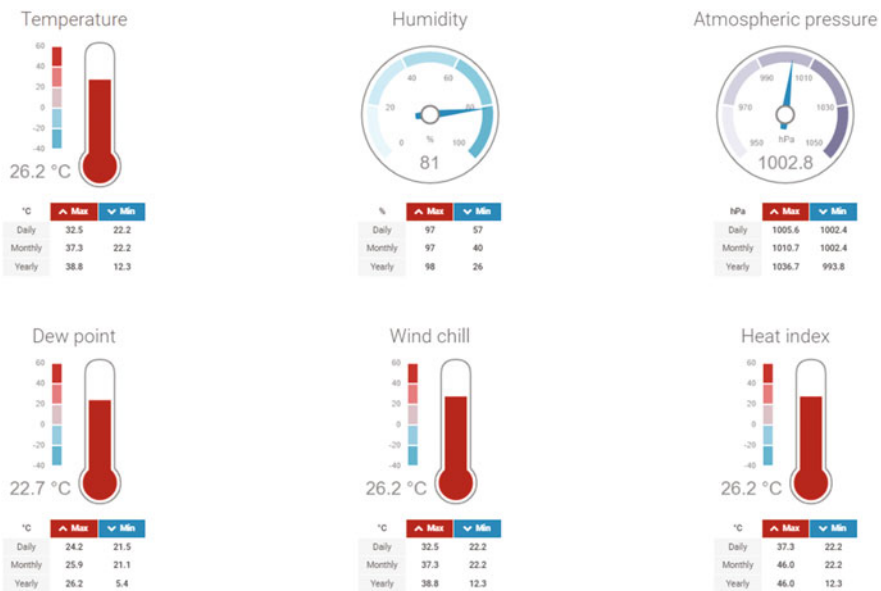
Fig. 4.10   Real-time weather data displayed in the weather cloud server

facilitate the automatic and real-time monitoring procedure for determining evapotranspiration.

$$ET_o = 0.0023R_a \left(\frac{T_{max} + T_{min}}{2} + 17.8\right)\sqrt{T_{max} - T_{min}} \qquad (4.3)$$

where ET_o is the reference evapotranspiration (mm/day), T_{max} and T_{min} are the maximum and minimum temperatures (°C), respectively, and R_a is the extraterrestrial radiation (MJ m^{-2} day^{-1}).

$$ET_o = \alpha_1 \left[3.87 \times 10^{-3} R_s (0.6 T_{max} + 0.4 T_{min} + 29)\right] \qquad (4.4)$$

where ET_o is the reference evapotranspiration, R_s is the solar radiation (MJ m^{-2}), T_{max} is the maximum temperature, and T_{min} is the minimum temperature. The most comprehensive method for calculating evapotranspiration is the FAO-Penman-Monteith method, where environmental data are used as inputs:

$$ET_o = \frac{0.408\Delta(R_n - G) + \gamma\frac{900}{T+273}u_2(e_s - e_a)}{\Delta + \gamma(1 + 0.34u_2)} \qquad (4.5)$$

where ET_o is the reference evapotranspiration (mm day^{-1}), R_n is the net radiation at the crop surface (MJ m^{-2} day^{-1}), G is the soil heat flux density (MJ m^{-2} day^{-1}), T is the mean daily air temperature at a 1.5–2 m height (°C), u_2 is the wind speed at 1.5–2 m height (m s^{-1}), e_s is the saturation vapor pressure (kPa), e_a is the actual vapor pressure (kPa), $e_s - e_a$ is the saturation vapor pressure deficit (kPa), Δ is the slope vapor pressure curve (kPa °C^{-1}), and γ is the psychrometric constant (kPa °C^{-1}). After the calculation of reference evapotranspiration, the crop-specific coefficient required to obtain the crop evapotranspiration is determined using the following equation:

$$ET_c = ET_o \times KC \qquad (4.6)$$

where ET_c is the crop evapotranspiration, ET_o is the reference evapotranspiration, and KC is the crop coefficient factor. The crop coefficient factor varies in terms of the types of plants and stages of growth. Researchers often employ irrigation experiments based on evapotranspiration, especially in outdoors. This work is conducted for several reasons, such as greater reliability, feasibility, and ease of application in advanced and automated research approaches. Goap et al. (2018) developed an IoT-based irrigation system using the soil moisture content, air temperature, relative humidity, precipitation, and UV to frame an algorithm through which irrigation operations are run. The calculation of evapotranspiration is also performed using the ET_o calculator or CROPWAT software, which were developed by FAO based on the FAO 65 Penman–Monteith equation. Furthermore, integration with a microcontroller through programming can help to calculate ET_o automatically.

4.2.3.1.3 Plant-Based Irrigation Scheduling

In plant-based methods, irrigation managers use measurable physiological plant traits such as canopy temperature, leaf water potential, sap flow, and stem shrinkage to monitor the plant water status and the time of irrigation. Steppe et al. (2008) successfully integrated the plant stem and diameter into a mathematical model to continuously predict apple tree water requirements. The crop water stress index has also been used by researchers to estimate plant water requirements by studying canopy-temperature and vapor pressure deficit relationships and has produced satisfactory results (Steppe et al., 2008). O'Shaughnessy et al. (2017) investigated the combined applicability of the plant water stress index and neutron probe readings to provide solid feedback for irrigation scheduling, especially in arid regions. This method is sometimes not feasible for monitoring plant symptoms, where in some cases, it may be too late to recover the plant from water stress. In plant-based methods, devices usually monitor the symptoms designated by the plants. Sometimes, it takes a long time until the plant exhibits any symptoms.

As shown in the study by Cremona et al. (2004), irrigation scheduling based on the water stress index is quite similar to irrigation scheduling based on the water content both in terms of yield and water use efficiency. In Spain, the application of automatic irrigation systems using plant transpiration to irrigate tomato inside the greenhouse reduced water consumption by 20% (Pawlowski et al., 2017). Irrigation scheduling for monitoring plants in the greenhouse and in the open field (Osroosh et al., 2015) has produced good results. De la Rosa et al. (2015) investigated plant water stress as an irrigation base for scheduling with the intention of increasing water use efficiency and resulted in 17%, 15%, and 37% water savings in 3 successive years (2009, 2010, and 2011), respectively. Some researchers used a constant value as the irrigation threshold, while others used an adaptive irrigation threshold based on the theoretical crop water stress index (Osroosh et al., 2015). Gu et al. (2017) developed a new crop water stress model based on the RZWQM2 scenario, which saved 30.5%, 17.3%, and 7.1% in low-frequency irrigation, whereas in high-frequency irrigation, 35%, 30%, and 16% of water was saved in successive years.

4.3 Discussion

Through an extensive review of outdoor and indoor monitoring, control and irrigation systems in recent years, IoT- and intelligent-based monitoring systems play a crucial role in increasing water use efficiency, increasing yields, saving energy and quality production, and reducing adverse environmental impacts. IoT-based technologies assist with remote, real-time, and robust monitoring and collection of data to make the irrigation operation efficient and effective. Deploying different sensors provides real-time monitoring data that significantly minimize water loss, enabling better control of the amount of water in the root zone, plant, and evapotranspiration rate and maintain the moisture content within an ideal range of growth. The other

part, which is intelligent protocols, helps control the irrigation application or predict plant water requirements for the upcoming hours, days, or weeks. This system enables growers to achieve on-demand control of irrigation operations and to predict the future moisture content in the soil, water status in plants and the proportion of water that will be lost due to various factors over the coming periods. However, many challenges remain to be addressed through further studies to ensure that the systems more cost-beneficial for researchers and growers. Challenges include standardizing the IoT-and intelligent-based system protocol. More specifically, the system should be user friendly, secure, require low power consumption (battery-operated), and use precalibrated sensors. Sometimes both researchers and growers may find the systems confusing. Regarding security problems, for example, if we consider LoRaWAN technology, connecting the end nodes with the gateways may sometimes be mixed with or receive nearby LoRa Node data, and thus the system is unsuitable. In future studies developing long-range protocols, security issues should be considered.

Calibration is another major challenge that requires in situ adjustment because it may vary in terms of the soil texture and presence of organic matter. This issue should be the focus of future studies. On the other hand, through this review, we also found that most of the technologies are employed in greenhouses rather than open fields. The abovementioned technologies should also be utilized in outdoor farming systems.

4.4   Concluding Remarks

Considering the increased food demands and water scarcity, water management within irrigation systems is the key to securing water for industrial, municipal and daily utilization, as well as meeting world food demands. Through this review, we realized that precision irrigation systems are evolving, and their application makes irrigation management more efficient and convenient. Practicing precision irrigation whenever it is incorporated with IoT where different sensors are deployed in the field to monitor and collect the data and send them to microcontrollers and cloud servers further facilitates real-time irrigation monitoring and leads to increased water use efficiency. When controlling irrigation systems, which is a challenge worldwide, the use of advanced technologies such as artificial intelligence is important to take irrigation operations to the next level. An artificial intelligent control system enables smart control of irrigation operations, which minimizes water loss. The system will become more comprehensive and robust when combined with IoT-based irrigation systems, especially under conditions where the laborers are aging and smart and remote-control systems are immediately needed. However, further research is required to devise new instruments with high accuracy and low cost to make the irrigation operation much more user friendly and robust.

References

Abioye, E. A., Abidin, M. S. Z., Mahmud, M. S. A., Buyamin, S., Ishak, M. H. I., Rahman, M. K. I. A., Otuoze, A. O., Onotu, P., & Ramli, M. S. A. (2020). A review on monitoring and advanced control strategies for precision irrigation. *Computers and Electronics in Agriculture, 173*, 105441. https://doi.org/10.1016/J.COMPAG.2020.105441

Adeyemi, O., Grove, I., Peets, S., & Norton, T. (2017). Advanced monitoring and management systems for improving sustainability in precision irrigation. *Sustainability (Switzerland), 9*(3), 353. https://doi.org/10.3390/SU9030353

Azaza, M., Tanougast, C., Fabrizio, E., & Mami, A. (2016). Smart greenhouse fuzzy logic based control system enhanced with wireless data monitoring. *ISA Transactions, 61*, 297–307. https://doi.org/10.1016/J.ISATRA.2015.12.006

Belaqziz, S., Khabba, S., Er-Raki, S., Jarlan, L., Le Page, M., Kharrou, M. H., El Adnani, M., & Chehbouni, A. (2013). A new irrigation priority index based on remote sensing data for assessing the networks irrigation scheduling. *Agricultural Water Management, 119*, 1–9.

Buttaro, D., Santamaria, P., Signore, A., Cantore, V., Boari, F., Montesano, F. F., & Parente, A. (2015). Irrigation management of greenhouse tomato and cucumber using tensiometer: Effects on yield, quality and water use. *Agriculture and Agricultural Science Procedia, 4*, 440–444. https://doi.org/10.1016/j.aaspro.2015.03.050

Campos, N. G. S., Rocha, A. R., Gondim, R., da Silva, T. L. C., & Gomes, D. G. (2020). Smart & green: An internet-of-things framework for smart irrigation. *Sensors (Switzerland), 20*(1), 190. https://doi.org/10.3390/s20010190

Cesari de Maria, S., Bischetti, G. B., Chiaradia, E. A., Facchi, A., Miniotti, E. F., Rienzner, M., Romani, M., Tenni, D., & Gandolfi, C. (2017). The role of water management and environmental factors on field irrigation requirements and water productivity of rice. *Irrigation Science, 35*(1), 11–26.

Cremona, M. V., Stützel, H., & Kage, H. (2004). Irrigation scheduling of kohlrabi (Brassica oleracea var. gongylodes) using crop water stress index. *HortScience, 39*(2), 276–279. https://doi.org/10.21273/hortsci.39.2.276

Croock Al-Janabi, M. S., Hal, A. K., & Khatri, V. (2018). Application of Fuzzy logic in water irrigation system Related papers Fuzzy based Decision Support Model for Irrigat ion Syst em Management Application of Fuzzy logic in water irrigation system. *International Research Journal of Engineering and Technology, 5*, 3372. Retrieved from www.irjet.net

De la Rosa, J. M., Domingo, R., Gómez-Montiel, J., & Pérez-Pastor, A. (2015). Implementing deficit irrigation scheduling through plant water stress indicators in early nectarine trees. *Agricultural Water Management, 152*, 207–216. https://doi.org/10.1016/j.agwat.2015.01.018

De Pascale, S., Incrocci, L., Massa, D., Rouphael, Y., & Pardossi, A. (2019). Advances in irrigation management in greenhouse cultivation. In L. Marcelis & E. Heuvelink (Eds.), *Achieving sustainable greenhouse cultivation* (pp. 241–284). Burleigh Dodds Science Publishing. https://doi.org/10.19103/as.2019.0052.12

Deka, P. C., & Chandramouli, V. (2009). Fuzzy neural network modeling of reservoir operation. *Journal of Water Resources Planning and Management, 135*, 5. https://doi.org/10.1061/ASCE0733-94962009135:15

Domínguez-Niño, J. M., Oliver-Manera, J., Girona, J., & Casadesús, J. (2020). Differential irrigation scheduling by an automated algorithm of water balance tuned by capacitance-type soil moisture sensors. *Agricultural Water Management, 228*, 105880. https://doi.org/10.1016/J.AGWAT.2019.105880

Elaydi, H. (2017). An automated irrigation system for greenhouses. *American Journal of Electrical and Electronic Engineering, 5*(2), 48–57. https://doi.org/10.12691/ajeee-5-2-3

Falamarzi, Y., Palizdan, N., Huang, Y. F., & Lee, T. S. (2014). Estimating evapotranspiration from temperature and wind speed data using artificial and wavelet neural networks (WNNs). *Agricultural Water Management, 140*, 26–36. https://doi.org/10.1016/j.agwat.2014.03.014

Goap, A., Sharma, D., Shukla, A. K., & Rama Krishna, C. (2018). An IoT based smart irrigation management system using Machine learning and open source technologies. *Computers and Electronics in Agriculture, 155*, 41–49. https://doi.org/10.1016/j.compag.2018.09.040

Gu, Z., Qi, Z., Ma, L., Gui, D., Xu, J., Fang, Q., Yuan, S., & Feng, G. (2017). Development of an irrigation scheduling software based on model predicted crop water stress. *Computers and Electronics in Agriculture, 143*, 208–221. https://doi.org/10.1016/j.compag.2017.10.023

Gu, Z., Zhu, T., Jiao, X., Xu, J., & Qi, Z. (2021). Neural network soil moisture model for irrigation scheduling. *Computers and Electronics in Agriculture, 180*, 105801. https://doi.org/10.1016/J.COMPAG.2020.105801

Gutiérrez, J., Villa-Medina, J. F., Nieto-Garibay, A., & Porta-Gándara, M. Á. (2013). Automated irrigation system using a wireless sensor network and GPRS module. *IEEE Transactions on Instrumentation and Measurement, 63*(1), 166–176.

Hargreaves, G. H., Samani, Z. A., & Abstract, A. (1985). Reference crop evapotranspiration from temperature. *Applied Engineering in Agriculture, 1*, 96.

Hemming, S., de Zwart, F., Elings, A., Righini, I., & Petropoulou, A. (2019). Remote control of greenhouse vegetable production with artificial intelligence—Greenhouse climate, irrigation, and crop production. *Sensors (Switzerland), 19*(8), 1807. https://doi.org/10.3390/s19081807

Incrocci, L., Thompson, R. B., Fernandez-Fernandez, M. D., de Pascale, S., Pardossi, A., Stanghellini, C., Rouphael, Y., & Gallardo, M. (2020). Irrigation management of European greenhouse vegetable crops. *Agricultural Water Management, 242*, 106393. https://doi.org/10.1016/j.agwat.2020.106393

Khriji, S., el Houssaini, D., Kammoun, I., & Kanoun, O. (2021). Precision irrigation: An IoT-enabled wireless sensor network for smart irrigation systems. In T. Hamrita (Ed.), *Women in precision agriculture. Women in engineering and science* (pp. 107–129). Springer. https://doi.org/10.1007/978-3-030-49244-1_6

Kişi, Ö. (2006). Daily pan evaporation modelling using a neuro-fuzzy computing technique. *Journal of Hydrology, 329*(3–4), 636–646. https://doi.org/10.1016/j.jhydrol.2006.03.015

Kodali, R. K., & Sarjerao, B. S. (2017, July). A low cost smart irrigation system using MQTT protocol. In *2017 IEEE Region 10 Symposium (TENSYMP)* (pp. 1–5). IEEE.

Krishnan, R. S., Julie, E. G., Robinson, Y. H., Raja, S., Kumar, R., Thong, P. H., & Son, L. H. (2020). Fuzzy logic based smart irrigation system using internet of things. *Journal of Cleaner Production, 252*, 119902. https://doi.org/10.1016/j.jclepro.2019.119902

Liao, Y., Cao, H. X., Xue, W. K., & Liu, X. (2021). Effects of the combination of mulching and deficit irrigation on the soil water and heat, growth and productivity of apples. *Agricultural Water Management, 243*, 106482.

Lichtenberg, E., Majsztrik, J., & Saavoss, M. (2013). Profitability of sensor-based irrigation in greenhouse and nursery crops. *HortTechnology, 23*, 770.

Moghaddamnia, A., Ghafari Gousheh, M., Piri, J., Amin, S., & Han, D. (2009). Evaporation estimation using artificial neural networks and adaptive neuro-fuzzy inference system techniques. *Advances in Water Resources, 32*(1), 88–97. https://doi.org/10.1016/j.advwatres.2008.10.005

Mohamed, E. S., Belal, A. A., Abd-Elmabod, S. K., El-Shirbeny, M. A., Gad, A., & Zahran, M. B. (2021). Smart farming for improving agricultural management. *The Egyptian Journal of Remote Sensing and Space Science*.

Nikolaou, G., Neocleous, D., Katsoulas, N., & Kittas, C. (2019). Irrigation of greenhouse crops. *Horticulturae, 5*(1), 7. https://doi.org/10.3390/horticulturae5010007

O'Shaughnessy, S. A., Andrade, M. A., & Evett, S. R. (2017). Using an integrated crop water stress index for irrigation scheduling of two corn hybrids in a semi-arid region. *Irrigation Science, 35*(5), 451–467. https://doi.org/10.1007/s00271-017-0552-x

Osroosh, Y., Troy Peters, R., Campbell, C. S., & Zhang, Q. (2015). Automatic irrigation scheduling of apple trees using theoretical crop water stress index with an innovative dynamic threshold. *Computers and Electronics in Agriculture, 118*, 193–203. https://doi.org/10.1016/j.compag.2015.09.006

Owen, J. S., Jr., Lebude, A. V., Chappell, M., & Hoskins, T. (2016). Advanced irrigation management for container-grown ornamental crop production. Virginia Coop. Ext. Hort 218-P. pp 1–18. Retrieved from HORT-218.pdf (vt.edu)

Pacco, H. C. (2022). Simulation of temperature control and irrigation time in the production of tulips using Fuzzy logic. *Procedia Computer Science, 200*, 1–12. https://doi.org/10.1016/j.procs.2022.01.199

Palmer-Jones, R., & Sen, K. (2003). What has luck got to do with it? A regional analysis of poverty and agricultural growth in rural India. *The Journal of Development Studies, 40*(1), 1–31.

Pawlowski, A., Sánchez-Molina, J. A., Guzmán, J. L., Rodríguez, F., & Dormido, S. (2017). Evaluation of event-based irrigation system control scheme for tomato crops in greenhouses. *Agricultural Water Management, 183*, 16–25. https://doi.org/10.1016/j.agwat.2016.08.008

Pirkner, M., Dicken, U., & Tanny, J. (2014). Penman-Monteith approaches for estimating crop evapotranspiration in screenhouses-a case study with table-grape. *International Journal of Biometeorology, 58*(5), 725–737. https://doi.org/10.1007/s00484-013-0653-z

Prenger, J. J., Ling, P. P., Hansen, R. C., Keener, H. M., & Prenger, J. J. (2005). Plant response-based irrigation control system in a greenhouse: System evaluation. *Transactions of ASAE, 48*(3), 1175–1183.

Seelig, H. D., Stoner, R. J., & Linden, J. C. (2012). Irrigation control of cowpea plants using the measurement of leaf thickness under greenhouse conditions. *Irrigation Science, 30*(4), 247–257. https://doi.org/10.1007/s00271-011-0268-2

Selmani, A., Outanoute, M., Oubehar, H., Ed-Dahhak, A., Lachhab, A., Guerbaoui, M., & Bouchikhi, B. (2019). An embedded solar-powered irrigation system based on a cascaded fuzzy logic controller. *Asian Journal of Control, 21*(4), 1941–1951. https://doi.org/10.1002/asjc.2220

Sidhu, R. K., Kumar, R., & Rana, P. S. (2020). Machine learning based crop water demand forecasting using minimum climatological data. *Multimedia Tools and Applications, 79*(19), 13109–13124.

Steppe, K., de Pauw, D. J. W., & Lemeur, R. (2008). A step towards new irrigation scheduling strategies using plant-based measurements and mathematical modelling. *Irrigation Science, 26*(6), 505–517. https://doi.org/10.1007/s00271-008-0111-6

Thompson, R. B., Gallardo, M., Valdez, L. C., & Fernández, M. D. (2007). Using plant water status to define threshold values for irrigation management of vegetable crops using soil moisture sensors. *Agricultural Water Management, 88*(1–3), 147–158. https://doi.org/10.1016/j.agwat.2006.10.007

Trajkovic, S. (2005). Temperature-based approaches for estimating reference evapotranspiration. *Journal of Irrigation and Drainage Engineering, 131*, 316. https://doi.org/10.1061/ASCE0733-94372005131:4316

Trajkovic, S., Todorovic, B., & Stankovic, M. (2022). Forecasting of reference evapotranspiration by artificial neural networks. *Journal of Irrigation and Drainage Engineering, 129*, 454. https://doi.org/10.1061/ASCE0733-94372003129:6454

Tseng, D., Wang, D., Chen, C., Miller, L., Song, W., Viers, J., Vougioukas, S., Carpin, S., Ojea, J. A., & Goldberg, K. (2018). *Towards automating precision irrigation: Deep learning to infer local soil moisture conditions from synthetic aerial agricultural images*. IEEE. Retrieved from https://github.com/BerkeleyAutomation/RAPID

Van Iersel, M. W., Chappell, M., & Lea-Cox, J. D. (2013). Sensors for improved efficiency of irrigation in greenhouse and nursery production. *HortTechnology, 12*, 735.

Wang, H., Sánchez-Molina, J. A., Li, M., Berenguel, M., Yang, X. T., & Bienvenido, J. F. (2017). Leaf area index estimation for a greenhouse transpiration model using external climate conditions based on genetics algorithms, back-propagation neural networks and nonlinear autoregressive exogenous models. *Agricultural Water Management, 183*, 107–115. https://doi.org/10.1016/j.agwat.2016.11.021

Chapter 5
Strategic Short Note: Artificial Intelligence and Internet of Things: Application in Urban Water Management

Martin Anda, Roberta Fornarell, and Stewart Dallas

Abstract AI and machine learning techniques have already demonstrated significant outcomes in various water industry applications such as water quality monitoring, chemical dosing, prioritising active leakage detection areas, intelligent network optimisation, and the prediction of water pipe failure. Can these techniques be extended from water utility operations (Anda, 2017) into home and commercial water usage (Schmack et al., 2019)? The introduction of a reward credit system to those residents who actively save energy-intensive mains water and wastewater, whilst optimally managing aquifer recharge, can support localised, hybrid water sources at residential and community scale (Fornarelli et al., 2019). While currently, machine-learning algorithms are being used to detect inaccuracies or anomalies in water meter data, in the future, AI and machine learning techniques can be used to better manage the use of alternate water sources in cities to achieve sustainable hybrid water systems (Fornarelli et al., 2021).

Keywords Artificial intelligence · Internet of things · Urban water · Smart meter · Hybrid water systems · Water trading

5.1 Introduction

AI and machine learning techniques have already demonstrated significant outcomes in various water industry applications such as water quality monitoring, chemical dosing, prioritising active leakage detection areas, intelligent network optimisation, and the prediction of water pipe failure. Can these techniques be extended from water utility operations (Anda, 2017) into home and commercial water usage (Schmack et al., 2019)?

M. Anda (✉) · R. Fornarell · S. Dallas
Murdoch University, Perth, WA, Australia
e-mail: M.Anda@murdoch.edu.au

The rapid rate of global development in ultrasonic, revenue grade advanced metering technology is allowing the emergence of a new cost-effective approach to smart urban water management (Haning & Anda, 2016). In particular, it can enable hybrid water systems—those utilising alternative and non-potable water sources (Byrne et al., 2019)—to be accurately quantified in a continuous manner without meter fouling. This in turn paves the way for hybrid water systems to be utilised in urban water trading—a novel market-based approach to a specified band of water use to enable higher levels of water efficiency (Fornarelli et al., 2019).

The introduction of a reward credit system to those residents who actively save energy-intensive mains water and wastewater, whilst optimally managing aquifer recharge, can support localised, hybrid water sources at residential and community scale (Fornarelli et al., 2021). While currently, machine-learning algorithms are being used to detect inaccuracies or anomalies in water meter data, in the future, AI and machine learning techniques can be used to better manage the use of alternate water sources in cities to achieve sustainable hybrid water systems.

5.2   Methods

Over 60 ultrasonic smart water meters were deployed in 2018–2019 across 40 participating households within the City of Fremantle, Perth, Western Australia as part of the RENeW Nexus project (Resilient Energy and Water Systems) which is a federally-funded initiative being delivered under the national Smart Cities and Suburbs program (Fornarelli et al., 2019). The approach adopted for the water component of RENew Nexus integrates the smart metering of hybrid water systems, household participation, and data analytics at the residential scale within the traditional centralised urban water network (Fornarelli et al., 2021) (Fig. 5.1).

5.3   Results

Since installation, water volume data has been recorded every 30 min with each meter uploading this data daily via the Telstra NB-IoT network to a dedicated data management platform. The water usage data is then analysed and water balance modelling undertaken (Fornarelli et al., 2019). The water balance quantifies the volume of water used by source (rainwater, groundwater, greywater, mains), the volume of wastewater produced, as well as the amount of local abstraction and recharge to aquifer. This in turn enumerates mains water savings, reductions in discharge to sewer, and abstraction/discharge to aquifer for each site (Fornarelli et al., 2021).
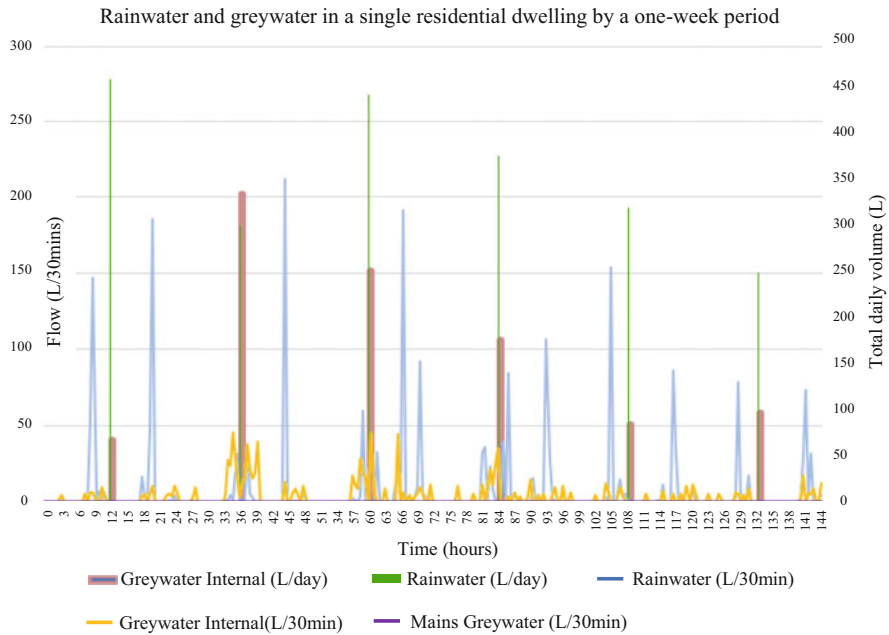
Fig. 5.1 Example of smart metering output (five meters deployed) from a Fremantle residence maximising alternative water sources

5.4 Conclusion

In all the aspects analysed in this paper (e.g., community engagement, water efficiencies, water value and water tariff), the role of digital technologies in the water sector is vital to assist with the new shift from centralised water networks to a community-empowered, integrated, centralised-hybrid water system (Fornarelli et al., 2021).

References

Anda, M. (2017). Decentralised water and energy infrastructure: Integration into compact urban form. In J. H. P. Bay & S. Lehmann (Eds.), *Growing compact: Urban form, density and sustainability* (pp. 204–218). Routledge.

Byrne, J. J., Anda, M., & Ho, G. E. (2019). Water sustainable house: Water auditing of 3 case studies in Perth, Western Australia. *Water Practice & Technology, 14*(2), 435–443. https://doi. org/10.2166/wpt.2019.028

Fornarelli, R., Anda, M., Dallas, S., & Morrison, G. M. (2021). Smart metering technology and community participation: Investigating household water usage and perceived value of hybrid water systems. In *Water supply*. IWA.

Fornarelli, R., Anda, M., Dallas, S., Schmack, M., Dawood, F., Byrne, J., Morrison, G. M., & Fox-Reynolds, K. (2019). Enabling residential hybrid water systems through a water credit-debit system. In *Water supply*. IWA.

Haning, A., & Anda, M. (2016). The post carbon city and smart metering. In M. Eissa (Ed.), *Smart metering technology and services – Inspirations for energy utilities* (pp. 63–83). InTech. ISBN 978-953-51-2452-8.

Schmack, M., Anda, M., Dallas, S., & Fornarelli, R. (2019). Urban water trading – Hybrid water systems and niche opportunities in the urban water market – A literature review. *Environmental Technology Reviews, 8*(1), 65–81. https://doi.org/10.1080/21622515.2019.1647292

Chapter 6
Purification of Agricultural Polluted Water Using Solar Distillation and Hot Water Producing with Continuous Monitoring Based on IoT

P. D. Kahandage, Mohammad Hussain Seyar, Ryozo Noguchi, and Tofael Ahamed

Abstract Considerable number of people are still suffering from the scarcity of clean drinking water as most of the available surface water bodies are already polluted. The difficulty is severe, especially in underdeveloped countries in which majority is facing for energy crisis as well. Therefore, when finding the solutions, it is very important to pay attention for both problems as water purification and management is always bound with energy. Even though many of household water purification methods have been invented, solar water distillation is a simple and affordable for many countries which are blessed with sufficient solar power and concern on energy. Continuous monitoring of water quality is an important task of a purification system as the keeping of quality of water within a recommended range is the main goal. IoT devises such as sensors, controllers, communication, and cloud flat forms facilitate the real-time monitoring of water quality and quick responding remotely with smart phone applications. Most of the domestic level water purification systems can be more advanced with IoT technology. Different water quality

P. D. Kahandage
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering and Soil Science, Faculty of Agriculture, Rajarata University of Sri Lanka, Mihintale, Sri Lanka

M. H. Seyar
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Soil Science and Irrigation, Kabul University, Kabul, Afghanistan

R. Noguchi
Laboratory of Agricultural Systems Engineering, Division of Environmental Science and Technology, Graduate School of Agriculture, Kyoto University, Kyoto, Japan

T. Ahamed (✉)
Faculty of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

sensors such as pH sensors, Temperature sensors, Turbidity sensors, Dissolved Oxygen (DO) sensors, Conductivity sensors, Salinity sensors, and Total Dissolved Solids (TDS) sensors are available and possible to be controlled by popular controller types such as Arduino microcontroller board or Raspberry Pi minicomputer board. This chapter covers the concept of designing and fabrication of a solar system which can be used for two purposes as water distillation and hot water producing and controlled by IoT technology with real time data monitoring. Temperature of the water and air of the solar still, pH, and TDS of water were measured using relevant sensors and monitored. The solar still of the system is also acting as the storage tank by insulating the glass top when the system is working as a hot water system. The evaporation capacity of the still were improved by several strategies such as connecting a flat plate collector, placing sponge cubes in the still and placing a sand layer under the still. The performance of the solar water distillation system with each strategy was evaluated separately and the combined effect of all the strategies was also checked. All the introduced evaporation enhancement methods have performed well, and the maximum distillation capacity was 3.98 L/m^2/day, and it has been achieved when the system was running with all the strategies together. The maximum water temperature was also achieved when all the strategies are deployed, and it was recorded as 58 °C by indicating the potential for using the system for making hot water for domestic purposes such as washing and cleaning. The measured water quality parameters show that the purified water by the system is suitable for drinking.

Keywords Continuous monitoring of water quality · Hot water producing · IoT · Solar water distillation · Water quality

6.1 Introduction

The application of the IoT in water management is an innovative trend that brings many advantages for both consumers and administrators. Ensuring the availability and sustainable management of water and sanitation for all is one of the Sustainable development goals (SDGs) adopted by all United Nations Member States in 2015 to alleviate poverty and improve livelihood (United Nations, 2020). Water is recognized as an essential factor for better health and as a key factor in many sectors, such as poverty reduction, and improved food security, peace and human rights, ecosystems, and education. However, many developing countries still face a considerable challenge in finding sufficient clean water throughout the year for drinking and sanitation purposes. In many agricultural countries, agricultural pollutants are the major sources of surface water pollution, especially in rivers and streams. Additionally, groundwater may be polluted due to the leaching of pesticides and fertilizers into aquifers. The irresponsible overuse of fertilizers and other agrochemicals in farming areas to enhance yield accounts for this pollution. Consequently, there is a high concentration of heavy metal ions such as As, Cd, Cu, and Zn in groundwater. The accumulation of nutrients in water bodies leads to eutrophication, threatening

Table 6.1 Accessibility of people for drinking water in 2020

Water source	Description	Number of people
Safely managed services	Improved water sources, available anytime, free from contaminants	5.8 billion
Basic services	Improved water sources, located within a round trip of 30 minutes	1.2 billion
Limited services	Improved water sources, located within a round trip of more than 30 minutes	0.282 billion
Unprotected sources	From wells and springs	0.368 billion
Untreated surface water	From lakes, ponds, rivers, and streams	0.122 billion

aquatic biodiversity by increasing the biological oxygen demand. Another water-related problem is the hardness of the water due to the presence of calcium and magnesium ions. This occurs when water percolates over banks of limestone, chalk or gypsum. Although it is generally safe to drink, the taste is unpleasant due to the excess minerals.

The total amount of available water on Earth is estimated to be 1.4 billion km^3, of which approximately 97.5% is seawater, while only 2.5% is clean water (Modi & Modi, 2019). As reported by the World Health Organization, nearly 2.1 billion people do not have access to safe drinking water, 3.4 million people die each year due to water-related issues, and millions of women and children spend 3–6 h each day collecting water from distant water sources, even those that are not in good condition for consumption (WHO, 2022). Although ensuring equitable access to clean and affordable water is targeted by the SDGs, a considerable number of people still do not have access to clean drinking water and use polluted and unprotected sources without any treatments for their water requirements. Table 6.1 shows the accessibility of drinking water for people in 2020 (WHO, 2022).

Treated water services are classified under three categories: safely managed, basic, and limited services. Safely managed services are characterized by having drinking water from improved sources on the premises and available anytime as well as free from fecal and chemical contaminants. The sources of basic services are almost the same as those of safely managed services, but they are located within a round trip of 30 minutes. In limited services, improved water sources are located far away, and collection takes more than 30 minutes. Contaminated drinking water is a major burden on human health, and there is the greatest risk of waterborne diseases among infants and young children, weakened people and elderly individuals, especially when living in unhygienic conditions (Gorchev & Ozolins, 2011).

Therefore, available hard water and contaminated water must be purified to cater to the water demand in many countries that suffer from water scarcity. Thus, efficient and sustainable utilization of existing surface water sources with the support of advanced available technologies are essential.

The available water purification methods, such as reverse osmosis (RO) and vapor compression, are not appropriate for domestic levels in rural areas of developing countries. Water distillation is a purification method that can remove heavy metals and other impurities from water. Many tropical countries with sufficient solar energy throughout the year have a higher likelihood of utilizing free energy for distilling both hard and contaminated water.

Energy is also a word that is always bound together with water management, as pumping, purification, storage, and delivery depend on electricity. Ensuring affordable, reliable, and sustainable energy is also one of the sustainable development goals (United Nations, 2020). Therefore, promoting the use of solar energy in water management is highly justifiable, as it provides emission-free, reliable, freely available, and renewable energy. A solar water distillation unit requires simple technology, no requirement of highly skilled labor for maintenance work and low energy consumption; hence, it is a great alternative for obtaining fresh water from hard water. The most beneficial advantage of solar energy is low operation and maintenance costs due to zero fuel cost energy, but it requires more space for its collection and high initial investments (Tiwari et al., 2003). Nevertheless, it is the best solution for water purification in remote areas and small communities in dry zones with a lack of drinking water.

As even low-price sensors allow for easy measurement of water quality and other parameters and the possibility of using available communication technologies with few configurations, IoT-based water purification and management systems can be considered low-cost solutions, which facilitate distant monitoring and regulation (Singh & Ahmed, 2020). Therefore, the goal of this chapter is to discuss the possibilities to use IoT techniques for continuous monitoring and implementation of decisions in solar water purification and hot water systems.

6.2   The Architecture of the Proposed IoT-Based Solar Water Distillation and Hot Water System

The main purpose of this chapter is to discuss the possibility of using IoT technology in solar water purification and hot water production. Therefore, the main concept of employing IoT improvement technologies with the support of recently carried out relevant studies and the results of evaluating some components of the system is discussed. Figure 6.1 shows the conceptual diagram of the IoT-based solar water distillation system and hot water system.

The most salient feature of this concept is switching the system between purification by distillation and hot water production based on the requirements and remotely measured parameters by the sensors. As the main purpose of this system is purification, it contains a water basin, solar preheating system, cooling system, distilled water collection tank, and various sensors for water quality and water level determination.
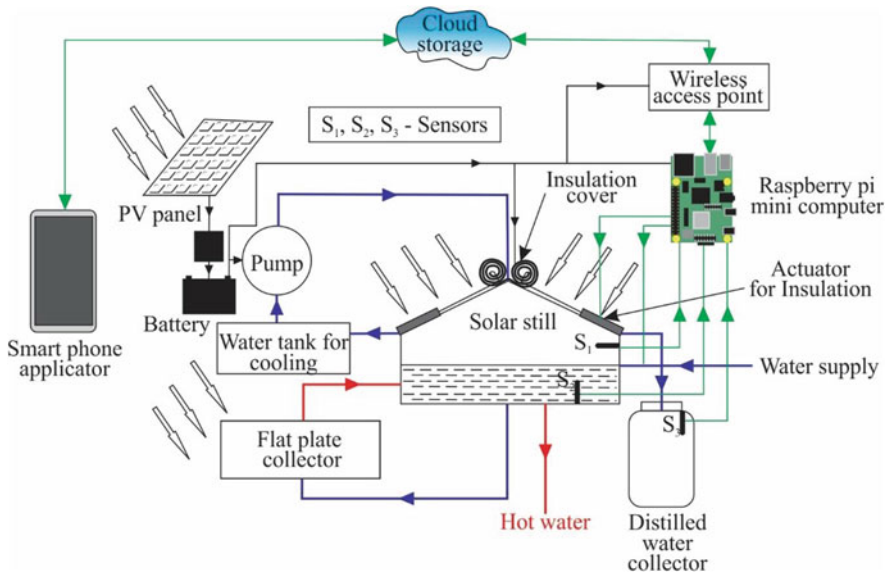
Fig. 6.1   Schematic diagram of the solar water distillation unit with IoT devices

In this IoT-based purification and hot water system, distillation with solar power has been selected as the purification technique so that obtaining hot water for domestic purposes is also facilitated with solar power. The flat plate solar collector, which is one of the major components of a solar water heating system, is suggested to connect with the solar still of the distillation system to increase water evaporation. The double-sloped water basin is connected to a continuous supply of water from the source, and the supply is under control with the support of a flow control valve. The water inside the basin is allowed to flow continuously through the flat plate solar collector mounted on the outside, increasing its temperature to facilitate evaporation. This flow is a natural phenomenon that occurs with only the difference in water density with temperature. The preheated water in the basin constantly evaporates due to solar heat, and the evaporated water condenses again by hitting the glass cover above the basin. To facilitate the condensation process, the glass cover is regularly cooled by the cooling system powered by a pump operating with solar power, and the distilled water deposited on the glass cover is freely collected into the distilled water collector with the assistance of gravity.

In contemplation of switching the system for hot water production, the glass cover cooling system is shut off and covered by insulation allowing the water basin to work as a hot water storage tank. At the same time, the basin should be filled with water. The switching process is expected to monitor the water levels and other parameters of the system with the support of IoT. The IoT system comprises thermal sensors and ultrasonic water level sensors, Wi-Fi communication through a Raspberry Pi minicomputer, and cloud storage controlled by a smartphone application.

6.3 Basic Architecture of an IoT-Based Water Purification System

The main components of an IoT-based system are sensors, controllers, communication systems, and IoT platforms. The main purpose of a water purification system is to maintain water quality within the purest range by removing externally accumulated impurities. Therefore, continuously monitoring the water quality is essential to confirm that the system is working properly. IoT technology facilitates this requirement with different water quality sensors. The basic architecture of the IoT-based water purification system is shown in Fig. 6.2.

The most important water quality parameters to be measured continuously are pH, temperature, turbidity, dissolved oxygen (DO), conductivity, salinity, and total dissolved solids (TDS) (Singh & Ahmed, 2020). Table 6.2 gives the information on the water quality parameters, reference values, and sensors used to measure them.

As several possible communication systems, controllers, and IoT flat forms are available, and they can be selected considering the requirements.

The Arduino microcontroller board or Raspberry Pi minicomputer controller is preferable in this system as the controller. External hardware is required to connect to the internet when an Arduino microcontroller board is used.
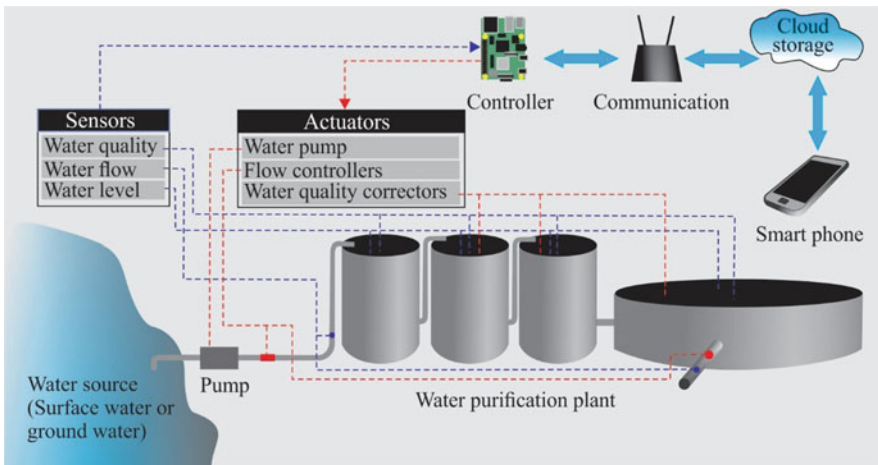
Fig. 6.2 Basic architecture of the IoT-based water purification system

Table 6.2 Basic water quality parameters of a water purification system and sensors in use

Water quality parameter	Description	Recommended value/range for drinking water	Sensors used
pH	Measures the degree of acidity or basicity of water using a scale from 0 to 14. The pH value of normal water should be 7 indicating the neutrality	6.5–8.5	pH sensors Ex: Combination pH sensors, differential pH sensors, laboratory pH sensors, process pH sensors
Temperature	An important factor for growth of organisms (ex. legionella) in water	20–50 °C Most preferable −30 °C	Temperature sensors Ex: Thermocouples, resistive temperature devices (RTDs, thermistors), infrared radiators, bimetallic devices, liquid expansion devices, molecular change of state devices, silicon diodes
Turbidity	Turbidity is the criterion that determines the clarity of water. It is measured by Formazin Nephelometric Unit (FTU) and Nephelometric Turbidity Unit (NTU)	Less than 5 NTU	Turbidity sensors Ex: Nephelometric sensors, suspended solid sensors, absorption sensors
Dissolved Oxygen (DO)	Amount of gaseous oxygen present in water	6.5–8.0 mg/L	Dissolved oxygen sensors
Conductivity	Measures the capability of water to pass a flow of electric current. It gives an idea about how much of substances, chemicals, and minerals are in the water	400 μS/cm	Water conductivity sensors
Salinity	Measures how much of salt is dissolved in water	Less than 200 ppm	Salinity sensors Ex: Electrode sensors, inductive sensors
Total Dissolved Solids (TDS)	TDS is the total amount of organic and inorganic substances present in the water	500 mg/L	TDS sensors

6.4 Water Purification Methods and the Possibility of Using IoT

Water purification is the process of removing contaminants from raw water to make it suitable for drinking. Dirt, parasites, bacteria, algae, viruses, fungi, and minerals, including heavy metals, are removed from the water during the purification process. Boiling, chlorination, filtration, coagulation and sedimentation, ion exchange, solar

treatments, distillation, and combined treatment systems are some of the small-scale purification methods in use at the domestic level (Agrawal & Bhalwar, 2009).

The boiling point of water is 100 °C at sea level and is highly influenced by the surrounding pressure. A vast range of waterborne pathogens is vulnerable to destruction when water is boiled. However, during the pasteurization process, heating to 60 °C for 10 min also destroys most of the waterborne pathogens.

Therefore, heating the water to the desired level and maintaining it for a considerable duration can minimize the energy utilization for boiling. Therefore, there is good potential for using IoT techniques to monitor the temperature and duration of a water heating system to control the use of energy and maintain water quality.

Chlorination is the most affordable and easy water purification method, especially when a large quantity of water is purified. In this process, a measured amount of chlorine (Cl) or chlorine compounds, such as sodium hypochlorite, is added to water and allows for a specified contact time (30 min) to destroy pathogens such as bacteria, viruses, and cysts. As the lethal effect of chlorine is determined by pH, temperature, and contact time, IoT technology can be applied to monitor these parameters in real-time (WaterProfessionals, 2022).

Filtration is a physical method of eradicating suspended particles, some pathogens, and algae from water to make the water safer and cleaner. Sand, charcoal, membrane, and ceramic filters are used to filter the water. Although the lifetime of a filter is determined by several factors, the total volume of the water filtered and the level of impurity of the water are major determinants of the lifetime. Therefore, sensors and IoT devices can be deployed to monitor filters.

In both coagulation and sedimentation methods, separation of suspended solid particles is anticipated. Coagulation is the phenomenon that increases the particle size of solid particles by combining them and allowing them to accumulate on the bottom of the tank. In sedimentation, solid particles are supposed to settle on the bottom of the tank due to their weight. Allum potash is one of the coagulation agents that can be effectively used to control cholera (Khan et al., 1984). As this process is time-consuming and excess coagulant agent gives the water an unpleasant taste, water quality sensors can be used to determine the exact quantity of required coagulant, and IoT devices can be used to provide coagulant precisely and on time.

Ion exchange is one of the most effective technologies to remove dissolved inorganic ions. In this process, ionic contaminants are removed from water using another tolerable ionic substance. This method has been widely used to remove the water hardness due to the presence of Ca and Mg ions. As regular maintenance and cleaning are essential for ion exchange purification systems, IoT technology can be used to monitor and determine the required maintenance (Mazille, 2020).

In combined treatment systems, two or more purification methods discussed above are combined.

6.5 Solar Water Distillation and Potential for Improvement with the Latest Innovations

Nature utilizes solar energy to convert saline water into freshwater by heating water from oceans, lakes, rivers, and other large bodies of water in the hydrologic cycle. Vapors are generated by increasing the temperature of water transported by wind to distant places, which condense and precipitate in the form of rain. This natural phenomenon can be achieved in an artificially constructed system, and condensed water can be collected easily for day-to-day uses (Sanserwal et al., 2020). Solar distillation is a promising and alternative method to supply water for small communities in rural areas. The major component of a solar water distillation system is the solar still, which contains water for distillation and provides conditions to absorb solar energy to make water vapor for desalination. The black paint inside the solar still improves the solar energy absorption, and the glass top (transparent cover) of the solar still allows the short wavelength of sunlight to enter the solar still while blocking the longwave infrared radiation reflected by the water surface, consequently increasing the temperature inside the solar still. The transmitted radiation into the solar still is partially reflected and absorbed by the water, increasing its temperature compared to the glass top. The absorbed radiation finally reaches the black painted surface, generally known as the basin inside layer, where it is largely absorbed. Then, most of the thermal energy is transmitted to the water, and a small quantity may be lost to the atmosphere by conduction. Due to this, the temperature difference between the water and glass top is increased. Therefore, transferring heat from water to the glass top commences via three modes of heat transfer, radiation, convection, and evaporation from the water surface. The evaporated water is deposited on the inner surface of the glass top and is condensed after releasing the latent heat. The condensed water stream into the collecting pipes is provided at the lower ends of the glass top under gravity. The collected water is distilled water and ready for further use. The thermal energy received by the glass cover, through radiation, convection, and latent heat, is lost to the ambient environment by radiation and convection (Tiwari et al., 2003). A perforated tube can be used to provide water flow on the upper surface of the glass top to accelerate the cooling process of the glass top to accelerate the evaporation process inside the solar still. Figure 6.3 shows two types of solar stills, a single slope and a double slope, with their basic functions.

When the glass top of the solar still is inclined on one side, as shown in Fig. 6.3a. The solar still is known as a single-slope solar still. As shown in Fig. 6.3b, the glass top of the double-slope solar still is inclined on two opposite sides. When a solar tracking mechanism is not available, the single-slope stills are more efficient in one-half of the day, while the double-slope stills show fairly constant efficiency throughout the day.
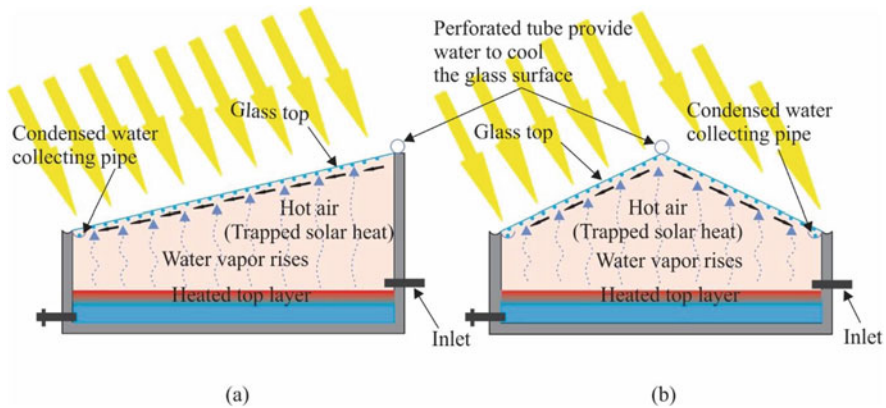
Fig. 6.3 (**a**) Single-slope and (**b**) double-slope solar stills

6.5.1 Active and Passive Solar Distillation

Based on solar energy utilization, solar distillation units or solar stills can be classified into two categories: active solar stills and passive solar stills. In passive solar stills, only the solar energy directly falling into the solar still contributes to water evaporation. In an active solar still, an external apparatus such as a flat plate solar collector, an inverted absorber, or an additional condenser is connected to the solar still to harness additional solar energy so that a faster rate of evaporation is assured compared to the active solar stills (Ansari et al., 2013). As shown by Alawee et al. (2021), the external heating process of active solar stills can be accomplished by using a flat plate collector, solar concentrator, heat pipes and mini solar ponds. In passive solar stills, the solar energy can be effectively harnessed to increase the basin water temperature using internal and external reflectors that reflect the sunlight on the absorption plate (Alawee et al., 2021). The efficiency and productivity of both active and passive stills can be increased by increasing the surface area of the solar distillation unit (Refalo et al., 2016). As shown by Aziz et al. (2017), several factors, such as solar intensity, wind velocity, ambient temperature, water-glass temperature difference, free water surface area, absorber plate area, glass angle, depth of water, and temperature of inlet water, affect the productivity of a solar still (Arunkumar et al., 2019a; Aziz et al., 2017). Figure 6.4 shows the schematic diagrams of active and passive solar stills.

Solar stills can be classified based on design and configurations as basin type, wick type, and special design such as hemispherical, spherical, concave, v shape, tubular conical, and pyramid type. The performance of these systems is determined by their design, elementary setup, and materials used to fabricate the distillation system (Sanserwal et al., 2020).
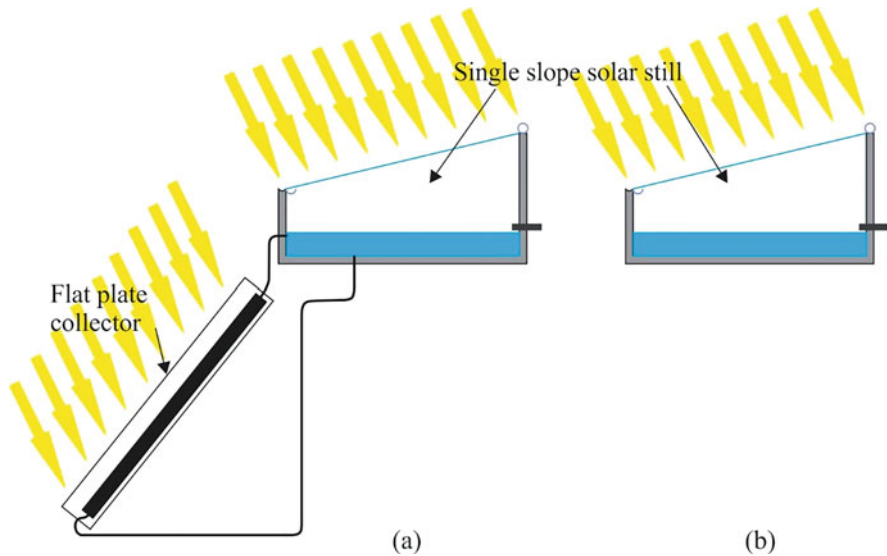
Fig. 6.4 (**a**) Active single-slope solar still and (**b**) passive single-slope solar still

6.5.2 Possible Innovations to Improve Vapor Generation in Solar Stills

The solar water distillation method is not popular as it is highly uneconomical because the distillation capacity of a simple basic solar still is very low, between 2 and 5 L/m^2/day (Manokar et al., 2018; Murugavel et al., 2010). Several studies have been conducted to improve the efficiency of vapor generation in solar stills concerning a higher output of distillation (more than 5 L/day). Experiments have shown that the use of solar concentrators internally and externally can increase evaporation in solar stills. Plain mirrors and parabolic reflectors can be used here to concentrate solar energy to improve productivity (Dev et al., 2011). Another method to improve a solar distillation system is using phase change material to store additional heat in the still. Here, different phase change materials, which can store heat as latent heat, can be applied. Paraffin wax and nanoparticle-enhanced paraffin are mostly used as latent heat storage materials (Kabeel & Abdelgaied, 2016).

In addition to latent heat storage, sensible heat storage materials can also be used in solar stills to increase productivity. Fins, sand, sponges, marbles, pebbles, iron scrap, wicks, charcoal, corrugated absorbers, black cotton, jute, clay, mild steel, and black gravel granite are used for sensible heat storage (Arunkumar et al., 2019b). Several studies have shown that the use of nanofluids in solar stills helps to increase evaporation. Mixing nanoparticles of Al_2O_3 and Cu_2O in wastewater still results in a significant improvement in distillation capacity (Omara et al., 2015).

Heat absorbing materials such as PV panels, black cotton and jute can be used as a higher heat-gaining material in solar stills to achieve a higher distillation (Manokar

et al., 2018; Pal et al., 2017). Several studies have been carried out by changing the design of the solar still to improve productivity. Among them, one study has shown that stepped-type solar stills can increase the absorbing area to improve the distillation output (Abdullah, 2013). Air blowers can be used to create a bubbling effect in the still to increase the evaporation rate by distributing heat energy equally throughout the basin (Joy et al., 2018). A condenser can be installed to improve the condensation process to accelerate distillation. A bank of tubes immersed in flowing fresh water is typically used as a condenser in solar stills (Refalo et al., 2016). One study, which was conducted by integrating a compression heat pump with a solar still in such a way that immersing the condenser in water and installing the evaporator in the air of the still to provide two sources of heat and two places for condensation, considerably improved the distillation capacity (Ben Halima et al., 2014).

6.6   Solar Water Heating Systems and Data Monitoring

Solar hot water systems or thermal systems have been designed specifically to harness the heat energy of the sun and transfer it to water to increase the temperature and store the warm water in an insulated tank while permitting the water to circulate through the system until it reaches the desired temperature. Waterflow can naturally occur due to the phenomenon of the water density changing with temperature or otherwise controlling the flow forcefully with the support of a pump. When a pump is in operation in the system, photovoltaic electricity from a solar panel is a good option to provide power for both the pump and the controlling system. Unlike using a natural flow system, when a controlled system is used, it is exceptionally easy to maintain the preferred water temperature by monitoring the system parameters with the assistance of thermal and flow sensors. Figure 6.5 gives a schematic illustration of a solar hot water system.

Two types of solar water heating collectors can be used in this system: evacuated tubes and flat plate collectors. Evacuated tubes are a bank of glass tubes mounted on a frame, and flat plate collectors are a bank of glass or metal tubes installed in an insulated absorbing box with a transparent top.

The day's solar intensity, ambient temperature, water temperatures at the lower and upper parts of the tank, lower and upper ends of the flat plate collector, and the outlet are important parameters to check regularly. Temperature sensors can be used to monitor the important temperature values in the system. Figure 6.6 shows how continuous data monitoring of a solar water heating system with IoT technology can be used to control the temperature of water output.
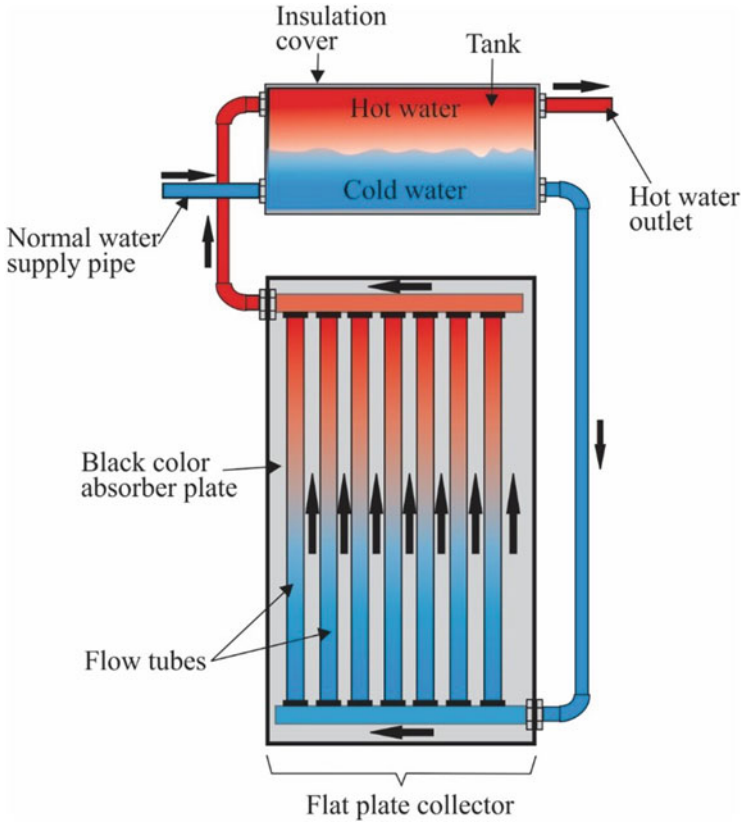
Fig. 6.5 A schematic diagram of a simple solar water heating system

6.7 IoT-Based Solar Water Distillation and Hot Water System

6.7.1 Study Conducted to Test the Performance of Solar Stills Under Different Improvement Strategies

The solar water distillation system comprised several units, such as a solar still or basin on a sand layer to store and evaporate water, a cooling system for condensation of distilled water, a flat plate solar collector to preheat the water, and a frame to carry all the components together. Materials were selected for fabricating each component based on the designed factors, such as simplicity, reliability, and stability. The basin was 0.072 m^3 in capacity with a 0.5 m^2 effective evaporation area and was composed of a double-sloped glass top and a water supply pipe with a floater valve. All the
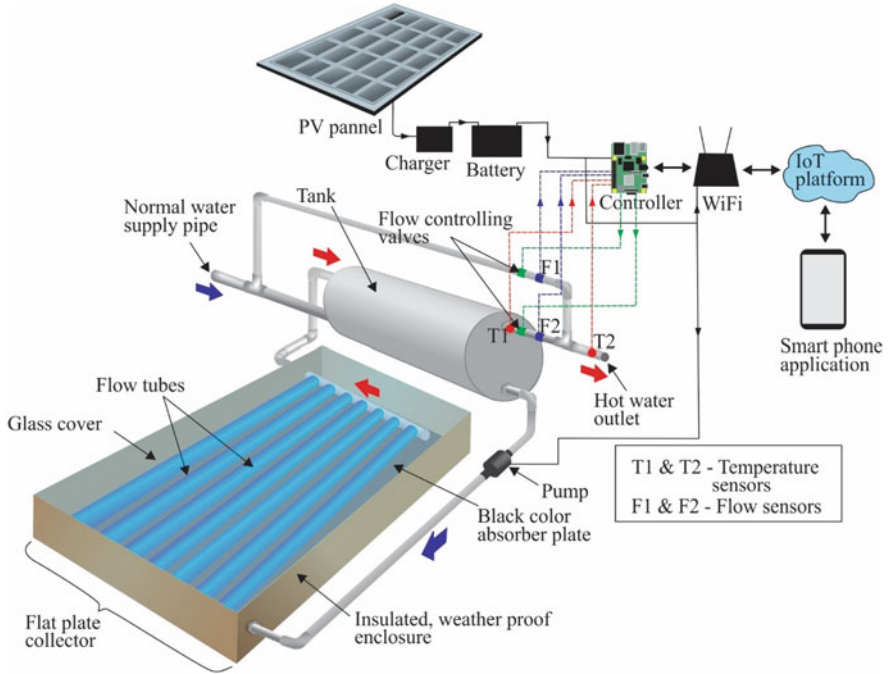
Fig. 6.6 Connection of sensors and IoT devices to control the temperature of output water of a solar water heating system

surfaces of the basin were fabricated with stainless steel and painted black for higher absorption of a large amount of solar radiation and very low transmissivity of solar energy. The sidewalls of the basin were insulated with 50 mm extended polystyrene to obtain the advantages of sensible heat storage (Abu-Hijleh & Rababah, 2003). Sponge cubes with dimensions of 60 mm × 60 mm × 30 mm were placed in the basin water to increase the wetted surface area and reduce the water surface tension. A river sand layer with a height of 2 cm was incorporated under the solar still to store the additional heat. A cooling system was installed to support the condensing process. The flat plate solar collector, which comprises a bank of copper tubes, was designed to increase the temperature of the basin water by preheating to increase the rate of distillation. Figure 6.7 shows a schematic diagram of the solar still with all the new features and sensors.

Plates 6.1, 6.2, and 6.3 show how sponge cubes, flat plate collectors, and sand layers have been set in the distillation system.
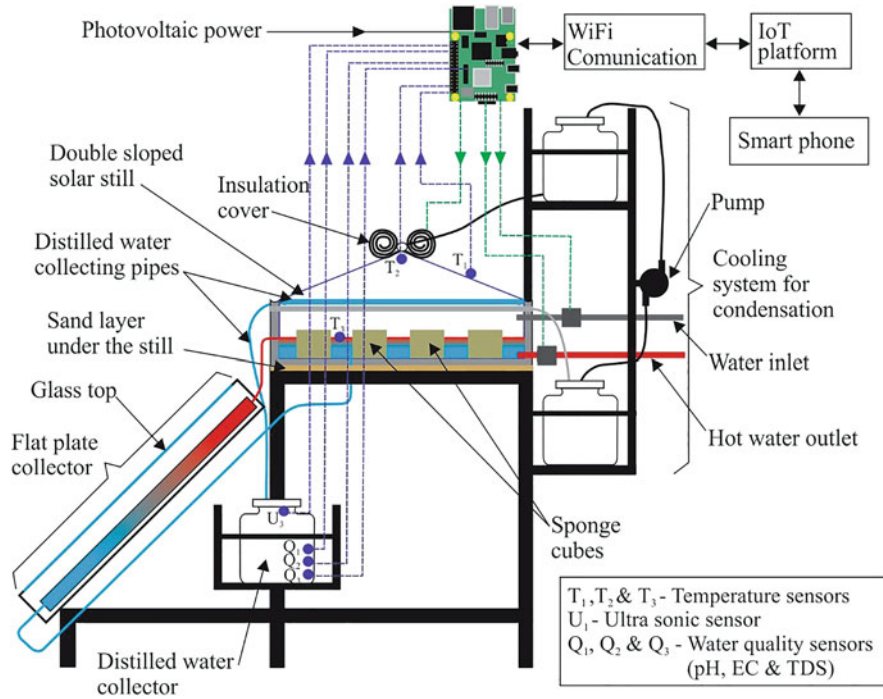
Fig. 6.7 Schematic diagram of the solar water distillation system with improved features

6.7.2   Performance Evaluation of the Solar Still

The solar still performance was evaluated under five conditions to estimate the contribution of each introduced feature separately.

In this study, only the passive solar still without a flat plate collector, sponge cubes and sand layer was considered as T1. The solar still with only the flat plate solar collector was considered T2. The solar still with only the sand layer was considered T3. The solar still with only sponge cubes was taken as T4, and the solar still with all the features of flat plate collector, sand layer and sponge cubes was considered as T5.

The total average distilled output, maximum and average air temperatures of the solar still, maximum and average water temperature of the solar still, and increase percentage of each treatment compared to T1 are demonstrated in Table 6.3.

Both basin air and basin water temperatures increased to the maximum just after noon because the absorption of heat was higher than the losses to the atmosphere. In the early hours of the morning, the difference in glass temperature and the water temperature was smaller, which caused lower productivity. This is because of the small amount of energy absorbed by the water during these periods. Figure 6.8 shows the variation in the distilled output of all the treatments throughout the day.

Plate 6.1 Arrangement of sponge cubes in the solar still

The highest output value and temperature values were observed in T5. Since the flat plate solar collector more efficiently absorbs heat than the solar still, the basin water temperature markedly increased, giving a higher production. When the sand bed was incorporated under the metal solar still, it absorbed heat energy from water so that the water and air temperatures inside the solar still decreased, resulting in lower production in the morning. However, during the afternoon period, the sand bed retained more heat energy due to its high thermal capacity. Therefore, during the afternoon period, the distilled output and air and water temperatures increased more conspicuously than during the morning period. In T3, the height of the sponge cubes exposed above the water surface was 1 cm. Abu-Hijleh & Rababah (2003) showed that at shorter exposure heights, more water rose to the top of the sponge by capillary forces, thus increasing the evaporation rate of the solar still. Applying sponge cubes increased the wetted surface area for evaporation and reduced the surface tension between water molecules with the help of small openings in the sponge cubes; thus, it was easier for the water molecules to evaporate. The graph in Fig. 6.8 clearly shows that the distilled output increased in the morning even at low basin air and water temperatures.

Plate 6.2 Connection of
flat plate collector with solar
still

6.7.3   Water Quality Results

Table 6.4 shows the measured pH, EC, and TDS values of water before and after all
the treatments compared to the laboratory distilled water.

As the water used for the experiment was obtained from the public water supply,
all the water quality parameters of the water before treatments were within the
standard range recommended by the World Health Organization (WHO). According
to the water quality results, the initial EC and TDS values of water considerably
decreased by solar water distillation, but they were still within the recommended
range. Therefore, this improved solar water distillation system can also be used for
highly polluted water.

Several studies based on the application of IoT in water purification were carried
out with different IoT devices. Table 6.5 shows some of the previous studies related
to this study and IoT devices that were used in those studies along with major
findings.

Plate 6.3 Laying of a sand
layer under the solar still

Table 6.3 Air temperature, water temperature, distilled output, and improvement percentage

Treatment	Water temperature (°C)		Air temperature (°C)		Total distilled output (L/m²/day)	Improvement percentage (%)
	Max.	Average	Max.	Average		
T1	48.0	38.85	51.6	40.80	1.67	–
T2	58.6	45.50	61.6	48.10	2.87	72.9
T3	52.6	41.10	54.6	41.45	2.01	19.9
T4	49.5	48.60	55.3	42.20	2.47	48.2
T5	58.3	43.60	64.3	46.30	3.98	138.3
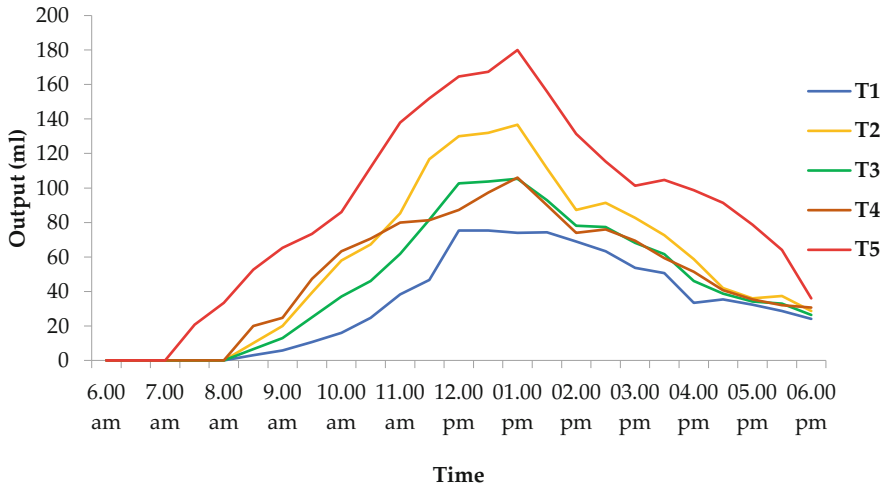
Fig. 6.8 Variation in the distilled output throughout the day

Table 6.4 pH, EC, and TDS values of the water with WHO standards

Quality parameter	Before Treatment	After treatment					Laboratory distilled water	WHO standard
		T1	T2	T3	T4	T5		
pH	7.38	6.90	7.06	7.12	7.22	7.21	7.66	6.5–8.5
EC (µS/cm)	431	8.17	7.22	6.69	6.60	7.01	1.41	Max. 750
TDS (ppm)	207.9	3.47	3.41	3.22	3.19	3.43	0.61	Max. 500

6.8 Conclusions

Solar water distillation can be used efficiently for small-scale water purification for drinking. There is considerable potential for using IoT technology in many water purification methods to save energy and increase efficiency by reducing human involvement. There is a possibility to integrate solar water distillation and solar water heating systems, as the essential components and requirements of both systems are interchangeable. In the conducted experiment, all the evaporation enhancement strategies introduced for the distillation unit contributed to increasing the distilled capacity of the system. The distilled production of the newly designed distillation unit of 0.5 m^2 increased by 138% with the combined effect of the sand layer, flat plate solar collector, and sponge cubes. The flat plate solar collector is suitable for increasing basin water temperature and basin air temperature. The calculated improvement of the flat plate solar collector over the passive solar still type was 72.9%. The river sand layer can store more heat energy during the morning period due to its high thermal capacity and release during the afternoon period. The efficiency of the distilled output increased by 48.2% when sponge cubes were

Table 6.5 Studies on IoT based water purification systems

Study	Purification method	IoT devices	Findings
IoT-based performance analysis of hybrid solar heater-double slope solar still (Benghanem et al., 2022)	Solar distillation	*Sensors*—Temperature, Humidity, Water level *Controller*—Arduino Mega 2560 *Communication*—WiFi *IoT Platform*—ThingSpeak	Distilled output—12 L/m2/day User can easily check whether the system works correctly or not
Water Purifier Quality Monitoring Using IOT (Deshmukh et al., 2020)	UV lamp along with s reverse osmosis or carbon block filters	*Sensors*—pH, Temperature, Turbidity *Controller*—Arduino *Communication*—WiFi	Water qualities are monitored real time and updated to the authorized person
An IoT based Control System for a Solar Membrane Distillation Plant used for Greenhouse Irrigation (Gil et al., 2019)	Solar membrane	*Sensors*—Temperature, Humidity, Solar radiation, CO_2 concentration *Controller*—MPC *Communication*—WiFi *IoT Platform*—Orion	Development of a model predictive control strategy in combination with an IoT platform and a smart grid framework
Cloud and IoT based smart architecture for desalination water treatment (Alshehri et al., 2021)	Dual nano membrane filtration	*Sensors*—Water level, pH, Turbidity, Conductivity *Controller*—Raspberry Pi *Communication*—WiFi	The desalination capacity was 0.47 m^3/l of freshwater from a saline concentration of 10 g/l, at 8.31 KWh/m^3

incorporated into the solar still because the area for water evaporation increased and the surface tension of the water molecules was reduced. The total average daily production of the improved solar water distillation unit was 3.98 L/m^2/day. The production increased by increasing the area of the basin according to the requirement. The distilled output has safe pH, EC, and TDS levels so it is suitable for drinking purposes.

References

Abdullah, A. S. (2013). Improving the performance of stepped solar still. *Desalination, 319*, 60–65. https://doi.org/10.1016/j.desal.2013.04.003

Abu-Hijleh, B. A., & Rababah, H. (2003). Experimental study of a solar still with sponge cubes in basin. *Energy Conversion and Management, 44*, 1411–1418.

Agrawal, V. K., & Bhalwar, R. (2009). Household water purification: Low-cost interventions. *Medical Journal Armed Forces India, 65*(3), 260–263. https://doi.org/10.1016/S0377-1237(09)80019-1

Alawee, W. H., Mohammed, S. A., Dhahad, H. A., Essa, F. A., Omara, Z. M., & Abdullah, A. S. (2021). Performance analysis of a double-slope solar still with elevated basin — Comprehensive study. *Desalination and Water Treatment, 223*, 13–25. https://doi.org/10.5004/dwt.2021.27125

Alshehri, M., Bhardwaj, A., Kumar, M., Mishra, S., & Gyani, J. (2021). Cloud and IoT based smart architecture for desalination water treatment. *Environmental Research, 195*, 110812. https://doi.org/10.1016/j.envres.2021.110812

Ansari, O., Asbik, M., Bah, A., Arbaoui, A., & Khmou, A. (2013). Desalination of the brackish water using a passive solar still with a heat energy storage system. *Desalination, 324*, 10–20. https://doi.org/10.1016/j.desal.2013.05.017

Arunkumar, T., Ao, Y., Luo, Z., Zhang, L., Li, J., Denkenberger, D., & Wang, J. (2019a). Energy efficient materials for solar water distillation - A review. *Renewable and Sustainable Energy Reviews, 115*, 109409. https://doi.org/10.1016/j.rser.2019.109409

Arunkumar, T., Raj, K., Rufuss, D. D. W., Denkenberger, D., Tingting, G., Xuan, L., & Velraj, R. (2019b). A review of e ffi cient high productivity solar stills. *Renewable and Sustainable Energy Reviews, 101*, 197–220. https://doi.org/10.1016/j.rser.2018.11.013

Aziz, S. N. S., El-Hadad, O., Rahim, S. A., & Chew, F. N. (2017). Solar still; unrevealed facts and reasons causing its low productivity. *Journal of Engineering Research, 5*(1), 181–199.

Ben Halima, H., Frikha, N., & Ben Slama, R. (2014). Numerical investigation of a simple solar still coupled to a compression heat pump. *Desalination, 337*(1), 60–66. https://doi.org/10.1016/j.desal.2014.01.010

Benghanem, M., Mellit, A., & Emad, M. (2022). IoT-based performance analysis of hybrid solar heater-double slope solar still. *Water Supply, 22*(3), 3027–3043. https://doi.org/10.2166/WS.2021.414

Deshmukh, T., Lokhande, H. N., Raj, M., & Sadegaonkar, R. (2020). Real time internet of things (IoT) based water quality. *International Journal of Innovative Research in Technology, 7*(2), 78–83.

Dev, R., Abdul-Wahab, S. A., & Tiwari, G. N. (2011). Performance study of the inverted absorber solar still with water depth and total dissolved solid. *Applied Energy, 88*(1), 252–264. https://doi.org/10.1016/j.apenergy.2010.08.001

Gil, J. D., Munoz, M., Roca, L., Rodriguez, F., & Berenguel, M. (2019). An IoT based control system for a solar membrane distillation plant used for greenhouse irrigation. In *Global IoT summit, GIoTS 2019 - Proceedings*. https://doi.org/10.1109/GIOTS.2019.8766370.

Gorchev, H. G., & Ozolins, G. (2011). WHO guidelines for drinking-water quality. *WHO Chronicle, 38*, 104–108. https://doi.org/10.1016/S1462-0758(00)00006-6

Joy, N., Antony, A., & Anderson, A. (2018). Experimental study on improving the performance of solar still using air blower. *International Journal of Ambient Energy, 39*(6), 613–616.

Kabeel, A. E., & Abdelgaied, M. (2016). Improving the performance of solar still by using PCM as a thermal storage medium under Egyptian conditions. *Desalination, 383*, 22–28. https://doi.org/10.1016/j.desal.2016.01.006

Khan, M. U., Khan, M. R., Hossain, B., & Ahmed, Q. S. (1984). Alum potash in water to prevent cholera. *The Lancet, 3*(4), 1032. https://doi.org/10.1177/004051756203200411

Manokar, A. M., Winston, D. P., Kabeel, A. E., & Sathyamurthy, R. (2018). Sustainable fresh water and power production by integrating PV panel in inclined solar still. *Journal of Cleaner Production, 172*, 2711–2719. https://doi.org/10.1016/j.jclepro.2017.11.140

Mazille, F. (2020). *Disaster situations: Planing and preparedness, water purification, ion exchange*. Retrieved June 22, 2022, from https://sswm.info/sswm-university-course/module-6-disaster-situations-planning-and-preparedness/further-resources-0/ion-exchange

Modi, K. V., & Modi, J. G. (2019). Performance of single-slope double-basin solar stills with small pile of wick materials. *Applied Thermal Engineering, 149*, 723–730. https://doi.org/10.1016/j.applthermaleng.2018.12.071

Murugavel, K. K., Sivakumar, S., Ahamed, J. R., Chockalingam, K. K. S. K., & Srithar, K. (2010). Single basin double slope solar still with minimum basin depth and energy storing materials. *Applied Energy, 87*(2), 514–523. https://doi.org/10.1016/j.apenergy.2009.07.023

Omara, Z. M., Kabeel, A. E., & Essa, F. A. (2015). Effect of using nanofluids and providing vacuum on the yield of corrugated wick solar still. *Energy Conversion and Management, 103*, 965–972. https://doi.org/10.1016/j.enconman.2015.07.035

Pal, P., Yadav, P., Dev, R., & Singh, D. (2017). Performance analysis of modified basin type double slope multi–wick solar still. *Desalination, 422*, 68–82. https://doi.org/10.1016/j.desal.2017.08.009

Refalo, P., Ghirlando, R., & Abela, S. (2016). The use of a solar chimney and condensers to enhance the productivity of a solar still. *Desalination and Water Treatment, 57*, 23024–23037. https://doi.org/10.1016/j.enconman.2014.05.021

Sanserwal, M., Kumar Singh, A., & Singh, P. (2020). Impact of materials and economic analysis of single slope single basin passive solar still: A review. *Materials Today: Proceedings, 21*, 1643–1652. https://doi.org/10.1016/j.matpr.2019.11.289

Singh, M., & Ahmed, S. (2020). IoT based smart water management systems: A systematic review. *Materials Today: Proceedings, 46*, 5211–5218. https://doi.org/10.1016/j.matpr.2020.08.588

Tiwari, G. N., Singh, H. N., & Tripathi, R. (2003). Present status of solar distillation. *Solar Energy, 75*(5), 367–373. https://doi.org/10.1016/j.solener.2003.07.005

United Nations. (2020). *The sustainable development goals report 2020. United Nations Statics Division*. Retrieved from https://unstats.un.org/sdgs

WaterProfessionals. (2022). *Water purification and chlorination*. Retrieved June 22, 2022, from https://www.waterprofessionals.com/learning-center/chlorination/

WHO. (2022). *Drinking-water*. Retrieved June 23, 2022, from https://www.who.int/news-room/fact-sheets/detail/drinking-water

Chapter 7
Long Range Wide Area Network (LoRaWAN) for Oil Palm Soil Monitoring

Yee Nie Goh, Diyana Jamaludin, Hazreen Haizi Harith, Alfadhl Yahya Alkhaled, Nurul Adilah Abdul Latiff, and Samsuzana Abd Aziz

Abstract As the global agricultural sector gradually moves towards industry 4.0, the implementation of the Internet of Things (IoT) in precision agriculture allows farmers to improve the overall management with real-time monitoring. With wireless sensor networks (WSN), information regarding variability of soil electrical conductivity (EC) and pH that crucially affect nutrient uptake are easily monitored. However, the challenge of adopting these technologies in agriculture is the need for a scalable and wide coverage wireless communication system. IoT implementations in agriculture are restricted by limited network coverage, for example, in oil palm estates that are vast and usually located in rural areas with limited internet access. This study aims to propose a system design based on long range wide area network (LoRaWAN) communication protocol to monitor soil EC and pH in an oil palm nursery. The template for LoRaWAN network is laid out in four parts; sensor node, gateway, network server, and application server. LoRaWAN is perfect for outlying regions without cellular network coverage or for establishing private networks covering long distances with minimum power consumption and maintenance. It provides long range communication of small amounts of data with high immunity to interference under low-power consumption. In this chapter, LoRaWAN is implemented as the transmission protocol that meets the need of the IoT services in oil palm nurseries to monitor soil EC and pH of growing mediums. A comparison of LoRaWAN performance in a young oil palm plantation, an oil palm nursery, an urban area and a greenhouse nursery shows that strong LoRaWAN transmission is

Y. N. Goh · D. Jamaludin · H. H. Harith · S. Abd Aziz (✉)
Smart Farming Technology Research Center, Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia
e-mail: samsuzana@upm.edu.my

A. Y. Alkhaled
Department of Horticulture, College of Agricultural and Life Sciences, University of Wisconsin – Madison, Madison, WI, USA

N. A. A. Latiff
Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, Kuala Terengganu, Terengganu, Malaysia

97

achievable with higher gateway elevation and clear line of sight. LoRa transmission successfully covered the entire oil palm nursery area of 6 hectare (ha) and reached up to more than 1 km distance in the urban area. For wider areas, this setup can be duplicated or scaled up. Both young oil palm plantation and nursery achieved strong LoRa transmission with received signal strength indication (RSSI) close to −100 and signal-to-noise ratio (SNR) values that are considered very minimally corrupted. From this study, the feasibility of LoRaWAN as a network protocol to be used in oil palm nurseries for soil EC and pH monitoring is proven to be successful.

Keywords Internet of Things (IoT) · Precision agriculture · Wireless sensor networks (WSN) · Long Range Wide Area Network (LoRaWAN) · Oil palm · Soil monitoring

7.1   Introduction

The agricultural sector particularly in Asia is gradually improving as farmers are encouraged to use modern techniques to increase yield and cut down on input cost. Precision agriculture is a farming management concept that uses technology to provide site specific care by observing, measuring and responding to inter and intra-field variability in crops. This approach applies information technology (IT) to ensure crops and soil receive all they need for optimum health and productivity. Furthermore, the use of modern technology not only increases crop productivity, but also helps in saving resources while keeping the environment clean and safe (Pierce & Nowak, 1999). Precision agriculture includes monitoring crops from various sensor nodes placed in agricultural fields using Internet of Things (IoT) and wireless sensor networks (WSN) to provide solutions based on collected data (Mohd Kassim et al., 2014).

Particularly in Malaysia and Indonesia, as the oil palm is the main contributing crop to the economy, efforts have been focused to modernise the farming techniques for higher productivity. In oil palm nurseries, monitoring electrical conductivity (EC) and pH levels of growing mediums are essential to provide crops with an optimum quantity of water and fertilisers based on their requirement to produce high quality seedlings. However, current fertigation systems in oil palm nurseries lack real-time crop requirement information as systems are monitored and controlled manually. In large oil palm nursery estates, the use of IoT is challenging as the coverage areas are vast and they are located in rural areas where internet access is limited and expensive, hence, making it less feasible for adoption. Long range wide area network (LoRaWAN) offers a very compelling combination of long range, low-power consumption, and secure data transmission. It provides the ability for sensors to connect at a range relatively further than other types of wireless sensor networks. LoRaWAN is expected to provide long range communication in large and outskirts oil palm nursery areas. Parameter data collected from sensors in growing mediums with the help of a LoRaWAN gateway are stored in the cloud which could be utilised for further analysis and automation to reduce labour while increasing

productivity. This chapter aims to provide real-time crop monitoring of EC and pH of growing mediums that is accessible through the cloud. The specific objectives were (1) to develop a soil EC and pH monitoring system based on LoRaWAN communication protocol, (2) to test the feasibility of LoRaWAN to transmit soil EC and pH data in different oil palms setups, and (3) to monitor and collect EC and pH data in growing medium of oil palm seedlings.

7.2 Internet of Things (IoT) in Agriculture

IoT refers to sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems over the Internet or other communication networks, which is a crucial component of digital advancement in precision agriculture. An IoT system consists of several functional blocks to facilitate various utilities to the system such as, sensing, identification, actuation, communication, and management. Agricultural IoT framework is a six-layered concept which includes hardware facilities, internet and communication technologies, IoT middleware, cloud services, big data analytics, and farmer experience (Ray, 2017) (Fig. 7.1).

Some of the domains of IoT for agriculture are farm monitoring, irrigation management system, pest and disease control, and animal tracking (Ray, 2017). As crop requirement varies throughout the field, real-time ecological information gathered remotely from agricultural surroundings provides precise input of location, timing, and amount required.

7.3 Wireless Sensor Network in Agriculture

Suitable communication technology for data transmission is important to ensure successful implementation of IoT in agriculture. WSNs, which is the network of sensors and actuators, are employed as a cost-effective application to increase agricultural yield. Sensor nodes communicate wirelessly through a communication protocol, forwarding data to an application server through a gateway. As more sensor nodes are added to WSN, parameters of agricultural monitoring and network scalability increase. Communication protocols form the backbone of IoT systems that enable network connectivity and coupling to applications. However, the limitations of most wireless communication technologies are high power consumption and short transmission distance that fail to meet the requirement of WSN applications in agriculture (Brewster et al., 2017).

To overcome this barrier, a low-power wide area network (LPWAN) is the recent solution in the communication sector. Three representative technologies of LPWAN that compete for large-scale IoT are long range (LoRa), narrow-band Internet of Things (NB-IoT), and Sigfox. In agriculture, devices only update sensed data a few

		Physical Layer
• Sensors (soil moisture, water level) • Actuators (sprinklers, water pump)	• Microcontroller (datalogger) • Network Equipment (gateway)	Physical Layer

		Network Layer
• WiFi • LTE	• GSM • LoRa	Network Layer

		Middleware Layer
• Database • Message-oriented	• Application Server • Transaction	Middleware Layer

IoT Server

		Service Layer
• Machine Identification • Field and Crop Management • Pesticide Control • Livestock Management	• Notification Service • Statistical Analysis • Crop Disease Information • Video Surveillance	Service Layer

Big Data Analytics

Dynamic Analysis

		Analytics Layer
• Biotechnology • Horticulture	• Integrated Farming • Tissue Culture	

Predictive Analysis

• Crop Yield Analysis • Pest, Disease, Weed Detection	• Weather Prediction • Logistics	

Farmer Experience

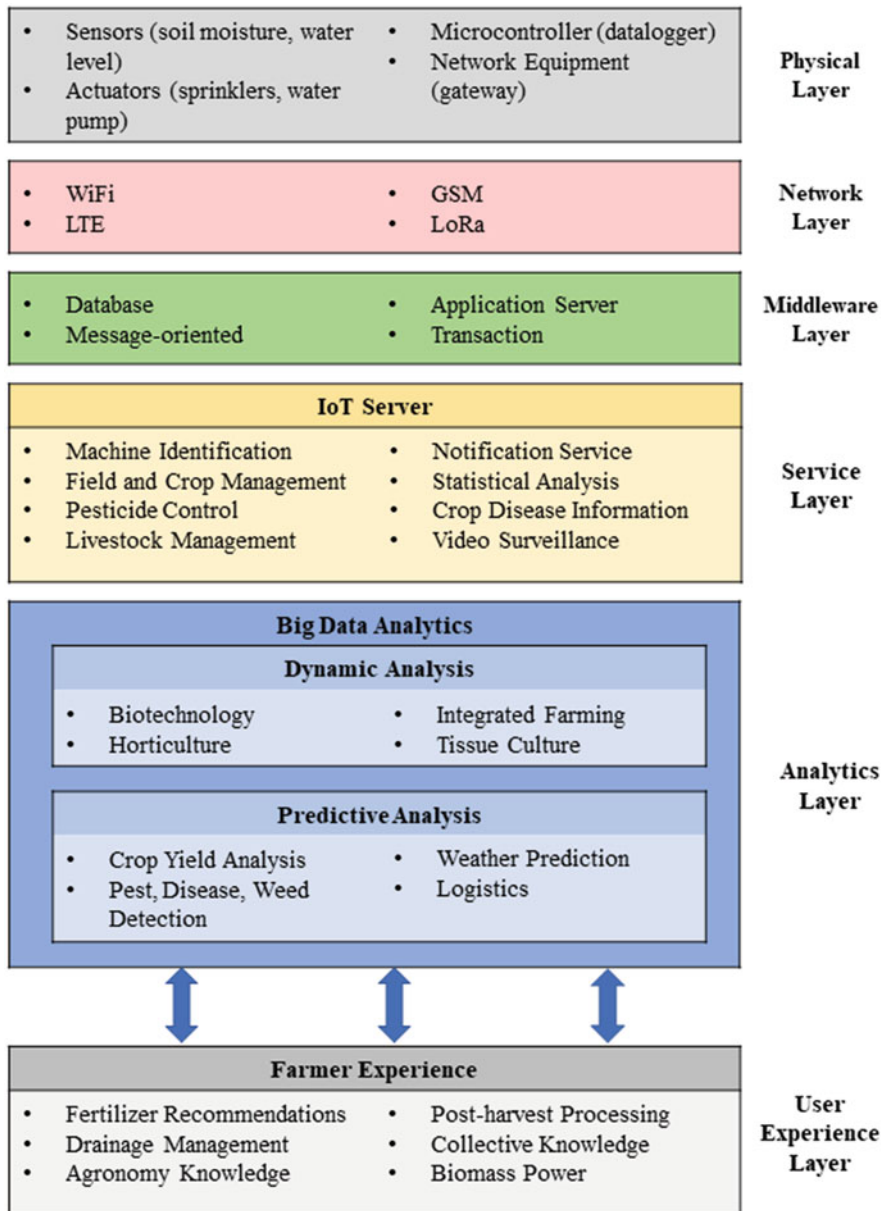|		User Experience Layer
• Fertilizer Recommendations • Drainage Management • Agronomy Knowledge	• Post-harvest Processing • Collective Knowledge • Biomass Power	User Experience Layer

Fig. 7.1 IoT architecture framework in agriculture

times per hour as the environmental conditions do not radically change. LPWAN does not focus on enabling high data rates per device; rather, the key performance metrics defined for LPWAN are energy efficiency, scalability, and coverage where

Sigfox and LoRa are ideal for this application. NB-IoT is the least favourable solution to use in agriculture, as it requires cellular coverage which many farms do not have (Mekki et al., 2019).

Sigfox has a slightly longer network coverage and range than LoRa, however, LoRa performs better in battery life and latency (Jawad et al., 2017). Besides, LoRa technology operates in the unlicensed frequency band, where end users are free to set up LoRa gateways that are similar to house-owned WiFi routers (Song et al., 2017). Therefore, LoRa technology is perfect for outlying regions without cellular network coverage or for establishing private networks covering long distances with minimum power consumption and maintenance.

7.4 Soil Electrical Conductivity (EC) and pH in Oil Palm

Oil palm (*Elaeis guineensis*) is one of the most important crops used for biodiesel production and is a major cash crop income for ASEAN countries such as Indonesia and Malaysia. To ensure optimum growth and high yield of oil palm, balanced and adequate supply of fertilisers are required. Since fertiliser recommendations are based on calibrated soil and crop requirement, therefore it is essential to monitor the variability of EC and pH level in soil (Behera et al., 2017). Fertigation systems provide balanced nutrition level to crop by monitoring and modifying pH and EC level of water mixed fertilisers with respect to soil parameters (Samsuri et al., 2010).

EC is a measure of the total amount of salts in the growing medium. High soil salinity restricts the crop's ability to uptake water from soil while high EC levels have a toxic effect on plant's metabolism (Hanlon & Bartos, 1993). On the other hand, pH is a measure of acidity or alkalinity that affects the availability of macro and micronutrients in the growing medium (McCauley et al., 2009). Almost all crops prefer a slightly acidic pH between 5.4 and 6.0. Soil EC values less than 1000 mhos/ cm and pH values more than 4.0 are categorised in the desirable range of soil properties for oil palms (Goh et al., 2016). Furthermore, pH values above 5.5 correlate to high soil fertility for oil palm cultivation (Mutert, 1999). Thus, regular monitoring of EC and pH of growing medium in oil palm nursery is essential to prevent almost all nutrient-related problems during oil palm growth.

7.5 LoRaWAN System Design for Soil EC and pH Monitoring

LoRaWAN defines the medium access control (MAC) communication protocol and the system architecture for the WSN network, standardised by the LoRa Alliance (Fremont, CA). The template for LoRaWAN network is laid out in four parts; (1) nodes, (2) gateways, (3) network servers, and (iv) application servers. In this
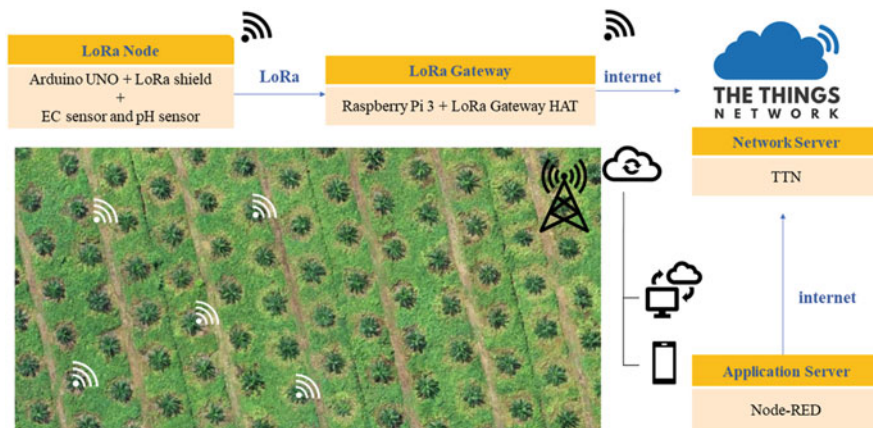
Fig. 7.2 LoRaWAN network architecture for soil EC and pH monitoring in oil palm plantation
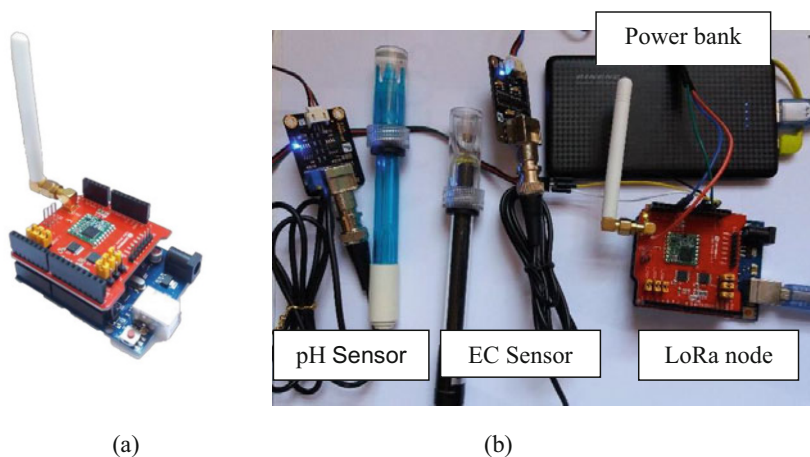
Fig. 7.3 (**a**) A LoRa node and (**b**) sensors connected to a LoRa node

study, a soil EC and pH monitoring system was proposed and developed based on LoRaWAN architecture as shown in Fig. 7.2. In the time of development, this setup was built on an Arduino based node and a single channel gateway in The Things Network (TTN) V2.

A LoRaWAN node is an endpoint physical hardware device that contains sensing capabilities, some computing power and a radio module for translating data into a radio signal. These end devices can send data to the gateway and also receive data via LoRa wireless protocol. The connection between a node and a gateway is bi-directional with very low bandwidth, between 0.3 and 50 kbps. The hardware components (Fig. 7.3) to setup a prototype node module in this study are as follows:

1. Arduino UNO Rev3
2. Dragino LoRa Shield 915 Mhz
3. DFRobot Analog Electrical Conductivity (EC) Sensor
4. DFRobot Analog pH Sensor

To build the LoRa node, LoRa shield is stacked onto Arduino UNO with an antenna installed (Fig. 7.3a). An EC sensor is connected at analog pin A1 and pH sensor at analog pin A0 of the Arduino UNO at the node. The LoRa node firmware is set up by importing LoraWAN-MAC-in-C (LMIC) and EC sensor libraries into the complete node sketch in Arduino IDE software.

The flow chart in Fig. 7.4 elaborates the algorithm flow at the node. Initial settings like pin assignments for both sensors, baud rate and frequency are set according to a standard requirement. The frequency for this system is set to 921.2 Mhz which is within the allowable unlicensed frequency band for LoRa in Malaysia; from the range of 919 to 923 Mhz. Since the algorithm uses serial communication, availability of serial pins and acknowledgement from the gateway is checked to indicate successful communication between the LoRa node and the gateway.

The LoRa packet messages received from nodes are forwarded by a physical machine called gateway to the network server via the internet. These messages are converted to an array of bits that can be sent via the traditional internet protocol (IP) networks. The gateway is connected to The Things Network (TTN), a global collaborative IoT ecosystem that creates networks, devices and solutions using LoRaWAN to which it transmits all the messages. The hardware components to setup the single channel gateway module are as follow:

1. Raspberry Pi 3 Model B+
2. Dragino LoRa GPS HAT 915Mhz

To build the LoRaWAN gateway, LoRa HAT is stacked onto the Raspberry Pi with the antenna installed as shown in Fig. 7.5. The Raspberry Pi runs on Raspbian, the free operating system based on Debian, optimised for the Raspberry Pi hardware. The software for the single channel LoRa gateway is cloned from GitHub (zakibakar75/single_chan_pkt_fwd). The pin configuration in globalconf.json file is set as so; "pin_nss": 6, "pin_dio0": 7, "pin_rst": 0. Then, the server is defined as 13.66.213.36 (router.us.thethings.network), spreading factor is set to SF7 and the frequency is configured to 921.2 MHz in the main.cpp file. After the code is compiled, the gateway is executed to obtain Gateway ID: b8:27:eb:ff:ff:c8:55:a1.

Figure 7.6 elaborates the algorithm flow at the server side from receiving broadcast messages and sending back acknowledgement to the same LoRa node. The baud rate and frequency are set with the application programming interface (API) key which is exposed by the handler selected; ttn-handler-brazil. Data is checked continuously as it is displayed in the TTN application data window.

The TTN ecosystem provides the network for IoT devices based on LoRaWAN standards. It is based on an open community of developers, where LoRa gateways and applications are registered for free. The TTN Server receives data packets from the LoRa Gateway stored temporarily in TTN and reachable via Internet. The

Fig. 7.4 Flowchart for LoRa node communication set up
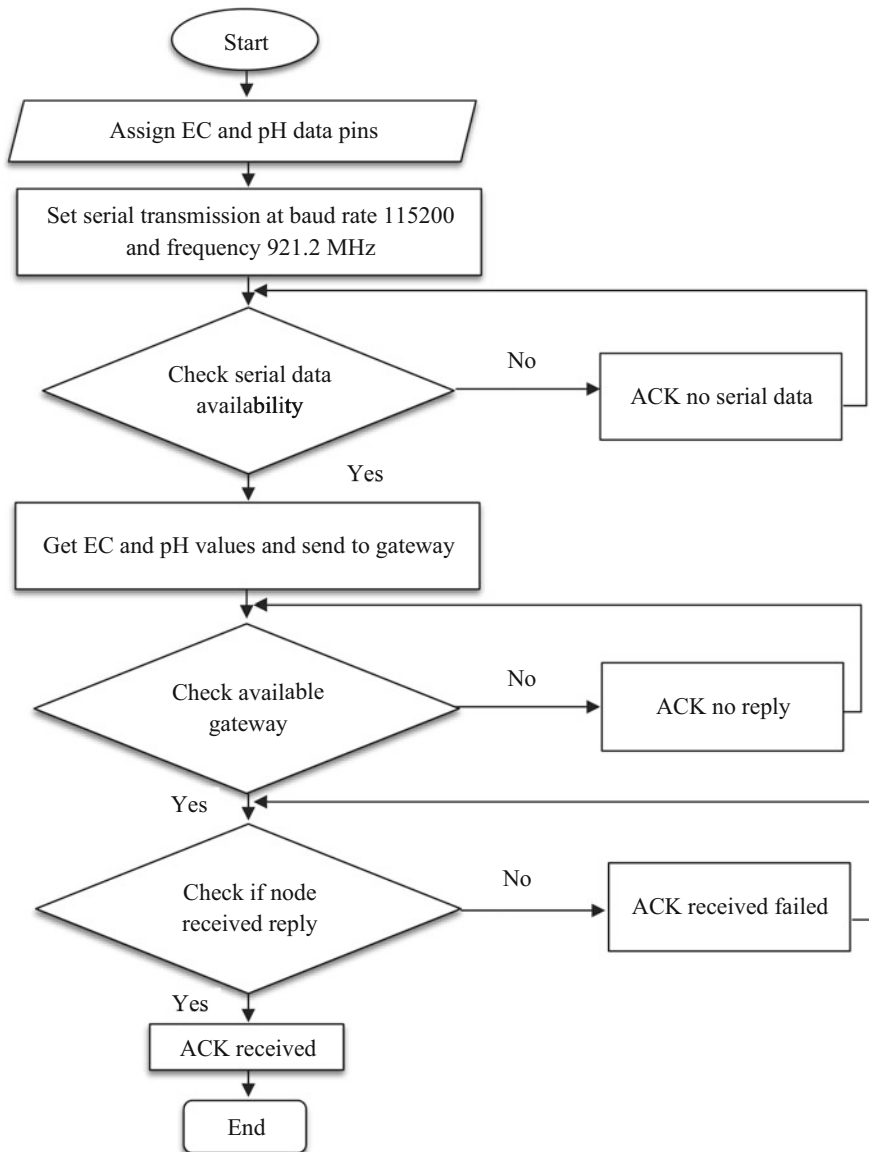
network servers can be used for both uplink (i.e., sensor to application) or downlink (i.e., application to sensor) communication.

A user account is registered at https://www.thethingsnetwork.org. To set up the node in TTN, an application is created, and the device is registered under activation by personalisation (ABP). The two session keys generated for this device, network

Fig. 7.5 LoRa gateway
with an antenna

session key and application session Key together with the device address are then hardcoded into the complete node sketch. To set up the gateway in TTN, the Gateway ID obtained previously from the gateway module is filled at gateway Extended Unique Identifier (EUI) (a 16-digit alphanumeric code that uniquely identifies a gateway in a LoRaWAN network) of the new registered gateway. The frequency of 921.2 MHz is set and "ttn-router-asia-se" is selected. Upon successful node and gateway registration, TTN data displays received packets from the LoRa node in the application data window. The payload shown in raw bytes in the application data window is decoded to display in ASCII format (Fig. 7.7). The raw byte data is taken and then a string that contains all of the characters corresponding to each byte is returned.

The LoRaWAN network server is then interfaced with the application server, in which application-specific processes are performed on cloud. Application server is where the data collected from the end devices, are available to browsers for viewing, manipulated and sent back out to nodes over the network server. The last segment of the system is the visualisation of received data using Node-RED, which is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs, and online services as part of the IoT. The Node-RED has extensions to get data from TTN and functions for dashboard user interface (UI). Figure 7.8 shows the browser-based flow editor for board voltage, board temperature, soil EC, and soil pH data obtained from TTN. Data from TTN are downloaded, modified into charts and gauges to be displayed in the dashboard UI.

The visualisation of the received data sensed by the node module is displayed in the dashboard UI. Figure 7.9 shows the results obtained from the LoRa node as EC and pH sensors are placed into the soil. The soil EC and pH readings of growing medium are obtained in real time. From this dashboard, users can easily track and monitor their crop condition to make better decisions for further controls.
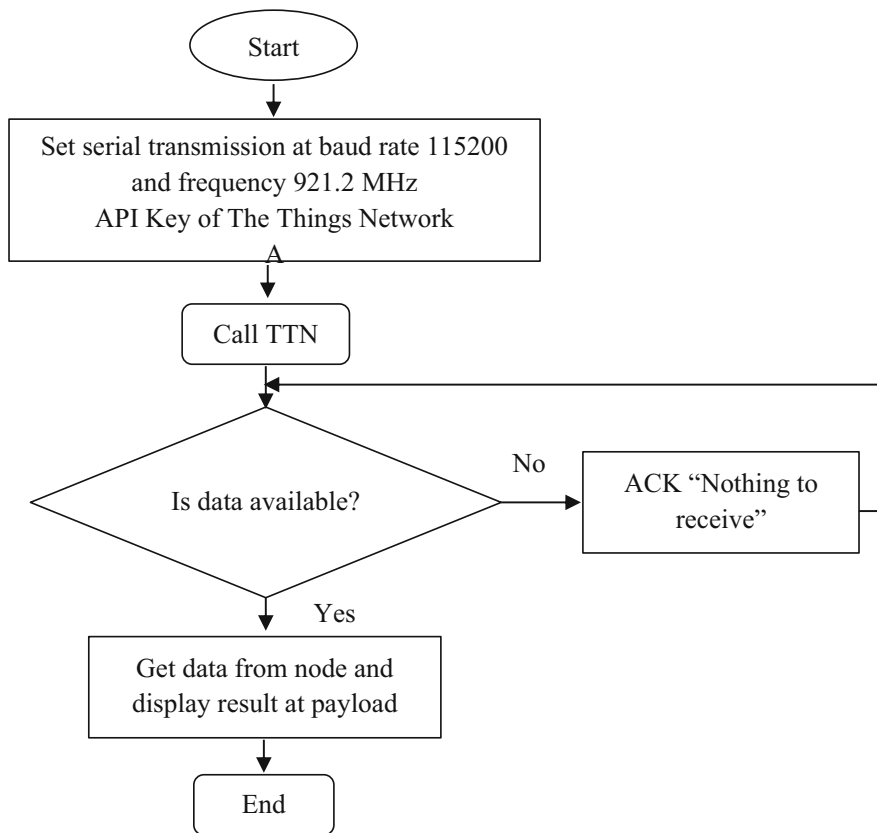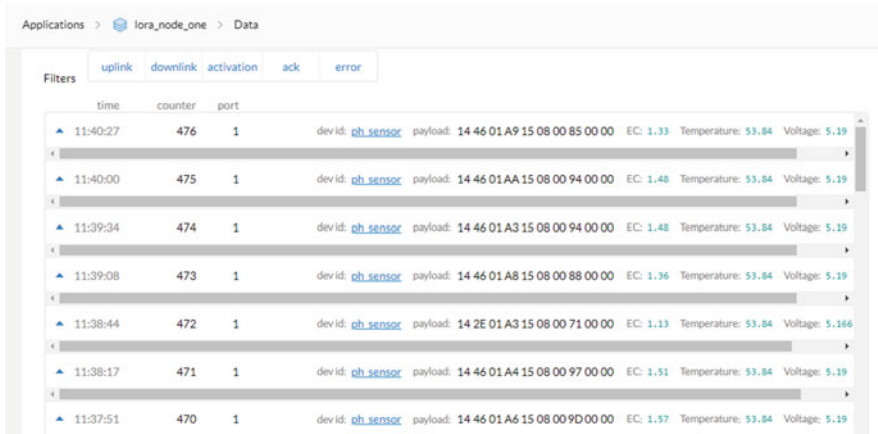
Fig. 7.6 Flow chart for receiving broadcast messages and sending back acknowledgement to the LoRa node

7.6   Signal Propagation Tests

A series of small-scale deployments was carried out to test the signal transmission feasibility of the developed LoRaWAN system within different environmental setups. The tests aimed to measure and compare the signal propagation performance through some foliage environments; a young oil palm plantation (Fig. 7.10), an oil palm nursery (Fig. 7.13), and an urban area (Fig. 7.15). All tests are carried out in Selangor, Malaysia. The gateway and the end-node device hardware of the system are battery powered. Hence, this portability allowed the experiment to be tested in various kinds of environments. From these experiments, relevant signal quality parameters such as the received signal strength indication (RSSI) and signal to noise (SNR) values together with the distance of each node from the gateway are extracted.

Fig. 7.7 Screenshot of TTN V2 payload data
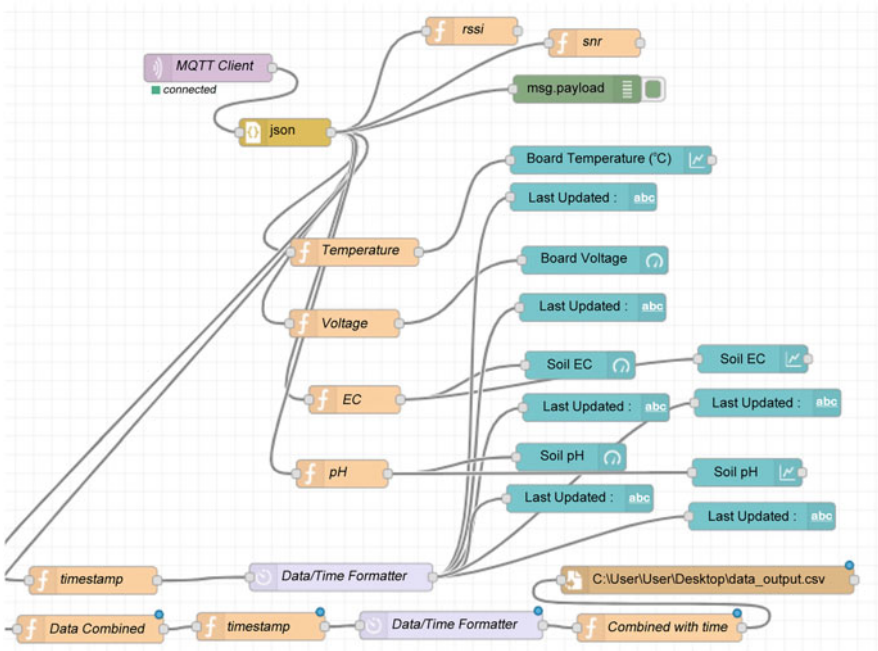
Fig. 7.8 Node-RED browser-based flow editor

EC   pH
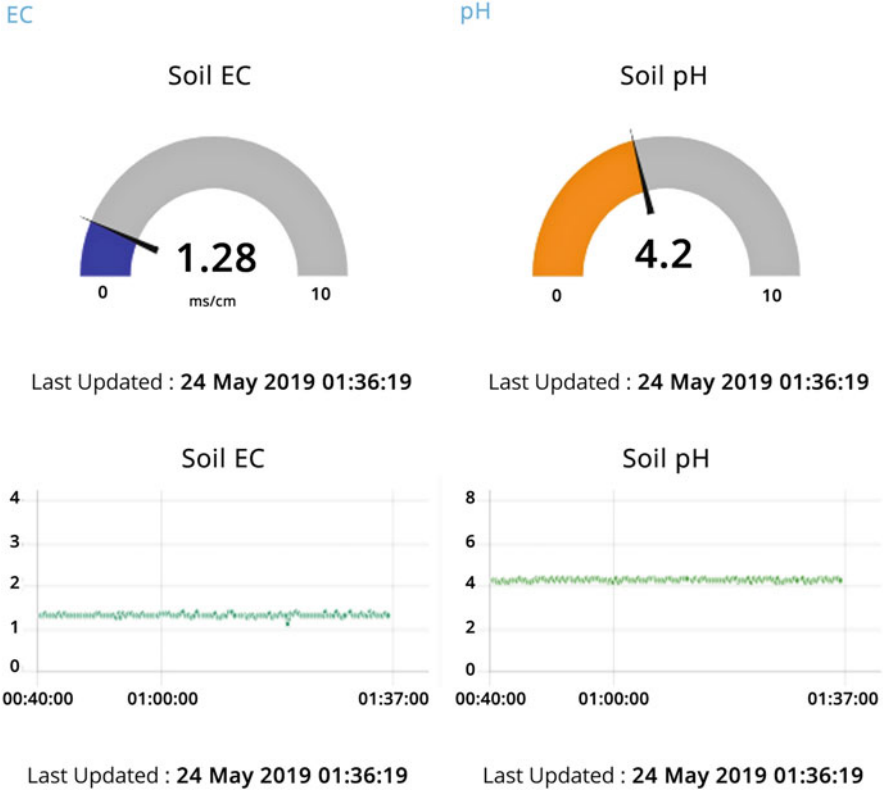
Fig. 7.9 Node-RED dashboard for soil EC and pH readings

7.6.1 Signal Propagation Test in a Young Oil Palm Plantation

The test area was at Universiti Putra Malaysia (UPM) Oil Palm Plantation, with a total area of four hectares. In total 850 7-year-old, around 5 m tall oil palm trees are planted in this block. The gateway is stationary and placed at 2.988439, 101.723713 on a 7 m high building. The ground elevation of the building is 81 m, hence the total elevation inclusive of building height is 88 m. The test is carried out by positioning the end-node device at different distances to test the maximum range of signal received.

The RSSI is the received signal power in milliwatts and is measured in dBm. This value is used as a measurement of how well the receiver (gateway) can "hear" a signal from the sender (node). RSSI values are negative and the closer it is to zero, the stronger the signal. For LoRa transmission, RSSI values from -30 dBm to -120 dBm are considered strong while values below -120 dBm are categorised as weak signals. On the other hand, SNR is the ratio between the received power signal

(a)   (b)

(c)
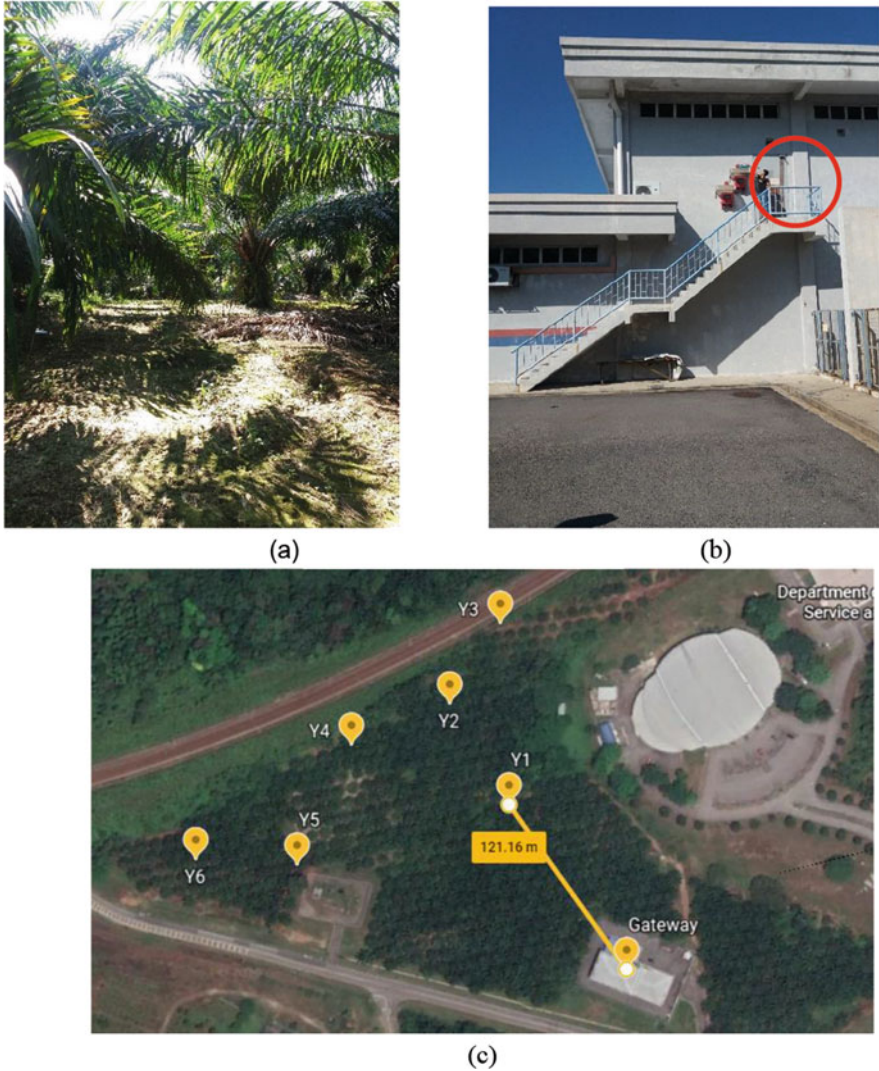
Fig. 7.10 (**a**) Environment in a young oil palm plantation with oil palms around five metres tall, (**b**) the LoRa gateway placement, and (**c**) the nodes and gateway locations in the oil palm plantation

and the noise floor power level. The noise floor is an area of all unwanted interfering signal sources which can corrupt the transmitted signal and therefore retransmission will occur. Normally the noise floor is the physical limit of sensitivity, however, LoRa works below the noise level. Typical LoRa SNR values are between -20 dB and $+10$ dB. Signals closer to $+10$ dB indicates that the received signal is less corrupted. LoRa can demodulate signals which are -7.5 dB to -20 dB below the noise floor. In all measurements (Fig. 7.11), the red colour points at zero does not
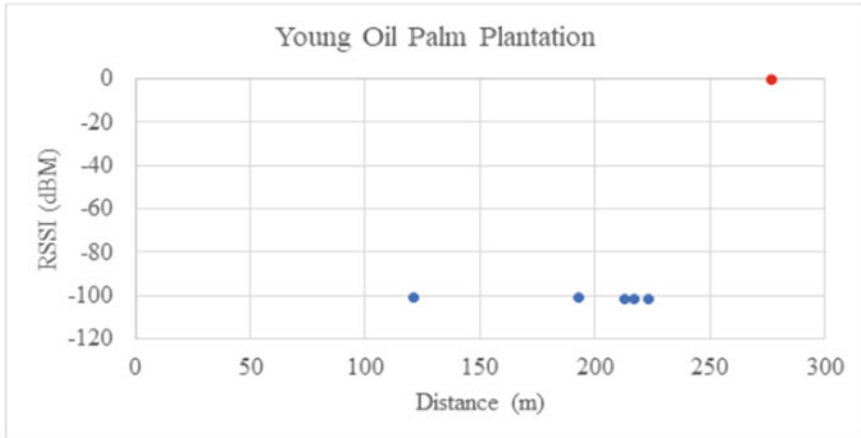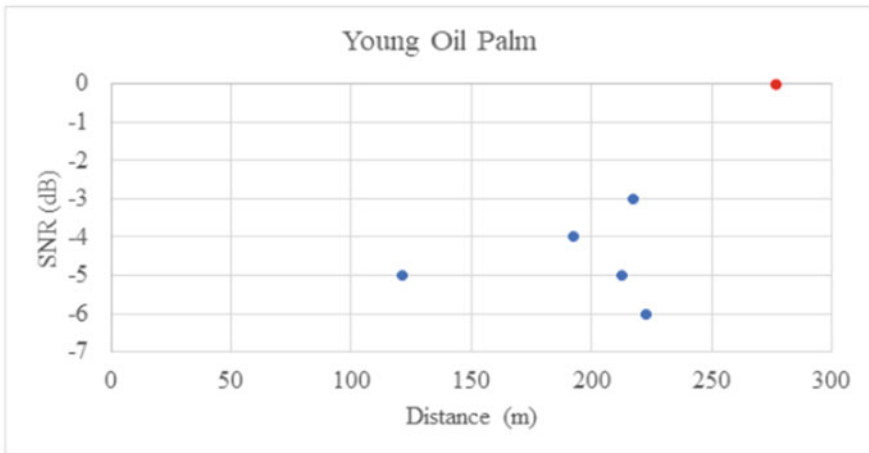
Fig. 7.11 (a) Received Signal Strength Indication (RSSI) and (b) Signal to noise ratio (SNR) in a young oil palm plantation

indicate the value zero but rather an indication of signal loss at that specific point location.

To effectively cover long ranges and achieve a good link budget, it is important to establish a direct line of sight between the transmitter and receiver as often as possible. In radio transmission, specific spatial areas in between the line of sight are referred to as Fresnel zones (Fig. 7.12). If there are objects in these zones, they can have a negative influence on the wave propagation, even when there is visual contact between the transmitting and receiving antennas. For each object located in
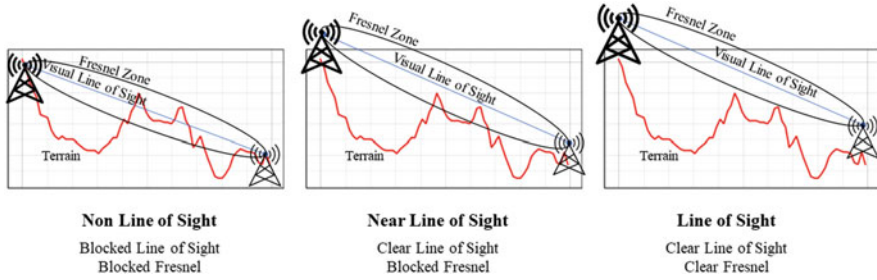
Fig. 7.12 Fresnel zone between transmitter and receiver

Table 7.1 LoRa transmission result in young oil palm plantation

Point	Coordinates	Distance from gateway (m)	Height (m)			RSSI (dBm)	SNR (dB)
			Ground Elevation (m)	Approximate Surrounding Tree Height (m)	Total Height (m)		
Y1	2.989326, 101.723095	121.16	76	5	81	−101	−5
Y2	2.989889, 101.722742	192.58	67	5	72	−101	−4
Y3	2.990331, 101.723019	223.04	66	5	71	−102	−6
Y4	2.989674, 101.722206	217.27	60	5	65	−102	−3
Y5	2.989009, 101.721890	212.98	59	5	64	−102	−5
Y6	2.989030, 101.721305	276.51	53	5	58	–	–

the Fresnel zone, the signal level is lowered, and the range is reduced (Jebril et al., 2018).

Based on the results for LoRa transmission in the young oil palm plantation, the RSSI values were almost constant as distance increased. Five of the points were in the range of strong LoRa signals except for point Y6 which has packet loss (Table 7.1). It is observed that there is a large variation in topography as ground elevation ranges from 81 m at the gateway to 53 m at where point Y6 is located. Since Y6 is at a lower elevation, the signal is disrupted before reaching the gateway as the signal must pass through a wide canopy and fronds of oil palm trees that were of similar height to the gateway. Thus, there is no clear line of sight between the node and the gateway. The area was secluded and the foliage attenuated LoRa signal propagation quite heavily. Signals that propagate through the oil palm trees are greatly affected by the distributed wide fronds that cause attenuation, scattering, diffraction and absorption of the radiated propagating waves (Ahmad et al., 2018).

In the experiment, the gateway placement was limited to a 7 m building which was the tallest building in that area. For LoRa transmission to reach further areas in

the young oil palm plantation, the gateway should be placed at a higher elevation for direct line of sight. The furthest point tested that received signal was as far as 223.04 m. The SNR values for all five points had negative values showing that noise level is higher than signal level throughout the transmission but is still within the range of a good signal. The transmission is affected by the nature of the plantation environment where large oil palm fronds attenuate LoRa signals.

7.6.2   Signal Propagation Test in an Oil Palm Nursery

The next test area was at Malaysian Palm Oil Board (MPOB) Nursery, Bangi with a total area of six hectares (Fig. 7.13). The oil palm seedlings are about 12–18 months old, ranging from 1–2 metres tall. The gateway is stationary and placed at 2.928454, 101.765041 on a 2.5 m high hut. The ground elevation of the hut is 22 m, thus the total elevation inclusive of hut height is 24.5 m. The test is carried out by positioning the end-node device at different distances to test the maximum range of the signal transmission.

In the oil palm nursery, typically there were no tall buildings and the seedlings had an average height of one to two metres as shown in Fig. 7.13a. As there were no attenuation factors, LoRa transmission at every node point was successful without packet loss, covering the whole area of six hectares (Table 7.2). For these distances, even with small obstacles like small storage buildings and oil palm seedlings, the transmissions were successful. In this nursery, communications with distance up to 261.01 m can easily be achieved even when the gateway is placed on a 2.5 m hut. However, within a similar distance, at 276.51 m of point Y6 in the young oil palm plantation, there were no signals received. This strongly proves that a clear and direct line of sight is required for successful LoRa transmission.

Based on the RSSI vs. distance graph in Fig. 7.14, as distance increases, RSSI gradually decreases from −91 to −104 dBm. All node points have positive SNR with higher received power signal than noise except for point N6 having negative SNR value indicating signal being lower than noise. This probably could be caused by the storage building at N1 which is directly in the line of sight between N6 and the gateway. Even at N4 there were few rows of 7 m high oil palm trees, the signal was still good because there was a clear line of sight as trees were planted sparsely with ample space for penetration. From this experiment, the feasibility of LoRaWAN as a network protocol is justified as the whole oil palm nursery area was successfully covered with strong LoRa transmission.

7.6.3   Signal Propagation Test in an Urban Area

In this test, the signal propagation performance is carried out in the urban area surrounded by tall buildings, houses and trees around UPM campus and Sri Serdang
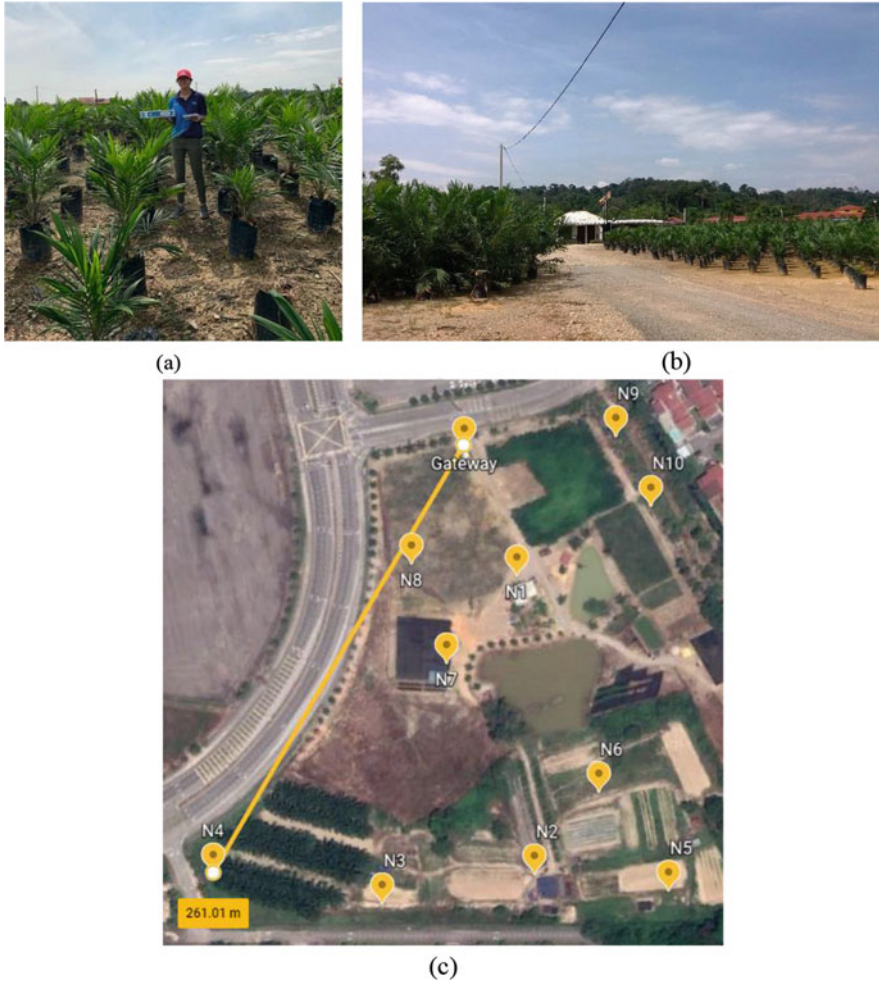
Fig. 7.13 (**a**) Environment in the Malaysian Palm Oil Board (MPOB) oil palm nursery, Bangi, (**b**) the LoRA gateway placement at a station building, and (**c**) the nodes and gateway location in the oil palm nursery

town with an estimated total area of 225 hectares (Fig. 7.15). The gateway is stationary and placed at 3.008647, 101.721345 on a 50 m (11 storeys) high building. The ground elevation of the building is 46 m, thus the total elevation inclusive of building height is 96 m above sea level. The test is carried out by positioning the end-node device at different distances to test the maximum range of signal transmission.

In the urban area, the gateway was placed at a much higher elevation than in both the young oil palm plantation and nursery as it was easier to get access to higher buildings. The gateway was placed indoors beside a window on the 11th floor at

Table 7.2 LoRa signal transmission results in oil palm nursery

Point	Coordinates	Distance from Gateway (m)	Height (m) Ground Elevation (m)	Approximate Surrounding Tree Height (m)	Total Height (m)	RSSI (dBm)	SNR (dB)
N1	2.9278347, 101.76529435	73.62	22	1	23	−91	8
N2	2.92640933, 101.76537634	227.58	23	1	24	−100	4
N3	2.92627145, 101.76467087	241.57	23	1	24	−96	6
N4	2.92642459, 101.76386446	261.01	21	7	28	−99	4
N5	2.92633962, 101.7660182	256.75	21	1	22	−95	7
N6	2.92681377, 101.76570588	194.66	23	1	24	−104	−3
N7	2.92740759, 101.76496567	115.84	23	1	24	−96	6
N8	2.92789339, 101.76479844	67.68	23	1	24	−93	8
N9	2.92850714, 101.76577265	80.36	20	2	22	−98	5
N10	2.92815334, 101.76593863	103.77	20	1	21	−101	2

about 50 m high. Therefore, LoRa transmission successfully reached up to 1.1 km in a city neighbourhood surrounded by trees, houses and some tall buildings that were made of a variation of materials. At point U6, though it was just 646.66 m away from the gateway, the signal was blocked by a seven storey UPM Engineering Faculty building. Point U3 had a similar distance of 1.1 km as U9, however, had signal loss due to the dense housing area surrounding U3. Furthermore, coverage radius was limited as the gateway was placed indoors. Almost all points transmitted data at negative SNR with RSSI lower than −100 dBm (Fig. 7.16) indicating there was higher interference in the transmission (Table 7.3).

The attenuation of signal happens due to it having to pass through or around several materials harder than wood (Rudd et al., 2014). Thus, in an urban environment, there is a heavy dependency on the placement of both gateway and end-node as a single house can have a large effect on the signal quality. This problem can be avoided by placing the gateway at higher elevations to reduce path loss by going around objects rather than through them. Height has a great impact on performance when there is free line of sight and the Fresnel zone is clear of obstacles. By applying this elevation in the young oil palm plantation and oil palm nursery, performance would very likely improve drastically.
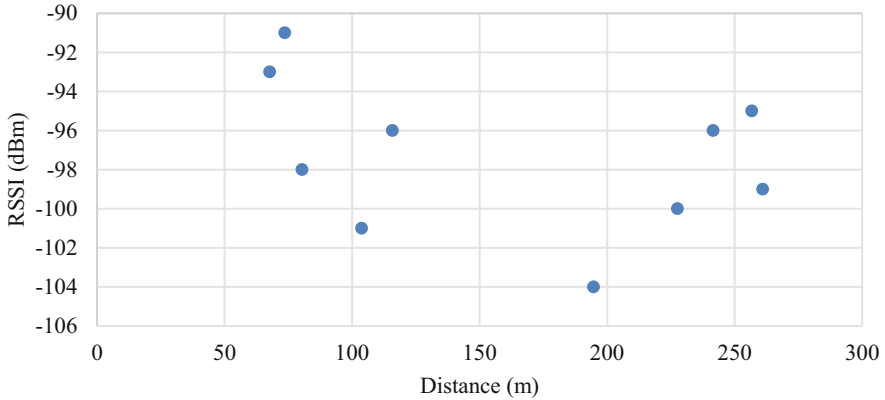
(a)
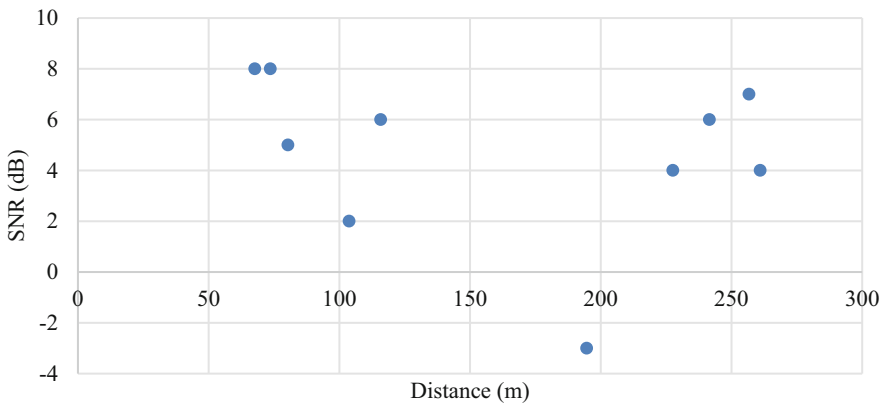
(b)

Fig. 7.14 (**a**) Received Signal Strength Indication (RSSI) and (**b**) Signal-to-noise ratio (SNR) across distances in oil palm nursery

7.7 Calibration of EC and pH Sensors

The calibration of EC and pH sensors were carried out to validate measurement accuracy. Errors are estimated during normal operation of a sensor and corrected by a suitable feedback mechanism of the calibration code.

For the EC sensor, a two-point calibration is done using standard buffer solutions of 1413 μS/cm and 12.88 mS/cm. The EC calibration code is uploaded to the Arduino board and commands are input into the serial monitor to view and calibrate electrical conductivity readings. The sensor probe is washed with distilled water, dried with paper and inserted into 1413 μS/cm standard buffer solution (Fig. 7.17).

Fig. 7.15 (**a**) The LoRa gateway placement at the building and (**b**) the nodes and gateway location in the urban area

(a)

(b)
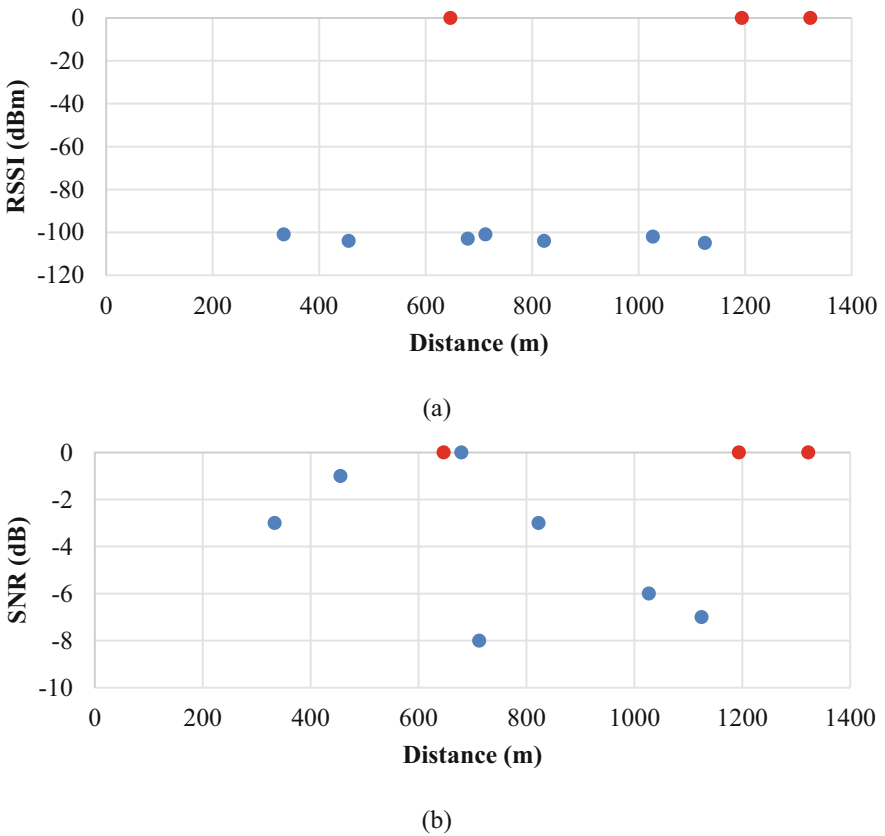
Fig. 7.16 (**a**) The received signal strength indication (RSSI) and (**b**) the signal-to-noise ratio (SNR) across distances in the urban area

Table 7.3 LoRa signal transmission in an urban area

Point	Coordinates	Distance from gateway (m)	Height (m)	RSSI (dBm)	SNR (dB)
U1	3.006262, 101.719660	333.51	45	−101	−3
U2	3.000516, 101.717129	1026.92	54	−102	−6
U3	3.004829, 101.711426	1194.04	41	–	–
U4	3.007306, 101.715167	712.52	47	−101	−8
U5	3.004845, 101.722722	455.51	46	−104	−1
U6	3.003334, 101.719087	646.66	47	–	–
U7	3.002631, 101.721258	679.38	48	−103	0
U8	3.001333, 101.721528	822.46	58	−104	−3
U9	2.998801, 101.723629	1124.55	77	−105	−7
U10	2.996780, 101.720701	1322.66	81	–	–

Fig. 7.17 EC sensor and conductivity buffer solutions

The probe is stirred gently in the solution, until the values are stable as the programme automatically identifies buffer solution 1413 μS/cm. After the first point calibration is completed, relevant parameters are saved and the second point calibration for buffer solution 12.88 mS/cm is performed.

For the pH sensor, buffer solutions 4.00, 7.00 and 10.00 are used to calibrate for neutral, acidic and alkaline solutions (Fig. 7.18). The buffer solutions are measured with a handheld pH meter to check pH value accuracy of the buffer solutions (Fig. 7.19).

The pH electrode is washed with distilled water and placed into the 7.00 buffer solution (Fig. 7.20). The pH calibration code is uploaded to the Arduino and the error of the pH value is printed on the serial monitor is compared with 7.00. The value

Fig. 7.18 pH buffer solutions
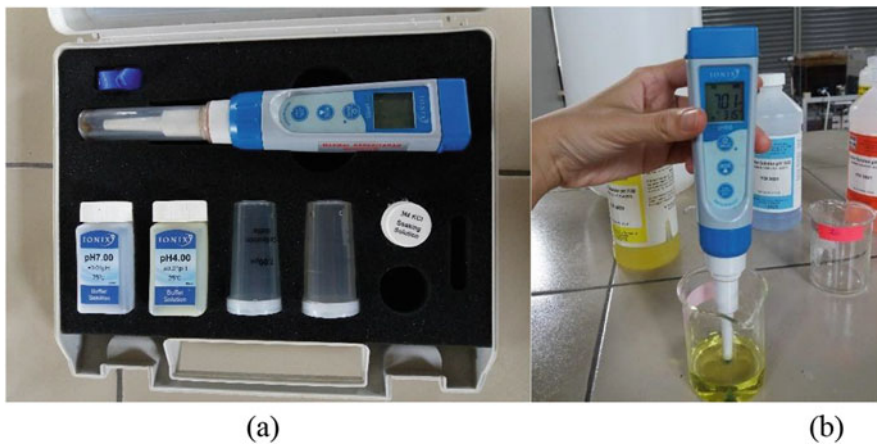
Fig. 7.19 (**a**) Ionix Instruments Premium Line Testers Series 5, (**b**) pH measurement

difference of 0.11 is input at "#define Offset" of the calibration code. Then the pH electrode is placed into the 4.00 buffer solution and waited for about 1 min. Then the potential gain of the sensor board is adjusted until it reaches 4.00 and the readings are

Fig. 7.20 pH sensor calibration

allowed to stabilise. After the acidic calibration is completed, alkaline calibration is then performed using 10.00 buffer solution.

7.8 Soil EC and pH Measurement Test

To test the practicality of using this system in an indoor environment to monitor soil parameters, a soil EC and pH measurement test is carried out in the UPM Transgenic Greenhouse (2.993361, 101.719596) (Fig. 7.21). A total of 60 oil palm seedlings of 9 months old are planted in this 78.04 m^2 greenhouse. Four different types of treatment are given to the oil palm seedlings in the greenhouse; uninoculated with water deficit; uninoculated and well-watered; inoculated and well-watered; and inoculated with water deficit. In this measurement test, five seedlings from each bench are monitored for soil EC and pH readings. The gateway is placed at five different points in the greenhouse (Fig. 7.22).

Soil EC (Fig. 7.23) and pH (Fig. 7.24) both show smooth line graphs for each seedling indicating stable sensor measurement and data transmission. Both EC and pH graphs display the differences in soil EC and pH values of seedlings in different growing conditions. Further observations and experiments can be carried out with the data obtained from soil monitoring to conduct plant health analysis. One of the factors which may affect soil salinity is the evapotranspiration rate. Seedlings located nearer to a ventilation fan will experience higher evapotranspiration rate.

Fig. 7.21 (**a**) Environment in greenhouse, (**b**) sensor node, and (**c**) EC and pH sensors placed in soil in the polybag of an oil palm seedling

Based on both charts, a healthy seedling that is uninoculated and well-watered has an average EC value of 419.2 uS/cm and an average pH value of 5.5.

Both the gateway and the node were placed in the greenhouse to evaluate the RSSI and SNR in an indoor environment while monitoring soil EC and pH of five different oil palm seedlings. According to Table 7.4, RSSI decreases as distance
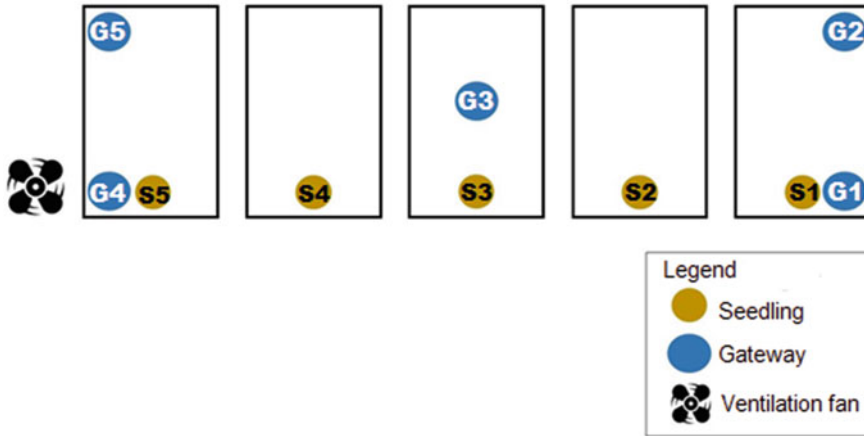
Fig. 7.22 Layout of greenhouse benches, gateway positions and tested seedlings Layout of greenhouse benches, gateway positions and tested seedlings
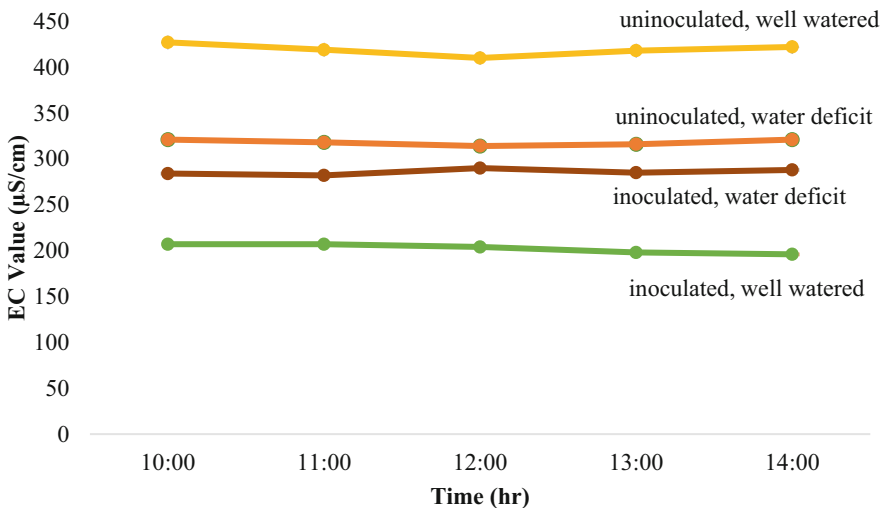
Fig. 7.23 Soil EC measurements of different treated oil palm seedlings transmitted using LoRa network in a greenhouse nursery

between gateway and node increases. Overall, the average RSSI of −63 dBm and SNR of 9 dB are considered as strong and good LoRa transmission. In the greenhouse, the oil palm seedlings were arranged with relatively wide spacing between the seedlings. The transmission did not face much interference as the signals from the node did not need to penetrate any walls to reach the gateway.
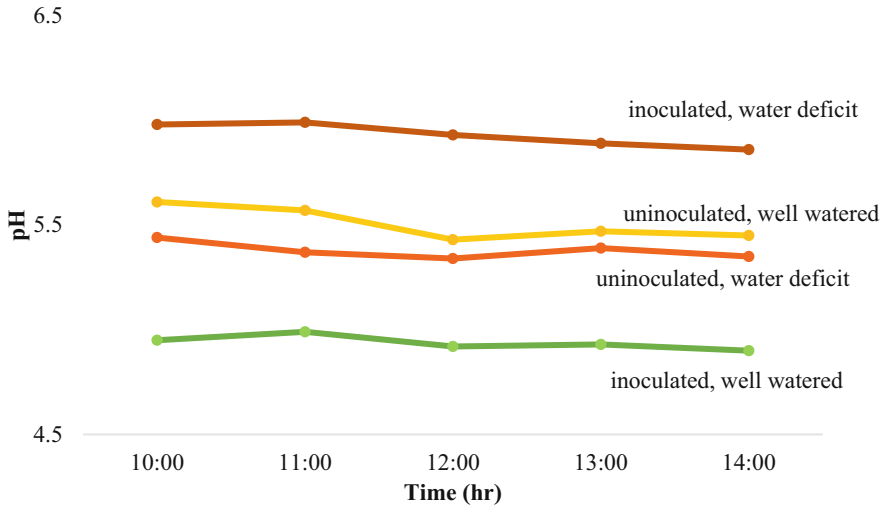
Fig. 7.24 Soil pH measurements of different treated oil palm seedlings transmitted using LoRa network in a greenhouse nursery

Table 7.4 Node RSSI and SNR at various gateway locations inside a greenhouse

	S1	S2	S3	S4	S5
Gateway/sensor node (RSSI)					
G1	−37	−59	−65	−65	−76
G2	−56	−68	−75	−64	−68
G3	−65	−80	−57	−60	−66
G4	−66	−60	−54	−51	−55
G5	−75	−67	−64	−65	−62
Gateway/sensor node (SNR)					
G1	10	9	9	9	10
G2	10	10	9	9	10
G3	9	9	9	9	10
G4	9	9	9	9	9
G5	9	9	9	10	9

7.9 Conclusion

A soil EC and pH monitoring system based on LoRaWAN protocol was successfully developed. In this study, a performance evaluation of the LoRaWAN network was conducted and from this evaluation, limitations of the performance in different environments were analysed. LoRaWAN transmission was able to cover the entire oil palm nursery area of 6 ha in this study with a maximum distance of 261 m. In the young oil palm plantation, as the area was secluded with wide canopy and fronds, the signal strength weakened as distance increased therefore achieving only a maximum

distance of 223 m. This problem can be circumvented by elevating the gateway or device to the point with free line of sight as proven in urban areas achieving up to more than 1 km distance. As for the test in the greenhouse setup (78.04 m^2), the transmission did not face much interference and LoRaWAN communication gave stable measurements throughout the experiment (10.00 am until 2.00 pm) with an average EC of 419.2 uS/cm and an average pH of 5.5. From this study, the feasibility of LoRaWAN as a network protocol to be used in oil palm nurseries for soil EC and pH monitoring is proven to be successful. The use of LoRaWAN will benefit farmers and overcome the restrictions for IoT implementation in the agriculture sector.

7.10   Recommendation

The scalability of LoRaWAN can be improved by increasing the spreading factor (SF) values. In this study, SF7 was used throughout the whole experiment. By increasing the SF, the bit rate will be slower but energy per data set becomes higher thus resulting in higher range. Furthermore, there should be optical visibility between the transceiver and receiver antennas by placing the gateway at a higher elevation for clear line of sight. The range can be improved by increasing the height of the antennas to achieve optical visibility between them.

For future investigation into this subject, more case studies are needed as this one is limited to short-term tests and conducted under similar weather conditions for all cases. For further application of the system, a fully automated fertigation system can be developed based on the LoRaWAN protocol. Complete data analytics can be performed by incorporating information from a weather station and sensor nodes to send feedback (downlink) to the actuator in fertigation systems. Another improvement to the current system is to add additional features such as resistance to the environment with waterproof, surge and lightning protection.

References

Ahmad, K. A., Salleh, M. S., Segaran, J. D., & Hashim, F. R. (2018, February). Impact of foliage on LoRa 433MHz propagation in tropical environment. In *AIP Conference Proceedings* (Vol. 1930, No. 1, p. 020009). AIP Publishing LLC.

Behera, S. K., Suresh, K., Rao, B. N., Ramachandrudu, K., Manorama, K., & Harinarayana, P. (2017). Soil fertility and yield-limiting nutrients in oil palm plantations of north-eastern state Mizoram of India. *Journal of Plant Nutrition, 40*(8), 1165–1171.

Brewster, C., Roussaki, I., Kalatzis, N., Doolin, K., & Ellis, K. (2017). IoT in agriculture: Designing a Europe-wide large-scale pilot. *IEEE Communications Magazine, 55*(9), 26–33.

Goh, K. J., Mahamooth, T. N., Ng, H. P., Teo, C. B., & Liew, Y. A. (2016). Managing soil environment and its major impact on oil palm nutrition and productivity in Malaysia. *Advanced Agriecological Research Sdn. Bhd*.

Hanlon, E. A., & Bartos, J. M. (1993). Soil pH and electrical conductivity: A country extension soil laboratory manual. *Circular (USA)*. no. 1081.

Jawad, H. M., Nordin, R., Gharghan, S. K., Jawad, A. M., & Ismail, M. (2017). Energy-efficient wireless sensor networks for precision agriculture: A review. *Sensors, 17*(8), 1781.

Jebril, A. H., Sali, A., Ismail, A., & Rasid, M. F. A. (2018). Overcoming limitations of LoRa physical layer in image transmission. *Sensors, 18*(10), 3257.

Kassim, M. R. M., Mat, I., & Harun, A. N. (2014, July). Wireless Sensor Network in precision agriculture application. In *2014 International conference on computer, information and tele-communication systems (CITS)* (pp. 1–5). IEEE.

McCauley, A., Jones, C., & Jacobsen, J. (2009). Soil pH and organic matter. *Nutrient Management Module, 8*(2), 1–12.

Mekki, K., Bajic, E., Chaxel, F., & Meyer, F. (2019). A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express, 5*(1), 1–7.

Mutert, E. (1999). Suitability of soils for oil palm in Southeast Asia. *Better Crops International, 13*(1), 37.

Pierce, F. J., & Nowak, P. (1999). Aspects of precision agriculture. *Advances in Agronomy, 67*, 1–85.

Ray, P. P. (2017). Internet of things for smart agriculture: Technologies, practices and future direction. *Journal of Ambient Intelligence and Smart Environments, 9*(4), 395–420.

Rudd, R., Craig, K., Ganley, M., & Hartless, R. (2014). *Building materials and propagation* (Final Report, Ofcom, 2604).

Samsuri, S. F. M., Ahmad, R., & Hussein, M. (2010, May). Development of nutrient solution mixing process on time-based drip fertigation system. In *2010 Fourth Asia international conference on mathematical/analytical modelling and computer simulation* (pp. 615–619). IEEE.

Song, Y., Lin, J., Tang, M., & Dong, S. (2017). An internet of energy things based on wireless LPWAN. *Engineering, 3*(4), 460–466.

Chapter 8
Strategic Short Note: Application of Smart Machine Vision in Agriculture, Forestry, Fishery, and Animal Husbandry

Kai-Rong Chang, Tsung-Hsiang Ma, and Yan-Fu Kuo

Abstract Food production is an increasingly important topic worldwide due to factors such as population increase and workforce aging. In conventional agriculture, which consists of forestry, fishery, and agriculture, and animal husbandry, manual observation is the main method used by farmers to monitor field and animal conditions. However, the younger generation is reluctant to engage in farming due to the high labor requirements and low wages. To solve this problem, smart machine vision, which is the combination of deep learning and machine vision, is applied for managing farms and increasing production. In this section, the architectures of smart machine vision applications are highlighted. Several examples of the applications are shown.

8.1 Introduction

Food security is always one of the top priorities globally. As estimated by the United Nations, the global population will reach 9.7 billion in 2050 (United Nations, 2022). However, climate change makes food production more challenging by reducing farmable land and worsening the environment for animal husbandry. In addition, food production is facing issues of labor shortage and aging workforce (United Nations, 2022). Nowadays, few in the young generation are willing to work in agriculture, forestry, fishery, and animal husbandry because of the harsh working environments and disproportionate wages.

K.-R. Chang · T.-H. Ma · Y.-F. Kuo (✉)

Department of Biomechatronics Engineering, College of Bioresources and Agriculture, National Taiwan University, Taipei, Taiwan
e-mail: ykuo@ntu.edu.tw

Past advances in farming have yielded new equipment and facilities designed to improve farming efficiency (e.g., tractors and greenhouses). However, the observation of farming or animal conditions still relies on manual observation. For example, farmers have to patrol in the field to check the growth condition of crops. In animal husbandry, farmers have to patrol regularly to monitor animal conditions. This is because the environments for crops and animal husbandry are usually complex. However, manual observation is slow and requires experience. More automatic monitoring approaches are needed.

In recent years, due to breakthroughs in computing speed, deep learning has become more popular as a method to solve complex machine vision problems in the fields of agriculture, forestry, fishery, and animal husbandry. The application of deep learning algorithms in machine vision is referred to as smart machine vision. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs; Rumelhart et al., 1985) are the common types of deep learning algorithms that are employed in smart machine vision. The use of smart machine vision is regarded as an automatic solution in the aforementioned fields.

This section introduces the workflow and applications of smart machine vision in agriculture, forestry, fishery, and animal husbandry. Firstly, different types of CNNs in various applications are introduced. Next, four components of smart machine vision applications are introduced. Last but not least, several examples of smart machine vision in agriculture, forestry, fishery, and animal husbandry are shown. These studies demonstrated how smart machine vision can help to resolve the food security problem.

8.2   Tasks of Smart Machine Vision

Smart machine vision applications can be categorized into static and dynamic tasks (Fig. 8.1). Static tasks include classification, localization and classification, semantic segmentation, and instance segmentation. Dynamic tasks are usually behavior recognition tasks. Typically, static tasks use images as the input. On the other hand, dynamics tasks use videos as the input.

Various types of CNNs are used for static tasks. For classification, CNNs containing convolution layers, pooling layers, and fully connected layers are used. These CNN models are usually referred to as backbone CNNs. Commonly used backbone CNNs include AlexNet, VGG, ResNet, etc. (Alzubaidi et al., 2021). For localization and classification, CNN models are usually composed of backbone CNNs, necks, and heads. Commonly used localization and classification CNN models include Fast R-CNN, YOLO, etc. (Liu et al., 2020). For semantic segmentation and instance segmentation, CNNs with encoder–decoder architectures are typically used. The commonly used semantic segmentation and instance segmentation CNNs include U-Net (Garcia-Garcia et al., 2018), YOLACT (Tian et al., 2021), etc.
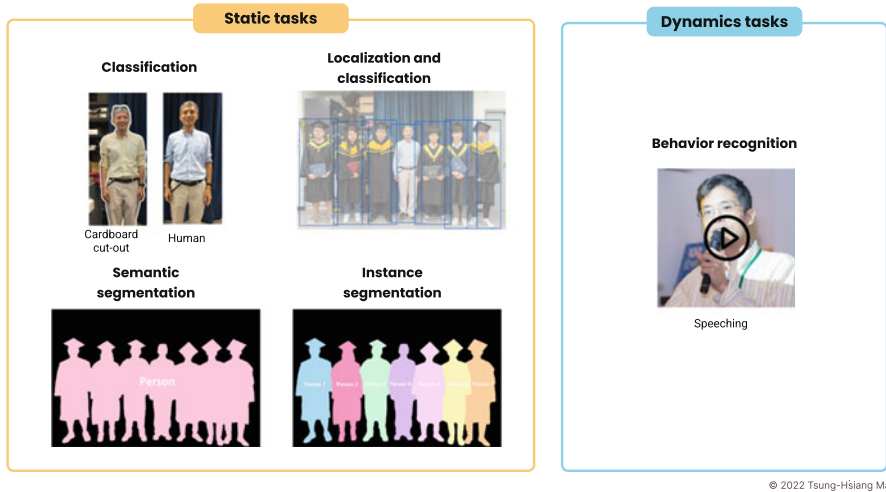
Fig. 8.1 Tasks of smart machine vision

Dynamic tasks are typically fulfilled using the combination of CNNs and RNNs. CNNs extract features from video frames, and RNNs determine the output by considering the features in consecutive frames of videos. A commonly used RNN is gated recurrent units and long short-term memory (Alzubaidi et al., 2021).

8.3   The Components of Smart Machine Vision

Typical smart machine vision applications in agriculture, forestry, fishery, and animal husbandry include four important components: image acquisition, machine learning, database, and user access (Fig. 8.2). Image acquisition is the first step in machine vision implementation. Images are collected by using cellphones manually or by using stationary cameras automatically. Typically, if the application requires only one image, cellphones are used for image acquisition. By contrast, if the application requires videos, stationary cameras are used for image acquisition.

The component of machine learning includes five steps, namely image collection, image augmentation, model architecture selection, model training, and model performance evaluation. To train a deep learning model, it is recommended to acquire at least 500 images for each category. The images are next annotated. The annotated images are then split into training, validation, and test with a ratio of typically 8:1:1. Image augmentation (e.g., flipping and rotation) is subsequently applied to the annotated training images to generalize the images and improve the robustness of the model to be trained. A CNN model for a specific task (e.g., classification, localization and classification, semantic segmentation, and instance segmentation) is then chosen. The training of the model involves hyperparameter selection. Typical
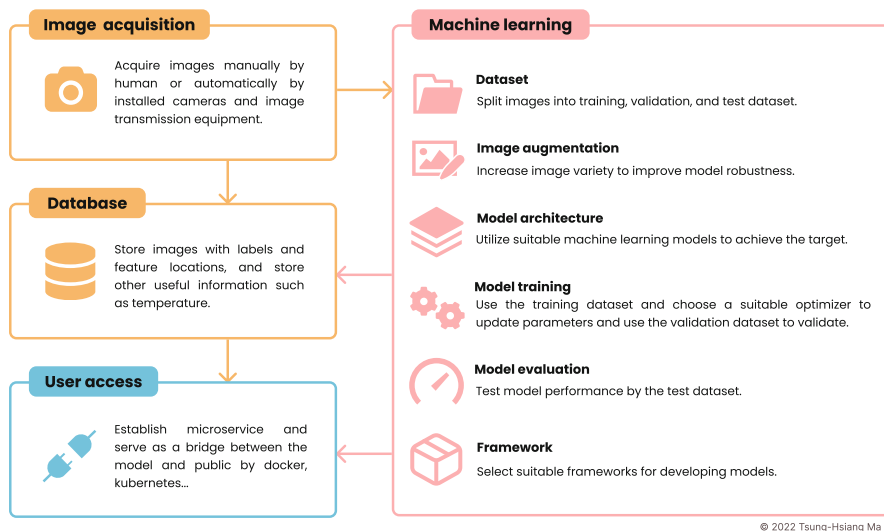
Fig. 8.2 Machine vision in agriculture implementation flow

hyperparameters include learning rate and weight decay. Appropriate hyperparameters improve the performance of the model to be trained. After the model is trained, test images are applied to the trained model to evaluate the model performance. The aforementioned procedure completes the component of machine learning.

A database is usually used in smart machine vision applications too. The database is an essential component because the data for model training (acquired images and labels of images) are stored in the database. The database can be used to store the images uploaded by end users too.

Another essential component of smart machine vision is user access. Typically, a microservice is established to serve as a bridge between the internal system (i.e., the trained model and database) and end users. Through the microservice, the trained model and database can be accessed by both internal and end users. The end users can also provide new data through user access.

8.4 Examples of Smart Machine Vision in Agriculture, Forestry, Fishery, and Animal Husbandry

Smart machine vision has been applied to the fields of agriculture, forestry, fishery, and animal husbandry. This is especially an increasing trend. Smart machine vision alleviates the issue of labor shortage. With smart machine vision, work patterns are changed, and work loadings are reduced. Below are some examples of smart machine vision.

Crops are vulnerable to pests and diseases, environmental changes, and storage conditions, resulting in economic losses. The application of smart machine vision is an easier and faster solution to control the quality of crops. Numerous studies have applied smart machine vision to identify plant diseases and pests (Abade et al., 2021) and superficial damages (Li et al., 2020). These applications help farmers to reduce economic loss and increase production.

Forests play an essential role in food security and daily necessities (Sunderland et al., 2013). People who live near tropical forests can acquire food surrounding specific tree species and even on the tree. Also, different trees can be made into a wide variety of daily necessities. Studies were conducted to identify consumable wood species. Yang et al. (2019) differentiated between morphologically similar species in genus *Cinnamomum* (Lauraceae). The species *C. osmophloeum* yields cinnamaldehyde and is used as a herbal plant. Pelletier et al. (2019) and Schiefer et al. (2020) identified tree species and mapped tree species in a forest, respectively (Hamedianfar et al., 2022), which can help those living nearby to reliably acquire food and earn a living.

Fish is a major source of protein globally. However, the biological sustainability of oceans has been brought to attention in recent years. Smart machine vision was applied to identify species of marine organisms to prevent overfishing or inadvertent illegal fishing (Aguzzi et al., 2020). Also, the length and species of harvested fish, which is required by some fisheries management organizations, can be estimated and recorded using smart machine vision (Tseng & Kuo, 2020; Tseng et al., 2020). Aquaculture is another way to raise seafood. Certain studies evaluated the frequency of fish feeding using smart machine vision (Zhao et al., 2021). Shrimp body length was estimated for feeding management using smart machine vision (Lai et al., 2022).

Economic animals are another major source of protein. Smart machine vision can be applied to alleviate the need for patrols and manual observation in animal farming. Related studies include a monitoring system for detecting sick chickens (Ojo et al., 2022), an observation system for identifying the tail-biting behaviors of pigs (Chen et al., 2021), an automatic monitoring of newborn piglets tracking and lactating frequency of sows (Ho et al., 2021), and an inspection system for identifying lameness behaviors of cows (Mahmud et al., 2021).

8.5 Conclusion

Food production is now affected by labor shortage globally. To meet the demand of food, smart machine vision is applied in agriculture, forestry, fishery, and animal husbandry to develop automatic solutions that can replace human power. The whole process can be simplified as image acquisition, machine learning, database, and user access. With the application of smart machine vision, farmers can manage their fields efficiently, harvest richly, and thereby improve food security worldwide.

References

Abade, A., Ferreira, P. A., & de Barros Vidal, F. (2021). Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture, 185*, 106125.

Aguzzi, J., Chatzievangelou, D., Company, J. B., Thomsen, L., Marini, S., Bonofiglio, F., Juanes, F., Rountree, R., Berry, A., Chumbinho, R., Lordan, C., Doyle, J., del Rio, J., Navarro, J., De Leo, F. C., Bahamon, N., García, J. A., Danovaro, P. R., Francescangeli, M., . . ., Gaughan, P. (2020). The potential of video imagery from worldwide cabled observatory networks to provide information supporting fish-stock and biodiversity assessment. *ICES Journal of Marine Science, 77*(7–8), 2396–2410.

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data, 8*(1), 1–74.

Chen, C., Zhu, W., & Norton, T. (2021). Behaviour recognition of pigs and cattle: Journey from computer vision to deep learning. *Computers and Electronics in Agriculture, 187*, 106255.

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez-Gonzalez, P., & Garcia-Rodriguez, J. (2018). A survey on deep learning techniques for image and video semantic segmentation. *Applied Soft Computing, 70*, 41–65.

Hamedianfar, A., Mohamedou, C., Kangas, A., & Vauhkonen, J. (2022). Deep learning for forest inventory and planning: A critical review on the remote sensing approaches so far and prospects for further applications. *Forestry: An International Journal of Forest Research, 95*, 451.

Ho, K. Y., Tsai, Y. J., & Kuo, Y. F. (2021). Automatic monitoring of lactation frequency of sows and movement quantification of newborn piglets in farrowing houses using convolutional neural networks. *Computers and Electronics in Agriculture, 189*, 106376.

Lai, P. C., Lin, H. Y., Lin, J. Y., Hsu, H. C., Chu, Y. N., Liou, C. H., & Kuo, Y. F. (2022). Automatic measuring shrimp body length using CNN and an underwater imaging system. *Biosystems Engineering, 221*, 224–235.

Li, Z., Guo, R., Li, M., Chen, Y., & Li, G. (2020). A review of computer vision technologies for plant phenotyping. *Computers and Electronics in Agriculture, 176*, 105672.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2020). Deep learning for generic object detection: A survey. *International Journal of Computer Vision, 128*(2), 261–318.

Mahmud, M. S., Zahid, A., Das, A. K., Muzammil, M., & Khan, M. U. (2021). A systematic literature review on deep learning applications for precision cattle farming. *Computers and Electronics in Agriculture, 187*, 106313.

Ojo, R. O., Ajayi, A. O., Owolabi, H. A., Oyedele, L. O., & Akanbi, L. A. (2022). Internet of Things and Machine Learning techniques in poultry health and welfare management: A systematic literature review. *Computers and Electronics in Agriculture, 200*, 107266.

Pelletier, C., Webb, G. I., & Petitjean, F. (2019). Temporal convolutional neural network for the classification of satellite image time series. *Remote Sensing, 11*(5), 523.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science.

Schiefer, F., Kattenborn, T., Frick, A., Frey, J., Schall, P., Koch, B., & Schmidtlein, S. (2020). Mapping forest tree species in high resolution UAV-based RGB-imagery by means of convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing, 170*, 205–215.

Sunderland, T., Powell, B., Ickowitz, A., Foli, S., Pinedo-Vasquez, M., Nasi, R., & Padoch, C. (2013). *Food security and nutrition*. Center for International Forestry Research (CIFOR).

Tian, D., Han, Y., Wang, B., Guan, T., Gu, H., & Wei, W. (2021). Review of object instance segmentation based on deep learning. *Journal of Electronic Imaging, 31*(4), 041205.

Tseng, C. H., & Kuo, Y. F. (2020). Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. *ICES Journal of Marine Science, 77*(4), 1367–1378.

Tseng, C. H., Hsieh, C. L., & Kuo, Y. F. (2020). Automatic measurement of the body length of harvested fish using convolutional neural networks. *Biosystems Engineering, 189*, 36–47.

United Nations. (2022). *Revision of world population prospects*. Author. Retrieved from https:// population.un.org/wpp/

United Nations. Department of Economic and Social Affairs, & United Nations Conference on Trade and Development. (2022). *World economic situation and prospects 2022*. Author.

Yang, H. W., Hsu, H. C., Yang, C. K., Tsai, M. J., & Kuo, Y. F. (2019). Differentiating between morphologically similar species in genus Cinnamomum (Lauraceae) using deep convolutional neural networks. *Computers and Electronics in Agriculture, 162*, 739–748.

Zhao, S., Zhang, S., Liu, J., Wang, H., Zhu, J., Li, D., & Zhao, R. (2021). Application of machine learning in intelligent fish aquaculture: A review. *Aquaculture, 540*, 736724.

Chapter 9
Artificial Intelligence in Agriculture: Commitment to Establish Society 5.0: An Analytical Concepts Mapping for Deep Learning Application

Victor Massaki Nakaguchi and Tofael Ahamed

Abstract Artificial intelligence (AI) was defined as a key component for the establishment of the Society 5.0. AI's rapid ascension and incorporation in all sectors of human activities is becoming the background of innovation. Therefore, the solution for many complex problems specially regarding to agri-food industry dynamics and climate change. Society 5.0 is an approach of the future society considering a projection over global population and the aspects related to sustainability and social welfare. AI is not a novel theme as it has been discussed from many decades ago. However, Big Data associated with highly efficient algorithms and powerful processing units are pushing AI into our reality. Prosper societies were built up above a solid agricultural base with well-organized systems of production and distribution, including all phases of industrial revolution as well. The advent of climate change and its consequences over crops is demanding innovative solutions to keep on increasing yield while mitigating the adverse effects on the ecosystem. The aim of this chapter is to provide an analytical concept mapping and framework about AI-based learning systems, in a quasi-philosophical way to establish Society 5.0 centered on deep learning techniques for agricultural solutions, specially addressing labor shortages and management of limited bioresources worldwide.

Keywords Artificial Intelligence · Deep learning · Digital agriculture · Autonomous machinery · Society 5.0

V. M. Nakaguchi
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

9.1 Introduction

Global overpopulation is a big concern for agricultural production considering it is necessary to enhance yield besides considering the effects of climate change over crops and its impacts on the ecosystem. In addition to it, the rural population is aging and not being replaced at the same rate, which represents an opportunity for innovation especially in the fields of management and automation of machinery. During the last two decades, data and information have been exponentially created and processed in a speed that have never seen before, besides that fast connectivity between countries and regions has overcome physical barriers, thus creating an interdependency among them including high flux of materials, goods, and people. Artificial intelligence (AI) came into hype and its rapid adoption in all sectors of our daily lives is transforming the lifestyle of people around world. Recommendation systems, virtual assistants, specific tasks robots, smart gadgets, and fully connected systems creating meta data all the time are just a few examples of the presence of this global trend, which also represents the change in habits and mentality of our society as well.

Historically, all human societies were developed based on local food security, and the improvement of agricultural production has been representing an upgrade version of that societies. The first one was composed of people able to hunt and collect fruits, the society 2.0 were made of agrarian communities able to grow their own food and exchange the excess, in the society 3.0, agricultural goods and natural resources were transformed in manufactured products in large scale (industrial society), and the contemporary society (information society) was established side by side with green revolution, responsible to provide food, fibers, and energy for the booming population (Fig. 9.1).

The agriculture of the twenty-first century is passing through a digital transformation where processes and management are being fully automatized and the
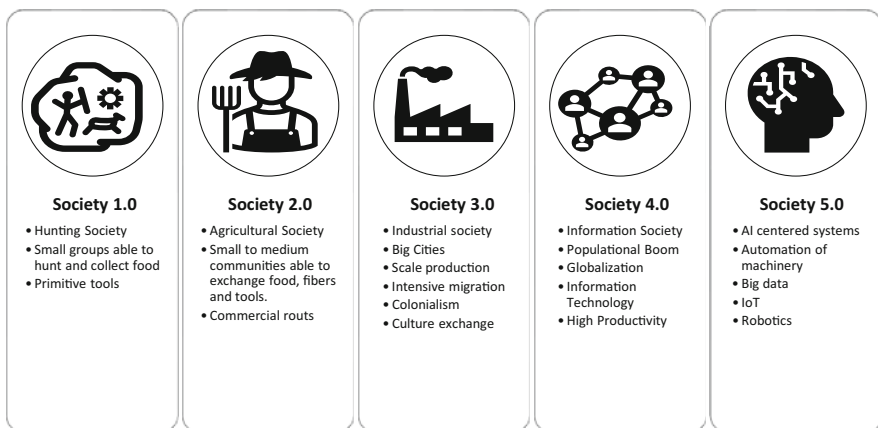
Fig. 9.1 From society 1.0–5.0, the main characteristics of each in a historical context

immediate effect is a high gain efficiency yield, costs reduction and by consequence, the sustainability of production. By incorporating the industry 4.0 elements inside farms, customers around the world can be aware of important aspects of the foods and fibers they are consuming, especially regarding traceability, quality, and standardization. On the other hand, farmers can compensate labor shortages by using robots, autonomous machines, and computational management systems. Moreover, by associating geo statistics methods, sensors (proximal and remote), and information technology (IT), the entire management of farming in real time can be done taking in consideration the spatial and temporal variability, which is contributing for precise use of inputs and bioresources in agriculture production (precision agriculture).

In this sense, the Society 5.0 has been proposed by the government of Japan (CSTI, 2016) as a prospect plan for the global society linked with sustainability and community-based welfare to reduce social gaps and improve life quality of the people (Fukuda, 2020), in this concept AI-based technologies innovations, IoT (internet of things), Big Data, and Blockchain, are fundamental components to its establishment.

Several challenges are involved in the implementation of this fifth version of our global community, principally those related to diminishing differences among countries development. In most of African countries, for instance, the last green revolution has not fully arrived yet and people are still dying by starvation, while in developed countries farming is getting into a completely industrial plants such as vertical farms and agribusiness stock market. Another challenge is to disrupt the AI paradigm, which means to make AI-based technologies a tool to be used for improvement of human's life quality including all people in the process equally, for example, since the first systems are being programed and trained by humans, some applications may contain human inherit cultural behavior like preconception, racism, and political stands.

AI-associated technologies may sound like a magic box, where the solution comes from after a single touch, however, it is not so easy as it looks like, the advancements in this field are being made gradually. In fact, deep learning (DL) is mainly responsible to bring AI to the edge, machines mounted with computer vision (CV) systems can perform tasks such as navigation systems for self-driving vehicles, object detection, mapping of behavior patterns, natural language processing (NLP), and augmentation and virtualization of the reality (as a part of Society 5.0). AI-based learning technologies produced a set of new applications to make possible a new revolution in agriculture. Due to its high capacity to process a great amount of data in real time, this is one of the eligible innovations to keep on upsetting Malthusian theory of population, once it permits through data management the improvement of sustainable production of food, fibers, and energy besides enhance the logistics and customer experiences for the coming years.

9.2   AI Mapping Concept

AI is a discipline inside Computer Sciences intended to develop systems able to mimic human's ability to resolve problems, its origin is traced from the middle of the last century as a theoretical area discussed by philosophers, mathematicians, and computer scientists. When AI was proposed by early 1950s, not enough technology was available at the time to bring it to the spotlight, nevertheless, the exponential progress in microelectronics, informatics, and engineering over the last decades, has allowed the arising of big data, high power processing units and advanced heuristic algorithms, a combination of these three made AI overcome its last winter and started to bloom for countless tasks and utilities.

AI aggregates many fields of study such as neural networks (NNs), evolutionary computation, and fuzzy logic. The employment of these fields we can find in data mining, CV, NLP systems, and autonomous machinery. AI's core uses machine learning (ML) algorithms which have revolutionized data analytics, especially for unstructured and multidimensional data, that by the way, represents the majority volume of data transferred throughout the internet. The output of ML algorithms is a model (Fig. 9.2), and the models of ML can be created by using at least one of three types of learning mechanisms: Supervised learning (predictive models), unsupervised learning (descriptive models), and reinforcement learning models.

Supervised learning in mathematical ways refer to classification or regression operations, the learning process is composed by two phases, training and testing as we are going to see further in this chapter, but in an effortless way to understand the concept, it has the presence of the "supervisor" character (labeled data), responsible to guide the learning process.

As for unsupervised learning method, the model is generated from a description of the whole data by using Association, Grouping (Clustering), Outliers Detection, Standard sequences, and Summarization algorithms, it requires post-analysis of the model to understand what the machine has learned, for unstructured data such as big data, and multidimensional data from radiometric sensors for instance, unsupervised learning algorithms are powerful tools.

Reinforcement learning is a class of ML algorithms where the system learns from its own experience due to interaction with the environment, this self-exploratory

Fig. 9.2 Definition of model in ML
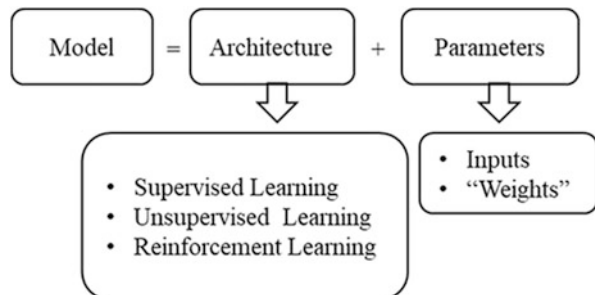
approach is based on penalties and rewards as guidance according to each assertive and failure behavior, respectively, when the machine is learning is trying to reach the given objective.

DL is a concept when applying NNs-based decision support systems, it can be seen as a variation approach of ML based algorithms, the main difference from ML methods remains in the fact that DL algorithms generate a model by self-learning the entire process, including the steps of feature extraction of inputs, adjustment of weights and bias. Thus, it is specially used to deal with big data analysis such as image classification, object detection, object segmentation, and natural speech processing.

In fact, ML techniques look after ways to solve problems by applying principally statistics and linear algorithms, on the other hand, DL looks like a human based approach dealing with nonlinear problems. For agriculture and livestock production, several applications including DL solutions are related especially for CV applications, for example, autonomous machinery, plant phenotyping, and unmanned aerial vehicles (UAVs) image analysis, cattle management, and animal welfare establishment.

9.3   Deep Learning (DL) and Neural Networks (NNs)

NNs are the main topic of study in DL, Artificial Neural Networks (ANNs) is a common name when referring to NNs for computer systems, it was summarized by Robert Hecht-Nielsen (1992) as a parallel and distributed processing structure consisting of elements connected unidirectionally where the information is processed from external inputs. ANNs emerged in a context where complex problems could no longer be solved by using pre-determined algorithms, e.g., image classification, where a small angle variation in the same image may change the whole values of pixels.

The ANNs was inspired by studies regarding the connection mechanism of the biological neuron (Synaptic connections), McCulloch and Pitts (1943), proposed a mathematical model for biological neurons with which are connected each other receiving and transmitting information as electrical pulses with some level of processing, consequently depending on the processing result the next neuron may either be activate or not (threshold logic). In 1958 Frank Rosenblatt introduced a model called perceptron (Fig. 9.3), which was designed to mimic some of the principles of intelligent systems in general. This model became the basis of future generations of ANNs, as recurrent neural networks (RNNs) and convolutional neural networks (CNNs) among others, besides that it is considered one of the very first feedforward NNs.

The working processing of an artificial neuron can be seen in Fig. 9.3, analog to biological neuron synapses, the result from the aggregate function produces a value that feeds the step-function (Eq. 9.1) from which may give an output (Exciting synapse) or not (Inhibition synapse), the weights are responsible to amplify or reduce
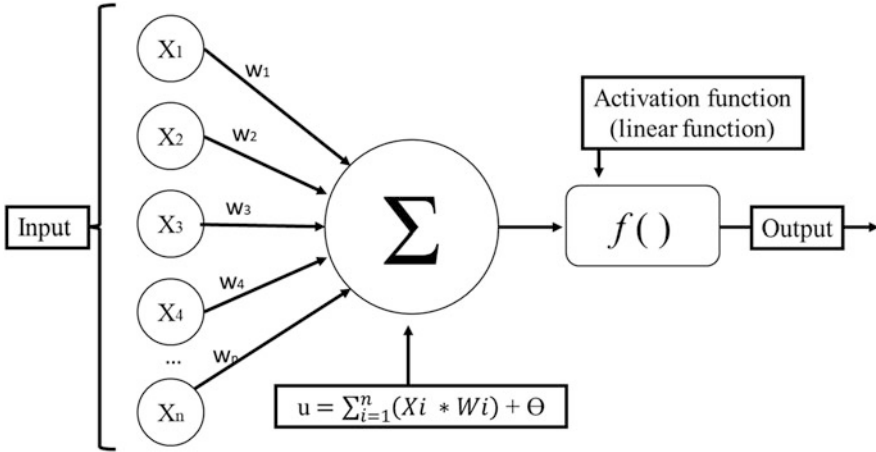
Fig. 9.3 One-layer perceptron model proposed by Rosenblatt (1958). u = the linear function as described, Θ is called bias, is the factor that represents the threshold for trigging the activation function

the input signal to the step-function, the process of adjusting weights is called "Training."

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ 1, & \text{for } x \geq 0 \end{cases} \qquad (9.1)$$

The one-layer perceptron is a good model to solve linear problems, however, for nonlinear problems such as simple xor logic, it cannot predict output values, since it is impossible to segregate correspondent values in only one dimension. The solution in these cases was the insertion of multiple layers, or also called hidden layers [multilayer perceptron (MLP)], in this type of ANNs the output value of one neuron may feed the input of the next neuron (Fig. 9.4). Besides that, other activation functions were introduced for better modeling the output of neurons such as sigmoid function (Eq. 9.2), hyperbolic tangent (Eq. 9.3), rectified linear unit (ReLU— Eq. 9.4), leaky ReLU (Eq. 9.5), and so on.

The choice of activation functions depends on what prediction value the ANN is being trained for, different activations functions can be used in the same architecture, for example, the hyperbolic tangent (tanh) is usually used in the hidden layers due to its range from -1 to 1, while the sigmoid function is often used for probabilities outputs, ReLU is nowadays one of the most used activation functions in ANN intended for images recognition systems due to their highly computational efficiency to not activate neurons with negative values.
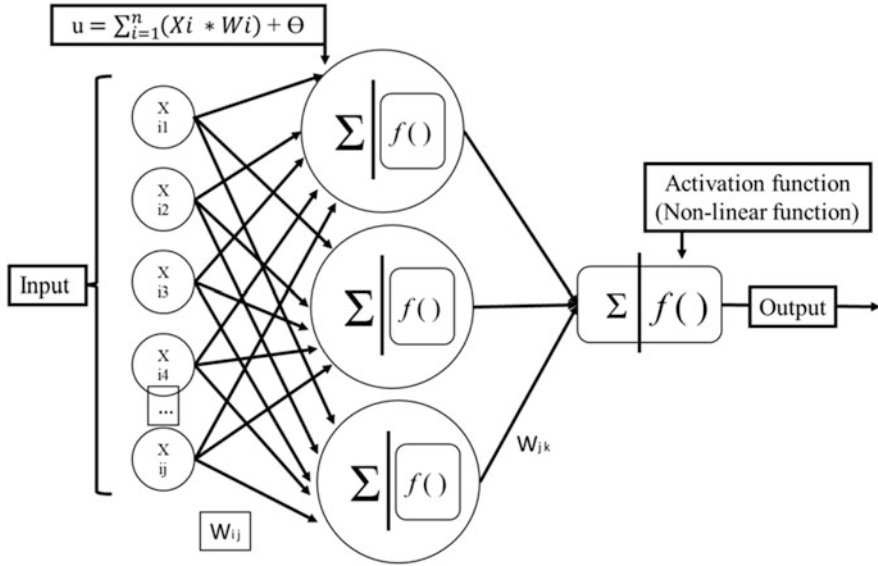
Fig. 9.4 MLP model with three hidden layers. In MLPs each input value connects with other neurons or layers depending on the result from activation function

$$f(x) = \frac{1}{1 + e^{-x}} \tag{9.2}$$

$$f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \tag{9.3}$$

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases} \tag{9.4}$$

$$f(x) = \begin{cases} ax, & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases} \tag{9.5}$$

9.3.1   Principles of the ANN Learning Process

ANNs are supervised learning models, therefore, it can be simply understood as a heuristic system that learns by doing mathematical equations and adjusting its results to reach the correct answer. The overall objective is to provide mechanisms for a computational system to extract, recognize, and classify patterns in multidimensional data with satisfactory level of accuracy.

The learning process can be divided into two steps or phases: training and testing (Fig. 9.5). Different datasets must be used as input values to feed each step, however,
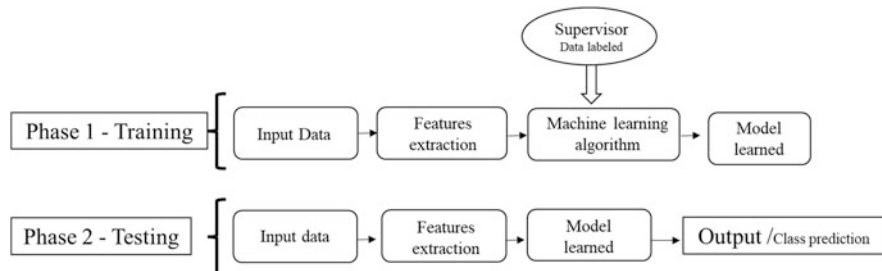
Fig. 9.5 Supervised learning chart flow, notice that blocks represent only the process, not the neurons associated with

in both cases we have labeled data (Images, sounds, etc.) that contains information about type, name, and position of the object that ANN is being modeling for. The labeled data are examples provided by humans to machines, like teaching kids by showing them an object and telling its respective name, e.g., when kids see, touch, and try the object, on the next time they see a similar one, based on their previous experiences, they can recognize it by associating the shapes, colors, textures, even the taste. As for ANNs these experiences are the adjustments of weights and extraction of features.

The training step is the tuning process to adjust the parameter "weights" to fit the inputs to the "correct answers" (Labeled data—"Supervisor"). The testing process is regarding the verification of the model assertiveness, this step "says" to the ANN how much the prediction is correct, and on which "direction" should the ANN goes to reduce the error. So basically, we say that an ANN have learned when the system has achieved a high precision ability to predict an object class or when it has decreased the assessment error at the minimum.

The most of ANNs are trained using backpropagation algorithm, besides that, the original work introduced backpropagation algorithm as a NN too, it consists of a sequence pace to update the weights after calculating the overall error (we call the cost function, which is the difference between "correct answer" or predicted value of true class and the obtained output value) the error is then multiplied by the slope of the activation function and finally backpropagated to the previous layers.

The aim of this algorithm is to reduce the cost function at each interaction (or epoch in DL language), backpropagation algorithm is seen as a descent algorithm as it looks to find the direction where the slope is minimum (Fig. 9.6). Here, two parameters are important to control how much should the weights be adjusted and for what direction (either backward or forward from the previous calculated weigh) it should go to: learning rate and momentum, the first one is regarding the "step size" that next weights should be decreased or increased for (As far from the minimum point on the gradient the learning rate should be big, as closer to the minimum it should be small). The momentum is about the direction the weights should go (vector moment) to reach the minimum slope. Note that the angular variation to
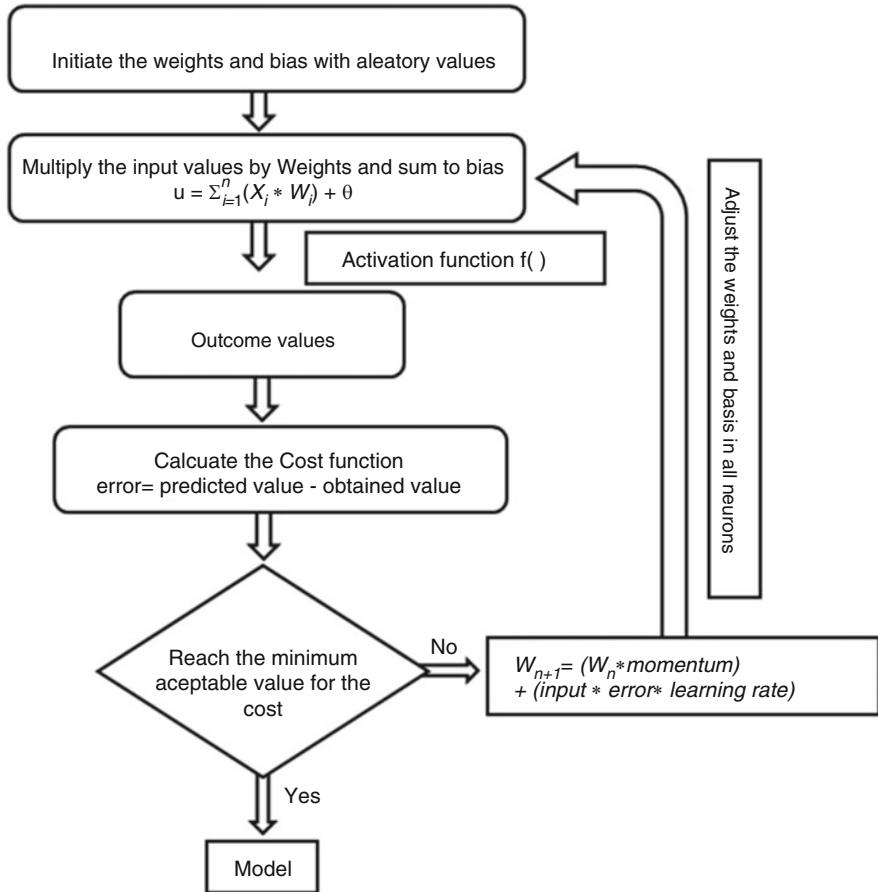
Fig. 9.6 Learning process using backpropagation algorithm

calculate the step size is defined according to the primary derivative of a point in relation to the previous one, that is how learning rate is adjusted in practice.

We should point out that a model pursuit should be able to generalize an object with a certain level of accuracy, good models are those capable of recognizing an unseen object as well as humans do. However, depending on complexity conditions of exposure, including occlusions and distortions, some objects are difficult even for us to do so, in other words, we should not have an ANN model to fit 100% in the training dataset, in DL language we do not want the model to overfit the training data during the learning process. When the model overfit the training dataset, it means the machine learned how to exactly describe the dataset given to it but in practice it may fail to predict new inputs. We need a model that can predict the objects of study based on similar characteristics among them.

DL techniques are specially used for image classification, detection, and segmentation (Fig. 9.7) as mentioned before. However, the challenge of program computer

Fig. 9.7 Distinct types of models for image recognition. Classification models predict the class name. Detection models find the object inside the image and draw a box on it. Segmentation models in this case, instance segmentation, put together segmentation of a pixel and object detection to individualize objects in the image

systems able to mimic human vision is not new, and the field of CV has been explored from decades ago, including several studies on mammals' vision proposed by Hubel and Wiesel (1959), which in fact, constituted the basis to understand the vision processing and hierarchy model of imaging recognition.

Kunihiko Fukushima (1980) proposed an ANN that he called Neocognitron, it was an updated version of his previous work Cognitron (Fukushima, 1975), in both works, the idea was to develop a system able to learn how to recognize image patterns, his hypothesis was that a successful computer vision would open the gates for further comprehension on human's vision. He considered in his work an association of previous hypothesis on patterns hierarchical recognition such as proposed by Hubel and Wiesel (1962, 1965), Gestalt theory (Wertheimer, 1938), and others. The main idea was that a biological brain recognizes an image by associating features of them layer by layer, from simple presence/absence of light and shapes to complex figures in the output layers.

Many years after K. Fukushima's work, Lecun et al. (1998) introduced a computer vision system able to recognize with a proficient level of accuracy written-digits, using CNN and gradient-based learning algorithm (Backpropagation), their work introduced a new generation of NNs once it could get over traditional CV methods for image classification based on ML and statistics approaches. Thus, reopened the interest and investments for DL industrial applications for images recognition through automatic learning instead of hand-designed algorithm. Currently, CV is a field of study in DL that reunites different science expertise (Fig. 9.8).
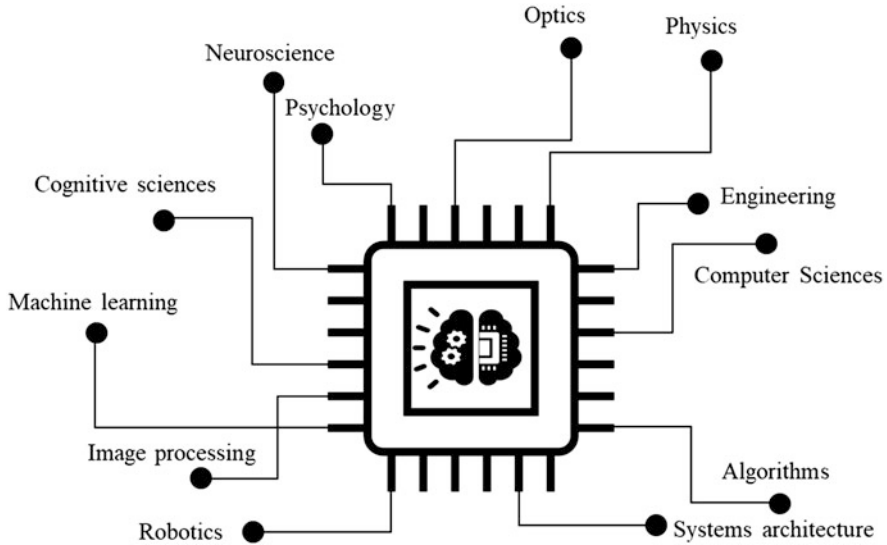
Fig. 9.8 Computer vision system is a field in DL that reunites several sciences

9.4 DL at the Edge with CNNs

Cameras are the most popular type of sensor present in our daily life; they are present in almost every place we can imagine. Over the years, the technology involved in these devices grew up exponentially, the improvement on spatial and radiometric resolution became a paradox for CV systems. The good side is regarding the exploratory opportunity for documentation details, especially to capture wide angles and several wavelength frequencies beyond human's vision capacity such as infrared bands. On the other hand, as for high resolution types ensure to capture images with millions of pixels, further sophisticated methods including best algorithms and faster processing power units to analyze the images are continually under on-growing demand. Comparing LeNet-5 in 1998, for instance, the input images had a dimension of 28×28 pixels, but currently, a 5 Mega Pixel camera for instance (common camera aboard on any ordinary Smartphone) can produce an image with 2592×1944 pixels, becoming a challenging factor to imaging processing. Fortunately, fresh solutions to get around these issues have been proposed during the last two decades, including new generations of graphic processing cards and better ANNs as well.

In CV systems, cameras are the source of data (input data) and actively participate in the training and deployment of AI-based systems. Over the years cameras has become accessible for everyone at an affordable price, the cost to produce self-datasets of images is too high, due to time consuming and storage costs, leading DL players to use common sources of data to train their applications. There are several free use dataset images available on the internet to serve as a parameter to evaluate

performance of CNNs models, the notorious and most appreciated are ImageNet (Stanford vision Lab), MS COCO (Provided by Microsoft®), and Open Images (Google LLC.).

Image classification and object detection are the most exciting areas of CV these days, in truth, this is the area where most of the efforts in DL have been concentrating so far. Enormous advancements have been made during the last 10 years since AlexNet (a type of CNN developed by Alex Krizhevsky et al., 2017) won the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC). By using the approach of multiple graphic processing units (GPUs) for parallel processing, the ReLUs activation function for modeling the neurons outputs and by increasing the training dataset by using image augmentation, their work became the turning point for the new generation of NNs that followed in sequence.

CNN is an ANN with many layers where convolutional operations are applied in some of that to extract features and produce a model that can predict an object with certain level of accuracy (Classification) or a position of an object inner an image or frame (object detection). By convolution, it is defined as a mathematical operation between two functions that produces a third function, in DL it stands for matrix multiplications (each pixel has a value associated to it, each value varies from 0 to 255 for each RGB color band).

In image classification algorithms, CNN is used to extract features of objects on each frame during the training process. The mechanism of feature extractions is done by applying "filters" (called Kernels), these filters can highlight main characteristics of pictures such as contours, edges, shadows, lines, and so on.

In the convolutional layers a grid of pixels is multiplied by the Kernel and the matrix resultant is used as input values to create the "feature maps," this one is a special layer that contains the relevant information to classify each class of image the CNN is being trained for (Fig. 9.9).

From 2012, year by year improvements were made, introducing new advancements in CNNs architectures such as ZFNet (Zeiler & Fergus, 2014), VGGNet (Simonyan et al., 2014), GoogLeNet "Inception" (Szegedy et al., 2015), and MobileNet (Howard et al., 2017). In 2015, ResNet (Kaiming He et al., 2016) was introduced and until now is one of the most used CNNs for object detection around the world. It won some competitions on ILSVRC reaching the minimum of 3.57% of error rate.

Many other CNNs are available for CV applications and the choice depends on the final goal, type of dataset, and power processing hardware. Historically, CNNs are heavy, and the computing is dense, not suitable for ordinary computers. Considering that, at the present, the greatest challenge for CNN architecture is to make it portable and suitable for low-end devices such as smartphones and nano computers.
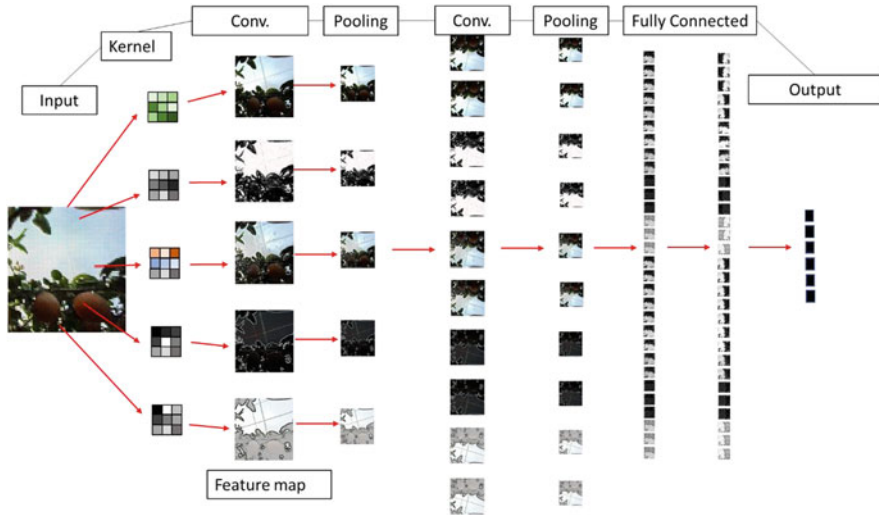
Fig. 9.9 CNN common architecture. In this example, a CNN with two convolutional and two fully connected layers. A kernel 3 × 3 is applied on the first layer, in AlexNet, for example, they used a 11 × 11 kernel

9.5 Deep Learning Algorithms for Object Detection

Image classification using CNNs proved to be faster comparing to other classification algorithms such as *K*-Nearest Neighbors (KNN), recently, specialized algorithms for image classification and object detection have been proposed to support CNNs to learn faster, especially when dealing with dense processing (huge datasets). A revolution for classification/object detection using CNNs was the region-based CNN or R-CNN (2014) and its family, including fast R-CNN (Girshick, 2015), these innovative approaches introduced the selective search method to generate region proposals before computation of feature maps, these region proposals hold the object to be classified (has the bounding box "b-box" included), that means it contains the object of interest inside one of the 2000 regions the image was divided into (the regions in this case represent the image cut in 2000 parts, each part is one region). The problem with these novel approaches was the low speed, making it impossible to execute real-time applications. However, a new generation of R-CNN called faster R-CNN (Ren et al., 2015), which does not use the region selective search method before CNN, was released to enhance the speed, it has introduced the region proposed network (RPN), which is a type of extra-CNN aggregated to the CNN to obtain the feature maps after CNN calculation (See details in Chap. 7).

The introduction of faster R-CNN makes possible the development of navigation systems supported by real-time CV, although this model requires high performance devices such as powerful GPUs and a lot of physical memory, the models provided by faster R-CNN are one of the most accurate available now (something strongly
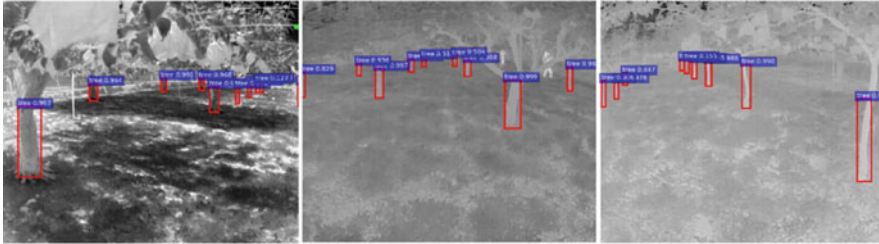
Fig. 9.10 Faster R-CNN object detection associated with thermal camera under different light conditions for navigation systems in a Japanese pear orchard

Fig. 9.11 Mask R-CNN instance segmentation for Apple harvesting automation in orchard

desirable especially for autonomous navigation). For agriculture purposes, especially in orchards, navigation systems based on CV are becoming the best alternatives to overcome limitations of systems based on traditional Global Navigation Satellite Systems (GNSS), especially under closed canopy (bad precision due to interference and scintillations of radio wavelength signals).

Guidance mechanisms based on CV are important for agricultural machines to defeat one of the most limitations imposed by the environment: the light dependency. Machines equipped with navigation systems including IR sensors can operate at night (Fig. 9.10), this characteristic allows gain in crop calendar, and autonomous night work shift, therefore optimizing the harvesting season.

By 2017 Mask R-CNN (He et al., 2017) came upon the stage presenting semantic segmentation and object detection joint together as Instance segmentation (Fig. 9.11), this new model is being explored in diverse tasks including harvesting robotic arms systems (see details in Chap. 11) and cattle monitoring for livestock management.

Fig. 9.12 YOLOv4 Object Detection in Apple Orchard for Apple Harvesting Vision Machine

Currently, the state of the art for image classification and object detection is the "You Only Look Once" (YOLO) neural network algorithm, YOLO is a model proposed in 2015 by Joseph Redmon, its concept is similar to the R-CNN regarding the region proposing, however, YOLO visualize the entire image during training and detection steps and compute the bounding boxes directly in the image using regression algorithms, YOLO introduces the idea of dividing the frame into $S \times S$ number of grids, so basically, if the probability of object of interest is higher to be in a cell grid, this cell grid becomes responsible for prediction of the object (see details in Chap. 9). Since its release in 2015 YOLO has reached several versions, being one of the most stable, accurate, and fast models for object detection (Fig. 9.12). Wang et al. (2022) listed six basic characteristics of a system to be considered as state of the art in deep learning, which includes faster and strong networks, effective integration methods, accuracy, robust loss function, efficient label assignment technique, and
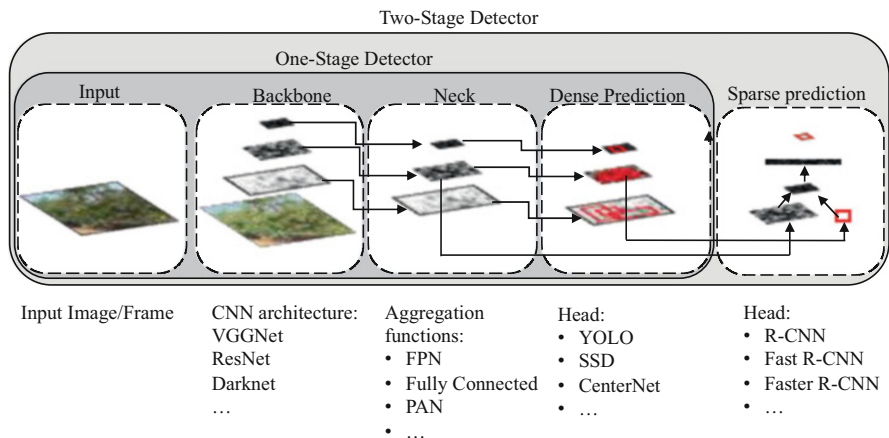
Fig. 9.13 Deep learning algorithms common architecture for object detection

training efficiency. The community of DL developers around the world have been improving YOLO family and releasing new upgraded versions online periodically.

It is important to note that CNNs are ANNs used for extraction of inputs features while DL algorithms for object detection such as Faster R-CNN, Single Shot Multibox Detector "SSD" (Liu et al., 2016), YOLO (Redmon et al., 2016) are specialized DL algorithms for object detection. In other words, CNNs and Classification/Object Detection/Segmentation models are two separated blocks that work together as part of an algorithm, that's why we say CNN is the "backbone" of DL algorithms.

Object detection algorithms architectures can be grouped into dense predictions and sparse prediction types, the common architecture of an object detection algorithm is composed by similar blocks for clear understanding (Fig. 9.13).

Object detection applications are limitless as it can be employed in every real-time CV system when tasks involve detecting objects, sorting, and tracking. Countless operations such as quality control, grading (Fig. 9.14) and packing of fruits, vegetables, eggs, and animal parts can be fully automatized in post harvesting procedures. Besides that, CV based on-the-go machines can distribute inputs (fertilizers, pesticides) directly on targets, as localized treatments, contributing to sustainability of agricultural production.

Another great opportunity to enhance the administration of rural properties is the adoption of DL based CV in central control management systems, as well as implemented in cities to monitor cars traffic and detect abnormal behavior of people it can be incorporated in agricultural fields to detect abnormal movements of animals or people inside lands, presence of wild bodily and individual monitoring of animals in flock, thus associate with animal welfare to stablish precision livestock farming (PLF).

Moreover, CV platforms can also assist traffic control of machines (Fig. 9.15) using tracking algorithms, best logistic planning can be enabled automatically by

Fig. 9.14 YOLOv4 object detection in Quail eggs industry for quality management and grading automation

Fig. 9.15 YOLOv4 object detection application for machine traffic control systems. In this example, YOLOv4 is used to qualify the distance between harvesting machines and truck transporter to prevent accidents

concentrating management tasks such as machines status, displacement, and maintenance prevention based on data of engines and mechanisms, all broadcasted in real time.

Systems designed to control traffic of vehicles, including tractors traffic are important to establish the concept of "rail" roads on production fields, as much bigger machines have become over time, the weight associated to them highlighted the problem of soil compaction. The idea of traffic control is important to reduce soil loss, and by consequence reduce heavy machinery operations like plowing and subsoiling. The direct plantation system introduced a few decades ago in big crops such as corn and soybeans in systematic rotation of cultivations have contributed to reduce the number of tractor operations in the same field, however, several machinery operations are fundamentally mandatory, including sowing, spraying, and harvesting, this last one usually is responsible to most of damage onto soil structure because often are made in partnership with truck trailers that often use normal set of tires, non-adequate to farm fields.

9.6   Conclusions: DL Serving a New Agriculture Revolution

The Society 5.0 consists of the first guidelines to a prospective future that have already began, if considering that we are currently living in the transition phase from the information society to AI-based society, many aspects of our daily life are still under debates and development such as privacy of data, monetary foundation, and autonomous machine safety protocols.

Global issues such as overpopulation, climate change, and environmental degradation are topics that demand immediate attention and active plans. Fortunately, these thematic have one common variable: the agriculture. There is a global consensus that best practices in agriculture can lead to the solution of those problems once setting up production to high efficiency use of bioresources and inputs we may reduce adverse impacts on the ecosystem caused principally by deposit of wastes and bad management of fields.

AI in agriculture is a vast topic that includes many fields of science, DL methods have gained attention recently due to its human vision sense approach. Countless innovations are being reported for machinery automation and management of fields. Regardless DL is quite a new advancement in the field of AI, it has already achieved astonishing improvements for CV, besides that, commercial solutions are being created and getting better continually. Innovations related to intelligent decision-making platforms, robotics and autonomous vehicles using DL base systems represents an opportunity to overcome the labor shortages in addition to enhance efficiency of agriculture production around the world, moreover, it stands for our best strategy to reach sustainability yield for the next decades.

Researchers, companies, and industries are gradually developing solutions supported by AI associated technologies to agricultural tasks. There are considerable challenges for implementation of such intelligent systems, especially regarding legal

limitations imposed to systems and applications, robustness, adaptability, connectivity, safety, and technical support as well. Moreover, the interaction between human–machine is a real concern since it will continue demanding at least a few of human supervision to "instruct" the machine to operate inside the fields or properties, so the role of human as supervisor is important to avoid machine decision-making modeling based on mislead bias datasets.

The movement of AI community for the next years is to create applications less specialists, with which final users would be able to customize features such as software, robots, and machines to their own necessities, for example, autonomous tractors able to mount implements and perform real-time adjustments to several tasks like sowing, spraying, and harvesting in different crops, orchards within a minimum human intervention. Another global tendency for technology companies especially hardware manufacturing industry is to reduce the cost for low-end AI deployment devices along with development of friendly user frameworks to make DL accessible to non-computer specialists, allowing common customers to emerge solutions for local scale problems, therefore meeting the requirement to establish the Society 5.0.

References

Council of Science. (2016). *Technology and innovation fifth science and technology basic plan*. Government of Japan. Retrieved from https://www8.cao.go.jp/cstp/

Fukuda, K. (2020). Science, technology, and innovation ecosystem transformation toward society 5.0. *International Journal of Production Economics, 220*, 107460. https://doi.org/10.1016/j.ijpe.2019.07.033. ISSN 0925-5273.

Fukushima, K. (1975). Cognitron: A self-organizing multilayered neural network. *Biological Cybernetics, 20*(3–4), 121–136. https://doi.org/10.1007/bf00342633

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics, 36*(4), 193–202. https://doi.org/10.1007/bf00344251

Girshick, R. (2015). Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1440–1448). IEEE.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/CVPR.2016.90

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2961–2969). IEEE.

Hecht-Nielsen, R. (1992). Neural networks for perception. Theory of the backpropagation neural network. Based on "nonindent" by Robert Hecht-Nielsen. In *Proceedings of the International Joint Conference on Neural Networks 1*, June 1989 (Vol. 593–611, pp. 65–93). IEEE. https://doi.org/10.1016/b978-0-12-741252-8.50010-8

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv*, 1704.04861.

Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology, 148*(3), 574.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction, and functional architecture in cat's visual cortex. *Journal of Physiology (London), 160*, 106–154.

Hubel, D. H., & Wiesel, T. N. (1965). Receptive fields and functional architecture in two nonstriate visual area (18 and 19) of the cat. *Journal of Neurophysiology, 28*, 229–289.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *ACM Communications Magazine, 60*(6), 84–90.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324. https://doi.org/10.1109/5.726791

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Cheng-Yang, F., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European Conference on Computer Vision* (pp. 21–37). Springer.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, 5*, 115–133. https://doi.org/10.1007/BF02478259

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 779–788). IEEE.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes et al. (Eds.), *Advances in Neural Information Processing Systems (NIPS)* (pp. 91–99). MIT Press.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65*(6), 386–408. https://doi.org/10.1037/h0042519

Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Learning local feature descriptors using convex optimisation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(8), 1573–1585.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015) Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1558–1748). IEEE. https://doi.org/10.1109/CVPR.2015.7298594

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

Wertheimer, M. (1938). Gestalt theory. In W. D. Ellis (Ed.), *A source book of Gestalt psychology* (pp. 1–11). Kegan Paul, Trench, Trubner & Company. https://doi.org/10.1037/11496-001

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In D. Fleet et al. (Eds.), *European Conference on Computer Vision (ECCV): Computer Vision* (pp. 818–833). Springer.

Chapter 10
Potentials of Deep Learning Frameworks for Tree Trunk Detection in Orchard to Enable Autonomous Navigation System

Ailian Jiang, Ryozo Noguchi, and Tofael Ahamed

Abstract The outdoor environment is more complex and uncertain than the indoor environment, resulting in the identification of natural landmarks remaining a challenge in the automation process. In the orchard, tree branches not only affect the signal accuracy of the global navigation satellite system (GNSS), but also affect the light conditions of the camera, making the recognition accuracy decrease. Thermal camera has the potential under low illumination, it can be used to recognize tree trunks using a deep learning system in different light conditions. Therefore, the objective of this study was to use a thermal camera to detect tree trunks at different times of the day under low-light conditions using deep learning. Thermal images were collected from the dense canopies of two types of orchards (conventional and joint training systems) under high-light (12–2 PM), low-light (5–6 PM), and no-light (7–8 PM) conditions in August and September (summertime) in Japan. Thermal imagery datasets were augmented to train, validate, and test using the faster R-CNN, YOLO-v3, and CenterNet deep learning model to detect a tree trunk. A total of 12,876 images were used to train the model, 9270 images were used for training, 2318 images were used to validate, and 1288 images were used to test the model. The mAP of the Faster R-CNN model was 0.8378, YOLO-v3 was 0.4077 and CenterNet was 0.9370 for the testing process. In the comparisons, CenterNet was achieved the highest accuracy in the three methods in tree truck detection using thermal camera. However, based on the analysis of the actual image results, CenterNet may have missed the target. Therefore, results showed that Faster R-CNN was the most suitable method can be used to recognize objects using thermal images in orchards under different lighting conditions.

A. Jiang
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

R. Noguchi
Faculty of Agriculture, Kyoto University, Kyoto-shi, Japan

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

Keywords Thermal image · Tree trunk detection · Low light conditions · Orchards · Faster R-CNN · YOLO · CenterNet

10.1 Introduction

Between 1995 and 2010, Japan's agricultural labor force has gradually declined from 4.14 to 2.39 million, and its average age increased from 59.1 to 65.8 years (Dong, 2018). Therefore, agricultural robotics have the potential to support agricultural labor shortages and increase agricultural productivity in this critical stage of transformation (Vadlamudi, 2019). In agricultural automation and robotics applications, vehicle navigation is important in outdoor environments, which are complex and uncertain compared to indoor conditions (Bergerman et al., 2016). Open field navigation has significant success using Real Time Kinematic Global Navigation Satellite System (RTK-GNSS) with higher accuracy (Takai et al., 2010). However, orchard navigation is the most complex, and interruption of RTK-GNSS signals due to high and dense canopies is frequently reported (Li et al., 2009). Because Japanese orchards have net and dense branches, GNSS signals may be affected, and many farmland orchards do not have base stations set up to use GNSS directly. In addition, the performance of GNSS-based navigation depends highly on GNSS signal quality.

Therefore, orchard-based navigation systems remain a challenge for the development and application of agricultural robots. Light Detection and Ranging (LiDAR) is used to scan the surrounding environment in real time and returns accurate distance information using the time-of-flight principle, it can overcome the limitations of low light and interruption of RTK-GNSS signal. Laser sensors have been used for agricultural position detection and automatic coupling using artificial landmarks concrete and grass surface ground for navigation (Tofael Ahamed et al., 2006a, b, 2009) or crop fields (Malavazi et al., 2018) by processing point clouds for crop and weed detection. However, LiDAR can only obtain distance and angular orientation information, cannot provide information about the object's type. To develop an intelligent robust system, LiDAR may not be sufficient for low-light conditions. Using a camera as a sensor is advantageous, due to its low cost and ease of installation in agricultural robots. Camera sensors are widely used in navigation systems in various ways, such as using color information to segment paths in citrus groves for vehicle navigation (Subramanian et al., 2006), or focusing on color information of shadows and soil texture, to distinguish traversable areas (furrows) from impenetrable areas (ridges) for navigation (Takagaki et al., 2013). However, common RGB cameras are easily affected by light, particularly in orchards, where the light differs between shaded and non-shaded areas, and the cameras typically experience exposure phenomena, making target identification unclear. Furthermore, low light is one of the most difficult aspects to create effective vision systems for agricultural robotics (Zhang et al., 2018). RGB cameras have high-light requirements and work with low accuracy or are even unsuitable for low-light environments (Guo et al., 2021).

A thermal sensor measures the temperature of an object and the amount of the heat it emits, based on this imaging principle, thermal camera was used to assess the health of tree trunks and detect damage in the interior (Abdullah & Mücahit Taha, 2021). Additionally, thermal camera has a detection function that is not affected by visible light; thus, this study uses a thermal camera to detect tree trunks in orchards for target detection. The target can be used to position a robot during navigation. In addition to thermal imagery, advancements in artificial intelligence and deep learning can be used for tree trunk detection to enable vehicle navigation under different lighting conditions. Similar targets may have the same range of temperatures that emit and can easily localize the target using computer vision-based deep learning approaches (Hespeler et al., 2021).

When convolutional neural networks (CNN) were successful in target detection (Krizhevsky et al., 2012), deep learning was attracted great attention because CNN can not only extract better expressive and more abstract features, but also integrate the functions of extraction, classification, and selection in one model. There are two main algorithms: one is the R-CNN (Regions with CNN features) series based on classification and the other is YOLO (You Only Look Once) based on regression. Those target detection methods first list the potential target location and then classify. CenterNet as an anchor-free target detection network, reduces the detect time by using key point estimation to find the center points and regress to other object properties, such as size, 3D location, orientation, and even pose (Zhou et al., 2019).

This study describes a solution for low-light conditions in orchards that allows the use of an effective vision system for vehicle navigation, which requires accurate target detection at low machine operating speeds. Therefore, this study uses Faster R-CNN, YOLO, and CenterNet for target detection after training using the same dataset. These three deep learning methods use different structures and have the potential to identify thermal images in low and no light conditions for agricultural orchard applications to improve efficiency. Thermal cameras use infrared information to detect targets in orchards. The purpose of this study is to use thermal camera for trunk detection as a target object, compare the results of Faster R-CNN, YOLO, and CenterNet to select a suitable method to develop an autonomous speed sprayer navigation system.

10.2   Materials and Methods

10.2.1   Field Data Collection

The FLIR ADK® thermal camera has a resolution of 640×512, and images were collected from pear orchards at the Tsukuba-Plant Innovation Research Center, University of Tsukuba, Tsukuba, Ibaraki (36°06′56.8″ N, 140°05′37.7″ E). In this study, two types of pear orchards were used for data collection: (1) a conventional pear orchard, and (2) a joint tree training pear orchard (Fig. 10.1). Images were

Fig. 10.1 Aerial view of experimental pear orchards: (**a**) conventional planted pear orchard and (**b**) joint tree training pear orchard in the Tsukuba-Plant Innovation Research Center, University of Tsukuba, Japan

Table 10.1 Dataset collection times and light conditions in orchards

Date	Time	Light condition
2021.8.24	19:00–20:00	No light
2021.8.26	13:00–14:00	Strong light
2021.9.06	17:00–18:00	Low light

collected three times under high-light (12–2 PM), low-light (5–6 PM), and no-light (7–8 PM) conditions in both orchards (Table 10.1). Thermal images under different light conditions, such as no-, high-, and low-light conditions, were then analyzed (Fig. 10.2).

10.2.2 Data Preparation

10.2.2.1 Image Frames from Videos

MATLAB® (Matrix Laboratory, Natick, Massachusetts, USA) was used to capture images every 15 frames and remove images that were blurred and that did not contain tree trunks. In total, 5313 images were analyzed, and the number of images of trunks in different periods was similar.

10.2.2.2 Labeling

LabelImg® was used to label datasets while preparing visual object classes (VOC) and COCO datasets. When the position of the tree trunk on the image was selected, information such as the image name, object classification, and pixel coordinates were recorded in an .xml and .txt file. .xml files were used to train Faster R-CNN model, and .txt files were used to train YOLO and CenterNet.
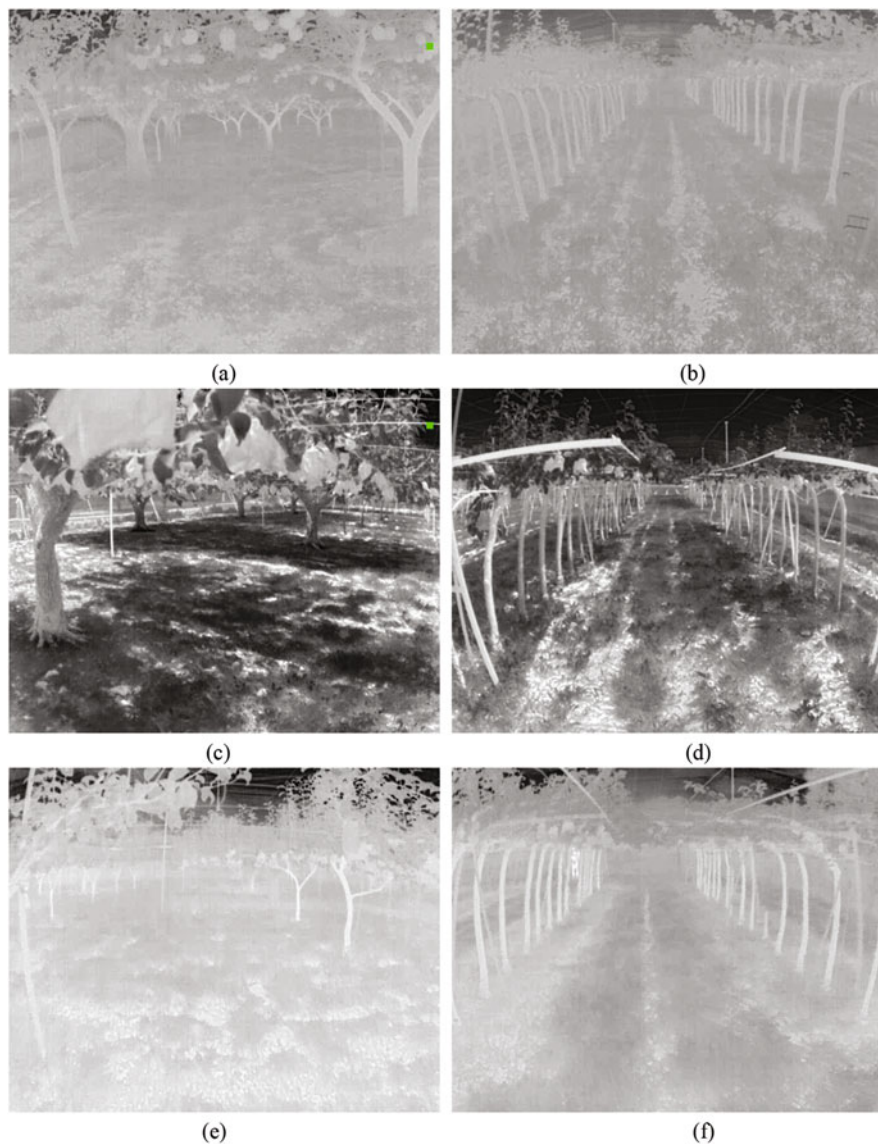
Fig. 10.2 Analysis of thermal images under different lighting conditions (**a**, **b**) no-light conditions (7–8 PM). (**c**, **d**) high-light conditions (12–2 PM). (**e**, **f**) low-light conditions (5–6 PM)

10.2.2.3   Data Augmentation

To obtain a larger database and improve training accuracy, the original 5313 images were randomly flipped and rotated at random angles to obtain 7563 new images, and

the labeling boxes of the images were also changed and saved as new .xml and .txt files.

10.2.2.4 Data Splitting

After labeling was completed, the data were divided into three sets: training, validation and testing. Training was conducted to adjust the model parameters, and the classifier was adjusted to increase accuracy. Validation was performed to check the state and convergence of the model developed during training, and validation sets were used to adjust hyperparameters and determine which set of hyperparameters was the most suitable. Testing was performed to evaluate the generalizability of the model, which was used for the validation set to determine the parameters.

10.2.3   Training Model Structure

10.2.3.1 Faster R-CNN (Faster Region Based Convolutional Neural Networks)

The faster R-CNN model was based on Fast R-CNN. Replacing the selective search in Fast R-CNN with Region Proposal Network (RPN), to achieve more accurate regional proposals, reduce the redundancy of network computation, and improve detection speed. The primary part of the model has convolutional layers, RPN, region of interest (ROI) pooling, and classification. In the Faster R-CNN model, the image's feature maps were got through convolution layers. RPN created region proposals and sent to Region of Interest (ROI) pooling. ROI pooling combine the feature maps and proposals to extract proposal feature maps. Classification part distinguished the classification of proposals and decided the exact location of bounding boxes (Fig. 10.3, 10.4).
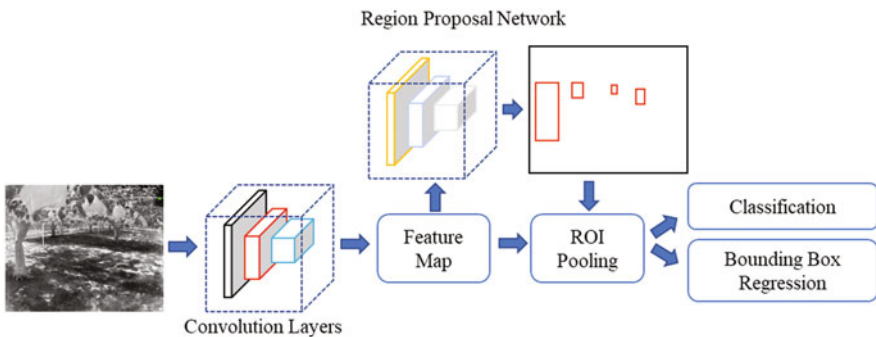
Fig. 10.3   Faster R-CNN structure for tree truck detection used in this research
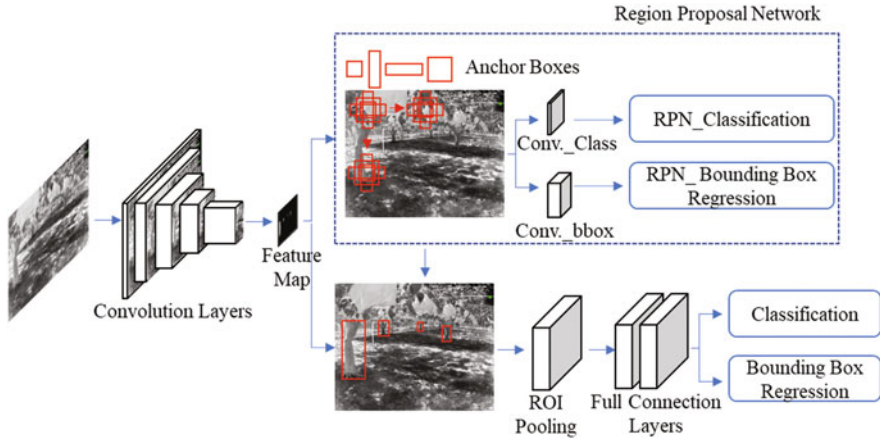
Fig. 10.4 Faster R-CNN network structure focusing regional proposal network for feature map

10.2.3.1.1 Convolutional Layers

As a CNN network target detection method, faster R-CNN first used a set of basic convolutional + ReLU + pooling layers to extract feature maps of images for training datasets. The feature maps were shared for subsequent RPN layers and fully connected layers. The feature extraction process in Faster R-CNN is the same as that in CNN and can be done using some common structures, such as the commonly used VGG and ResNet (Mahmud et al., 2021). In this study, VGG16 model was be used, which was a convolutional neural network model that was proposed (Simonyan & Zisserman, 2014). VGG can be divided into six configurations (A, A-LRN, B, C, D, E) according to the size of the convolutional kernel and the number of convolutional layers. This study used the D configuration with 13 convolutional layers, 3 fully connected layers, and 5 pooling layers. The convolutional layer and fully connected layer had weight coefficients called weight layers (Fig. 10.5).

The convolutional layer was used to extract features. A 3×3 convolutional kernel was used to slide over the image, and the output changed the image size because both the stride and padding setting value were 1. Because the linear model cannot solve all problems, the ReLU layer was used to add nonlinear factors to simulate more subtle changes (Oostwal et al., 2021). The pooling layer compressed the input feature map to make the feature map smaller and simplify the computational complexity of the network, and feature compression was used to extract the primary features. The pooling layer used a 2×2 convolutional kernel with a stride of 2 to perform the max pooling in the feature map. Therefore, the output feature map size was reduced by half. Finally, a fully connected layer connected all of the feature maps and transmitted the output to the softmax classifier.

In this VGGNet, there were five convolutional segments, all of which used 3×3 convolutional kernels because the 3×3 convolutional kernels can have fewer

224 x 224 x 3 224 x 224 x 64

112 x 112 x 128

56 x 56 x 256

28 x 28 x 512

7 x 7 x 512

14 x 14 x 512

1 x 1 x 4096 1 x 1 x 1000

☐ Convolution + ReLU
🟥 Max pooling
🟦 Fully connected + ReLU
🟨 Softmax
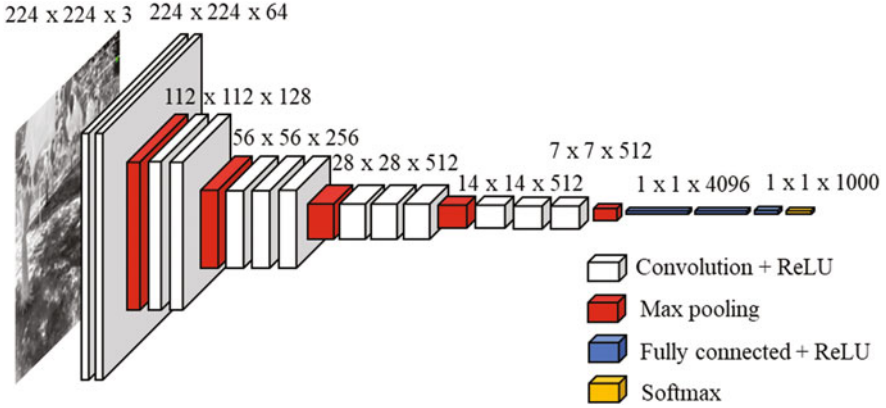
Fig. 10.5 VGG16 for target of tree trunk detection

parameters and have more nonlinear transformations to increase the CNN's ability to learn features (Simonyan & Zisserman, 2014).

10.2.3.1.2 RPN

This network model was a new structure proposed by Faster R-CNN, which determines the approximate position of the target from the feature maps. RPN first generates many anchors (the candidate boxes) on the feature map. Then, the RPN network was divided into two lines: the upper line distinguished whether it is a tree trunk by softmax classification and the lower line was used to calculate the bounding box regression offset for the anchors to obtain an accurate proposal (Li et al., 2021). The final proposal layer combined the anchors of the tree trunk, and the corresponding bounding box regression offsets to obtain the proposals and eliminate the proposals that were too small or out of bounds.

10.2.3.1.3 ROI Pooling

After the RPN network had created the proposals, ROI pooling was applied to the proposals, and the feature maps were generated by the last layer in the VGG16 network. Also, a fixed-size proposal feature map was obtained and sent to localization and recognition.

10.2.3.1.4 Classification

Softmax was used to classify the target objects of tree trunk detection. The output classification contained the probability of being the target as a tree trunk. Bounding

box regression offset each proposal and was used to predict the target detection box more accurately.

The total loss function of Faster R-CNN consisted of classification loss and regression loss. The classification loss was calculated by softmax in the RPN classification layer, which was used to classify the anchors as positive and negative for training the network. Conversely, regression loss was calculated by the RPN bounding box regression layer and used to train the bounding box regression network. Therefore, the total loss function can be expressed as Eq. (10.1):

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i,p_i^*) + \lambda \frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i,t_i^*) \qquad (10.1)$$

where i is the anchor index, p_i represents the positive probability of anchor I, p_i^* is the ground-truth label (if p_i^* is 1, the anchor is positive; if p_i^* is 0, the anchor is negative), t is the predicted bounding box, t_i is the vector representing the four parameterized coordinates of the predicted bounding box, and t_i^* is a positive anchor associated with the ground-truth box.

N_{cls} is the trunk image during the training process, and N_{reg} is the number of anchor locations. As the difference between N_{cls} and N_{reg} was too large in the real process, the parameter λ was used to balance the two parameters. Therefore, the network total loss calculation process was considered for two types of losses: L_{reg} is the regression and L_{cls} is the classification loss function for the detection of trunks. The smooth function (L1) was used to estimate regression loss (L_{reg}), which is calculated and expressed as Eq. (10.2):

$$L_{reg}(t_i, t_i^*) = \sum_{i\in\{x,\ y,\ w,\ h\}} \text{smooth}_{L1}(t_i - t_i^*) \qquad (10.2)$$

The smooth function is defined as Eq. (10.3):

$$\text{soomth}_{L1}(x) = \begin{cases} 0.5x^2 \dfrac{1}{\sigma^2} & |x| \le \dfrac{1}{\sigma^2} \\ |x| - 0.5 & \text{otherwise} \end{cases} \qquad (10.3)$$

where x is the prediction error of the bounding box and the parameter σ is used to control the smoothing area.

For the bounding box regression, we used the four coordinates in the following parameterized expressions (10.4)–(10.11):

$$t_x = \frac{(x - x_a)}{w_a} \qquad (10.4)$$

$$t_y = \frac{(y - y_{\mathrm{a}})}{h_{\mathrm{a}}} \tag{10.5}$$

$$t_w = \log\left(w/w_{\mathrm{a}}\right) \tag{10.6}$$

$$t_h = \log\left(h/h_{\mathrm{a}}\right) \tag{10.7}$$

$$t_x^* = \frac{(x^* - x_{\mathrm{a}})}{w_{\mathrm{a}}} \tag{10.8}$$

$$t_y^* = \frac{(y^* - y_{\mathrm{a}})}{h_{\mathrm{a}}} \tag{10.9}$$

$$t_w^* = \log\left(w^*/w_{\mathrm{a}}\right) \tag{10.10}$$

$$t_h^* = \log\left(h^*/h_{\mathrm{a}}\right) \tag{10.11}$$

where x, y, w, and h denote the center coordinates of the box and its width and height, respectively. The variables x, x_{a}, and x^* are used for the predicted box, the anchor box, and the ground-truth box, respectively, which can be considered a bounding box regression from the anchor box to the nearby real box.

10.2.3.2   YOLO (You Only Look Once)

YOLO's CNN network segments the input image into $S \times S$ grids, and predict the bounding boxes and the confidence score (Fig. 10.6). The confidence score contained two aspects, one was the probability of this bounding box containing the target, and the other was the accuracy of this bounding box. The confidence score
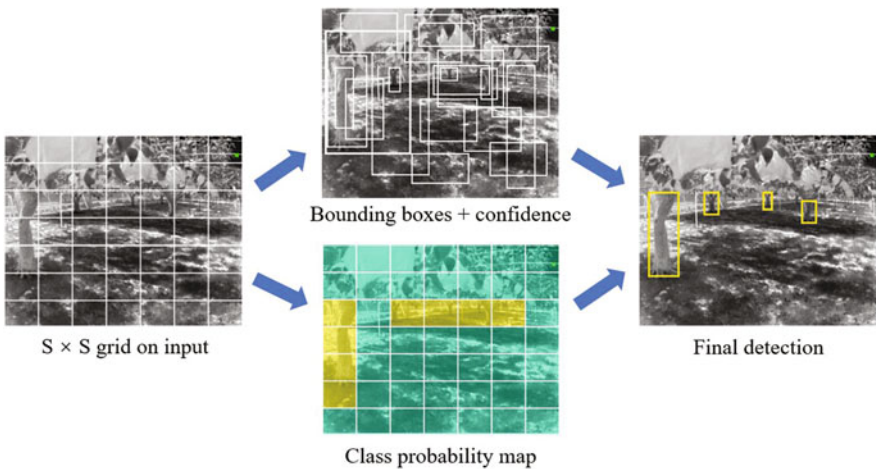
Fig. 10.6 The workflow of YOLO

reflected the probability that the target in the bounding box belongs to each category and how closely the bounding box matches the target.

In Chaps. 9 and 11 of this book, the structure of YOLO has been explained in detail, this chapter does not repeat.

10.2.3.3 CenterNet

CenterNet was an anchor-free target detection network, which was faster and more accurate because it located the detected target to a single point, saved the time and operation of listing bounding box. Compared with CornerNet, the same anchor-free detection method, CornerNet needed to perform two key point detection (upper left point and lower right point) to determine the position of the object, while CenterNet only needed to perform one key point detection (center point) to determine the position, which was simpler than CornerNet.

The structure of CentetNet was very simple (Fig. 10.7). The input image was scaled to 512×512 and then inputted into convolution neural network (e.g. Resnet, Hourglass), and the feature maps extracted by the neural network were sent to center point, offset, boxes size three branches for prediction to obtain the results.

The center got from the feature map as shown in Fig. 10.8, which will be transformed to a heatmap (Fig. 10.8(a)), the location of the object which had the highest value was the center (Fig. 10.8(b)). The center should be surrounded, the offset was used to assess the deviation of centers between the feature map and the original image (Fig. 10.8(c)) (Zhao & Yan, 2021).
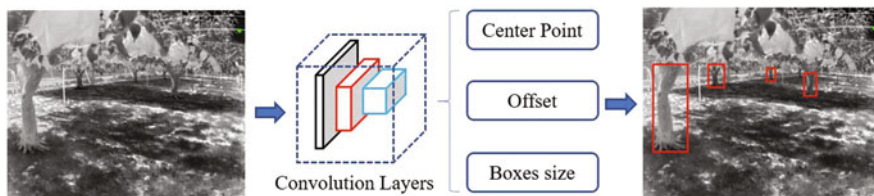
Fig. 10.7 The structure of CentetNet

a) keypoint heatmap   (b) local offset   (c) object size
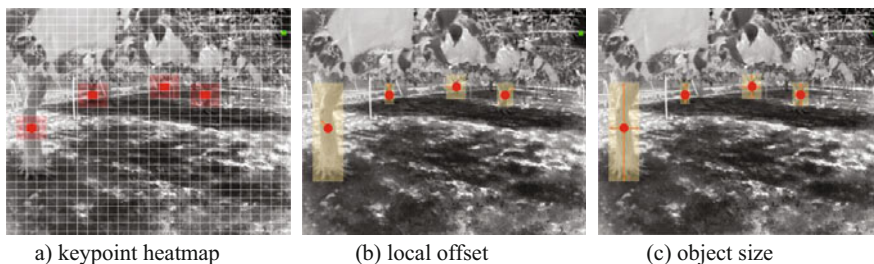
Fig. 10.8 (**a–c**) The workflow of CenterNet

The input image was denoted as $I \in R^{W \times H \times 3}$, W is the width of our image, the height is H, and 3 means picture RGB channels. The heatmap of the center keypoint is $\widehat{Y} \in [0, 1]^{\frac{W}{R} \times \frac{H}{R} \times C}$, R is the downsampling rate, C is the number of classes, in this research, just has one class of tree. $\widehat{Y}_{xyc} = 1$ means that the point is center keypoint for class C, $\widehat{Y}_{xyc} = 0$ means it is not detected or it is background. As show in Fig. 10.8 (a), the darker the color, the closer to the object.

The ground truth (GT) trained by CenterNet is a little different from other target detection networks, it is more similar to the segmentation network, the input GT is a feature map generated based on the target box. For example, the center point p of the ground truth needs to be calculated, $p = \left(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}\right)$, where x and y are the coordinates in the GT. But the feature map was downsampled, p is also downsampled by using R, $\widetilde{p} = \left[\frac{p}{R}\right]$. $\widetilde{p}$ is the center point of truth data in the feature map. To make the training process smoother, a Gaussian kernel was used to do the smoothing of GT, where the size of the Gaussian-processed values of the pixels around p points is

$$Y_{xyc} = \exp\left(-\frac{(x-\widetilde{p}_x)^2 + (y-\widetilde{p}_y)^2}{2\sigma_p^2}\right) \tag{10.12}$$

where σ_p is the standard deviation obtained by adapting to the size of the current target p.

10.2.4 Training Platform and Validation

The hardware environment used in this study included Windows 10, Intel(R) Core (TM) i7-10750H CPU @ 2.60 GHz, 32.0 GB of RAM, and an NVIDIA GeForce RTX 2060. The software environment included Anaconda 3, Python 3.5, and TensorFlow-GPU 1.13.1. In the experiment, a total of 12,876 infrared images of orchard trunks were used for training, 9270 images in the training datasets, and 2318 images in the validation datasets.

10.2.5 Model Testing

To verify model reliability and stability, 1288 images were selected as the test set for validation after the model was trained. In this paper, the precision-recall curve was obtained by calculating precision and recall to highlight the trade-off between precision (P) and recall (R) of the classification model. The area under the curve was calculated to obtain the average precision (AP) of the model. Because there was

only one category in this experiment, AP was the same as mAP (mean average precision). The P, R, and AP can be expressed as expressions (10.13)–(10.15):

$$P = \frac{TP}{TP + FP} \tag{10.13}$$

$$R = \frac{TP}{TP + FN} \tag{10.14}$$

$$AP = \int_0^1 P dR \tag{10.15}$$

where TP is the true positive value (i.e., the correct detection box); FP is the false-positive value (i.e., the false detection box that predicts the background as the target); and FN is the false negative value (i.e., the missed detection box). During testing, 1288 images were used as testing datasets in the three model.

10.3 Results

10.3.1 Faster R-CNN Testing

The number of iterations were 40,000, and the losses were reported as total loss, bounding box loss, classification loss, regional proposal network (RPN) classification loss, and RPN bounding box loss (Fig. 10.9). According to Fig. 10.9, the total loss function dropped to approximately 0.6 and oscillated near 0.6 when training reached 40,000 iterations, effectively reaching convergence. Also, both the bounding box loss and classification loss converged; thus, training was stopped at 40,000 iterations.

The mAP of the Faster R-CNN model was found to be 0.8378 (Fig. 10.10). The images for testing were considered based on the training results conducted for the high-, low-, and no-light conditions (Fig. 10.11). Two images were selected at different time periods.

10.3.2 YOLO Testing

Same as Faster R-CNN, YOLO's loss were tend to be smooth after 40,000 iterations. The training was stopped at 40,000 iterations and the loss value were decreased to 0.7 (Fig. 10.12). Precision-recall curve of YOLO is shown in Fig. 10.13, and its accuracy was only 0.4. The same images were selected for testing (Fig. 10.14), and it was found that YOLO could not detect tree trunks in no-light condition, and only some tree trunks in high-light and low-light can be found, and in the joint tree orchard, YOLO cannot be used at all.

(a)

(b)

(c)

(d)

(e)
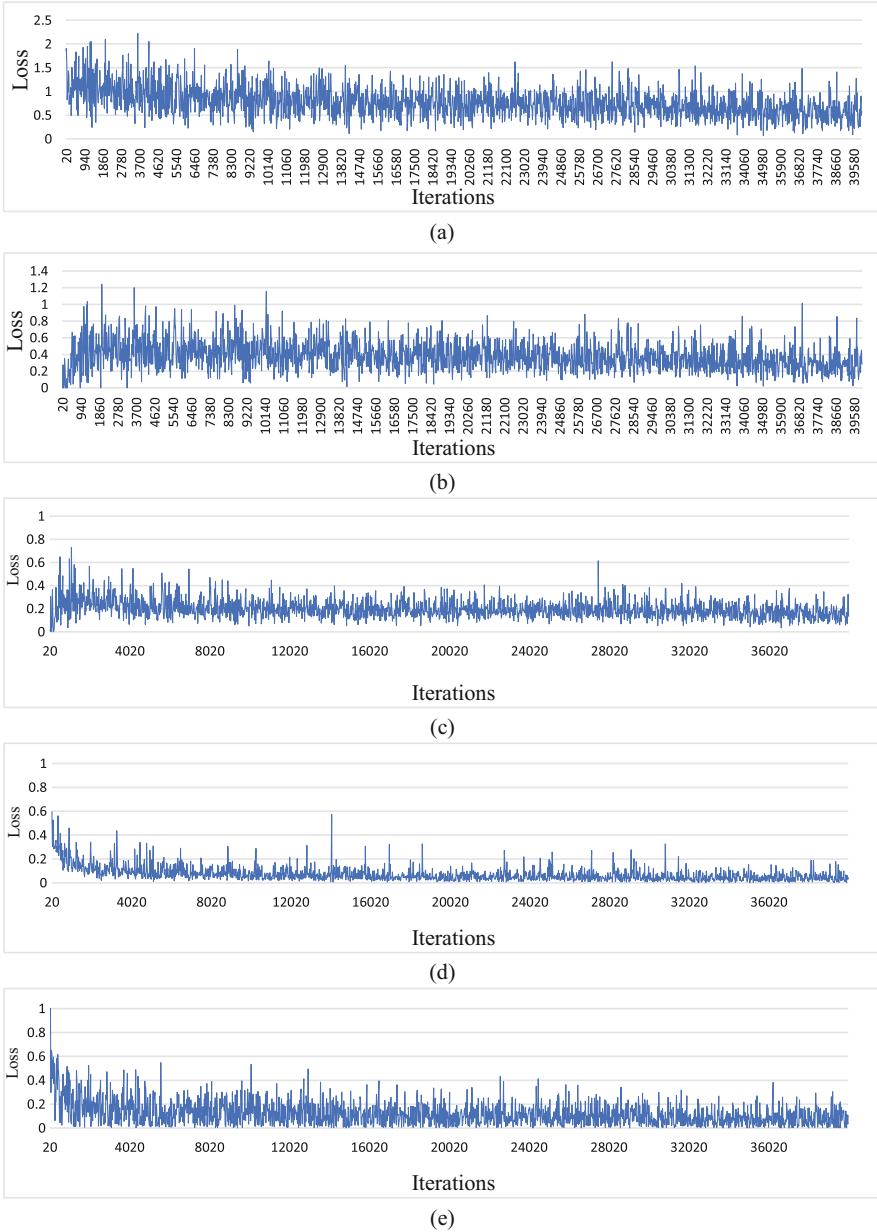
Fig. 10.9 Faster R-CNN loss images. (**a**) Total loss. (**b**) Bounding box loss. (**c**) Classification loss. (**d**) Regional Proposal Network classification loss. (**e**) Regional Proposal Network bounding box loss
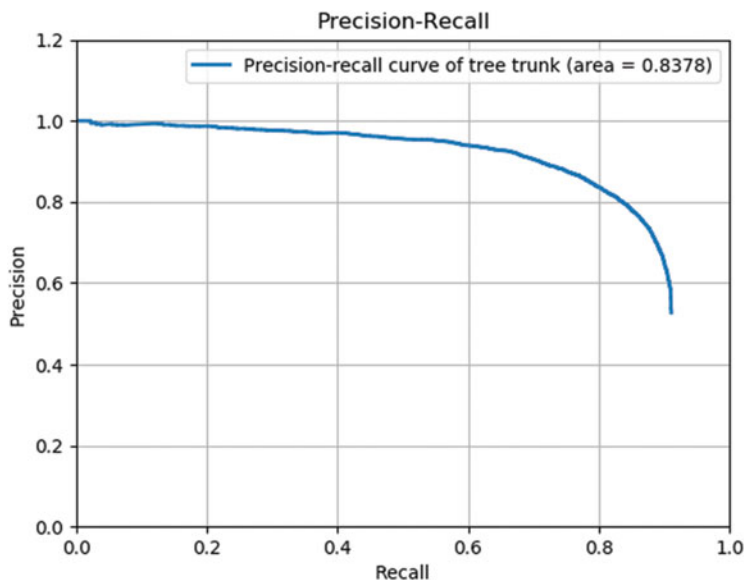
Fig. 10.10 Precision-recall curve of Faster R-CNN

10.3.3 CenterNet Testing

Different from the iteration times of Faster R-CNN and YOLO, CenterNet training was done by epoch. The training was set 60 epochs, and the loss had become smoothly at the end of the training (Fig. 10.15). CenterNet automatically saved the best result during training, and the best result was obtained at the 45th epoch in this study, the loss was 0.78. The precision-recall curve obtained with the best results is shown in Fig. 10.16, and its accuracy was reached at 0.94. Selected the similar images for testing, CenterNet also could accurately identify tree trunks (Fig. 10.17).

10.4 Discussion

Machine vision typically analyzed RGB imagery and can thus not manage low-light conditions to recognize natural landmarks. Conversely, GNSS suffered signal interruptions due to dense canopies inside the orchards. LiDAR also had difficulties recognizing tree trunks because the large scanning range decreases the mistake of recognizing tree trunks. To increase the robustness of these methods, this study used a thermal camera combined with deep learning to identify fruit tree trunks under high-, low-, and no-light conditions. The loss function shown in the results section represents the difference between the predicted value of the model and the training samples. A smaller value indicates that the closer the predicted sample is to the real

Fig. 10.11 Image results of Faster R-CNN test: (**a**–**d**) no-light conditions, (**e**–**h**) high-light conditions, and (**i**–**l**) low-light conditions

(i)

(j)

(k)

(l)
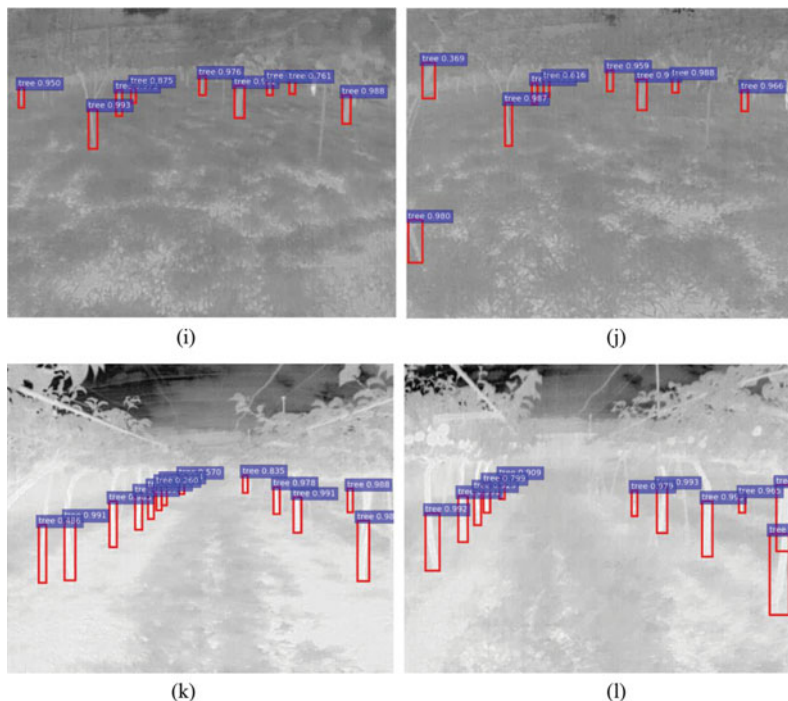
Fig. 10.11 (continued)

sample, the more robust the model was. In contrast, a larger value indicated that the difference between the predicted and real samples was larger (Zhang et al., 2020).

By training Faster R-CNN, YOLO, and CenterNet with the same dataset, through plotting precision-recall and testing the same images, it can be found that YOLO was less accurate in thermal images in this study, and Faster R-CNN and CenterNet can accurately identify the target. In the testing results in the conventional and joint orchards, this model was able to recognize tree trunks accurately.

Although the precision-recall curve showed that the accuracy of Faster R-CNN was lower than that of CenterNet, Faster R-CNNN was able to find more targets with higher target accuracy than CenterNet by image testing. In the joint tree orchard, the tree trunk covered by the shelf is detected by Faster R-CNN (Fig. 10.11g, h), but not by CenterNet (Fig. 10.17g, h). Faster R-CNN sometimes had detection errors, as in Fig. 10.11d, Faster R-CNN detected the orchard net as a target, but the detection error probability of Faster R-CNN was smaller compared to the targets missed by CenterNet. By the above comparison, Faster R-CNN was more suitable to be applied in this study.

However, results showed that the converged losses did not reach below 0.1, while other studies had less than 0.1 (Jiang et al., 2021); these results likely occurred was because the two types of trees (conventional and joint) were investigated in this study, which differ markedly in shape. Also, in the conventional orchard, the trees

Fig. 10.12 YOLO-v3 loss image
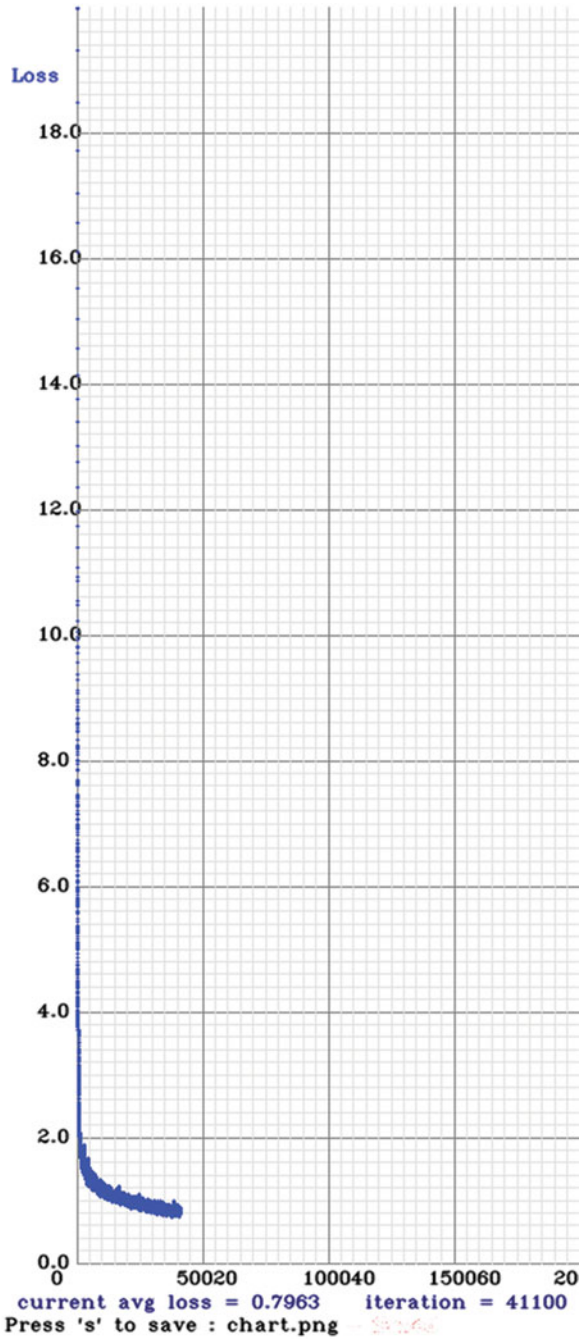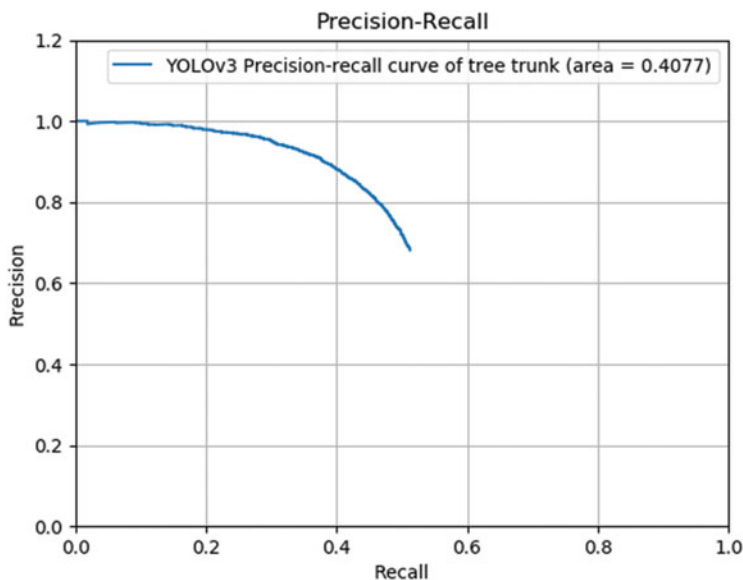
Fig. 10.13 Precision-recall curve of YOLO-v3

differed markedly in size, with some trees being 30 cm wide and some less than 5 cm, which led to a larger loss convergence result. In addition, randomly selected images were used to prepare the validation set. According to the validation results, the model works properly to detect the tree trunk in conventional and joint orchards under high-, low-, and no-light conditions, and had a higher accuracy rate. The poles and shelves in the orchards can be distinguished from the fruit tree trunk. In the rotated images, the trunk could be identified accurately.

In the joint tree system, using a thermal camera had more flexibility to detect the tree trunk due to the uniform shape of the tree and growth in rows. However, natural orchards have different types, and the shapes of the field and canopy coverages can be markedly different. Traditionally, orchard growers in Japan control pruning and training according to their heights of operations, which makes using machinery inconvenient. Joint tree training systems have the advantage of uniform growing and machinery automation systems. Therefore, the thermal camera used in the research has a high application potential in join tree training systems to use for orchard navigation in combination with other positional sensors such as LiDAR. In previous studies, LiDAR was able to detect artificial landmarks and cones for use in positional algorithms in orchard navigation (Thanpattranon et al., 2015, 2016). In future research, a thermal camera and LiDAR are planned to be used to navigate speed sprayers in an orchard.
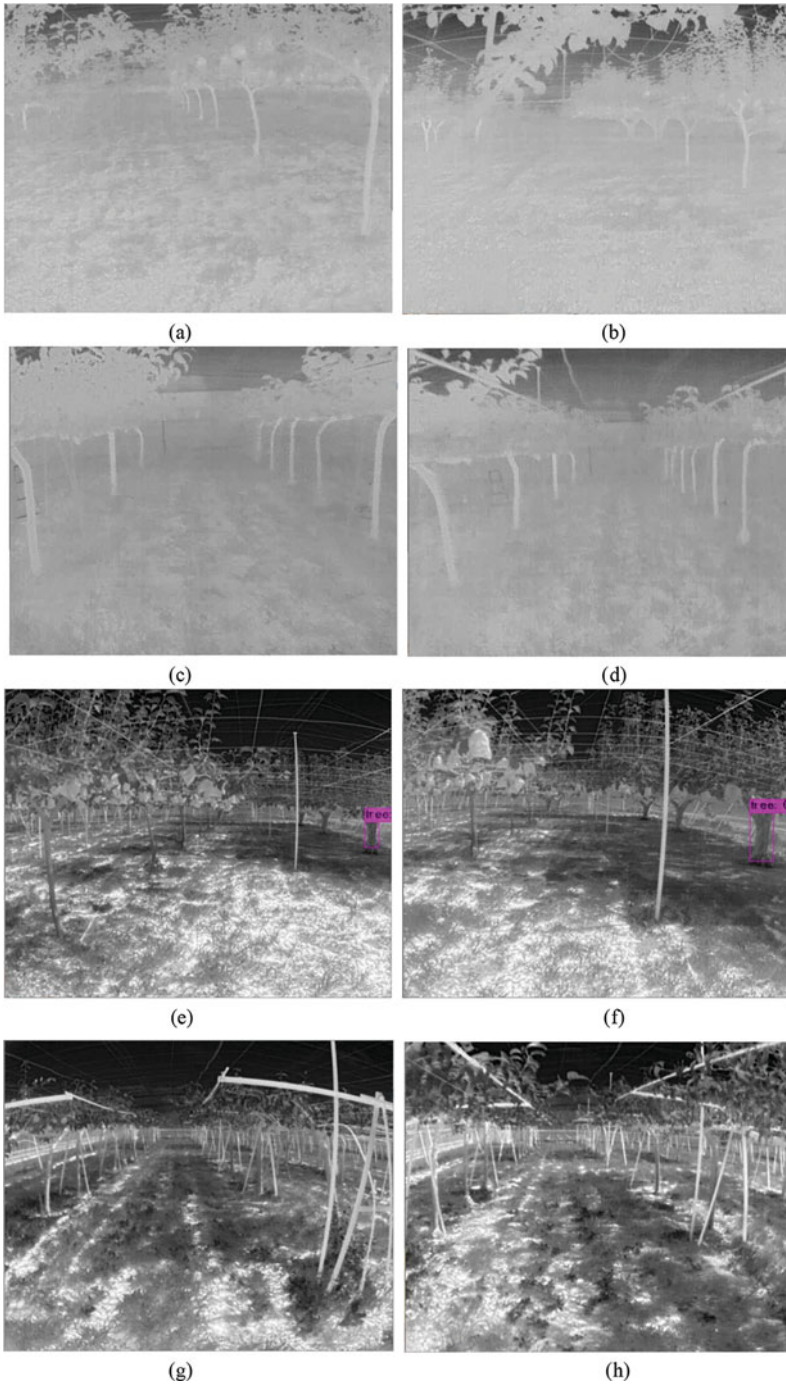
Fig. 10.14 Image results of YOLO-v3 test: (**a**–**d**) no-light conditions, (**e**–**h**) high-light conditions, and (**i**–**l**) low-light conditions
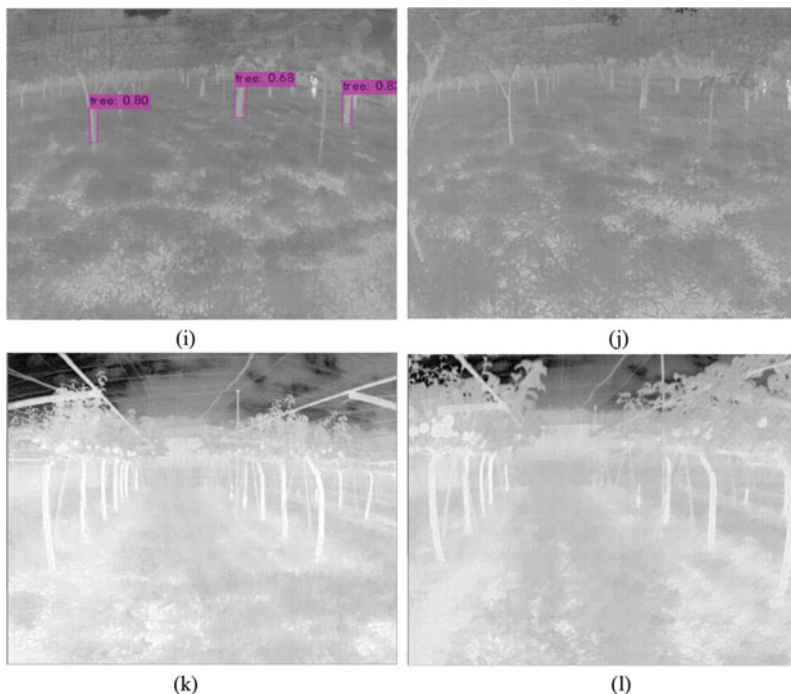
Fig. 10.14 (continued)

10.5   Conclusion

Thermal imagery had the potential to detect natural landmarks under different lighting conditions. This study proposes using the Faster R-CNN, YOLO-v3 and CenterNet for thermal images to detect pear tree trunks to enable navigation under various lighting conditions in orchards. Conventional and joint tree training system orchards were considered for dataset collection. The accuracy and reliability of the model was verified by training and testing the thermal images, indicating that target detection could be performed in high-, low-, and no-light conditions using a thermal camera. In this study, all 640×512-pixel images and videos were used. CenterNet had higher accuracy compared to YOLO-v3 and Faster R-CNN using the same dataset, but Faster R-CNN could detect more objects and was most suitable in this study. The model could distinguish poles and shelves even they are same size with tree trunk, indicating that the thermal camera can be used for orchard navigation to detect tree trunks under different light conditions, and using this system can allow machinery to operate in orchards at any time. Therefore, the thermal camera can be used in high-, low-, or no-light conditions inside orchards to detect tree trunks using deep learning algorithm. However, due to the unavoidable error in labeling, the
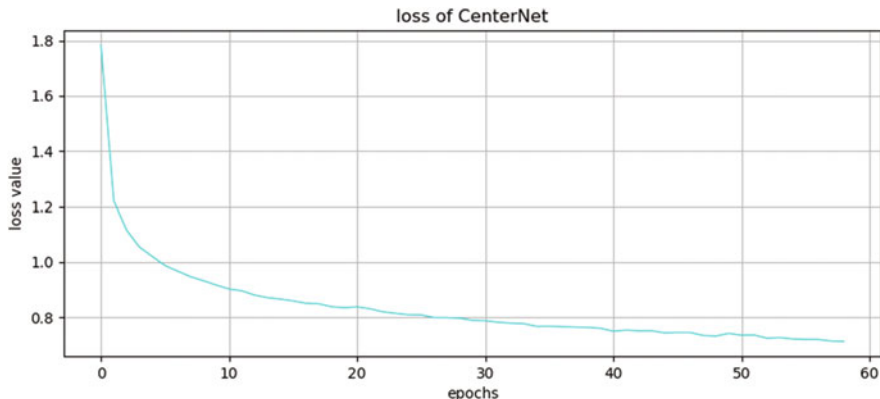
Fig. 10.15 CenterNet loss image
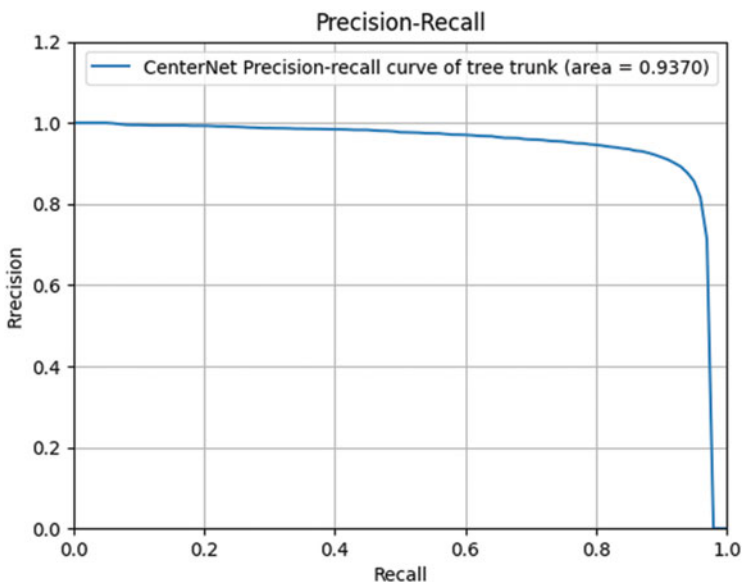
Fig. 10.16 Precision-recall curve of CenterNet

recognition accuracies decreased after 20m, such as omissions and identification errors. As the thermal camera is a monocular camera, distance measuring must be assisted by other sensors. In future research, an autonomous speed sprayer is planned to be used by installing the thermal camera and combine with LiDAR for orchard navigation under different light conditions to provide solutions in orchard automation to increase productivity.

Fig. 10.17 Image results of CenterNet test: (**a**–**d**) no-light conditions, (**e**–**h**) high-light conditions, and (**i**–**l**) low-light conditions

Fig. 10.17 (continued)

Conflicts of Interest The authors declare no conflicts of interest.

References

Abdullah, B., & Mücahit Taha, Ö. (2021). Canopy analysis and thermographic abnormalities determination possibilities of olive trees by using data mining algorithms. *Notulae Botanicae Horti Agrobotanici Cluj-Napoca, 49*(1), 12139. https://doi.org/10.15835/nbha49112139

Ahamed, T., Takigawa, T., Koike, M., Honma, T., Hasegawa, H., & Zhang, Q. (2006a). Navigation using a laser range finder for autonomous tractor (Part 2) navigation for approach composed of

multiple paths. *Journal of the Japanese Society of Agricultural Machinery, 68*(1), 78–86. https://doi.org/10.11357/jsam1937.68.78

Ahamed, T., Takigawa, T., Koike, M., Honma, T., Hasegawa, H., & Zhang, Q. (2006b). Navigation using a laser range finder for autonomous tractor (Part 1). *Journal of the Japanese Society of Agricultural Machinery, 68*, 68–77.

Ahamed, T., Tian, L., Takigawa, T., & Zhang, Y. (2009). Development of auto-hitching navigation system for farm implements using laser range finder. *Transactions of the ASABE, 52*(5), 1793–1803. https://doi.org/10.13031/2013.29120

Bergerman, M., Billingsley, J., Reid, J., & van Henten, E. (2016). Robotics in agriculture and forestry. In *Springer handbook of robotics* (pp. 1463–1492). Springer.

Dong, Y. (2018). *Japan: Aging of the agricultural labor force and its solutions*. World food Prize Foundation.

Guo, Z., Li, X., Xu, Q., & Sun, Z. (2021). Robust semantic segmentation based on RGB-thermal in variable lighting scenes. *Measurement, 186*, 110176. https://doi.org/10.1016/j.measurement.2021.110176

Hespeler, S. C., Nemati, H., & Dehghan-Niri, E. (2021). Non-destructive thermal imaging for object detection via advanced deep learning for robotic inspection and harvesting of chili peppers. *Artificial Intelligence in Agriculture, 5*, 102–117. https://doi.org/10.1016/j.aiia.2021.05.003

Jiang, D., Li, G., Tan, C., Huang, L., Sun, Y., & Kong, J. (2021). Semantic segmentation for multiscale target based on object recognition using the improved Faster-RCNN model. *Future Generation Computer Systems, 123*, 94–104. https://doi.org/10.1016/j.future.2021.04.019

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Paper presented at the Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, Lake Tahoe, Nevada.

Li, M., Imou, K., Wakabayashi, K., & Yokoyama, S. J. (2009). Review of research on agricultural vehicle autonomous guidance. *International Journal of Agricultural and Biological Engineering, 2*(3), 1–16.

Li, Z., Li, Y., Yang, Y., Guo, R., Yang, J., Yue, J., & Wang, Y. (2021). A high-precision detection method of hydroponic lettuce seedlings status based on improved Faster RCNN. *Computers and Electronics in Agriculture, 182*, 106054. https://doi.org/10.1016/j.compag.2021.106054

Mahmud, M. S., Zahid, A., Das, A. K., Muzammil, M., & Khan, M. U. (2021). A systematic literature review on deep learning applications for precision cattle farming. *Computers and Electronics in Agriculture, 187*, 106313. https://doi.org/10.1016/j.compag.2021.106313

Malavazi, F. B. P., Guyonneau, R., Fasquel, J.-B., Lagrange, S., & Mercier, F. (2018). LiDAR-only based navigation algorithm for an autonomous agricultural robot. *Computers and Electronics in Agriculture, 154*, 71–79. https://doi.org/10.1016/j.compag.2018.08.034

Oostwal, E., Straat, M., & Biehl, M. (2021). Hidden unit specialization in layered neural networks: ReLU vs. sigmoidal activation. *Physica A: Statistical Mechanics and its Applications, 564*, 125517. https://doi.org/10.1016/j.physa.2020.125517

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

Subramanian, V., Burks, T. F., & Arroyo, A. A. (2006). Development of machine vision and laser radar based autonomous vehicle guidance systems for citrus grove navigation. *Computers and Electronics in Agriculture, 53*(2), 130–143. https://doi.org/10.1016/j.compag.2006.06.001

Takagaki, A., Masuda, R., Iida, M., & Suguri, M. (2013). Image processing for ridge/furrow discrimination for autonomous agricultural vehicles navigation. *IFAC Proceedings, 46*(18), 47–51. https://doi.org/10.3182/20130828-2-SF-3019.00045

Takai, R., Barawid, O., Ishii, K., & Noguchi, N. (2010). Development of crawler-type robot tractor based on GPS and IMU. *IFAC Proceedings, 43*(26), 151–156. https://doi.org/10.3182/20101206-3-JP-3009.00026

Thanpattranon, P., Ahamed, T., & Takigawa, T. (2015). Navigation of an autonomous tractor for a row-type tree plantation using a laser range finder—Development of a point-to-go algorithm. *Robotics, 4*(3), 341–364.

Thanpattranon, P., Ahamed, T., & Takigawa, T. (2016). Navigation of autonomous tractor for orchards and plantations using a laser range finder: Automatic control of trailer position with tractor. *Biosystems Engineering, 147*, 90–103. https://doi.org/10.1016/j.biosystemseng.2016.02.009

Vadlamudi, S. (2019). How artificial intelligence improves agricultural productivity and sustainability: A global thematic analysis. *Asia Pacific Journal of Energy and Environment, 6*(2), 91–100. https://doi.org/10.18034/apjee.v6i2.542

Zhang, Y., Chen, H., He, Y., Ye, M., Cai, X., & Zhang, D. (2018). Road segmentation for all-day outdoor robot navigation. *Neurocomputing, 314*, 316–325. https://doi.org/10.1016/j.neucom.2018.06.059

Zhang, H., Tian, Y., Wang, K., Zhang, W., & Wang, F. Y. (2020). Mask SSD: An effective single-stage approach to object instance segmentation. *IEEE Transactions on Image Processing, 29*, 2078–2093. https://doi.org/10.1109/TIP.2019.2947806

Zhao, K., & Yan, W. Q. (2021). Fruit detection from digital images using CenterNet. Paper presented at the Geometry and Vision, Cham.

Zhou, X., Wang, D., & Krähenbühl, P. (2019). Objects as points. *arXiv*, 1904.07850.

Chapter 11
Real-Time Pear Fruit Detection and Counting Using YOLOv4 Models and Deep SORT

Addie Ira Borja Parico and Tofael Ahamed

Abstract This study aimed to produce a robust real-time pear fruit counter for mobile applications using only RGB data, the variants of the state-of-the-art object detection model YOLOv4, and the multiple object-tracking algorithm Deep SORT. This study also provided a systematic and pragmatic methodology for choosing the most suitable model for a desired application in agricultural sciences. In terms of accuracy, YOLOv4-CSP was observed as the optimal model, with an AP@0.50 of 98%. In terms of speed and computational cost, YOLOv4-tiny was found to be the ideal model, with a speed of more than 50 FPS and FLOPS of 6.8–14.5. If considering the balance in terms of accuracy, speed, and computational cost, YOLOv4 was found to be most suitable and had the highest accuracy metrics while satisfying a real time speed of greater than or equal to 24 FPS. Between the two methods of counting with Deep SORT, the unique ID method was found to be more reliable, with an $F1_{count}$ of 87.85%. This was because YOLOv4 had a very low false negative in detecting pear fruits. The ROI line is more reliable because of its more restrictive nature, but due to flickering in detection it was not able to count some pears despite their being detected.

Keywords YOLO · YOLOv4 · Deep SORT · Object counting · Real time · Object detection · Fruit detection

A. I. B. Parico
Graduate School of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan

Corpy & Co., Inc., Tokyo, Japan

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

11.1 Introduction

Pear farmers in typically count their yield manually and tend to have higher post-harvest losses due to short perishability and packaging arrangements. In addition, quicker decision making is necessary in an extreme climatic event situation. To make this process easier for farmers, a mobile phone-based application for counting pears in real time was conceptualized to support the logistics management of pears. However, this would require a fast and accurate detection method that is not computationally expensive. Taking a video from under the trees would require robustness due to challenges such as illumination and occlusion.

Deep learning algorithms have proven to be the most robust way for object detection (Kamilaris & Prenafeta-Boldú, 2018a, b). Considering accuracy and speed, YOLOv4 (You Only Look Once) (Bochkovskiy et al., 2020) has been the top performer for object detection models recently. YOLOv4 runs two times faster than a recent state-of-the-art object detection model, EfficientDet, at a similar accuracy. More importantly, YOLOv4 was designed to enable training on a single conventional GPU, unlike other models. After the development of YOLOv4, Wang et al. (2020a) modified the structure of YOLOv4 to enable scaling for different applications. YOLOv4-tiny was designed to maximize speed and to achieve the minimum computational cost possible. Then, YOLOv4-CSP and other larger versions of YOLOv4 were developed to maximize accuracy with varying computational requirements. In this study, the authors evaluated the speed-accuracy-memory tradeoff among YOLOv4, YOLOv4-tiny, and YOLOv4-CSP on detecting pear fruits.

One can use detection alone for counting objects. However, some of the most common problems in detection systems are flickering, failure to detect the object under occlusion and challenging illumination. Therefore, relying completely on the number of detections for the pear count could lead to errors, especially in a pear orchard scenario where the abovementioned challenges are common. For that reason, a back-up system should cover for this limitation to ensure accuracy in counting, which can be through object tracking. With object tracking, a unique ID can be assigned to a detection, thus, giving a more reliable measure of object count just in case the detection system fails. Deep SORT (Simple Online Realtime Tracking with Deep Association Metric) has been proven to be one of the fastest and most robust algorithms for multiple object tracking (Wojke et al., 2017). Deep SORT was found to have runtime speed of 25–50 FPS using recent conventional GPUs (Wojke et al., 2017). Due to its suitability for real-time tracking and robustness, Deep SORT was selected as the tracking algorithm in this study for counting the pear fruits in real time.

Several studies have utilized YOLO-based models for fruit detection and have demonstrated that YOLO models have a huge potential in accurate real time detection of fruits in an orchard (Koirala et al., 2019; Liu et al., 2020; Yan et al., 2021; Lawal 2021; Fu et al., 2020; Gai et al., 2021; Kang & Chen, 2020; Kuznetsova et al., 2020a; Kuznetsova et al., 2021; Wu et al., 2021). However, there were some

concerns found among these studies. First, no study simultaneously considered the parameters of detection accuracy, inference speed, and computational cost, which are very important for the optimization of the detection algorithm. Second, most of the studies did not report loss curves for their models, thus, it is difficult to verify if overfitting or underfitting of their dataset occurred. It is also difficult to confirm if they had truly achieved the highest possible performance metrics exhaustively without overfitting. Finally, reported research only focused on detection and did not address the possibility of counting the objects in real time, which is a completely different matter.

There was one study found where YOLO was combined with a classical tracking algorithm (Kalman filter) for counting the fruits with an F1 score of 0.972 (Itakura et al., 2021). However, the speed of the counting system was not mentioned. In addition, it was not stated if the information processing was done in one batch or sequentially. Thus, it is difficult to confirm if their system is truly applicable in real time.

Considering these limitations in related studies, the main contribution of this study was to develop a real time fruit counting system through detection and tracking while evaluating the computational cost. The concept is that the pear counter counts the pear fruits from a video taken in real time based on a two-part system with a state-of-the-art detection algorithm YOLOv4, and a leading tracking algorithm Deep SORT. To do that, detection performance, inference speed, and computational cost were considered concurrently as target metrics for optimizing the detection algorithm.

Furthermore, a comprehensive, systematic, and pragmatic guide in performing an object detection study in an agricultural or life sciences application was provided. First, target metrics were clearly defined. Then, a modified data splitting scheme was proposed for cases where data mismatch existed between the available training and test datasets. Next, this research also provided a methodological training strategy that can guide the researcher to objectively diagnose problems that exist during training, such as overfitting, and under fitting. With the systematic guide provided in carrying out object detection research, high quality and reproducible results are ensured.

This article provides the problem statement and contributions to the agricultural production system for minimizing post-harvest losses in Sect. 11.1. In Sect. 11.2, more details about the YOLO models, Deep SORT, and other related studies are discussed. Section 11.3 lays out the systematic methodology in preparing the data, defining the target metrics, carrying out the training, validation, and optimization of the detection models, comparison of the YOLOv4 models (YOLOv4, YOLOv4-CSP, and YOLOv4-tiny), integration with the tracking algorithm and, finally, evaluation of the pear counting system. In Sect. 11.4, the results of the training, error analysis, model comparison, and evaluation of the pear counting system are presented. Finally, Sect. 11.5 outlines the conclusion and future directions of the study.

11.2 Related Works

11.2.1 *You Only Look Once (YOLO)*

Deep learning algorithms have been shown to be one of the most robust ways for approaching object detection (Kamilaris & Prenafeta-Boldú, 2018b). Considering accuracy and speed, YOLOv4 (Wang et al., 2020a) has been the top performer for object detection models recently. Joseph Redmon in 2016 developed the predecessor of YOLOv4, You Only Look Once (YOLO), which was considered one of the first convolutional neural networks (CNNs) with real-time speed. Its speed was attributed to its one-shot detection mechanism, where it simultaneously predicted the bounding box coordinates and class probabilities from an image (Redmon et al., 2015); YOLO divides an input image into $S \times S$ grids then predicts bounding boxes with corresponding confidences of having detected an object of class C (Fig. 11.1). A
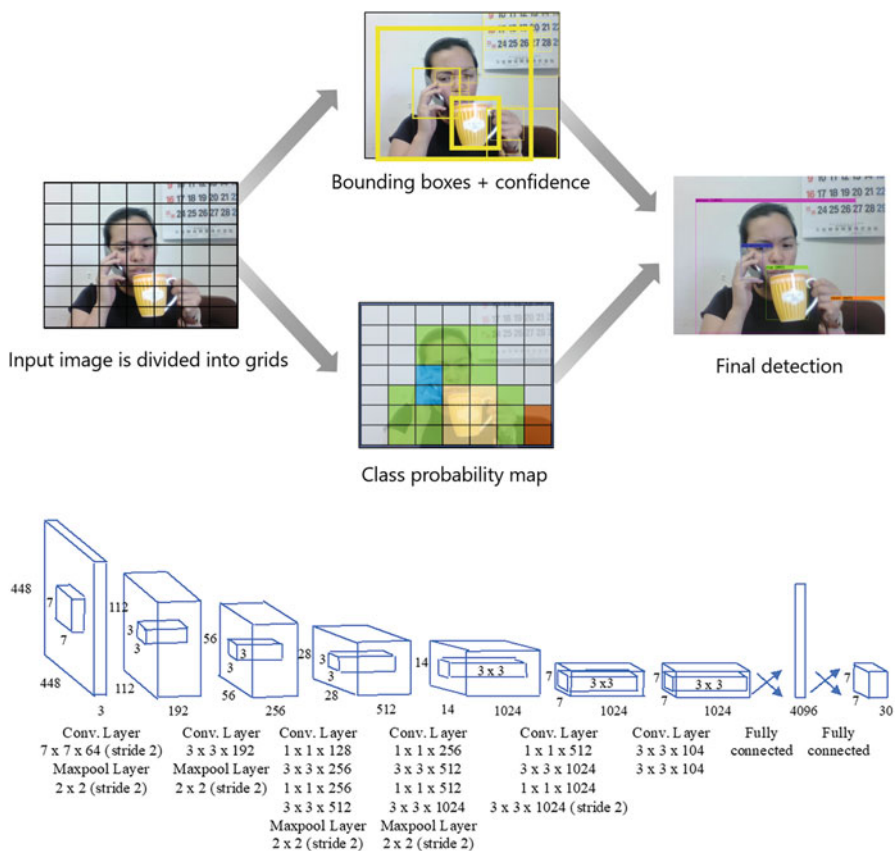
Fig. 11.1 (**Top**) General workflow of YOLO; (**Bottom**) YOLO Architecture

threshold value is set to eliminate bounding boxes of low confidence. Therefore, probabilities that are greater than the threshold value are considered detections.

However, the YOLO had its downsides. It had difficulties in detecting small objects and objects with unusual aspect ratios. It also made more localization errors compared to the state-of-the-art object detection model, fast R-CNN. In 2017, a more accurate counterpart of YOLO was introduced, which was called YOLOv2. The accuracy improvements in YOLOv2 were mainly due to the use of anchor boxes in predicting the location of objects in an image, batch normalization in the convolutional layers, and high-resolution classifier. Because of these improvements, YOLOv2 was able to outperform some state-of-the-art models such as Faster R-CNN with a mAP of 78.6% and an inference speed of 40 FPS on a Pascal VOC 2007 dataset and an mAP0.5:0.95 of 21.6% on an MS COCO test-dev 2015 (Redmon & Farhadi, 2017).

Then, a year after, several improvements were applied to YOLOv3. In this version, the previous backbone network of YOLOv2 (Darknet-19) was replaced with Darknet-53. Other than that, the following were also integrated into the system: (1) binary cross entropy in loss calculations, (2) use of logistic regression in predicting the "objectness score" for each bounding box, and (3) feature extraction at three different scales inspired by FPN (Lin et al., 2016; Redmon & Farhadi, 2018). Because of these enhancements, compared to YOLOv2, YOLOv3 had a better AP of 28.2% on an MS COCO dataset, which was on par with SSD but three time faster. However, the increased accuracy had some cost on the inference speed.

In early 2020, Bochkovskiy et al. introduced YOLOv4, which is more accurate and faster than YOLOv3 by 10% and 12%, respectively. YOLOv4 runs two times faster than a recent state-of-the-art object detection model, EfficientDet, at a similar accuracy. More importantly, YOLOv4 was designed to enable training on a single conventional GPU, unlike other models. The efficiency and increase in accuracy of YOLOv4 in object detection result mainly from several improvements incorporated into the model, which are: (1) cross-stage partial connections (CSP) in the new CSPDarknet53 inspired by CSPNet (Wang et al., 2020b), (2) use of Mish and leaky ReLU as an activation function (Maas et al., 2013; Misra, 2019), (3) adoption of a Path Aggregation Network (Liu et al., 2018) in place of the FPN that was used in YOLOv3, and (4) use of Spatial Pyramid Pooling (Huang et al., 2020) as a plug-in module.

After the development of YOLOv4, Wang et al. (2020a) modified the structure of YOLOv4 to enable scaling for different applications. YOLOv4-tiny was designed to maximize speed and to achieve the minimum computational cost possible. Then, YOLOv4-CSP and other larger versions of YOLOv4 were developed to maximize accuracy with varying computational requirements. In this study, the author compared the performances of YOLOv4, YOLOv4-tiny, and YOLOv4-CSP on detecting pear fruits.

Some common problems for detection systems are flickering and failure to detect the object under occlusion and challenging illumination. Therefore, relying completely on the number of detections for the pear count would be erroneous, especially in a pear orchard scenario where the mentioned challenges are common.

For that reason, a back-up system should cover for this limitation to ensure accuracy in counting, which can be through object tracking. With object tracking, a unique ID can be assigned to a detection, thus, giving a more reliable measure of object count in case the detection system fails.

11.2.2 Simple Online and Real Time Tracking with Convolutional Neural Networks (CNNs)

Among the algorithms for multiple object tracking, deep SORT has proven to be one of the fastest and most robust approaches (Wojke et al., 2017). It started as the Simple Online and Real time Tracking (SORT) algorithm (Bewley et al., 2016), which was developed to have a minimalistic approach in detection-based online tracking, which focused on efficiently associating object detections on each frame. It took advantage of the high reputation of convolutional neural networks in accurately detecting objects. In addition, two classic methods in motion prediction and data association, the Hungarian algorithm (Kuhn, 1955) and Kalman filter (Kalman, 1960), were implemented as the tracking components. Due to its modest complexity, SORT was 20 times faster than other state-of-the-art trackers (Bewley et al., 2016). Using Faster R-CNN (Ren et al., 2017) as the detector, it also had better performance compared to the traditional online tracking methods in the MOT (multiple object tracking) Challenge 2015 (Leal-Taixé et al., 2015).

The main drawback with SORT was occlusions and when viewpoints change. To solve this issue, Wojke et al. (2017) developed Deep SORT, which is an extended version of SORT (illustrated in Fig. 11.2). In Deep SORT, instead of relying only on motion-based metrics in data association, it also integrated a deep appearance-based metric derived from the convolutional neural network. This change resulted higher robustness from occlusion, change in viewpoint, and in using a nonstationary camera for lower identity switches. Using a modern GPU, Deep SORT was found to have a
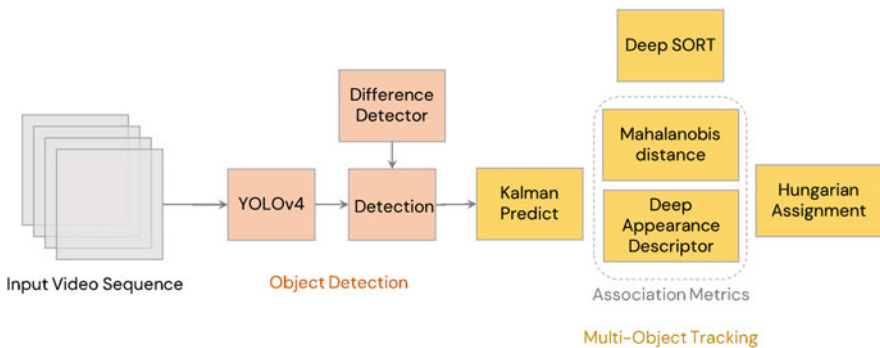
Fig. 11.2 Architecture of Deep SORT (Simple online and real time tracking with deep association metric)

runtime speed of 25–50 FPS using recent conventional GPUs (Wojke et al., 2017). Due to its suitability for real-time tracking and robustness, Deep SORT was selected as the tracking algorithm in this study for counting the pear fruits in real time.

11.2.3   Fruit Detection Using YOLO

Several studies have utilized YOLO-based models for fruit detection. Koirala et al. (2019) performed real time mango fruit detection with their Mango-YOLO model that exhibited an F1 score of 0.968, AP of 98.3% and an inference speed of 14 FPS on a NVIDIA GeForce GTX 1070 Ti GPU. Liu et al. (2020) and Lawal (2021) also proposed their own YOLOv3-based tomato detection systems that showed similarly remarkable AP values of 96.4% and 99.5%, respectively, and a faster speed of around 19 FPS for both studies using an NVIDIA GeForce GTX 1070 Ti GPU and an NVIDIA Quadro M4000 GPU, respectively, but this was still not sufficient for real time detection ($\geq$24 FPS). Li et al. (2021) also developed Lemon-YOLO for detecting lemon fruits, where they replaced Darknet-53 with an SE_ResGNet34 network. Their system had an AP of 96.28% and a detection speed of 106 FPS on the high-powered Tesla V100 GPU. Other studies also evaluated the performance of YOLO-based models on detecting other fruits such as apple, lemon, banana, and cherry (Fu et al., 2020; Kang & Chen, 2020; Kuznetsova et al., 2020a, b; Gai et al., 2021; Kuznetsova et al., 2021; Wu et al., 2021; Yan et al., 2021).

It is important to note that most of the studies did not report loss curves for their models, thus, it is difficult to verify if overfitting or under fitting of their dataset occurred. It is also difficult to confirm if they truly achieved the highest possible performance metrics exhaustively without overfitting.

11.2.4   Real-Time Fruit Counting Using YOLO and an Object Tracking Algorithm

Only one study combined YOLO with a multiple object tracking algorithm for counting fruits. Itakura et al. (2021) used YOLOv2 and Kalman filter to count pear fruits from a video to achieve an AP of 97% in detection and an F1 score in counting of 0.972. However, the speed of their counting system was not mentioned. It was also not stated if the tracking algorithm approach was online (current predictions rely only on past information) or offline (processes all information in one batch). Online tracking is more suitable for real time counting. Although offline tracking (also called batch tracking) can be more accurate, processing only occurs after all observations are obtained, thus making it an unattractive option for real time counting. Because of the lack of information about their tracking approach, it is difficult to confirm if their system is truly applicable in real time.

11.3   Materials and Methods

11.3.1   Field Data Collection

RGB video acquisition was done using two different cameras: a DJI Osmo Pocket, with a 12 MP, 1/2.3″ CMOS sensor and field of view of 80° F2.0, and a mobile phone camera (16 MP, 1/2.6″ BSI CMOS Sensor, F1.9 lens, with OIS). Video acquisition was done in a 0.15 ha joint-tree pear orchard (as seen in Fig. 11.3) in the Tsukuba-Plant Innovation Research Center, University of Tsukuba, Tsukuba, Ibaraki (36°06′56.8″ N, 140°05′37.7″ E) on a cloudy day and a partly cloudy day. The videos were taken from the bottom side of the trees. The details regarding the videos are outlined in Table 11.1.

11.3.2   Data Preparation

11.3.2.1   Videos Were Converted into Image Frames

Videos were converted into image frames using the "Scene video filter" of VLC. An automatic screenshot of the video was taken at an interval of half a second. For example, if the video had a frame rate of 60 FPS, one image frame was taken every after 30 frames. For a 30 FPS video, one image frame was taken every 15 frames.

Next was the elimination of images without pear fruits from the dataset. After that, the remaining image frames were 314 of 4k resolution images and

Fig. 11.3 Aerial view of the joint-tree pear orchard (**left**). Joint-tree pear orchard in the Tsukuba-Plant Innovation Research Center, University of Tsukuba, Tsukuba, Ibaraki (**right**)

Table 11.1 Details of video acquisition of pear fruits in the joint-tree pear orchard in the Tsukuba-Plant Innovation Research Center, University of Tsukuba, Tsukuba, Ibaraki

Date	Time	Weather condition	Device	Resolution, FPS
29 July 2020	12–1 p.m.	Cloudy	Mobile Phone Camera	1920 × 1080, 30
6 August 2020	9–10 a.m.	Intermittently cloudy	DJI Osmo Pocket	3840 × 2160, 60

134 of 1920 × 1088 resolution images, giving a total of 448 images. The dataset was further expanded through augmentation.

11.3.2.2   Labelling

Bounding box labelling was done using Supervisely® (Github, n.d.). Its labeling interface allows precise and efficient labelling, reducing the possibility of human-error in labelling ground truth data. However, the export format of the labels was only in Supervisely format (JSON). Other open-source Python-based annotation tools that export directly in YOLO format include LabelImg (Tzutalin LabelImg, 2021) and OpenLabeling (Cartucho et al., 2018).

Roboflow® was used to convert the Supervisely format labels to the desired frameworks (such as Darknet, Tensorflow™, PyTorch, etc.). An alternative to Roboflow is an open-source method of converting Supervisely labels to YOLO format (Shorten & Khoshgoftaar, 2019).

11.3.2.3   Data Augmentation

It is essential to make the pear fruit detection system more robust to different scenarios through diverse representation of pear fruits in joint-tree systems in one's dataset. However, there may be unavoidable bias present in the dataset that may not be obvious to the researcher, which can cause overfitting on the training dataset. To mitigate this possible concern, there is an assumption that more information can be extracted from the training dataset if the images are transformed in different ways. This is called data augmentation, which may simulate a wider representation of pear fruit data in orchards, thus avoiding possible overfitting to the training dataset. However, how does one decide which data augmentation techniques should be used?

There are two types of data augmentation: pixel-level and spatial-level. Pixel-level transformations change the images themselves but leave the bounding boxes unchanged. Some examples of pixel-level transformations are blurring, changing the brightness or exposure, adding noise, Cutout, Cutmix, and so on. This is useful if the researcher wants to preserve the bounding boxes themselves and intends to not distort the shape of the target object. Spatial-level transformations, on the other hand, change both the image and the bounding box, which makes the transformation slightly more complicated to code compared to pixel-level transformations. However, spatial-level transformations were shown to be more effective in improving the performance of object detection systems (Shorten & Khoshgoftaar, 2019). In this study, both kinds were used.

The following image transformations were done to the 4k images.

- Random flip (horizontal or vertical.)
- Random brightness adjustment from −25% to +25%.

- Random adjustment of gamma exposure from -20% to $+20\%$.
- Coarse Dropout: up to 6% of the image's pixels were subject to noise.

Images were resized to the following sizes: 416×416, 512×512, and 608×608. The aspect ratios of the images were preserved by adding black padding to avoid distorting the aspect ratio of the pear fruits. Through augmentation, the dataset was expanded from 448 images to 1337 images.

11.3.3   Data Splitting

As mentioned in the previous section, the dataset was comprised of 1337 images. We adapted a rule-of-thumb in data splitting for dataset sizes from 100 to 10,000, which was 70:30 for training and validation sets. This is typical for datasets that have even distribution, meaning the training and validation set are not too different from each other. However, in this study, the dataset had uneven distribution. High-resolution images were used in training the neural networks to enable them to detect smaller objects. Then, the trained neural networks were tested on the target application of this study, which were lower resolution mobile phone images. One may think that validating the trained model with a dataset that has a different distribution does not truly evaluate the performance of the trained model. However, how can the "learning" performance be truly measured?

Considering the uneven distribution, the dataset was split into four parts in a 70: 10:10:10 ratio: training, training-validation, validation and test set. The training and training-validation set contained the high-resolution images. The training-validation was the unseen high-resolution images, and was used to check if the trained model had overfit the training images. Again, the main target of the detection was pear counting using mobile phones. Thus, the validation and test sets were comprised of mobile phone images. The validation set's purpose was to check if there was a huge data mismatch between the Osmo images and the mobile phone images. The test set, on the other hand, was used to determine if the model had overfit the validation set.

11.3.4   Setting the Target Metric

Before training a network, it is important to set the desired error rate or accuracy. The whole point of training, validation, and optimization is to achieve the desired error rate or accuracy in detection. The desired error rate can be set at the same level as the human-level error in application. In pear fruit counting, the error rate was close to zero.

Accuracy is important in counting fruits. However, a machine that is adept in counting would be deemed useless if the speed was not real time. Moreover, for mobile phone use, it is important to consider if the inference should be done using

the phone's local computational resources or through cloud computing. For example, if the inference is desired to be done on the device itself, the pear fruit counter should not require too much computational power. Therefore, for a pear fruit counter using mobile phones, the aim is to have maximum accuracy that satisfies the minimum speed requirement, while considering the GPU consumption.

Thus, the goal was the following:

- Maximize the accuracy metric given the time-constraints, hardware, and dataset size available.
- Determine the inference speed of the YOLOv4 family and find out which one has inference speed close to real time ($\geq$24 FPS).

Find out the GPU consumption of the YOLOv4 family in terms of FLOPs. The CPU and GPU consumption is proportional to the number of FLOPs used (Sehgal & Kehtarnavaz, 2019).

11.3.5 Evaluation Metrics for the Detection

The performances of the models were evaluated based on the metrics used in the Pascal VOC Challenge (Everingham et al., 2010), which are listed in Table 11.2. The first metric is Intersection over Union (IoU), which is the proportion of the overlapping area and combined area of the bounding boxes of the prediction and the ground truth object.

True positive, false positive, and false negative values are prerequisites of the other performance metrics. A detection is considered a true positive (TP) detection if the IoU is equal to or greater than 0.5. False positive (FP) predictions are those having IoU with values below 0.5. False negative (FN) detections were the ground truth objects that were not detected at all, or those assigned with low confidence in predictions (eliminated by a certain threshold, which was considered 0.25 in this research). After calculating TP, FP, and FN, recall, precision, F1 score, and average precision can be derived. Recall is the sensitivity of the detection system. This metric is the ratio of true positive detections to total ground truth objects. Precision is the

Table 11.2 Performance metrics that were used to evaluate the YOLOv4 models are described below. This evaluation is based on the Pascal VOC Challenge (Everingham et al., 2010)

Performance metrics
Intersection over Union (IoU) $= \frac{\text{area of overlap}}{\text{area of union}}$
Recall $(R) = \frac{\text{TP}}{\text{TP+FN}}$
False Negative Rate (FNR) $= 1.00 - R$
Precision $(P) = \frac{\text{TP}}{\text{TP+FP}}$
False Positive Rate (FPR) $= 1.00 - P$
F1 score $= \frac{2 \cdot P \cdot R}{P+R}$
Average Precision (AP) $= \frac{1}{11} \sum_{R_i} P \cdot R_i$

TP true positive, *FP* false positive, *FN* false negative

correctness of the predictions, which is the ratio of the true positive detections to all positive detections. Next is the F1 score, which summarizes the overall performance of detection by incorporating both precision and recall. Finally, average precision (AP) is the area under the precision-recall curve interpolated from 11 points of recall and precision at different confidence thresholds. It is similar to the F1 score in the sense that it is one metric that summarizes the accuracy of a model. However, AP considers the confidence level of the predictions. Thus, this metric is more often used as a target metric for evaluating the performance of the models during training, and also for decision making in choosing the best model among the YOLOv4 models.

11.3.6   Components of the YOLOv4 Models

In this study, the authors compared the performances of YOLOv4, YOLOv4-tiny, and YOLOv4-CSP on detecting pear fruits. Table 11.3 shows the differences among these models in terms of their architectural components. In this section, how the elements of these models contribute to their respective characteristics is discussed more in detail.

11.3.6.1   Cross-Stage Partial (CSP) Connection

Cross-stage Partial (CSP) Connection is a technique to reduce computational complexity, which is originally derived from CSPNet (Wang et al., 2020b). To "CSP-ize" a network divides the feature map of the base layer into two parts then merges the two parts through transition → concatenation → transition (see Fig. 11.4). CSP-ization improves the accuracy and reduces the inference time through truncation of gradient flow (Wang et al., 2020a, b). Also, CSP-ization enables scaling of the model. Because of these reasons, CSP connections were incorporated into the backbone of the YOLOv4 models. CSPDarknet53 was chosen as the YOLOv4 backbone despite having lower accuracy in image classification compared to CSPResNext50 (Bochkovskiy et al., 2020). The next section explains why.

Table 11.3 Comparison of YOLOv4, YOLOv4-CSP, and YOLOv4-tiny in terms of their architectural elements

Model name	Backbone	Neck	Plug-in module
YOLOv4	CSPDarknet53 + Mish activation	PANet + Leaky activation	SPP
YOLOv4-CSP	CSPDarknet53 + Mish activation	CSPPANet + Mish activation	CSPSPP
YOLOv4-tiny	CSPOSANet + Leaky activation	FPN + Leaky Activation	–
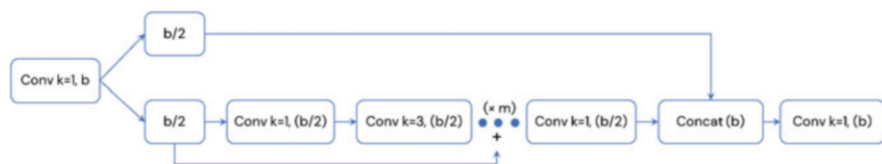
Fig. 11.4 Cross-Stage Partial Connection Block in YOLOv4-CSP
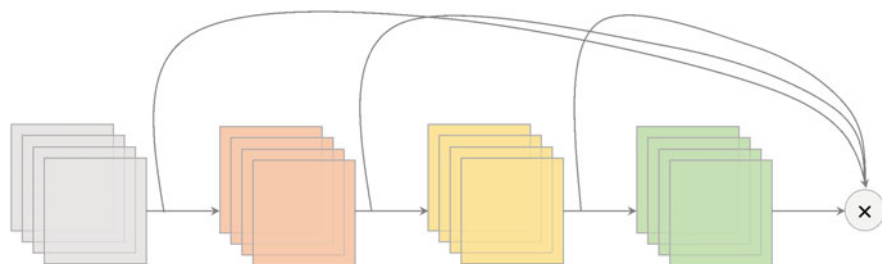
Fig. 11.5 One shot aggregation (OSA)

11.3.6.2 CSPDarknet53: YOLOv4 and YOLOv4-CSP's Backbone

Despite CSPResNext50's better performance in image classification, it was not the case for object detection. CSP-ization of Darknet53 led to higher accuracy in object detection due to the following (Wang et al., 2020a):

Higher input network size, which led to the ability to detect more small-sized objects. More convolutional layers 3×3, which led to a larger receptive field to cover the increased input network size. Larger number of parameters for greater capacity to detect multiple objects of different sizes in a single image.

Other than CSP-ization, several techniques were used to improve the performance of CSPDarknet53 without putting a burden on the computational requirement: (1) data augmentation techniques such as CutMix (Yun et al., 2019) and Mosaic (Bochkovskiy et al., 2020), (2) DropBlock (Ghiasi et al., 2018) as a regularization method, and (3) Class label smoothing (Bochkovskiy et al., 2020). Then, the following techniques were used to make the use of expensive GPUs no longer necessary in training: (1) Mish (Misra, 2019) as the activation function (further explained in Sect. 11.3.6.4), and (2) Multi-input weighted residual connections (Sandler et al., 2018).

11.3.6.3 YOLOv4-tiny's Backbone: CSPOSANet

For YOLOv4-tiny, it is important to make the computations efficient and fast without sacrificing much the accuracy. Thus, one shot aggregation (OSA) (shown in Fig. 11.5), which is derived from VoVNet (Lee et al., 2019), was implemented between the calculation modules of YOLOv4-tiny's backbone CSPOSANet for

smaller computation complexity. This resulted in the reduction of the size of the model and the number of parameters through the removal of an excess amount of duplicate gradient information. A leaky rectified linear unit was used as the activation function for CSPOSANet due to its faster speed in convergence (Maas et al., 2013).

11.3.6.4   Why Were Leaky Rectified Linear Unit and Mish Used as the Activation Functions for the YOLOv4 Models?

The leaky rectified linear unit (or leaky ReLU) is a modified version of ReLU. The difference is that the former allows a small nonzero gradient over its entire domain, unlike ReLU (Fig. 11.6). Deep neural networks utilizing leaky ReLU were found to reach convergence slightly faster than those using ReLU. However, leaky ReLU is slightly less accurate, but has lower standard deviations compared to its more novel counterparts (Misra, 2019). However, leaky ReLu has better performance with under a 75% IoU threshold and with large objects and has lower computational cost due to lower complexity (Misra, 2019).

Mish, on the other hand, is a smooth, continuous, self-regularized, nonmonotonic activation function that enables smoother loss landscapes which helps in easier optimization and better generalization. It has a wider minimum, and thus can achieve lower loss. Because of these benefits, neural networks implementing Mish led to higher accuracy and lower standard deviations in object detection. Moreover, it retains the feature of its predecessors (Swish and leaky ReLU) in terms of unbounded above and bounded below. The former avoids saturation (which generally causes training to slow down), whereas the latter results in stronger regularization effects (fits the model properly).

Thus, leaky ReLU would be more suitable if the goal was to maximize speed without sacrificing much of the accuracy. Then, if accuracy should be maximized, Mish would be the better option. Table 11.4 summarizes the activation functions used and their corresponding effects on each YOLOv4 model.

$$f(x) = \max(0, x)$$

$$f(x) = 1(x < 0)(\alpha x) + 1(x \geq 0)(x)$$
$$\text{where } \alpha = \text{small constant}$$

$$f(x) = x \tanh(\ln(1 + e^x))$$
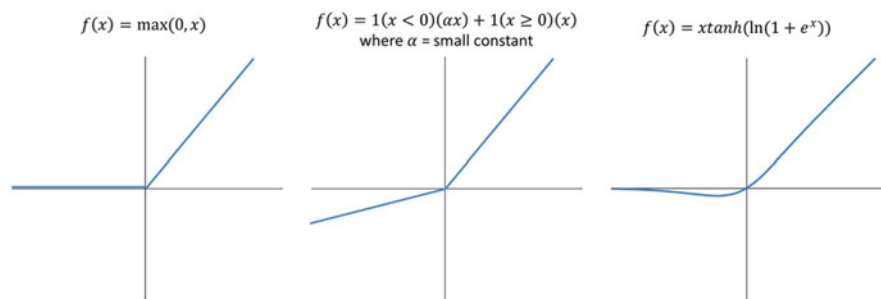
Fig. 11.6 Activation functions. (**Left**) Rectified Linear Unit (ReLU); (**Center**) Leaky ReLU; (**Right**) Mish

Table 11.4 Summary of activation function used and the reason why the specified activation functions were used

Model	Activation function	Effect
YOLOv4	Mish + Leaky ReLU	Balanced accuracy and speed
YOLOv4-CSP	Mish	Maximized accuracy
YOLOv4-tiny	Leaky ReLU	Maximized speed and Minimum computational cost
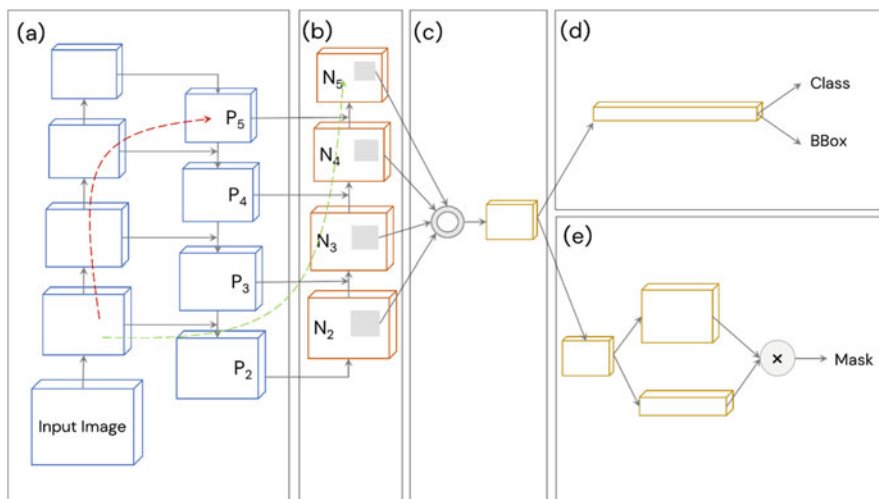
Fig. 11.7 Architecture of PANet, which inspired the path aggregation in YOLOv4's neck. (**a**) FPN backbone; (**b**) bottom-up path augmentation; (**c**) adaptive feature pooling; (**d**) box branch; (**e**) fully connected fusion (concatenation is done instead of addition for YOLOv4)

11.3.6.5 YOLOv4's Neck: Path Aggregation Network (PANet)

Path aggregation (shown in Fig. 11.7), originally proposed by Liu et al. (2018), was used as the neck for YOLOv4 and YOLOv4-CSP in place of FPN (which was used in YOLOv3). This technique aggregates parameters from different backbone levels for different detector levels through bottom-up path augmentation and adaptive feature pooling. Bottom-up path augmentation shortens the information path and enhances the feature pyramid by making fine-grained localized information available to top layers (the classifiers). On the other hand, adaptive feature pooling recovers the broken information path between each proposal and all feature levels (cleaner paths are created). It fuses the information together from different layers using an element-wise max operation. Thus, PANet ensures that important features are not lost. For these reasons, PANet was used as the neck for YOLOv4 and YOLOv4-CSP.

11.3.6.6   YOLOv4's Plug-In Module: Spatial Pyramid Pooling (SPP)

Spatial pyramid pooling (or SPP) is another feature of YOLOv4 and YOLOv4-CSP that eliminates the need for a fixed-size input image, making them more robust and practical. SPP is added on top of the last convolutional layer of YOLOv4 and YOLOv4-CSP. SPP pools the features and generates outputs with fixed length, which are then fed into the classifier layer (Fig. 11.8). In this study, the pooling was done through the spatially division of feature maps into different scales of $d \times d$ equal blocks, where d can be $\{1, 2, 3, \ldots\}$. These different scales of division forms
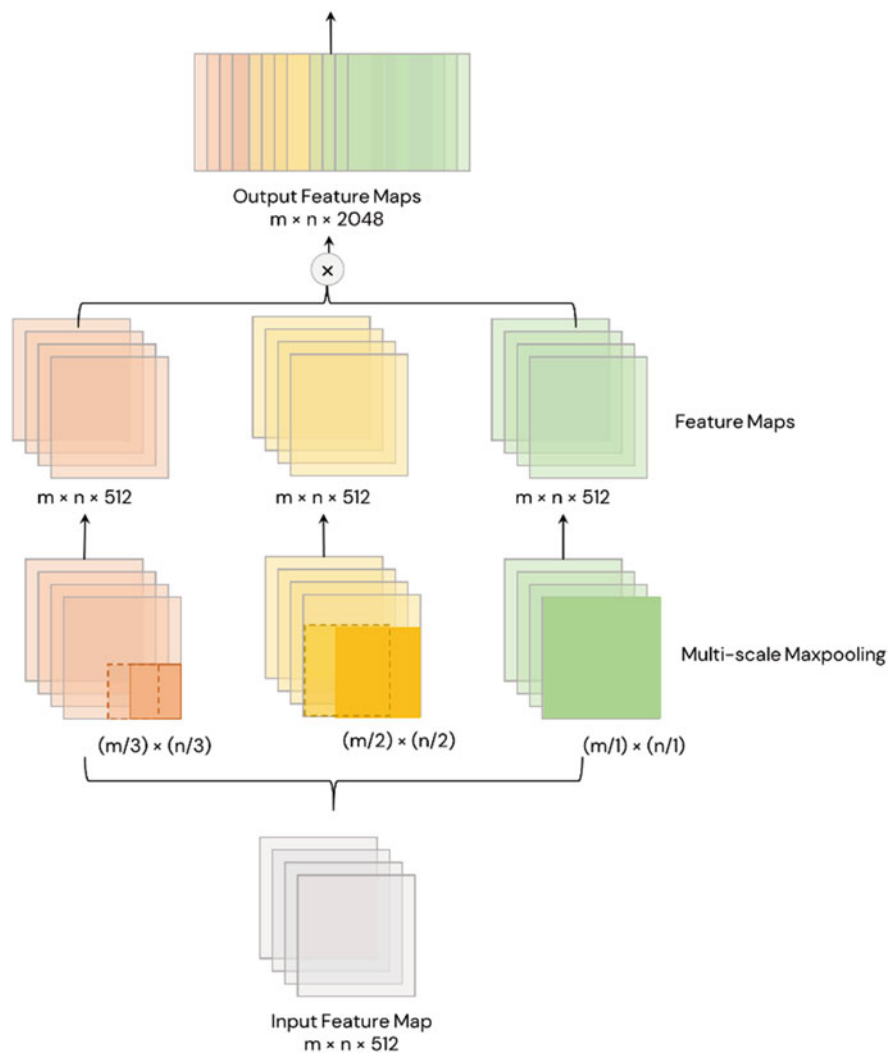
Fig. 11.8 Spatial Pyramid Pooling in YOLOv4

are called spatial pyramids. Then, max pooling was done for each level of division to produce a concatenated 1D vector (originally). SPP works similarly in the YOLOv4 models, but the difference is the input feature map size is equal to the output feature map size through padding.

11.3.7 Training, Validation, and Optimization

The complete training process was composed of two stages: stage-1 training and stage-2 training.

11.3.7.1 Stage-1 Training

The Darknet framework (Redmon et al., 2019) was used to train the YOLOv4 models within the Google Colab™ Notebook environment. The GPU used for training, validation, and inference was Tesla T4. Custom anchors were calculated using k-means clustering (Table 11.5 for the custom anchors used). Training-time-augmentation was enabled. The total number of iterations for stage-1 training was 6000.

Table 11.5 Anchors of the YOLOv4 models

Model name	Anchors	Anchor mask indices		
		First	Second	Third
YOLOv4-tiny-416	8, 8, 17, 17, 35, 27, 40, 42, 60, 50, 77, 72	5	0, 1, 2, 3, 4	–
YOLOv4-tiny-512	8, 8, 15, 15, 28, 25, 46, 42, 66, 58, 93, 85	4, 5	0, 1, 2, 3	–
YOLOv4-tiny-608	12, 12, 28, 24, 44, 45, 66, 54, 82, 78, 117, 100	4, 5	0, 1, 2, 3	–
YOLOv4-416	5, 5, 9, 9, 14, 14, 22, 19, 29, 33, 52, 22, 42, 40, 56, 53, 80, 67	0, 1, 2, 3	4, 5, 6, 7	8
YOLOv4-512	7, 8, 12, 12, 18, 18, 28, 25, 39, 43, 61, 29, 56, 54, 81, 69, 102, 97	0, 1, 2, 3	4, 5, 6	7, 8
YOLOv4-608	8, 9, 15, 15, 26, 22, 37, 34, 24, 60, 61, 43, 63, 64, 85, 78, 117, 100	0, 1, 2	3, 4, 5	6, 7, 8
YOLOv4-CSP-512	7, 8, 12, 12, 18, 18, 28, 25, 39, 43, 61, 29, 56, 54, 81, 69, 102, 97	0, 1, 2, 3	4, 5, 6	7, 8
YOLOv4-CSP-608	8, 9, 15, 15, 26, 22, 37, 34, 24, 60, 61, 43, 63, 64, 85, 78, 117, 100	0, 1, 2	3, 4, 5	6, 7, 8

11.3.7.2 Hyperparameters

Stage-1 training used the linear warmup policy for the first 1000 iterations and multi-step decay (other terms: piecewise constant decay, stepwise annealing) as the learning rate schedule policies. The update rule for the multi-step learning rate schedule was as follows:

$$LR_{n+1} = \begin{cases} d \cdot LR_n, & \text{if } n \text{ in [steps]} \\ LR_n, & \text{otherwise} \end{cases} \tag{11.1}$$

where n is the iteration step, LR_n is the previous learning rate, d is the decay rate ($d \in \mathbb{R} \mid 0 < d < 1$), and [steps] is the set of iterations when to decrease the learning rate. For this study, $d = 0.1$, [steps] = [4800, 5400], $LR_0 = 0.001$ for YOLOv4 and YOLOv4-CSP and 0.00261 for YOLOv4-tiny. The learning rate schedules for the YOLOv4 models are illustrated at Fig. 11.9.

Regarding the optimizer, Nesterov Accelerated Gradient, momentum, and weight decay were implemented. The momentum and weight decay were set as 0.949 and 0.0005, respectively, for all the YOLOv4 models. The localization loss was based on Complete IoU (CIoU) (Zheng et al., 2020), which is illustrated in Fig. 11.10. A complete list of the hyperparameters is shown in Table 11.6.
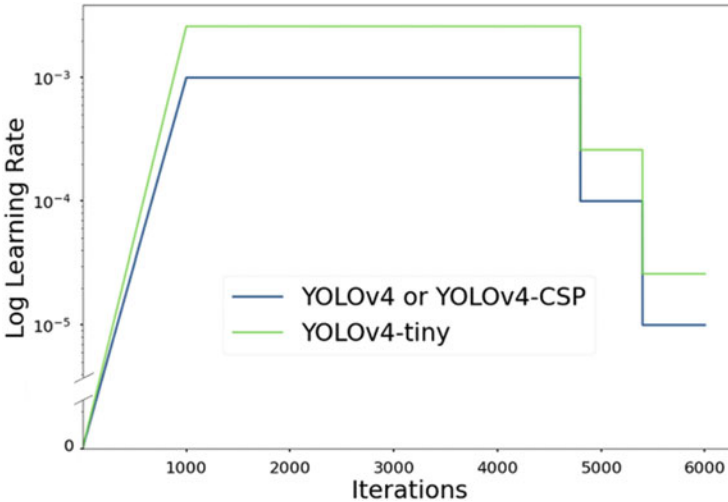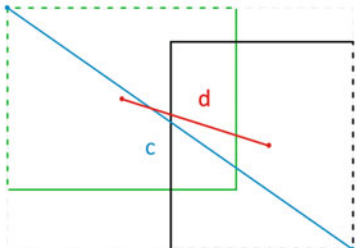
Fig. 11.9 Learning rate schedule of YOLOv4, YOLOv4-CSP, and YOLOv4-tiny, where maximum iteration = 6000. For the first 1000 iterations, a linear warm up policy was done, which is a slow rise of the learning rate. After the 1000th iteration, a multistep decay policy was done

Localization loss

$$LOSS = \boxed{1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v}$$

Confidence Loss

$$-\sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\hat{C}_i \log(C_i) + \left(1 - \hat{C}_i\right) \log\left(1 - C_i\right) \right] -$$

$$\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} \left[\hat{C}_i \log(C_i) + \left(1 - \hat{C}_i\right) \log(1 - C_i) \right] -$$

$$\sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{c \in classes} \left[\hat{p}_i(c) \log(p_i(c)) + \left(1 - \hat{p}_i(c)\right) \log(1 - p_i(c)) \right]$$

Classification Loss

Euclidean distance (d)

$$LOSS = 1 - IoU + \boxed{\frac{\rho^2(b, b^{gt})}{c^2}} + \alpha v$$

$$\alpha = \frac{v}{1 - IoU + v},$$

$$v = \frac{4}{\pi^2} \left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h} \right)^2.$$

Localization Loss is Based on CIoU loss

Fig. 11.10 (**Left**) The complete YOLOv4 Loss Function; (**Right**) details of the localization loss based on Complete Intersection-over-Union loss

Table 11.6 Hyperparameters of the YOLOv4 models

Model name	Batch	Subdivisions	Momentum	Decay	Initial learning rate
YOLOv4-tiny	64	16	0.9	0.0005	0.00261
YOLOv4 or YOLOv4-CSP	64	16	0.949	0.0005	0.001

11.3.7.3 Stage-2 Training

Stage-2 training involved a fine-tuning process, which confirms if the weights with the best mAP from stage-1 training had reached its maximum value. The details of the fine-tuning process can be found in Algorithm 11.1.

11.3.7.4 Error Analysis

For a supervised learning algorithm, the best performance achieves low bias and low variance. In typical 70:30 data splitting, the purpose of the error analysis is to determine if the model has achieved the highest possible accuracy without overfitting to the training data.

However, the aim of data splitting at 70:10:10:10 is to train on high resolution images (for more robustness) with the goal of having good performance on low resolution images. Thus, through comparison of each pair of errors in Fig. 11.11, the error analysis answers the following questions:

- Has the model achieved the lowest possible bias?
- Did the model overfit on the training data?
- Does the train-validation set have high mismatch in data distribution compared to the validation set?
- Did the model overfit on the validation set?

To avoid data mismatch at Stage C (Fig. 11.11, Table 11.7), data augmentation was done to the training-validation set to simulate a lower image quality, like that of a mobile phone image. After passing all the stages in the error analysis, the optimized models were compared based on their performance on the test set.

11.3.8 Model Comparison

After going through training, validation and optimization, the model that satisfied the following criteria on the test set was chosen to be the YOLOv4 model for the pear counting stage of this study. First, it should be the highest in all evaluation metrics
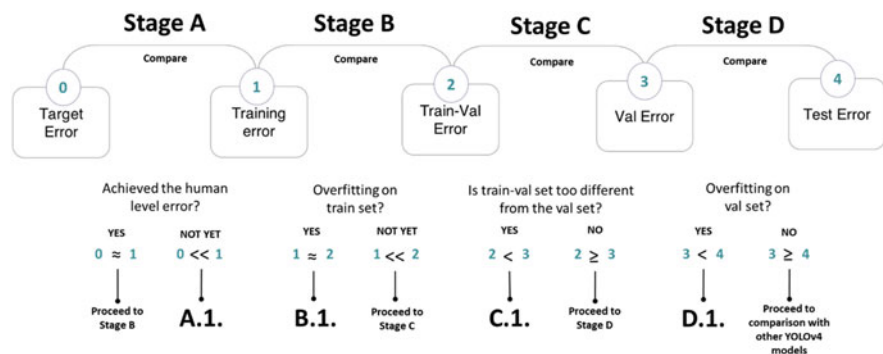
Fig. 11.11 Error analysis to achieve the best possible performance for the model from Stage A to Stage D. The goal of the error analysis is to reduce the gap on each pair of errors by performing the strategies found in Table 11.7. The target error (also called Bayes error) is the lowest possible detection error, which in this case can be considered as the human error rate in detecting pears

Table 11.7 Strategies in reducing error gap in the error analysis. The stages were defined in Fig. 11.11

Stage	How to reduce the gap?
A.1	• Train longer
	• Train a bigger model
	• Use more effective optimization algorithms
	• Momentum, RMS Prop, Adam
	• Try other neural network architectures or hyperparameters
B.1	• Add more to training set
	• Perform regularization methods to improve generalizing ability
	• L2, Dropout, Data augmentation
	• Try other neural network architectures or hyperparameters
C.1	• Make training data more similar to val/test set
	• Collect more data similar to val/test set
	• Consider generating synthetic dataset
	• Note: overfitting to the synthetic dataset is a risk
D.1	• Increase the val set size

and second, the inference speed should be close to real-time ($\geq$24 FPS). Lastly, the authors noted the GPU consumption of the chosen model to consider if in-device inference or cloud computing inference would be ideal for mobile phone platform implementation.

11.3.9 Pear Counting Using the Selected YOLOv4 Model and Deep SORT

The best performing YOLOv4 model that satisfied the criteria in the model comparison was converted to the Tensorflow™ format. Deep SORT, in combination with YOLOv4, was implemented locally to track the pears in an unseen test mobile phone video of resolution 1920 × 1080, 32 s long, with a frame rate of 30 FPS. The hardware specification was as follows: Quad-core Intel® Core™ i7-7700HQ @ 2.80 GHz, 16.0 GB RAM and NVIDIA GeForce GTX 1060.

Two counting methods were compared in this study: (1) region-of-interest (ROI) method and (2) unique object ID method. The ROI method was based on the number of unique object centroids tracked by Deep SORT that would cross the ROI, which is a horizontal line. Different ROIs were tested, and 50% of the height of the video was deemed to be the optimal ROI. For the second method, the counts were based on the number of unique object IDs generated by Deep SORT's tracking mechanism. Figure 11.12 illustrates the pear counting system.

Fig. 11.12 Pear counting system based on YOLOv4-512 and Deep SORT using the ROI-line method. The number on the top left is the number of objects that crossed the horizontal ROI line. Unique IDs can be seen on the top left corner of each bounding box

Table 11.8 The performance metrics used to evaluate the pear counting methods. This evaluation is based on the CLEAR MOT Challenge (Bernardin & Stiefelhagen, 2008)

Performance metrics
$R_{\text{count}} = \frac{\text{TP}}{\text{TP+FN}}$
FN Rate$_{\text{count}} = 1.00 - R$
$P_{\text{count}} = \frac{\text{TP}}{\text{TP+FP}}$
FP Rate$_{\text{count}} = 1.00 - P$
$\text{F1}_{\text{count}} = \frac{2 \cdot P \cdot R}{P + R}$
$\text{MOTA} = 1 - \sum \frac{\text{FN+FP+mismatches}}{\text{total count}}$

TP true positive, *FP* false positive, *FN* false negative, *MOTA* multiple object tracking accuracy

11.3.10 Evaluation Metrics for the Pear Counting

The performance metrics for pear counting are similar to the detection's evaluation metrics. However, the authors used the subscript *count* to denote metrics associated with pear counting. Additionally, metrics from CLEAR Multiple Object Tracking (MOT) (Bernardin & Stiefelhagen, 2008), as seen below, were used but modified in the case of this study. The metrics for pear counting are summarized in Table 11.8. In this study, the objects themselves were not moving, thus, mismatches = 0.

11.4 Results and Discussion

The goal of the study was to compare YOLOv4, YOLOv4-tiny, and YOLOv4-CSP in terms of accuracy, speed, and memory usage. After evaluating which YOLOv4 model had the best performance in combination with Deep SORT, they were also evaluated for pear counting use.

11.4.1 Training Details

Table 11.9 outlines the details about the training of the YOLO4 models. YOLOv4-tiny took less than half an hour for 1000 iterations for all the input sizes. This is because smaller models take less time in training due to less computational complexity. In total, it took around 1.4 h to train YOLOv4-tiny completely, which is a remarkably short time. On the other hand, YOLOv4 had a different training speed to that of YOLOv4-CSP despite the similar size. The latter was able to train from 2.0 to 2.8 h for 1000 iterations, compared to YOLOv4, which spanned from 2.0 to 3.1 h. This difference is due to the CSP-ized PAN and SPP architecture of YOLOv4-CSP, which effectively reduced 40% of the computation (Wang et al., 2020a).

However, examining the loss curves in Fig. 11.13, one gets a clue if the models have achieved minimum losses and maximum mAP. Based on the loss graphs, YOLOv4-tiny converged in a quicker manner compared to YOLOv4 and YOLOv4-CSP. This may be due to the fact that it had a higher learning rate and it used the leaky ReLU as the activation function. Among the models, YOLOv4-CSP seemed to converge the slowest which may be due to the computational cost of CSPization and the use of Mish as the activation function. Upon seeing the

Table 11.9 Details regarding training YOLOv4 models

Model name	Average training time per 1000 iterations (h)	GPU memory usage in training (GB)	Weight sizes (MB)
YOLOv4-tiny-416	0.22[a]	2.22[a]	22.96
YOLOv4-tiny-512	0.23[a]	2.34[a]	22.97
YOLOv4-tiny-608	0.23[a]	2.45[a]	22.97
YOLOv4-416	2.00	3.08	249.986
YOLOv4-512	2.24	3.18	249.998
YOLOv4-608	3.10	3.34	250.016
YOLOv4-CSP-512	1.91	3.06	205.298
YOLOv4-CSP-608	2.80	4.52	205.316

[a] Subdivisions = 16
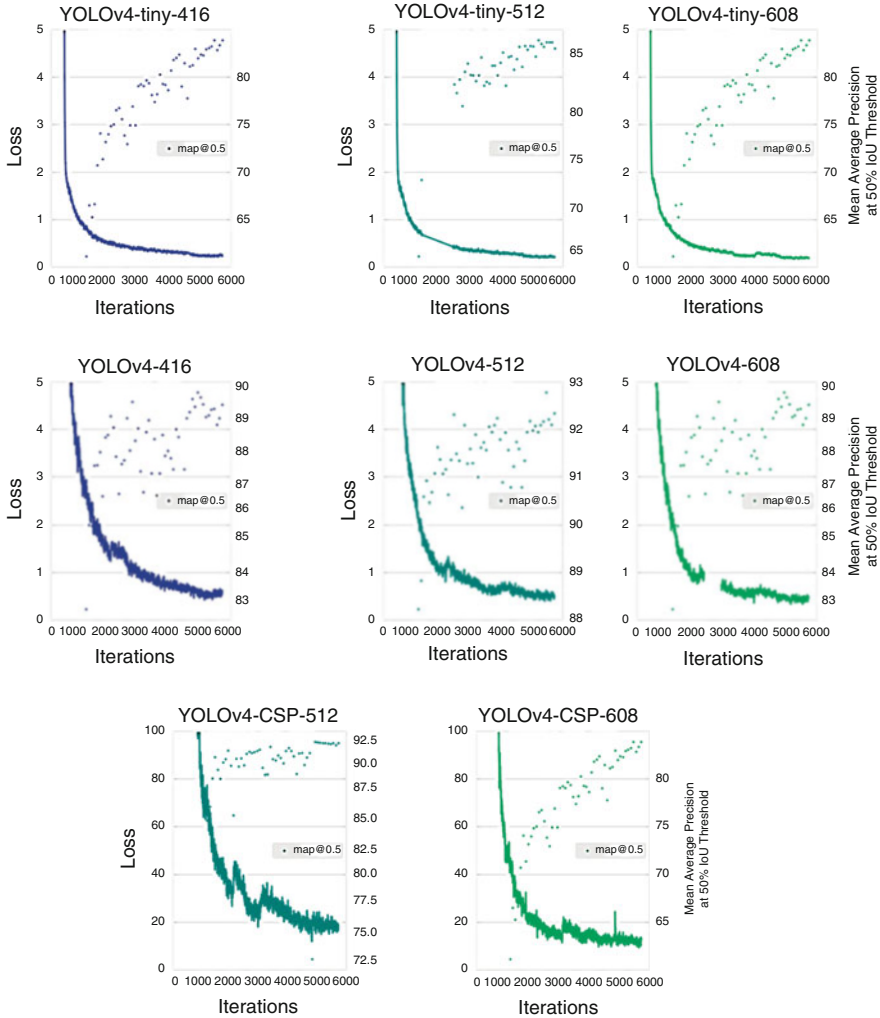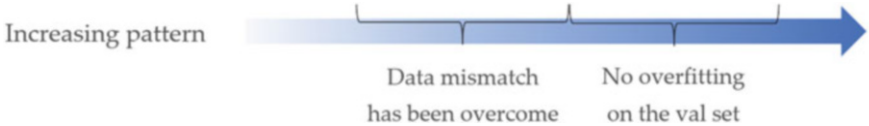
Fig. 11.13 Average Loss and Mean Average Precision (mAP) over 6000 iterations for each of the YOLOv4 models

differences in convergence rate within 6000 iterations, YOLOv4 and YOLOv4-CSP were further trained following the fine-tuning algorithm from Algorithm 11.1 to get the best possible weight.

After the fine-tuning process, error analysis was done, and the results are shown in Table 11.10. It can be observed that there is an increasing pattern among the average precision values. Comparing the AP_{50} train-val and AP_{50} val, it can be concluded that the data mismatch between the high-resolution data and lower resolution data was overcome. Looking at the AP_{50} val and AP_{50} test, the values were also increased. This is a good sign that no overfitting occurred on the validation

Table 11.10 Average precision for the training-validation set (train-val), validation set (val), and test set (test) IoU threshold = 50% and Confidence Threshold = 0.5

Model Name	AP_{50} Train-Val	AP_{50} Val	AP_{50} Test
YOLOv4-tiny-416	83.78	92.91	94.09
YOLOv4-tiny-512	86.23	93.08	93.53
YOLOv4-tiny-608	87.61	92.11	94.19
YOLOv4-416	90.39	93.72	93.76
YOLOv4-512	92.86	94.61	96.64
YOLOv4-608	91.32	95.39	96.76
YOLOv4-CSP-512	92.74	93.48	97.16
YOLOv4-CSP-608	92.60	94.51	98.32

Increasing pattern →

Data mismatch has been overcome

No overfitting on the val set

set. Confirming from the error analysis, it is possible to perform comparisons between the performances of the models on the test dataset.

11.4.2 Model Performance Comparison

To reiterate, the authors set the following goals:

- Maximize the detection performance metrics, which were mentioned at Table 11.2.
- Determine which YOLOv4 family has an inference speed close to real time (≥24 FPS).
- Find out the GPU consumption of the YOLOv4 models.

YOLOv4, YOLOv4-tiny, and YOLOv4-CSP were compared based on the criteria above. Table 11.11 shows the detection performance of the YOLOv4 models in terms of P, FPR, R, FNR, F1, and Average IoU on the test dataset. YOLOv4-CSP-608 exhibited the best performance among the metrics. This may be due to the Mish used as the activation function and higher network resolution. Thus, the intricate features of the object could be learned at a deeper sense compared to models at lower network resolution.

Interestingly, in terms of P and FPR, most of the models performed well, including YOLOv4-tiny. This shows that even the smallest model had promising accuracy. The second in place in the performance ranking were YOLOv4-512 and YOLOv4-608. An unexpected outcome was that YOLOv4-512 performed better compared to YOLOv4-CSP-512.

Table 11.11 Performance metrics of the YOLOv4 models in terms of Precision (*P*), False Positive Rate (FPR), Recall (*R*), False Negative Rate (FNR), F1 score and average Intersection-over-Union (IoU) on the test dataset. Ranking is indicated by color, where green is first, yellow is second, and orange is third best

Model Name	P	FPR	R	FNR	F1	Average IoU
YOLOv4-tiny-416	1.00	0.00	0.87	0.13	0.93	83.06
YOLOv4-tiny-512	0.98	0.02	0.88	0.12	0.93	83.13
YOLOv4-tiny-608	1.00	0.00	0.89	0.11	0.94	85.85
YOLOv4-416	0.98	0.02	0.90	0.10	0.94	82.37
YOLOv4-512	1.00	0.00	0.94	0.06	0.97	85.77
YOLOv4-608	1.00	0.00	0.94	0.06	0.97	86.75
YOLOv4-CSP-512	1.00	0.00	0.92	0.08	0.96	86.16
YOLOv4-CSP-608	1.00	0.00	0.95	0.05	0.98	87.18
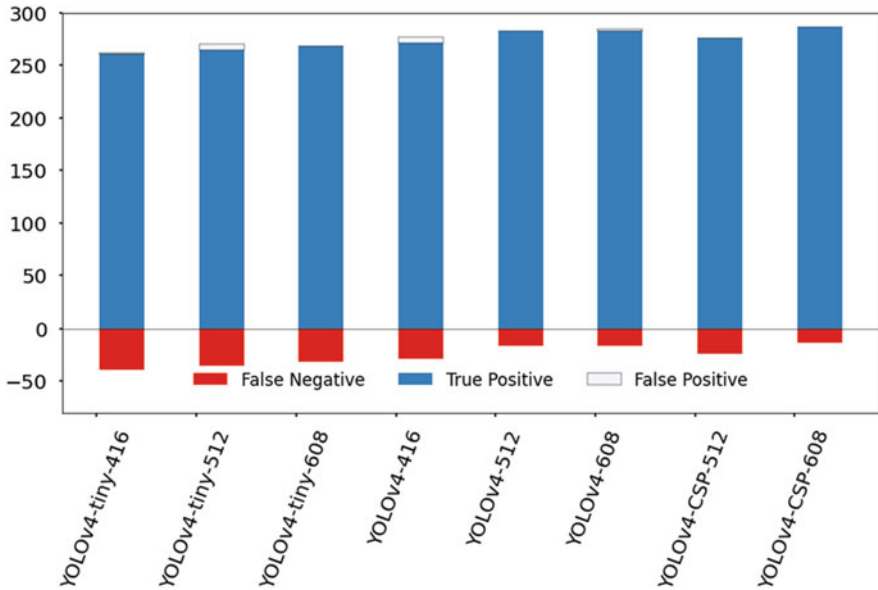
Fig. 11.14 Comparison of YOLOv4 models in terms of the frequency of True Positive, False Positive, and False Negative detections. The specific values are in Table 11.12

In counting fruits, FNR was deemed to be better compared to FPR. Looking at Fig. 11.14, consistent with the results from Table 11.11, the top performing models were YOLOv4-CSP-608, YOLOv4-512, and YOLOv4-608. However, it is difficult to ignore the fact that YOLOv4-tiny-608 had comparable performance to YOLOv4-416 with less false positive detections. This shows the potential of YOLOv4-tiny if a

higher network resolution is set. It is possible that YOLOv4-tiny may have satisfactory performance if the network resolution is increased.

11.4.3   Speed-Accuracy Tradeoff in the YOLOv4 Models

Figure 11.15 shows the tradeoff in speed and accuracy in terms of AP_{50} on the test dataset. The specific values can be found in Table 11.12. YOLOv4-512, 416, and YOLOv4-tiny satisfied the requirement for real-time speed. However, YOLOv4-CSP seemed to have traded off high accuracy with some speed, although it is a good thing to note that the speed of YOLOv4-CSP-512 was very close to real time, which was 21.4 FPS. Another observation was that YOLOv4-608 satisfied the real time speed requirement of ≥ 24 FPS at 26.4 FPS. Thus, so far, the best model in terms of accuracy while satisfying the real-time speed requirement was YOLOv4-608. The next thing to consider are the metrics in computational power.

11.4.4   Average Precision at Different Thresholds

Average precision (AP) became more widely used as an accuracy metric because of the PASCAL VOC Challenge and COCO Challenge. AP@0.50 is officially used by the PASCAL VOC Challenge whereas AP@0.75 is considered the "strict metric" for the COCO Challenge. Huang et al. (2017) compared different convolutional object
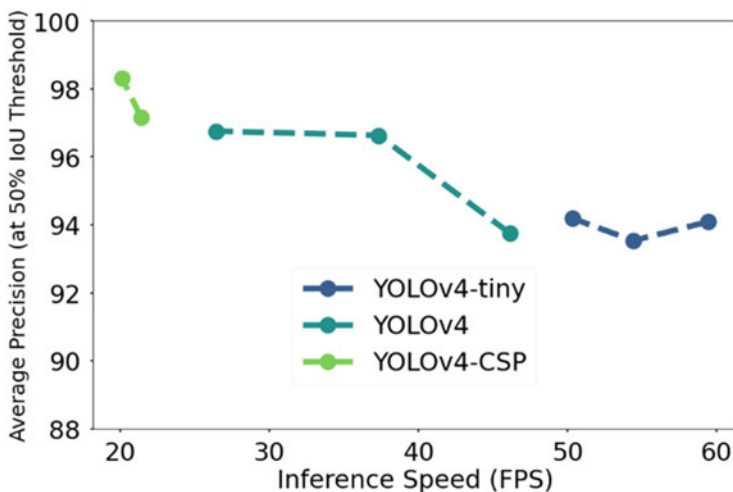
Fig. 11.15 Comparison of the YOLOv4 models in terms of accuracy (in this case, average precision at 50% intersection-over-union threshold) and inference speed (frame per second)

Table 11.12 Performance metrics of the YOLOv4 models on the Test Dataset (Complete)

Model name	P	R	F1	Average IoU	AP$_{25}$	AP$_{50}$	AP$_{75}$	BFLOPs	Inference speed (FPS)	Inference memory usage (GB)	TP	FP	FN
YOLOv4-tiny-416	1.00	0.87	0.93	83.06	95.1897	94.0893	70.6543	6.789	59.4	2.07	261	1	39
YOLOv4-tiny-512	0.98	0.88	0.93	83.13	95.7535	93.5346	77.9119	10.283	54.4	2.11	264	6	36
YOLOv4-tiny-608	1.00	0.89	0.94	85.85	95.5740	94.1936	80.1209	14.500	50.3	2.15	268	0	32
YOLOv4-416	0.98	0.90	0.94	82.37	95.9052	93.7559	79.6057	59.571	46.1	2.60	271	6	29
YOLOv4-512	1.00	0.94	0.97	85.77	96.9504	96.6365	87.5749	90.235	37.3	2.78	283	0	17
YOLOv4-608	1.00	0.94	0.97	86.75	97.4674	96.7600	87.9839	127.232	26.4	3.04	283	1	17
YOLOv4-CSP-512	1.00	0.92	0.96	86.16	97.8568	97.1558	86.8072	76.142	21.4	2.62	276	0	24
YOLOv4-CSP-608	1.00	0.95	0.98	87.18	99.0726	98.3243	86.6339	107.359	20.1	2.79	286	0	14
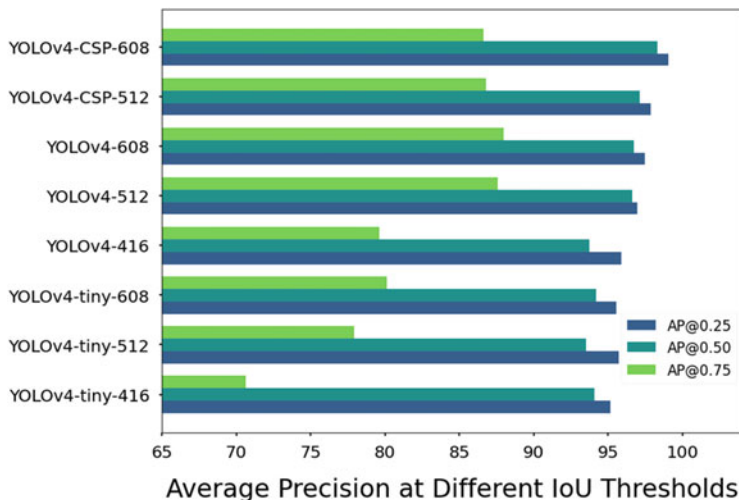
Fig. 11.16 Average precision (AP) of the YOLOv4 models at 25%, 50%, and 75% Intersection-over-Union (IoU) thresholds for the test set. The specific values are in Table 11.12

detectors in terms of mAP@0.50 and mAP@0.75. Their results showed that their models with low AP at more restrictive IoU thresholds (mAP@0.75) always showed low AP at less restrictive IoU thresholds (mAP@0.50).

However, for the YOLOv4 models, this seemed not to be the case (see Fig. 11.16). A possible reason for the different pattern is the differences in receptive field. Based on observations among state-of-the-art object detection models, the increase in receptive field was found to be associated with higher classification accuracy (Araujo et al., 2019). Increase in the receptive field is a natural consequence of a higher number of layers, so this is one possible reason why YOLOv4 and YOLOv4-CSP had higher AP.

Nevertheless, it is important to note that receptive field size is not the only contributing factor to the differences in the AP values of the YOLOv4 models. Another factor that may have affected the performance could be the presence of residual connections. As seen in Fig. 11.16, YOLOv4-tiny-608 performed better, if not on par, with YOLOv4-416. Other than an increased receptive field caused by the higher network resolution, the use of a one-shot aggregation technique in pooling data in YOLOv4-tiny and CSP connections could have contributed hugely to its competitive performance. Thus, if YOLOv4-tiny was trained at network resolutions higher than 608 × 608, it might have very satisfactory accuracy metrics while maintaining low computational cost.

11.4.5 FLOPS Analysis

Other than metrics measuring the correctness and sensitivity of detection, another important thing to consider is the computational cost of the YOLOv4 models. The inference GPU memory usage for each model was noted and can be seen in Fig. 11.17. To also take into account a platform-independent measure of computation, FLOPS (floating point operations per second) was plotted against the inference GPU memory usage data. As confirmed from the FLOPs vs. GPU memory usage plot, YOLOv4-tiny had significantly lower computational requirement compared to YOLOv4 and YOLOv4-CSP. The bigger models YOLOv4 and YOLOv4-CSP had comparable computational requirements.

In addition, FLOPs was plotted against the inference speed, where the speed-memory tradeoff was observed. YOLOv4-CSP had the slowest inference speed and was observed to have lower FLOPs values at similar network resolutions compared to YOLOv4, which had relatively higher memory consumption but faster speed. YOLOv4-tiny had the best values in terms of speed and computational cost. A similar pattern is seen in Fig. 11.18, where instead of FLOPs, the GPU memory usage was plotted against the inference speed of the YOLOv4 models.

11.4.6 YOLOv4 Models on Illumination and Occlusion Challenges

In an orchard environment, interobject occlusion naturally occurs. Thus, it is important to be able to detect pears despite this challenge. Moreover, in joint tree systems, data acquisition was done from the bottom side of the trees, which caused a high contrast characteristic for the images, making pear detection challenging even for humans.
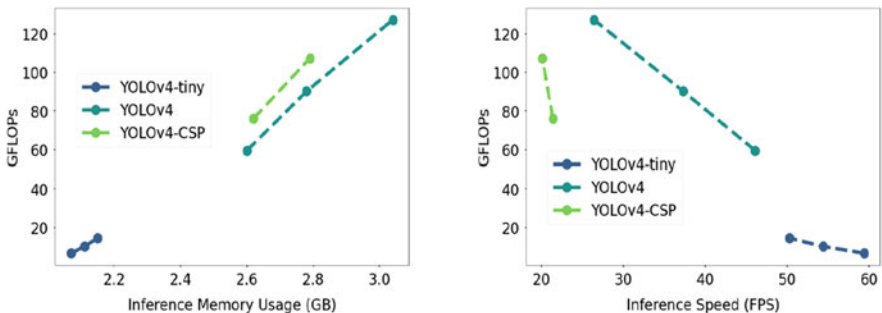
Fig. 11.17 Relationship of Floating Point Operations per Second (FLOPs) with Inference GPU Memory Usage (**left**) and Inference Speed (**right**) for each YOLOv4 model. The specific values are given in Table 11.12
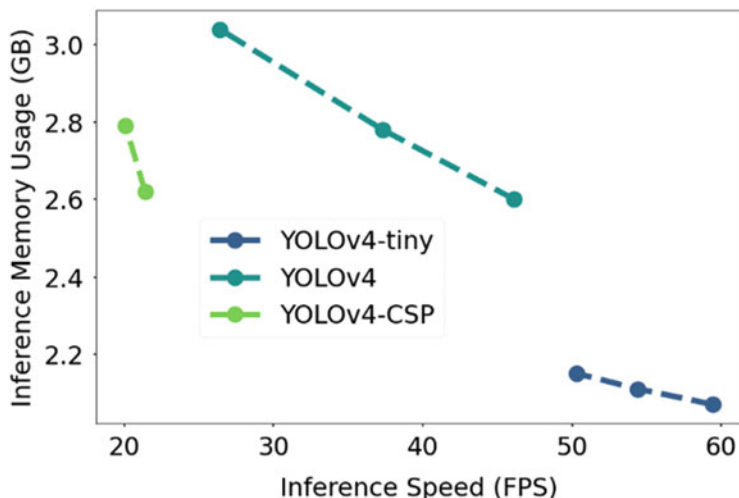
Fig. 11.18 Comparison of YOLOv4 models in terms of speed and GPU memory usage at inference time. The specific values are in Table 11.12

As seen from some sample detections in Fig. 11.19, where there is some slight occlusion and potentially challenging illumination, all the YOLOv4 models successfully detected the pears in the image despite having a considerable density of leaves in the background. With higher degrees of occlusion, the differences became more evident. In a still challenging illumination condition, but with a moderate degree of occlusion in Fig. 11.20, YOLOv4-CSP-608 and YOLOv4-tiny-406 (interestingly) successfully detected all the pears. However, in a good illumination condition and with a high degree of occlusion, only the YOLOv4-CSP models were able to successfully detect the pears (Fig. 11.21). YOLO is known to have some difficulty in detecting small objects (Du, 2018). From an image with pears that appear smaller and with some degree of occlusion, it was confirmed that YOLOv4-608 and 512 overcame this limitation of YOLO (Fig. 11.22). However, the limitation of YOLO in detecting small pears was exacerbated by the presence of occlusion for the other models.

11.4.7 Comparison of the Pear Counting Methods

Two methods of pear counting were evaluated: through an ROI line and through counting the unique IDs. The differences between the two method's performances are summarized in Table 11.13. The unique ID-based method performed better for most performance metrics in counting, specifically on MOTA, FN rate, $Recall_{count}$ and $F1_{count}$. Since the unique ID-based method has a less restrictive nature, it is more sensitive in counting, thus, had lower false negative rate and higher recall. However,
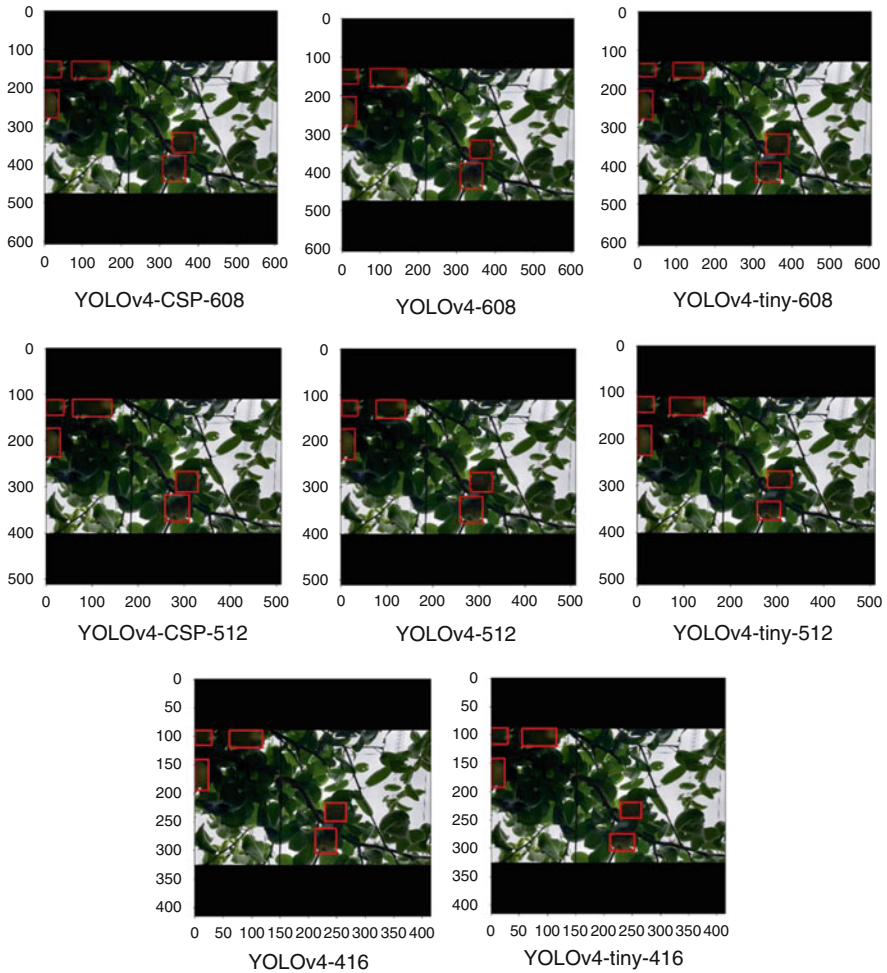
Fig. 11.19 Example detections from the different YOLOv4 models using an image with slight occlusion, potentially-challenging illumination and considerable density of leaves

the ROI line-based method filtered out false positive detections effectively, as shown by its FP rate of 1.89%. Due to its more restrictive nature, it also more correct detections than the unique ID-based method, as shown by its higher FP rate and $Precision_{count}$. However, the lack of sensitivity of the ROI-line based method was too disadvantageous, resulting in a very low $Recall_{count}$ of 58.49%. Thus, overall, the unique ID-based method had the best sensitivity and correctness tradeoff, as shown by its higher $F1_{count}$ of 87.85%.
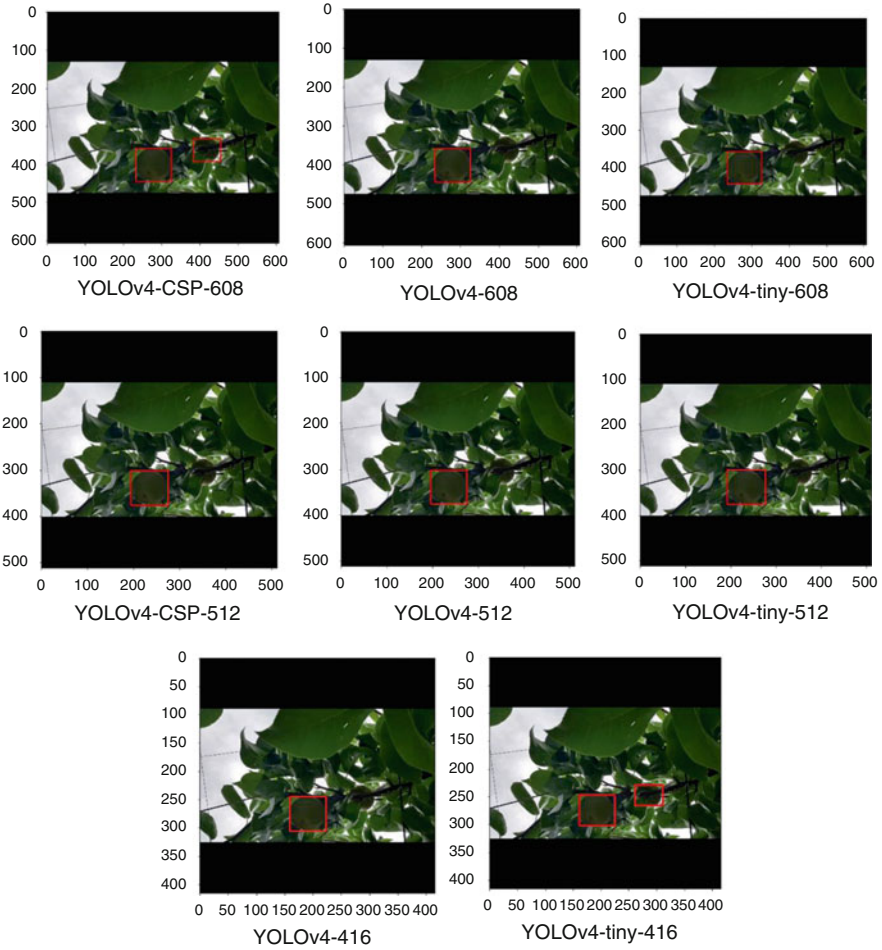
Fig. 11.20 Example detections from the different YOLOv4 models using an image with moderate degree of occlusion and potentially challenging illumination

11.4.8 Breakdown of the False Negative Counts in the ROI Line-Based Counting

The ROI line-based method had a low sensitivity in its counts but a very high correctness. How can we improve the sensitivity of an ROI line-based system? The authors observed the behavior of the pear objects that were detected by YOLOv4 but missed by the ROI line, which is summarized in Fig. 11.23. Of the false negative counts, 73% were actually detected by YOLOv4. Of the false negative counts 50% were detected only after passing the ROI Line. This might be due to the limitation of the computational resources, which might have been overcome using a higher-end
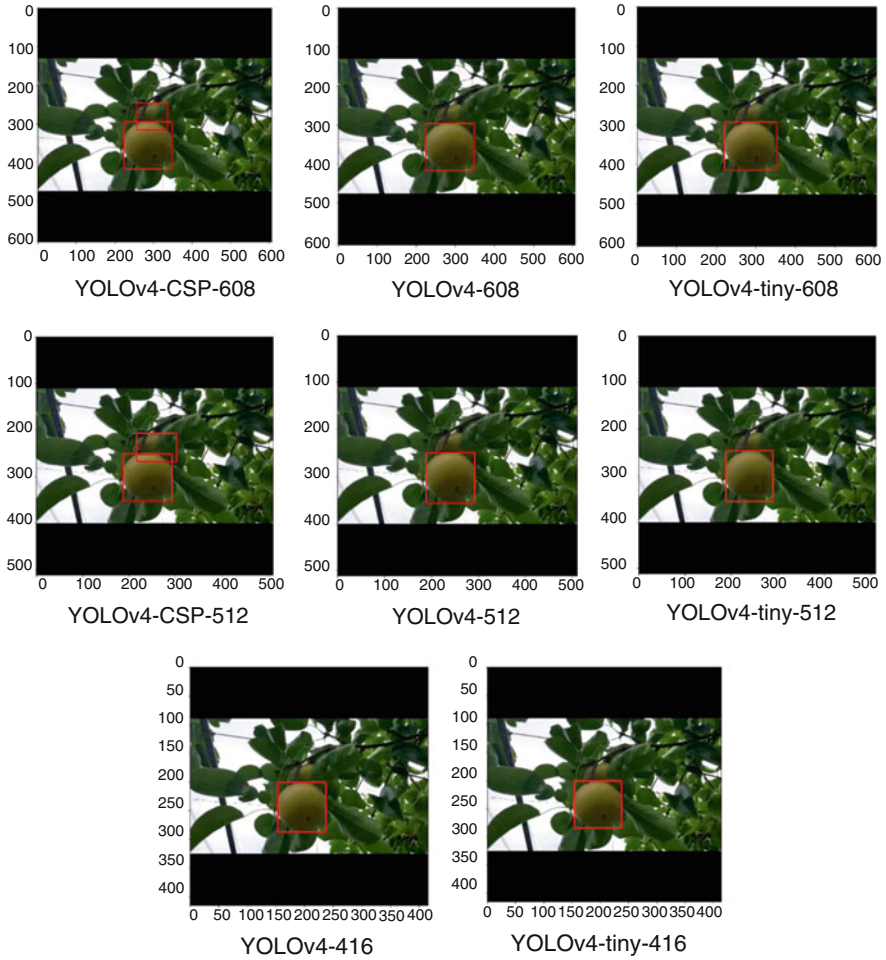
Fig. 11.21 Example detections from the different YOLOv4 models using a close-up image with a high degree of occlusion but good illumination

GPU device. Twenty-three percent of the false negative counts were detected just before or while crossing the line, and this could be attributed to the limitation of Deep SORT's tracking ability in challenging illumination and increased occlusion due to its reliance on appearance information in tracking. To tackle this limitation, it is recommended to switch the priority to motion information instead of appearance in challenging illuminations to achieve better performance and robustness.
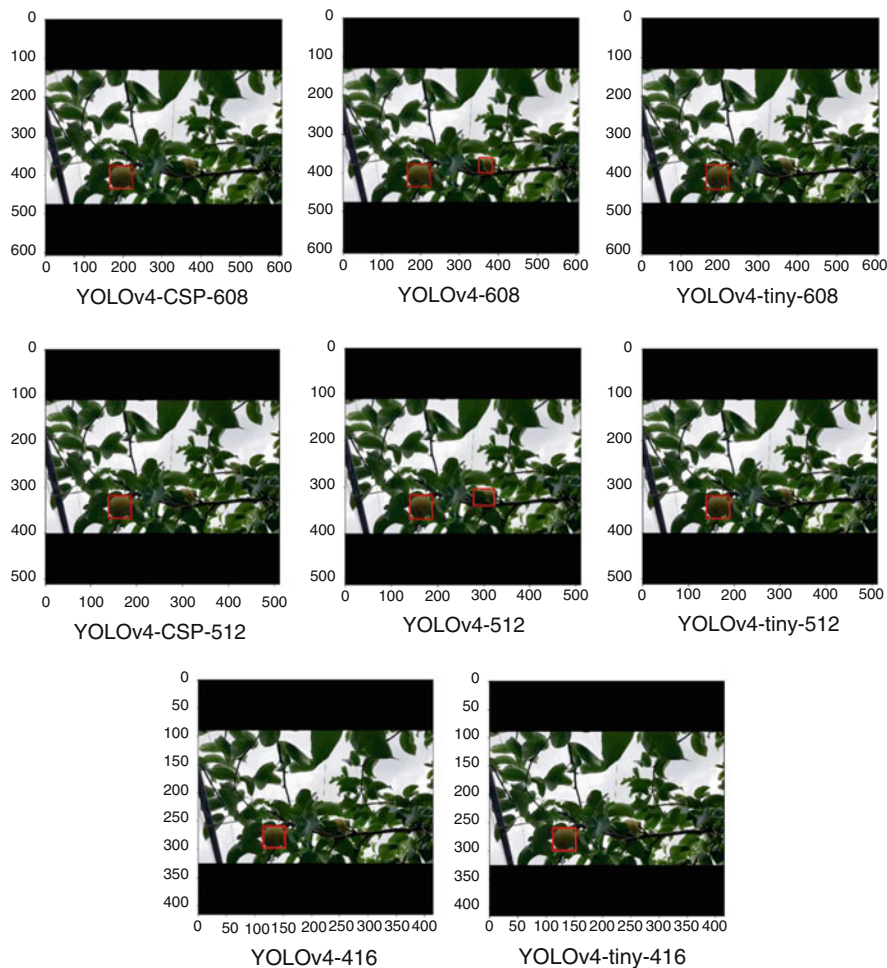
Fig. 11.22 Example detections from the different YOLOv4 models using a nonclose-up image with a moderate degree of occlusion

11.5   Conclusions

This study aimed to produce a robust real-time pear fruit counter for mobile applications using only RGB datasets with state-of-the-art object detection models (YOLOv4 models) and the MOT algorithm Deep SORT. In addition, we provided a systematic and pragmatic method for choosing the most suitable model for a desired application in agricultural sciences for further application. In terms of accuracy, YOLOv4-CSP was the optimal model with an AP of 98%. In terms of speed and computational cost, YOLOv4-tiny showed a very promising performance at a comparable rate with YOLOv4 at the lower network resolutions. If considering the balance in terms of accuracy, speed and computational cost, YOLOv4 was found to

Table 11.13 Pear counting performance metrics of the pear counting system based on YOLOv4 and Deep SORT on a 1920 × 1080 video between the two approaches in counting: unique-ID based and region-of-interest (ROI) based. Values in bold face are the higher ones between the counting approaches

Counting metrics	%	
	Unique-ID based	ROI-line based
MOTA	**75.47**	56.60
FN rate	**11.32**	41.51
FP rate	13.21	**1.89**
Precision$_{count}$	87.04	**96.88**
Recall$_{count}$	**88.68**	58.49
F1$_{count}$	**87.85**	72.94
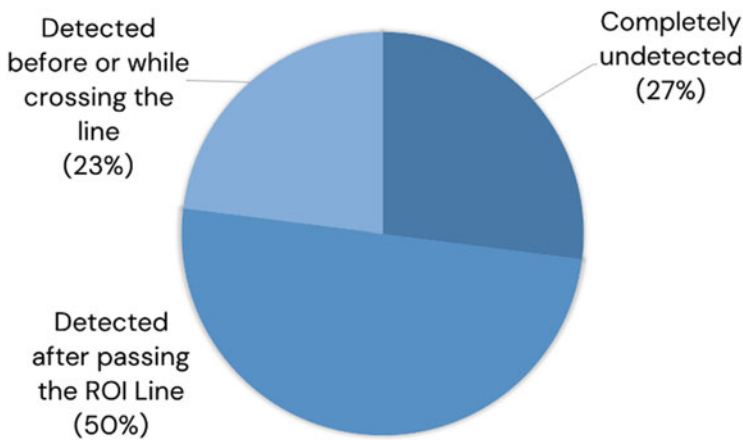
Fig. 11.23 Breakdown of the false negative counts in the region-of-interest line-based counting method

be the most suitable, with an AP > 96%, inference speed of 37.3 FPS and FN Rate of 6%. Thus, YOLOv4-512 was chosen as the detection model for the pear counting system with Deep SORT. Between the unique ID method and ROI Line method in counting, the former was found to be more reliable compared to the ROI-line method in counting the pears, as characterized by its F1$_{count}$ of 87.85%. It is important to note that this is the case because YOLOv4 had very low false negative detections. The ROI line could be more reliable because of its more restrictive nature, but due to flickering in detection it was not able to count some pears despite their being detected.

To fully maximize the accuracy of detection, cloud computing is recommended with YOLOv4-CSP in mobile applications instead of using the local resources of the mobile phone. If the cost of running a cloud computing service is the concern, Amazon's cloud services (AWS) support YOLOv4, where 1 million inferences is charged USD 1.362 (2 h) using the server inf1.xlarge in the us-east-1 region. The downside would be requirement to have an internet connection. If on-device inference is preferred, training YOLOv4-tiny at higher network resolutions may be the

best option. However, even YOLOv4-tiny requires at least about 2 GB of GPU; thus, low-end mobile phones would not be able to utilize it.

To combat possible flickering problems of the tracking algorithm, counting the unique IDs that tracked for a specific lifespan duration, such as more than 80% of the lifespan, is recommended.

Conflicts of Interest The authors declare no conflicts of interest.

Appendix

Algorithm 11.1 Pseudocode of the Fine-Tuning Process in Stage 2 Training of the YOLOv4 Models

---

$i = 0$ *for 1st training*, 1 *for 2nd training*
$LR_i = $ *learning rate for* $(i + 1)^{th}$ *training*
$mAP_i = $ *best mAP for* $(i + 1)^{th}$ *training*
$N = $ *iteration with the best mAP*
$w_i = $ *weights with the best mAP for* $(i + 1)^{th}$ *training*
$d = $ *decay rate*
$step1 = $ first time the learning rate was decreased
if $N_o < step1$ then
   train at $LR_1 = d \times LR_0$ for $N_o + 1000$
   if $mAP_0 < mAP_1$ then
      train at $LR_1 = d^2 \times LR_0$ for $N_1 + 1000$
      if $mAP_1 < mAP_2$ then
         select w_2
      else then
         select w_1
   else then
      select w_0
else then
   train at $LR_1 = d^2 \times LR_0$ for $N_o + 1000$
   if $mAP_0 < mAP_1$ then
      select w_1
   else then
      select w_0

References

Araujo, A., Norris, W., & Sim, J. (2019). Computing receptive fields of convolutional neural networks. *Distill*. https://doi.org/10.23915/distill.00021

Bernardin, K., & Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: The CLEAR MOT metrics. *EURASIP Journal on Image and Video Processing, 2008*, 1–10. https://doi.org/10.1155/2008/246309

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016) Simple online and realtime tracking. In *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*, 25–28 September 2016 (pp. 3464–3468). IEEE.

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv*, 2004.10934.

Cartucho, J., Ventura, R., & Veloso, M. (2018). Robust object recognition through symbiotic deep learning in mobile robots. In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, 1–5 October 2018 (pp. 2336–2341). IEEE.

Du, J. (2018). Understanding of object detection based on CNN family and YOLO. *Journal of Physics Conference Series, 1004*, 012029. https://doi.org/10.1088/1742-6596/1004/1/012029

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision, 88*, 303–338. https://doi.org/10.1007/s11263-009-0275-4

Fu, L., Duan, J., Zou, X., Lin, J., Zhao, L., Li, J., & Yang, Z. (2020). Fast and accurate detection of banana fruits in complex background orchards. *IEEE Access, 8*, 196835–196846. https://doi.org/10.1109/access.2020.3029215

Gai, R., Chen, N., & Yuan, H. (2021). A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-021-06029-z

Ghiasi, G., Lin, T.-Y., & Le, Q. V. (2018). DropBlock: A regularization method for convolutional networks. In *Proceedings of the NIPS'18: Proceedings of the 32nd International Conference on Neural Information Processing Systems* (pp. 10750–10760). Curran Associates.

Github. (n.d.). *Lyjourney supervisely2yolo*. Retrieved January 3, 2021, from https://github.com/lyjourney/supervisely2yolo

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S. et al. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3296–3297). IEEE.

Huang, Z., Wang, J., Fu, X., Yu, T., Guo, Y., & Wang, R. (2020). DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection. *Information Sciences, 522*, 241–258. https://doi.org/10.1016/j.ins.2020.02.067

Itakura, K., Narita, Y., Noaki, S., & Hosoi, F. (2021). Automatic pear and apple detection by videos using deep learning and a Kalman filter. *OSA Continum, 4*, 1688. https://doi.org/10.1364/OSAC.424583

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering, 82*, 35–45. https://doi.org/10.1115/1.3662552

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018a). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture, 147*, 70–90. https://doi.org/10.1016/j.compag.2018.02.016

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018b). A review of the use of convolutional neural networks in agriculture. *The Journal of Agricultural Science, 156*, 312–322. https://doi.org/10.1017/S0021859618000436

Kang, H., & Chen, C. (2020). Fast implementation of real-time fruit detection in apple orchards using deep learning. *Computers and Electronics in Agriculture, 168*, 105108. https://doi.org/10.1016/j.compag.2019.105108

Koirala, A., Walsh, K. B., Wang, Z., & McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of 'MangoYOLO'. *Precision Agriculture, 20*, 1107–1135. https://doi.org/10.1007/s11119-019-09642-0

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly, 2*, 83–97. https://doi.org/10.1002/nav.3800020109

Kuznetsova, A., Maleva, T., & Soloviev, V. (2020a). Detecting apples in orchards using YOLOv3 and YOLOv5 in general and close-up images. In *Lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 12249, pp. 233–243). Springer. ISBN 9783030587987.

Kuznetsova, A., Maleva, T., & Soloviev, V. (2020b). Using YOLOv3 algorithm with pre- and post-processing for apple detection in fruit-harvesting robot. *Agronomy, 10*, 1016. https://doi.org/10.3390/agronomy10071016

Kuznetsova, A., Maleva, T., & Soloviev, V. (2021). YOLOv5 versus YOLOv3 for apple detection. In A. G. Kravets, A. A. Bolshakov, & M. Shcherbakov (Eds.), *Cyber-physical systems: Modelling and intelligent control. Studies in systems, decision and control* (Vol. 338, pp. 349–358). Springer.

Lawal, M. O. (2021). Tomato detection based on modified YOLOv3 framework. *Scientific Reports, 11*, 1447. https://doi.org/10.1038/s41598-021-81216-5

Leal-Taixé, L., Milan, A., Reid, I., Roth, S., & Schindler, K. (2015). MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv*, 1504.01942.

Lee, Y., Hwang, J., Lee, S., Bae, Y., & Park, J. (2019). An energy and GPU-computation efficient backbone network for real-time object detection. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Long Beach, CA, USA, 16–17 June 2019 (pp. 752–760). IEEE.

Li, G., Huang, X., Ai, J., Yi, Z., & Xie, W. (2021). Lemon-YOLO: An efficient object detection method for lemons in the natural environment. *IET Image Processing, 15*, 1–12. https://doi.org/10.1049/ipr2.12171

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2016). Feature pyramid networks for object detection. *arXiv*, 1612.03144.

Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. 2018.

Liu, G., Nouaze, J. C., Mbouembe, P. L. T., & Kim, J. H. (2020). YOLO-tomato: A robust algorithm for tomato detection based on YOLOv3. *Sensors (Switzerland), 20*, 1–21. https://doi.org/10.3390/s20072145

Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, Atlanta, GA, USA, 16–21 June 2013 (Vol. 28).

Misra, D. (2019). Mish: A self-regularized non-monotonic neural activation function. *arXiv*.

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In *30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), CVPR 2017* (pp. 6517–6525). IEEE. https://doi.org/10.1109/CVPR.2017.690

Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv*, arXiv:1804. 1–6.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2015). You only look once: Unified, real-time object detection. *Journal of Chemical & Engineering Data, 27*, 306–308. https://doi.org/10.1021/je00029a022

Redmon, J., Bochkovskiy, A., & Sinigardi, S. (2019). *Darknet: Yolov3 - Neural network for object detection*. Github. Retrieved June 17, 2019, from https://github.com/AlexeyAB/darknet

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*, 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the 2018 IEEE/CVF Conference on*

Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018 (pp. 4510–4520). IEEE..

Sehgal, A., & Kehtarnavaz, N. (2019). Guidelines and benchmarks for deployment of deep learning models on smartphones as real-time apps. *Machine Learning and Knowledge Extraction, 1*, 450–465. https://doi.org/10.3390/make1010027

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data, 6*, 60. https://doi.org/10.1186/s40537-019-0197-0

Tzutalin LabelImg. (2021). Retrieved January 3, 2021, from https://github.com/tzutalin/labelImg

Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2020a). Scaled-YOLOv4: Scaling cross stage partial network. *arXiv*, arXiv:2011.08036.

Wang, C.-Y., Mark Liao, H.-Y., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., & Yeh, I.-H. (2020b). CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Seattle, WA, USA, 14-19 June 2020 (pp. 1571–1580). IEEE.

Wojke, N., Bewley, A., & Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. In *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, 17–20 September 2017 (pp. 3645–3649). IEEE.

Wu, L., Ma, J., Zhao, Y., & Liu, H. (2021). Apple detection in complex scene using the improved YOLOv4 model. *Agronomy, 11*, 476. https://doi.org/10.3390/agronomy11030476

Yan, B., Fan, P., Lei, X., Liu, Z., & Yang, F. (2021). A real-time apple targets detection method for picking robot based on improved YOLOv5. *Remote Sensing, 13*, 1619. https://doi.org/10.3390/rs13091619

Yun, S., Han, D., Chun, S., Oh, S. J., Yoo, Y., & Choe, J. (2019). CutMix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea, 27 October–2 November 2019 (pp. 6022–6031). IEEE.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., & Ren, D. (2020). Distance-IoU loss: Faster and better learning for bounding box regression. *Proceedings of the AAAI Conference on Artificial Intelligence, 34*, 12993–13000. https://doi.org/10.1609/aaai.v34i07.6999

Chapter 12
Pear Recognition System in an Orchard from 3D Stereo Camera Datasets Using Deep Learning Algorithms

Siyu Pan and Tofael Ahamed

Abstract In orchard fruit picking systems for pears, the challenge is to identify the full shape of the soft fruit to avoid injuries while using robotic or automatic picking systems. Advancements in computer vision have brought the potential to train for different shapes and sizes of fruit using deep learning algorithms. In this research, a fruit recognition method for robotic systems was developed to identify pears in a complex orchard environment using a 3D stereo camera combined with mask region-convolutional neural networks (Mask R-CNNs) deep learning technology to obtain targets. This experiment used 9054 RGBA original images (3018 original images and 6036 augmented images) to create a dataset divided into a training, validation, and testing sets. Furthermore, we collected the dataset under different lighting conditions at different times which were high light (9–10 am) and low light (6–7 pm) conditions at JST, Tokyo Time, August 2021 (summertime) to prepare training, validation, and test datasets at a ratio of 6:3:1. All the images were taken by a 3D stereo camera which included PERFORMANCE, QUALITY, and ULTRA models. We used the PERFORMANCE model to capture images to make the datasets; the camera on the left generated depth images and the camera on the right generated the original images. In this research, we compared the performance of different types with the R-CNN model (Mask R-CNN and Faster R-CNN), and we also compared the performance of same types of instance segmentation model (Mask R-CNN and YOLACT) the mean Average Precisions (mAP) of Mask R-CNN, Faster R-CNN and YOLACT were compared in the same datasets with the same ratio. Each epoch in Mask R-CNN was set at 500 steps with total 80 epochs, Faster R-CNN was set at 40,000 steps, the YOLACT was set at 400,000 iterations for training. For the recognition of pears, the Mask R-CNN, had the mAPs of 95.22% for validation set and 99.45% was observed for the testing set. On the other hand,

S. Pan
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

mAPs were observed 87.9% in the validation set and 87.52% in the testing set using Faster R-CNN. The mAPs of YOLACT was 87.07% in validation set, 97.89% in testing set. The different models using the same dataset had differences in performance in gathering clustered pears and individual pear situations. Mask R-CNN outperformed Faster R-CNN and YOLACT when the pears are densely clustered at the complex orchard. Therefore, the 3D stereo camera-based dataset combined with the Mask R-CNN vision algorithm had high accuracy in detecting the individual pears from gathered pears in a complex orchard environment.

Keywords Mask R-CNN · 3D stereo camera · Pear detection

12.1 Introduction

Modern fruit harvesting is mainly conducted by human labor and is roughly the same in different regions of the world. However, it requires human involvement and thus, complexity and labor hiring from overseas. The globalization of the COVID-19 pandemic and its economic impact has wreaked havoc on all economies around the world, pushing many into recession and possibly even economic depression (Barua, 2020). Furthermore, the aging and availability of labor are concerns. Among common fruits, the pear stands out as an essential fruit type for daily life. For example, the Japanese pear (such as Pyrus pyrifolia Nakai) is one of the most widely grown fruit trees in Japan and has been used throughout the country's history (Saito, 2016). Regardless of the harvest season, due to the need for a large number of laborers for picking and a shortage of labor, the cost of pear picking has gradually increased.

The world labor force is predicted to decline by approximately 30% between 2017 and 2030 (Schrder, 2014). With the development of agricultural machinery, modern agricultural technology has gradually evolved from manual planting and picking to full automation and intelligence. Since the 1990s, with the development of computers and information technology, artificial intelligence and machine vision in agricultural machinery have become more effective and popular (Wei et al., 2014). Since most agricultural work involves repetitive content operations, one of the most popular agricultural robots is the picking robot. Over time, most countries in the world have developed intelligent picking robots through different methods and techniques to load and unload agricultural products and detect fruit and positioning issues (Bechar & Vigneault, 2016). Therefore, for relatively delicate and soft fruits such as pears, the use of picking robots can greatly increase productivity. However, in recent studies, object detection in picking robots was reported to cause injuries due to grasping or using shear to detach the fruit from the branch (Hannan & Burks, 2004). The successful picking of soft pears depends on the recognition of the shape of the pears to understand the curved surface of the fruit. In classical image processing, it is challenging to recognize fruits, as shapes and sizes vary in orchards. In addition, illumination is a concern in dense canopies. Variability occurs in the detection of pears due to their size, shape, and illumination. Therefore, a large

number of training datasets including size, shape, and illumination variabilities are needed to address the challenges of pear detection in complex orchard environments.

Deep learning has become a potential method to overcome the limitation of conventional segmentation in image analysis. It is one of the subfields of machine learning and has now developed a variety of different architectures (Ertam & Aydın, 2017). Self-Organizing Feature Map (SOFM) is the ability of the discussed neural network to determine the degree of similarity that occurs between classes. It is also a method that belongs to deep learning. Among other things, SOFM networks can be used as detectors that indicate the emergence of a widely understood novelty. Such a network can also look for similarities between known data and noisy data (Boniecki & Piekarska-Boniecka, 2004). Additionally, deep learning includes artificial neural networks (ANNs) (LeCun et al., 2015) and neural networks extracted by convolutional neural networks (CNNs) by fully connected layers (FCNs), where CNNs preserve the spatial relationships between pixels by learning internal features using small pictures of the input data (Krizhevsky et al., 2012).

Intelligent robot vision processing of target plants has become an indispensable step in agricultural intelligence and many excellent target detection methods are now widely used in the development of agricultural robots as target detection continues to develop. The first types was two-stage method (detection and segmentation) included Fast R-CNN (Girshick, 2015) and Faster R-CNN (Ren et al., 2015), which have roughly the same principle of selecting the region of interest by region feature network (RPN) (He et al., 2017) and then transmitting to the head layer to generate the edges as well as the species. With the demand for accuracy in target detection, Mask R-CNN (He et al., 2017) was introduced, which adds FPN (He et al., 2017) to the backbone layer based on Faster R-CNN and adds a new branch in the head layer to generate more accurate masks. The second type is one-stage detection included You Only Look Once (YOLO) (Redmon et al., 2016) and You Only Look At Coefficient Ts (YOLACT) (Bolya et al., 2019) which focuses on the detection of targets; all detection results are lower than the above models but faster than the above models. However, due to the demand for accuracy in target detection, the Mask R-CNN detection speed is slower than that of other detection models (Dorrer & Tolmacheva, 2020).

Some identification techniques identify by evaluating, extracting, and recognizing color, because, in the food industry, color is an identifier used by producers and processing engineers as well as consumers and is the most direct way of identification (Sobol et al., 2020). Therefore, color extraction is also widely used in identification technology. Boniecki et al. (2010) analyzed the classification ability of Kohonen-type neural models learned using "unsupervised" methods. Classification of three selected apple varieties frequently found in Polish orchards was carried out. The neural classification was based on information encoded in the form of a set of digital images of apples and dried carrots. Representation in the form of a palette of the main colors occurring in fruits and dried vegetables and selected shape coefficients were used as a basis for the classification (Boniecki et al., 2010).

However, it is not enough for deep learning. Most of the deep learning methods were limited to RGB images which have a limitation of depth information. In a

recent study, a thermal camera was used to detect tree trunks in a complex orchard. However, in comparing a tree trunk to some fruit, there is more complexity in detecting and measuring the distance for picking information (Jiang et al., 2022). A 3D stereo camera has further advantages in addition to conventional camera sensors. The 3D stereo camera mimics and imitates the human eye imaging principle. With the powerful visual system of the human eye, the perception of the third dimension (depth) is derived from the difference between the image formed by the left eye and the right eye. Because of this difference, the human eye visual system introduces the third dimension (depth), and the 3D stereo camera receives biological inspiration to detect the depth information of an object by extracting three dimensions of information from the digital image and using it for 3D reconstruction. In addition, the camera perceives the depth of objects in the range of 1–20 m at 100 FPS (Ortiz et al., 2018). By detecting complex situations in orchards with a 3D stereo camera combined with Mask R-CNN vision algorithms, specified fruits can be detected.

Mask R-CNN is conceptually simple and was proposed by He et al. (2017). It is a flexible, pass-through object instance segmentation framework. This method can efficiently detect objects in images while generating high-quality segmentation masks for each instance. Mask R-CNN was also used for instance segmentation of detected objects and the evaluation of human poses (He et al., 2017). Several studies have shown that Mask R-CNN can be used for the detection of some fruits. Jia et al. (2020) used a series of apple images with a size of 6000 × 4000-pixel resolution under natural light using a Canon camera for cloudy and sunny weather conditions (Jia et al., 2020). Yu et al. (2019) proposed a Mask R-CNN-based algorithm to detect and quantify wild strawberries, and the fruit detection results of 100 test images showed an average detection accuracy of 95.78% and a recall rate of 95.41% (Yu et al., 2019). All of the above results showed that Mask R-CNN can be used for instance segmentation. In the above study, RGB images were used, which did not cover the depth information of the distance. However, Mask R-CNN with a 3D stereo camera can be further used for complex canopy and the weight files produced by the dataset produced by the common dataset. However, the 3D stereo camera has problems such as recognition errors and difficulty in obtaining depth information when detecting in real time. If the additional function has the depth information of the garden, then masking in terms of shape and size is still possible.

Mask R-CNN extended the object detection framework of Faster R-CNN by adding an additional branch at the end of the model, thus achieving instance segmentation for each output suggestion frame using a fully connected layer (Cai & Vasconcelos, 2019). Unlike ROI-Pooling of Faster R-CNN, ROI-Pooling inputs an image and multiple regions of interest (ROIs) into a feature map of fixed size, which was then mapped to a feature vector by a fully connected network (FCN) (Krizhevsky et al., 2012). However, ROI-Align in Mask R-CNN canceled the quantization of ROI-Pooling twice and retained the decimals, and then used bilinear interpolation (Kirkland, 2010) to obtain the image values on pixel points with floating-point coordinates. This was because although the quantization did not affect

the classification in the work, it had a significant negative impact on predicting the exact mask for pears in the orchard (He et al., 2017).

However, the complexity of orchards causes difficulty in detection, such as the presence of leaf shading, overlapping fruits, insufficient light, interruption of light due to nets over the canopy, and more shadows in orchards, which affect the detection results. Faster R-CNN was used to detect peppers, melons, and apples using multiple vision sensors (Sa et al., 2016), and although high detection accuracy was achieved, the detection of overlapping fruits was greatly reduced. Mask R-CNN has the potential to help overcome problems with size, shape, and illumination. Since the Mask R-CNN uses instance segmentation, it can over detect different individuals of the same species, so overlapping parts of the fruit can also be detected precisely and variability in shape can be adjusted, thus improving the accuracy of detection. Therefore, the purpose of this research is to develop a pear recognition system using instance segmentation based on a Mask RCNN from 3D camera datasets. The expected recognition of pears can be implemented as a fruit picking mechanism with fewer injuries to the surface with the recent advancements of manipulators and robots.

12.2   Materials and Methods

12.2.1   Field Data Collection

In this study, a 3D stereo camera named ZED (Stereolabs Inc. San Francisco, CA, USA) was used to collect 3018 (4-channel) original pictures from the T-PIRC (36° 07′04″ N, 140°05′45″ E) on a sunny day. The video shot with the ZED camera simulated the movement of the trajectory of the manipulator, and the observation distance from the pear was less than 50 cm. The video was trimmed into a frame-by-frame binocular image through the ZED camera protocol. The right side of the camera was showed the images in depth images, and the left side was 4-channel RGBA images.

Considering the influences of different light intensities in the natural environment and the camera parameters, in this research, 4 videos were taken at 9–10 am and 4 video were taken at 6–7 pm from Tsukuba-Plant Innovation Research Center (T-PIRC). The videos were taken during pears were at the fruit stage grown in the orchards covered with nets (Fig. 12.1a, b). The total number of original images used was 3018 in the training, validation, and testing process. Among these datasets, there were 1818 images used for training, 900 images for validation, and 300 images for testing. As mentioned before, that we collected the data from different times with different light intensities included high light and low light conditions (Table 12.1).
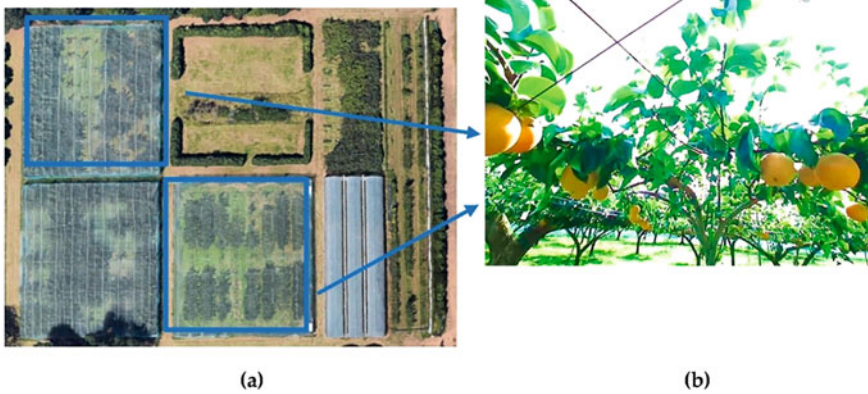
Fig. 12.1 Aerial view of orchards for data collection located at the Tsukuba-Plant Innovation Research Center (T-PIRC), University of Tsukuba, Tsukuba, Ibaraki. (**a**) Satellite view of Tsukuba-Plant Innovation Research Center (T-PIRC); (**b**) the view of pear orchard in T-PIRC

Table 12.1 Dataset collection times and light conditions in the complex orchard

Date	Time	Light condition
24 August 2021	9:00–10:00	High light
24 August 2021	18:00–19:00	Low light

12.2.2 *Instance Segmentation*

Image segmentation techniques consisted of object detection, semantic segmentation, and instance segmentation. Object detection solved the problem of identifying the content and location of images. Semantic segmentation was used to label each object with classes. However, instance segmentation was a combination of object detection with boundaries and semantic segmentation with classes (Fig. 12.2). In the case of instance segmentation, the object pear fruit was recognized as individual pears inside the same class compared to semantic segmentation. Many instance segmentation methods were based on segmentation proposals (He et al., 2017). Deep-Masks (Sa et al., 2016) proposes segmentation candidates followed by Fast R-CNN for classification. These methods were very slow and inaccurate. Mask R-CNN performs parallel prediction based on the masks and labels, making its instance segmentation method simpler and faster (He et al., 2017).

12.2.3 *Mask R-CNN*

Segmentation algorithm was gradually developed. There were two different methods of segmentation algorithm. One was single-stage (detection). A model that directly obtained the classes and boxes information of the objects presented in the input
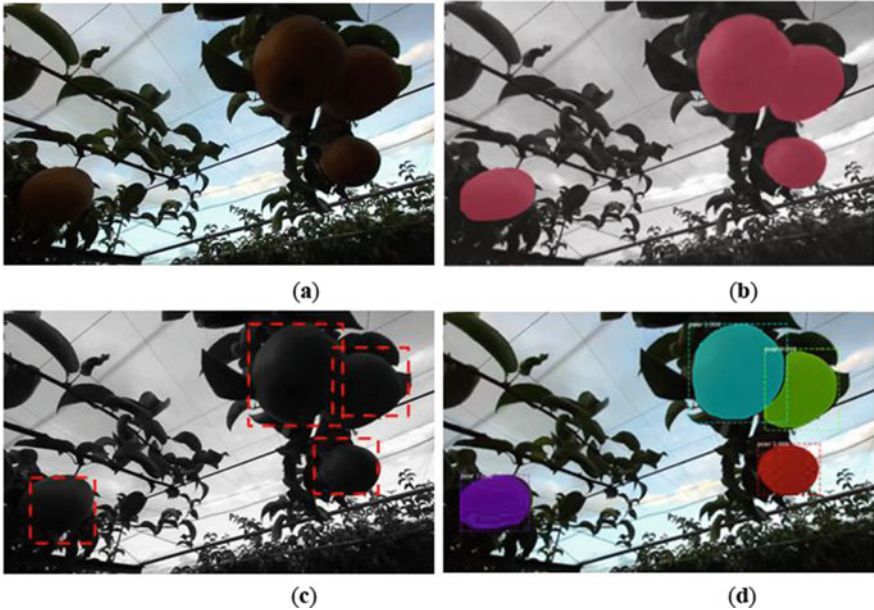
Fig. 12.2 Different segmentation in pear detection using 3D camera datasets, (**a**) original image; (**b**) semantic segmentation; (**c**) object detection; and (**d**) instance segmentation

image (YOLO and YOLACT), while the other was two-stage (detection + segmentation). First it located the frame of the target object and then segmented the target object within the frame (Faster R-CNN and Mask R-CNN). In the Mask R-CNN process, three main parts were followed: first, the backbone network extracts feature maps from the input image; second, the feature map outputs from the backbone network were sent to the region proposal network (RPN) (Ren et al., 2015) to generate regions of interest (ROIs); third, the ROI maps were output from the RPN, mapped to the shared feature maps to extract the corresponding target features, and then output to the FC and full convolutional networks (FCN) for target classification and instance segmentation (Fig. 12.3). This process generated classification scores, bounding boxes, and segmentation masks. With the evidence of the presented research, Mask R-CNN was used to detect the fruit.

The original images enter the backbone network for selection and screening to get the feature maps. Then, the foreground and background are extracted in the RPN network, enter the ROI-Align network for standardization, and finally enter the head network to generate classes, boxes, and masks for pear detection.
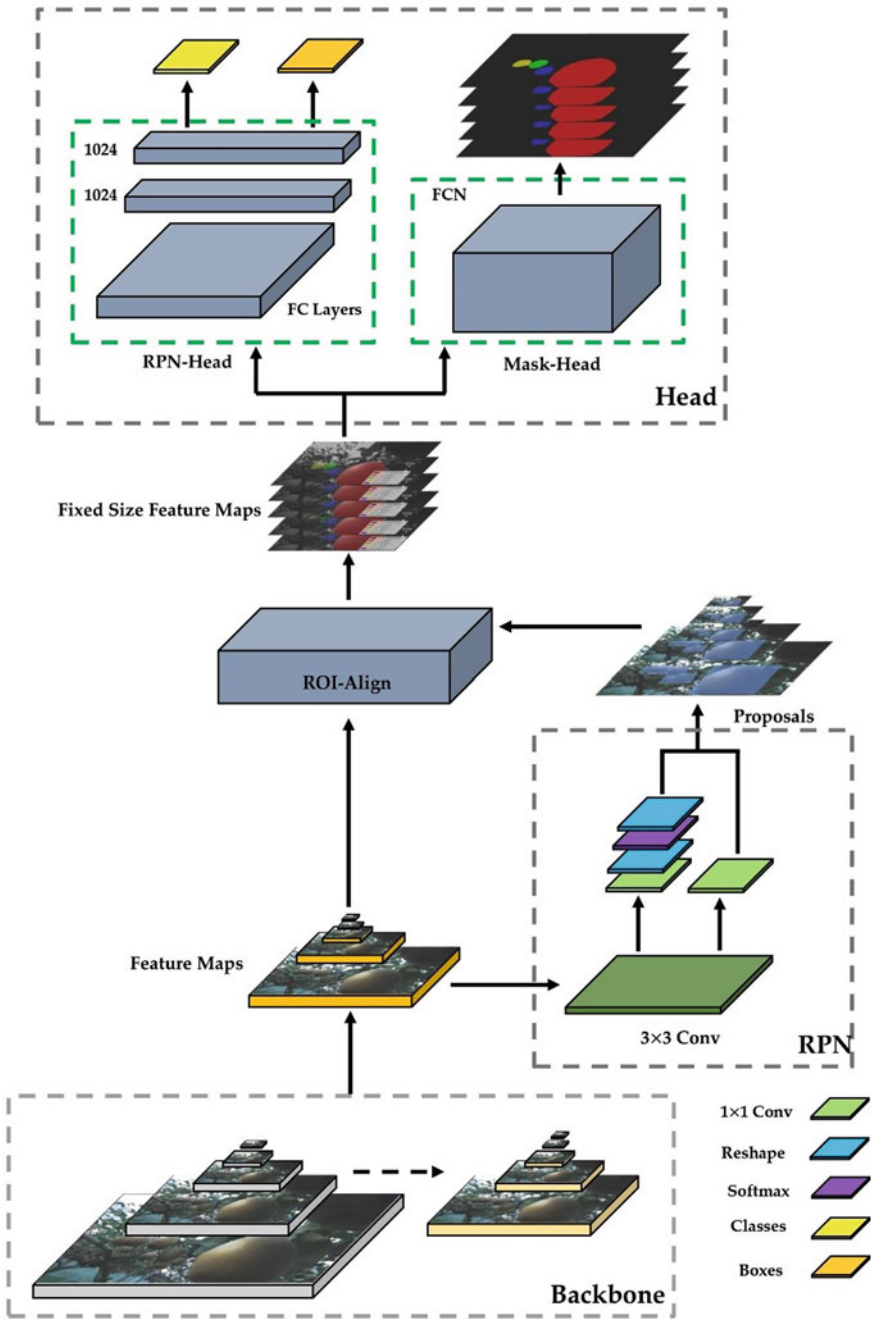
Fig. 12.3 Mask R-CNN structure for pear quantity in orchards from 3D camera datasets

12.2.4   ZED AI Stereo Camera

This research used a stereo camera as the main camera for data collection. The stereo camera was an integrated binocular camera that used advanced sensing technology based on a stereo vision to provide video acquisition, depth information, real-time location information, and other technologies. It has been applied to target reconstruction, position acquisition, and other fields (Tran et al., 2020). A stereo camera was used to obtain the distance of the pear through its depth functions, such as a 3D point cloud map. Therefore, the distance between the target fruit with the mask to the camera is measured and transmitted to the upper computer. Then, to realize the simulated grasping of the robot, Mask-RCNN was used for apple pickings and implemented in the intelligent platform.

12.2.5   Data Preparation

12.2.5.1   Deep Learning Environment

This experiment used a processor 11th Gen Intel(R) Core (TM) i7-11700F @2.50 GHz(16CPUs), ~2.5 GHz, 16,384 MB RAM, and Nvidia GeForce RTX 3060 GPU with Windows® 10 home edition™, CUDA 10.0, cuDNN 7.4, and Visual Studio™ 2019 as the training base. The environment configuration was created based on Mask R-CNN environment under anaconda, where the TensorFlow version was used in TensorFlow-gpu2.14.0, Keras 2.6.0, and Python 3.6.

12.2.5.2   Video to Image Conversion

The videos were taken with a stereo camera and images were converted to a specific format. Since the stereo camera came with its own shooting software, ZED Explorer™, which could shoot videos in three modes (ULTRA, PERFORMANCE, and QUALITY), the PERFORMANCE mode was chosen for this experiment, shooting a number of videos in HD720 stored in SVO format. The ZED protocol was used to trim the images to PNG format, and the image resolution was 1280 × 720.

12.2.5.3   Image Annotation

LabelMe® was used as the image annotation tool for semantic segmentation written in JavaScript for online labeling (Russell et al., 2008) The difference from LabelImg® was that the target was plotted in detail, and then a target mask was generated in LabelMe. LabelMe labeled all targets under the software interface, and

different classes were named as different label tags and different entities of the same class were named at once in order.

12.2.6   Data Splitting

The dataset consisted of 3018 images taken at different times of the day. The Mask R-CNN dataset was divided into training, validation, and testing sets, with the ratio set at 6:3:1.

In earlier experiments, three sets of videos with 1080 P resolution and .avi format were taken with an iPhone™ 11 mobile phone. We observed that the training set taken with the ZED camera made it difficult to test the videos and images taken with the mobile phone. This was due to the different camera calibration modules used on the stereo and the mobile phone camera as well as the different apertures and light transmission of the cameras making accurate identification difficult. This was why all of the experiments were conducted using stereo cameras to produce the dataset and test set. Second, as in the video taken by the stereo camera, different colors of pears existed in different shadows, so the individual pears in the dataset also showed three colors: bright yellow, yellow, and dark yellow. In this study, all colors of pears were calibrated and placed in the training set and the validation set to achieve adaptation to each angle and each color of pears.

Additionally, due to the homogenization of the original dataset, the shape of the pears and leaves tended to be similar under the dark light condition. Therefore, we decided to perform data augmentation on this dataset. Since the shape of the pear is similar to a sphere, and the shape of the leaf was irregular, we flipped and rotated each image of the original dataset so that the pear still tended to be spherical at different angles, but the leaves presented different shapes at different angles. The data set was expanded to 9054 images with the training set having 5054 images, the validation set having 2700, and the testing set having 900 at the ratio of 6:3:1 by data augmentation. We also rotated each image of the original dataset by 30° and flipped each image by 180° with the same method.

12.2.7   Training Process of Mask R-CNN

12.2.7.1   Feature Extraction (Backbone: ResNet101 + FPN)

A deep convolutional network referred to a network with different depths that could be constructed by constructing different weights. This was widely used in image feature extraction. However, with further deepening of the convolution network, the more convolution layers, the higher the corresponding training error. For the original network, simply increasing the depth led to gradient dispersion or exploding gradients. To address this issue, we represented this layer as an input-based learned
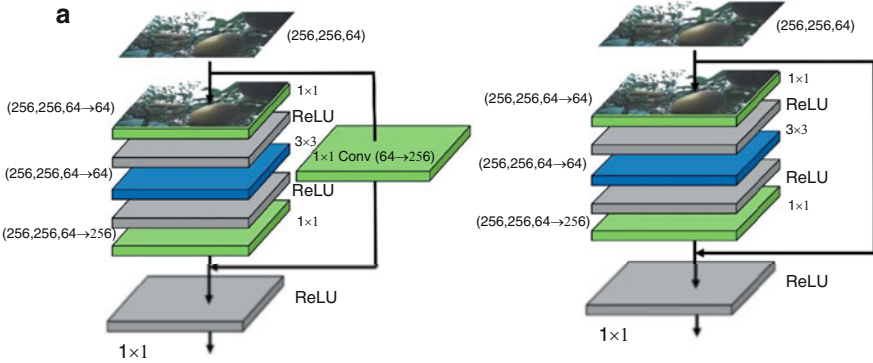
Fig. 12.4 The inner structure of ResNet101 as an example of second layers (C2): (**a**) is conv block and (**b**) is identity block. The images which were inputted into the ResNet have changed the channels. Conv block is the first stage of each layer, and the identity blocks and conv blocks were combined to the ResNet

residual function. Experiments showed that residual networks were easier to optimize and improved accuracy by adding considerable depth (He et al., 2016).

Residual networks (ResNet) were widely cited in the backbone networks of Faster R-CNN and Mask R-CNN, with ResNet50 and ResNet101 being the most common for Mask R-CNN. The Mask R-CNN using ResNet101 outperforms all previous basic variants of the state-of-the-art model (He et al., 2017), including the single-model variant of G-RMI (Huang et al., 2017) (Fig. 12.4).

These figures showed the different residual modules in a stage. The basic structure of these two modules met the standard residual structure. The difference between the convolutional block and identity block was that the convolutional block had a 1 × 1 conv layer. The shortcut of the convolutional block needed to go through a 1 × 1 conv for converting the number of channels. The identity block was directly connected to the upper output level. Therefore, ResNet101 was chosen as the backbone for the Mask R-CNN in this research.

The feature pyramid network (FPN) (Lin et al., 2017) as an elaborate multiscale detection method, had a structure consisting of three parts: bottom-up, top-down, and lateral connections. This structure allowed the features of each layer to be fused so that they had both strong semantic and strong spatial information. Feature extraction using the ResNet-FPN backbone for Mask R-CNN showed a great improvement in accuracy and speed (He et al., 2017). The structure of Mask-RCNN feature extraction was based on ResNet 101 and FPN (Fig. 12.5).

Top-down: no difference from the traditional feature extraction process, ResNet was used as the skeleton network and then divided into five stages according to the size of the feature map. These were named Stage 1, Stage 2, Stage 3, Stage 4, and Stage 5. In the convolution process, Conv2, Conv3, Conv4, and Conv5 are defined as C2, C3, C4, and C5. Next, the stage passes through the FPN on the right side from top to bottom and from left to right. Sampling starts from the last layer and samples to the nearest upper layer. The results of the previous layer were connected
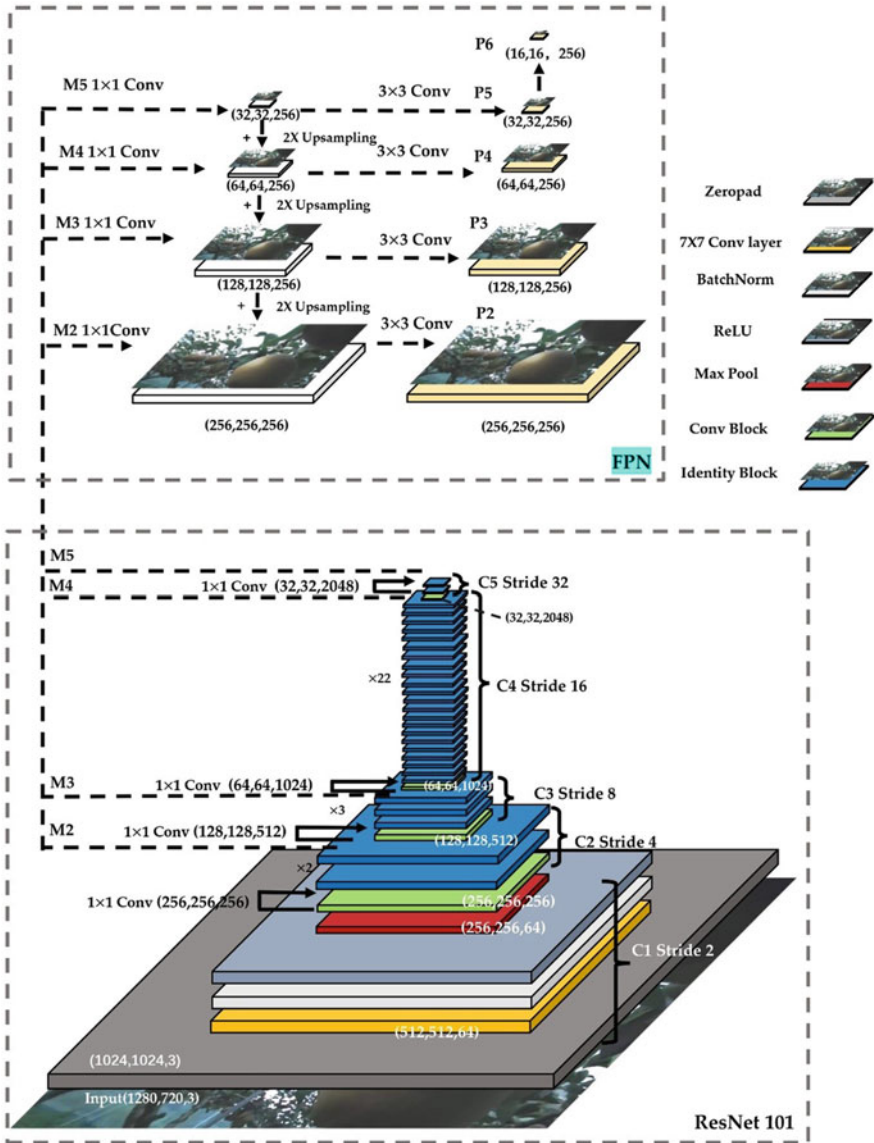
Fig. 12.5 ResNet101 + FPN for pear quantity recognition

horizontally by a layer conv 1 × 1, which was used to reduce its channels, and the results of all sampled layers were set to the same channels, fixed at 256. From there, it was processed through a layer of conv 3 × 3 to eliminate the aliasing effect. P2, P3, P4, P5, and P6 from FPN were used as RPN inputs, and P2, P3, P4, and P5 were used as subsequent Mask R-CNN inputs (Huang et al., 2017).

12.2.7.2   Region Proposal Network (RPN)

RPN, as a fully convolutional network (FCN) (Girshick et al., 2015), was specifically targeted for the task of generating detection suggestions and extracting candidate frames (Girshick et al., 2015). Based on P2, P3, P4, P5, and P6 obtained in Sect. 12.2.7.1, a series of anchors were generated in the RPN. Taking the P6 layer as an example, the feature map size of the P6 layer was (16, 16, and 256), and its step size relative to the original map was 64 so that each pixel point on P6 was generated with 3 transformations of anchors with aspect ratio {0.5, 1, 2}. There were a total of 1200 generated anchors (16 × 16 × 3 = 768), and by analogy, P2 (256 × 256 × 3 = 196,608), P3 (128 × 128 × 3 = 49,152), P4 (64 × 64 × 3 = 12,288), and P5 (32 × 32 × 3 = 3072); from P2 to P5, a total of 261,888 anchors were generated on the original image (Fig. 12.6).

The positive and negative classes for network training were established by the generated anchors and 256 were selected according to intersection-over-union (IoU) (Ren et al., 2015) for training the RPN, of which, 128 positive samples (foreground) and 128 negative samples (background) were guaranteed. This step was performed to calibrate the anchor box, in addition to calculating the offset between the anchor box and the ground truth (Fig. 12.7).

The IoU represented the overlap of the two bounding boxes, where the ground truth and the offset of the anchor box can be expressed as.

$$\Delta_X^* = (x^* - x_a)/w_a, \quad \Delta_Y^* = (y^* - y_a)/h_a \tag{12.1}$$

$$\Delta_X^* = (x^* - x_a)/w_a, \quad \Delta_Y^* = (y^* - y_a)/h_a \tag{12.2}$$

x_a, y_a represented the coordinate value of the center point of the anchor box; w_a, h_a represent the width and height of the anchor box; x^*, y^* represented the coordinate
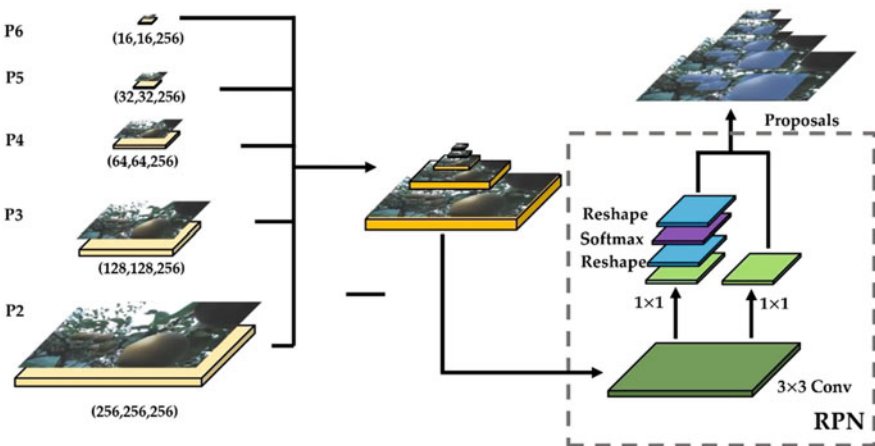
Fig. 12.6 RPN in Mask R-CNN for extracting proposals of original pear images
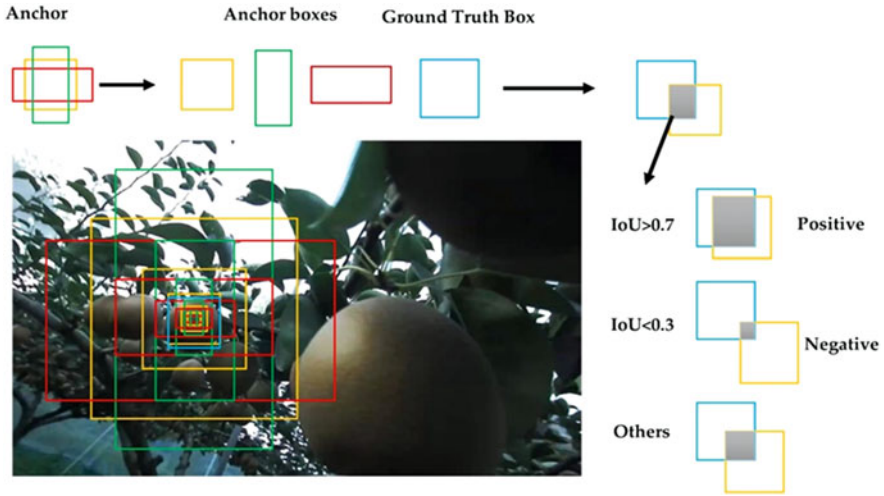
Fig. 12.7 Generation for IoU by comparing anchor boxes with ground truth boxes. If IoU > 0.7, then label = 1 positive; if IoU < 0.3, then label = −1 negative; others, label = 0

value of the center point of the ground truth; and w^*, h^* represented the coordinate value of the width and height of the ground truth.

Next, we entered the regression and classification of the RPN, which classified each layer of anchor into foreground and background and the four displacement quantities of regression. For example, in the P6 layer, the size of the feature map was 16×16, there were $16 \times 16 \times 3$ anchors, and the probability of each anchor as foreground and background was calculated separately. The array was ($16 \times 16 \times 3$, 2), and the regression of offset was ($16 \times 16 \times 3$, 4). The same operation was performed from P2 to P6, and the classification of (261,888 and 2) was obtained in total information and the regression information of (261,888 and 4).

The scores (probabilities) corresponding to the 256 positive and negative samples were found from (261,888 and 2); as a result, 256 positive and negative samples were obtained. The Softmax cross-entropy loss values were calculated using the scores and the label values of the positive and negative samples, which caused RPN to initially extract foreground anchors as candidate regions using anchors and the Softmax function.

The offset corresponding to the index where the 128 positive samples were located from the (261,888 and 4) regression array was found, and this offset was used with the offset calculated between the positive samples and the real frame to calculate the loss value to regress the proposals from the positive samples.

12.2.7.3   ROIs and ROI-Align

In this research, the ROI parameter was set for taking a certain amount from 261,888 anchors as the ROI during training; this parameter was set to 2000. Therefore, 2000 anchors from 261,888 anchors were taken as the ROI needed in the next stage.

Our method ranked the scores of the positive samples obtained in the previous stage of RPN from highest to the lowest, removed the top 2000 anchors with the highest scores, and accumulated the more accurate box coordinates by regression of the offset of RPN. Finally, a nonmaximal suppression (NMS) (Bodla et al., 2017) was performed on these 2000 anchors to eliminate duplicate boxes. Finally, 2000 matching ROIs were selected.

Since NMS processing was performed on the ROIs after the proposal, some of the layers with less than 2000 ROIs had supplemental 0 processing, so the 2000 ROIs obtained needed to be eliminated. Eliminating the ROIs filled with 0's and eliminating all the boxes with multiple objects in the ground truth at the same time, the IoU value of each ROI was calculated. Then, we calculated the IoU value of each ROI and ground truth, obtained 400 ROIs with positive and negative samples of 1:3, and finally returned its 400 samples, its displacement offset, and 400 masks.

ROI-Align of the standardized ROIs was used to obtain the final required feature maps. ROI-Align was a unique part of Mask R-CNN. Unlike ROI-Pooling in Faster R-CNN, ROI-Pooling was a standardized operation used to extract a small feature map from each ROI, and this feature had a fixed spatial range. This paper used 7×7 feature maps as fixed-size feature maps. First, the ROI of the floating-point number was quantized into a feature map of standard size, and the quantized ROI was quantized again to obtain an $N \times N$ integer feature map. Furthermore, for an image whose original size was 1280×720, after 8 samplings and resizing, the size of the obtained feature map was 160×160. Assuming that there was a 113×113 region proposal, the size of the feature map was 14.125×14.125 (113/8). At this time, after the first quantization, the region proposal size on the feature map was 14×14. Assuming that it eventually became a 7×7 fixed-size feature map, the feature map needed to be divided into 49 regions, and the size of each region was 14/7 = 2. At this time, the second quantization was performed, and the final small area size was 2×2. Finally, the maximum pixel quality was selected in each small 3×3 area to form a 7×7 fixed-size feature map.

Although ROI-Pooling did not change the categories in the data in the two quantization while it generated the mask, omitting the very small floating-point number also affected the area and size of the mask. Mask R-CNN canceled the two in ROI-Pooling-quantization operations and preserved floating-point numbers (He et al., 2017). For example, images were taken at a size of 1280×720. After 8 samplings, the size of the feature map (8×8) was obtained. Another 113×113 region proposal was also assumed to be mapped to the feature map. The size then changed to 14.125×14.125. The operation at this time was different from the above ROI-Pooling. The quantization operation was directly canceled, and the floating-point number was reserved. Assuming that the final required transformation was a
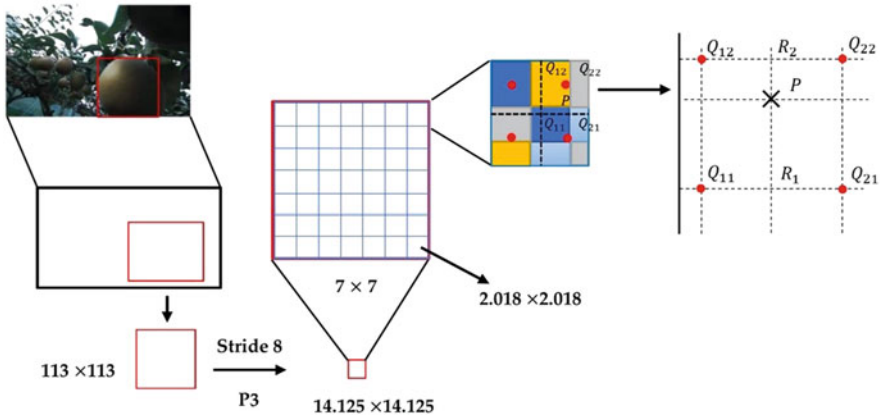
Fig. 12.8 Bilinear interpolation in ROI-Align was used to obtain fixed feature maps for pear recognition. *P* represents pixel coordinates that ROI-Align wanted to obtain after bilinear interpolation. Q11, Q12, Q22, and Q21 represent the four coordinates of known pixel points around point *P*

7 × 7 feature map, 49 small areas needed to be planned in the feature map with a feature map of 14.125 × 14.125. Since the size of each small area was changed to 14.125/7 = 2.018, the final minimum area was 2.018 × 2.018. Assuming that the number of sampling points was four, bilinear interpolation was used for calculation so that the pixel values of the four points could be obtained. The quantization operation was eliminated, and the errors were greatly reduced (Fig. 12.8).

The process of ROI-Align simply described and calculated the edge lengths of each ROI but does not round them. Each ROI region was divided into $K \times K$ bins, and the size of each bin was not rounded. The value of each bin was obtained by bilinear interpolation of the four values of the most adjacent feature map. Max pooling was used to obtain a feature vector of fixed length. Mask R-CNN uses bilinear interpolation, which is a useful operation that can be used to interpolate two-dimensional images to compare different pixel sizes or image spacings, which be used to remove ROI-Align in Mask R-CNN. Thus, the pixel values of four adjacent pixel points were obtained. For example, if the number of sampling points were 4, this could be expressed as follows: for each small area, the average was divided into four points, the center point position was taken for each part, and the pixel value of the center point position at this time was bilinear. The interpolation method was used for calculation. Through this method, the pixel values of the four points were obtained, thereby eliminating the need for two quantization operations and generating the corresponding mask for pear recognition.

12.2.7.4 Mask RCNN for Classification and Regression

The ROIs obtained after entering ROI-Align needed to pass the head architecture to perform category of classification, regression, and mask generation operations,
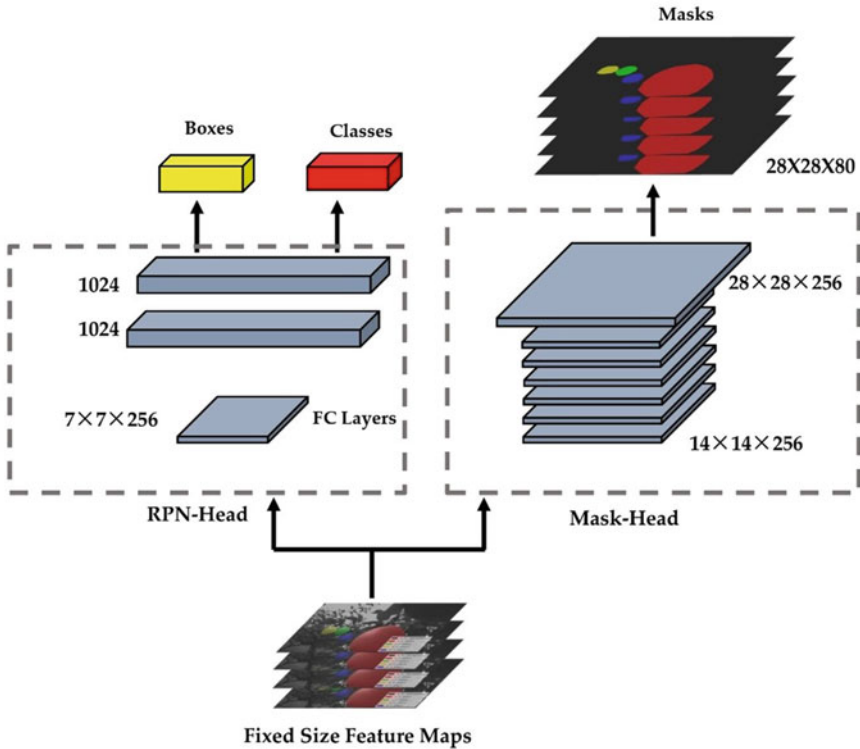
Fig. 12.9 Flow diagram of feature maps to produce boxes, classes, and masks for each pear in each fixed-size feature map after ROI-Align

which had two main branches. The upper branch indicated category classification and regression, and the lower branch generated the mask for each pear in each feature map.

After the ROI of the previous layer, a $7 \times 7 \times 256$ feature map was generated, and another branch was aligned to a $14 \times 14 \times 256$ feature map (Fig. 12.9). Since the outputs of P2, P3, P4, P5, and P6 were all 256, the final channel was still 256 after ROI-Align.

The branch of left side in Mask R-CNN had the same principle of classifying targets and generating frames with Faster R-CNN. The Faster R-CNN also calculated which category each proposal belongs to and outputs the classification result through the fully connected layer. Additionally, the position offset of each proposal was obtained by bounding box regression which was used to return a more accurate target detection frame. The branch of right side was based on the Faster R-CNN, and the full convolutional layer branch was added to obtain a more accurate mask. The difference between FCN and CNN is that the final FCN was replaced by a convolutional layer. Unlike the classical CNN, which uses FCN after the convolution layer to obtain a fixed-length feature vector for classification (fully connected

layer + Softmax output), FCN accepted an input image of arbitrary size and used deconvolution to sample the feature map of the last convolution layer to restore it to the same size as the output image. The feature map of a fixed size of the ROI region was generated by the ROI-Align operation. After 4 convolution operations, a 14 × 14 feature map was generated. Then, a 28 × 28 feature map was generated by up-sampling. Finally, a 28 × 28 feature map with a depth of 80 was generated by the deconvolution operation to obtain the exact mask.

12.2.7.5 Loss Function

The loss functions of the Mask R-CNN were divided into two main parts. The first part was the RPN loss function, which was similar to the Faster R-CNN. The RPN loss function consisted of two parts: the classification loss (L_CLS) and the bounding box regression loss (L_BOX).

The general formula was as follows (Girshick et al., 2015):

$$L = L_{\text{RPN}} + L_{\text{MASK}} \tag{12.3}$$

$$L_{\text{RPN}} = L_{\text{CLS}} + L_{\text{BOX}} \tag{12.4}$$

$$L(\{p_i\}, \{u_i\}) = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*) \tag{12.5}$$

$$L_{\text{CLS}} = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) \tag{12.6}$$

N_{cls}: Since the anchor generated in the RPN stage was only used to classify foreground and background and was set to 256, the value of N_{cls} is 256.

p_i was the probability of predicting the target, p_i^* was the group truth (GT) label.

$$p_i^* = \begin{cases} 0, & \text{negative label} \\ 1, & \text{positive label} \end{cases} \tag{12.7}$$

$L_{\text{cls}}(p_i, p_i^*)$ denotes two classes: the target and the nontarget logarithmic loss, in which

$$L_{\text{cls}}(p_i, p_i^*) = -\log\left[p_i^* p_i + (1 - p_i^*)(1 - p_i)\right] \tag{12.8}$$

$$L_{\text{BOX}} = \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*) \tag{12.9}$$

N_{reg} was the size of the feature maps, and $\lambda \frac{1}{N_{\text{reg}}}$ was used as the normalized weight to balance the classification loss and regression loss, which was taken as $\frac{1}{256}$ in the RPN training phase,

$t_i = \{t_x, t_y, t_w, t_h\}$ was a vector that represents the offset used for prediction in the RPN training phase,

t_i^* was the actual offset of the group truth corresponding to the positive anchors.

$$L_{\text{reg}}\left(t_i,\, t_i^*\right) = R\left(t_i - t_i^*\right) \qquad (12.10)$$

R was the smooth$_{L1}$ function

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \qquad (12.11)$$

L_{MASK} was the loss function resulting from adding the mask branch to the Mas R-CNN and had the functional expression

$$L_{\text{MASK}} = L\left(p_i,\, p_i^* t_i t_i^* s_i s_i^*\right) = \frac{1}{N'_{\text{cls}}} \sum_i L_{\text{cls}}\left(p_i p_i^*\right) + \lambda'^{\frac{1}{N'_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}\left(t_i t_i^*\right)}$$

$$+ \gamma \frac{1}{N_{\text{mask}}} \sum_i L_{\text{mask}}\left(s_i s_i^*\right) \qquad (12.12)$$

In this research, we conducted a discussion of the pseudo-code of Mask R-CNN for different datasets included training set (TRD), validation set (VAD), and testing set (TSD). We gave a brief logical explanation in the tabular format of the pseudocodes of the modified Mask R-CNN (Table 12.2).

12.2.7.6   Model Metrics Function

The results of the model prediction values were classified into four categories: true positive (TP), indicating a positive sample detected correctly; false negative (FN), indicating a negative sample predicted incorrectly; true negative (TN), a negative sample predicted correctly; and false positive (FP), a positive sample predicted incorrectly.

In this thesis, only pears needed to be detected, so pears were used as the only category. The task of Mask R-CNN was to detect the number of pears present in the pictures. Therefore, TP indicated the result that Mask R-CNN detected pears as pears in each image of the test set in the testing phase. FN pears were not identified. There was a missed identification: TN indicated that the part that was not identified as a pear, and FP indicated that the background or the leaves were identified as a pear (Figs. 12.7 and 12.10).

Table 12.2 Pseudocodes in pear recognition using Mask R-CNN, Faster R-CNN, and YOLACT in training and testing phases. Algorithm1 Training phase using Mask R-CNN; algorithm2 Testing phase using Mask R-CNN; algorithm3 Training phase using Faster R-CNN; algorithm4 Testing phase using Faster R-CNN; algorithm5 Training phase using YOLACT; algorithm6 Testing phase using YOLACT

Algorithm1 Training Phase using Mask R-CNN
1.**Inputs**:
2.   Dataset_train:$TRD = \{Image_i\}_{i=1}^{M}$,
3.   Datset_val: $VAD = \{Image_j\}_{j=1}^{N}$,where M, N are the number of images.
4.   **if** mode $=$ "training"
5.   Get object index in $Image_i$, $Image_j$
6.   Extraction from ResNet101 to FPN
7.   Anchor generation from $P2$, $P3$, $P4$, $P5$, $P6$
8.   BG and FG generation from RPN via (Eqs. 12.1 and 12.2)
9.   Calculated the L_{RPN} via (Eqs. 12.4–12.9)
10.   ROIs Generation from $ROI - Align$
11.   Masks, boxes, classes Generation from *the Head*
12.   Calculate the loss of the head layer L_{MASK} via (Eq. 12.12)
13.   Save_logs_weights(*mask _ rcnn _ shapes*. *h*5)
14.**Return**:
15. $logs$ _ $weights$, L_{RPN}, L_{MASK}, L
Algorithm2 Testing Phase using Mask R-CNN
1.**Inputs**:
2.   Dataset_Test:$TSD = \{Image_i\}_{i=1}^{M}$, where M is the number of images.
3.   GPU_COUNT $= 1$
4.   IMAGES_PER_GPU $= 1$
5.   **if** mode $=$" inference"
6.   Model.load_weights(*mask _ rcnn _ shapes*. *h*5)
7.   **For *i* in range** ($Image_i$):
8.   **Input** *Anchors*
9.   Generated *rpn _ ROIs*
10. $targer$ _ $ROIs = rpn$ _ $ROIs$
11.   Generated $target$ _ $class$ _ ids, $target$ _ $bbox$, $target$ _ $mask$
12.   Created *masks* for detections
13.**Return**:
14. $Image_i$, $masks$, $class$ _ id, $class$ _ $name$, $scores$
15.   Visualize.display_instances $Image_i$
Algorithm3 Training Phase using Faster R-CNN
1. **Inputs**:
2.   Dataset_train:$TRD = \{Image_i\}_{i=1}^{M}$,
3.   Datset_val: $VAD = \{Image_j\}_{j=1}^{N}$, where M, N is the number of images.
4.   **if** mode $=$ "training"
5.   Get object index in $Image_i$
6.   Extraction from VGG16 (*Visual Geometry Group Network*)

(continued)

Table 12.2 (continued)

7.	Region proposals generation from *RPN*
8.	ROIs generation from *ROI − Pooling*
9.	Classification from *the Head*
10.	Calculated *Loss*
11.	Save_logs_weights
12.**Return**:	
13.	*logs _ weights*, L_{RPN}
Algorithm4 Testing Phase using Faster R-CNN	
1.**Inputs**:	
2.	Dataset_Test:$TSD = \{Image_i\}_{i=1}^{M}$, where M is the number of images.
3.	if mode =" inference"
4.	Model.load_weights
5.	**For** i **in range** ($Image_i$) :
6.	**Input** *Anchors*
7.	Generated *rpn _ ROIs*
8.	*targer _ ROIs = rpn _ ROIs*
9.	Generated *target _ class _ ids*, *target _ bbox*
10. **Return**:	
11.	*Image$_i$, target _ class _ ids*, *target _ bbox*
12.	Visualize.display_instances *Image$_i$*
Algorithm5 Training Phase using YOLACT	
1.**Inputs**:	
2.	Dataset_train:$TRD = \{Image_i\}_{i=1}^{M}$,
3.	Datset_val: $VAD = \{Image_j\}_{j=1}^{N}$,where M, N are the number of images.
4.	**if** mode = "training"
5.	Get object index in *Image$_i$*, *Image$_j$*
6.	Extraction from ResNet101 to *FPN*
7.	Anchor generation from *P*3, *P*4, *P*5, *P*6, *P*7
8.	138×138×k Masks generation form Protonet
9.	Anchors generation in Feature maps from *P*3, *P*4, *P*5, *P*6, *P*7 in *Prediction Head*
10.	Mask and B-box coefficients generation based on anchors
11.	Anchors filtration from Fast NMS
12.	Matrix multiplication of Mask coefficient and prototype in *Crop & Threshold*
12.	Calculate the loss of L_{RPN}, L_{MASK}, L
13.	save_loss: L_{RPN}, L_{MASK}, Lnet
13.	save_logs_weights(*yolac _ customer _* 110 _ 400000. *pth*
Algorithm6 Testing Phase using YOLACT	
1.**Inputs**:	
2.	Dataset_Test:$TSD = \{Image_i\}_{i=1}^{M}$, where M is the number of images.
3.	**if** __name__ == '__main__':
4.	set_cfg(args.config)
5.	**if** args.images **is** None
6.	dataset = COCODetection(cfg.dataset.valid_images, cfg.dataset.valid_info,

(continued)

Table 12.2 (continued)

7.	transform=BaseTransform(), has_gt=cfg.dataset.has_gt)
8.	**else**:
9.	dataset = None
10.	net = Yolact()
11.	net.load_weights(args.trained_model)
12.	net.eval ()
13.	evaluate (net, dataset,train_mode=False)
14.	visualize.display_instances $Image_i$
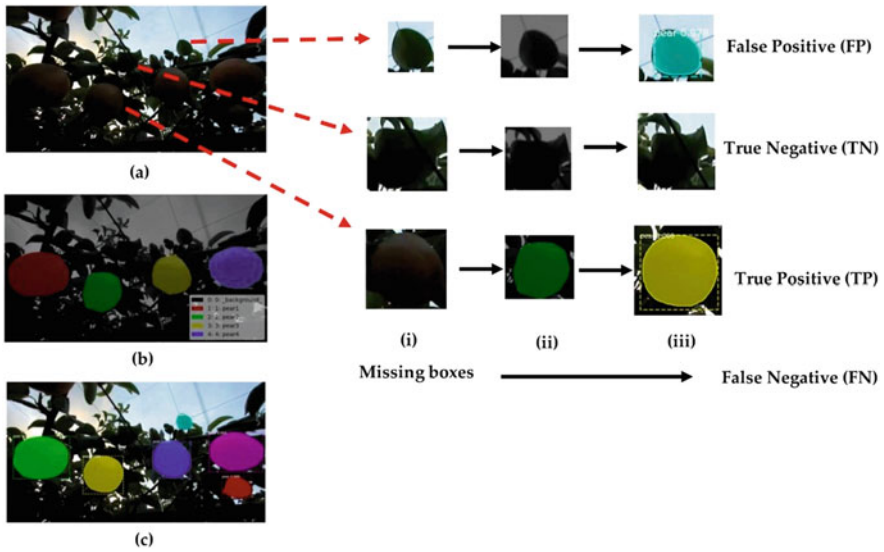
Fig. 12.10 Pear prediction for determining FP, TN, TP, and FN using Mask R-CNN, (**a**) original image; (**b**) cv_mask input image before testing and (**c**) mask image after testing. Precision overreflected the proportion of correct classification in the number of positive samples classified by the model. Its expression is

$$P = \frac{TP}{TP + FP} \tag{12.13}$$

Recall was the ratio of the number of correct samples to the number of positive samples, and its expression is

$$R = \frac{TP}{TP + FN} \tag{12.14}$$

Since there was only one category of fruit in this experiment, AP = mAP. Its value was between [0, 1], and the closer to 1, the better the model recognition effect.

12.3 Results

12.3.1 Training Details

In this research, 9054 four-channel RGBA images (3018 images were original images and 6036 images were augmented images) in PNG format were used, and all images were taken by the same 3D stereo camera. The size of the validation set was adjusted by the loss function of the validation set. Initially, the training, validation, and testing sets were divided into a ratio of 6:3:1, that was, 5054 images for the training set, 2700 images for the validation set, and 900 images for the test set. The epoch was set to 80 with 500 steps in each epoch. During the experiment, since the Mask-RCNN could only use three-channel RGB images for the predicted images, the channels of the test set of 900 RBGA images were modified to RGB after the error was found. The following figures showed the loss diagram of each partial function for this study (Fig. 12.11a–f).

In this research, comparison experiments were conducted on the same datasets at different learning rates. From the training results, when the learning rate was set to 0.001, the training loss dropped to 0.3099 and the validation set loss dropped to 0.4637. Additionally, the Mask R-CNN head bounding box loss dropped to 0.0434 in the training set and the validation loss dropped to 0.0601 and the Mask R-CNN head class loss dropped to 0.0656 in the training set and the validation loss dropped to 0.1119; the Mask R-CNN mask loss dropped to 0.1260 in the training set and the validation loss dropped to 0.1310; the RPN bounding box loss dropped to 0.0677 in the training set and the validation loss was 0.1077; the RPN class loss in the training set was 0.0071 and the validation loss was 0.0432 (Fig. 12.11a–f).

Figure 12.11a indicates the overall loss; by 80 epochs, each epoch was trained with 500 steps, which indicates that the model was good for this training. The Mask R-CNN bounding box loss denoted the loss of Mask R-CNN bounding box refinement, Mask R-CNN class loss denoted the head layer loss of classifier of Mask R-CNN, Mask R-CNN mask loss denoted the head layer mask binary cross-entropy loss of Mask, the RPN bounding box loss denoted the RPN bounding box loss, and the RPN class loss denoted anchor classifier loss.

Classification loss indicated how close the training model was to predicting the correct class. Mask R-CNN class loss was used as the head layer, and all objects were covered, while RPN class loss only covered the foreground and background of images. The border loss, on the other hand, responded to the distance between the real boxes and the predicted boxes. The Mask R-CNN mask loss responds to how close the model was to the predicted correct class mask. The sum of the above five losses constituted the overall loss.
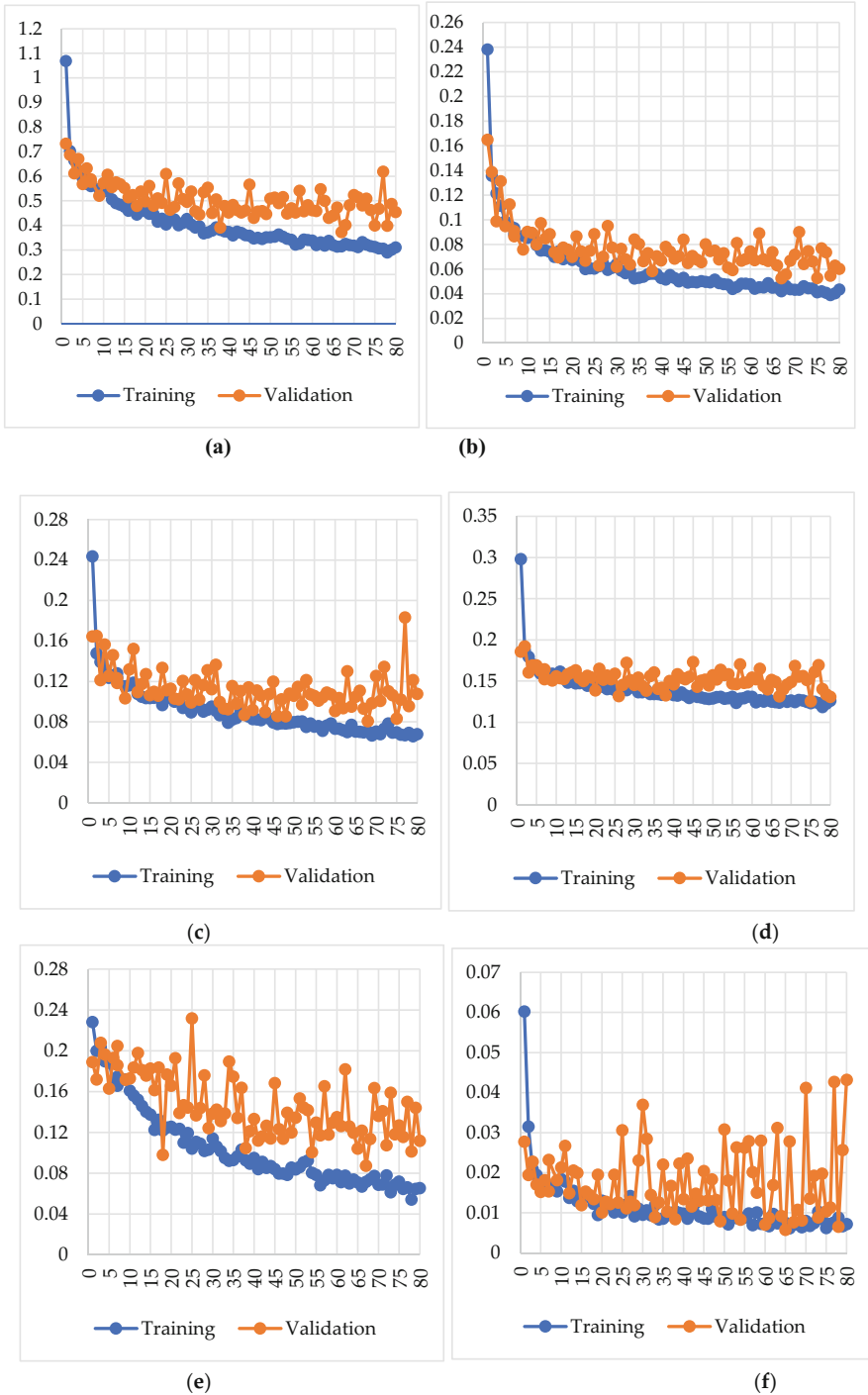
Fig. 12.11 Mask R-CNN loss results from training losses and validation losses, (**a**) Total loss; (**b**) Mask R-CNN head bounding box loss; (**c**) Mask R-CNN head class loss; (**d**) Mask R-CNN mask loss; (**e**) RPN bounding box loss; and (**f**) RPN class loss

12.3.2 Evaluation of Model Metrics

A series of weight files were obtained from the Mask R-CNN training and were used to evaluate the Mask R-CNN training model. The weight files left from the last training in the training process were selected to evaluate the test set.

The Precision (P), Recall (R), Average Precision (AP), and mean Average Precision (mAP) were used as the main parameters to evaluate the model in this research. We tested the different performances of the test set using the weights obtained from the training set after 80 epochs at different learning rates and the response plots of the Precision-Recall (PR) curves at a learning rate of 0.001. We also tested the same operation on the validation set with the weight trained by training sets, with overall mAP (IoU = 0.5). In addition, we tested different parts of the validation set, which was divided into three sections. One was original image datasets in the validation set, and another two were datasets after doing augmentation. We found that the results were nearly similar using the same weight tested in three sets: the original images (89.76%), rotation augmentation (84.47%), and flipped augmentation images (89.67%). However, while using all the datasets, including originals and two other augmented imageries in the validation process, the accuracy was increased (92.02%).

Table 12.3 shows the comparison of mAPs in Mask R-CNN and Faster R-CNN. The Precision-recall curve of Faster R-CNN and Mask R-CNN from the testing set at different learning rates after 80 epochs (Fig. 12.12).

12.3.3 Evaluation of Model Effectiveness

In this study, by creating a dataset and Mask R-CNN model using a 3D stereo camera, we found the best weights by comparing the fit of Mask R-CNN, Faster R-CNN and YOLACT with the same learning rate at lr = 0.001. By testing 900 images of the test set taken at different times, we obtained the following results by comparing the different effects of aggregating pears and separating pears under different illumination. Due to the problem of the light, branch, and leaf shading in the orchard, this research compared the test results of Mask R-CNN, Faster R-CNN, and YOLACT under the light intensity from multiple pears in gathering and individual situations (Figs. 12.13 and 12.14).

Table 12.3 mAP (IoU = 50%) results from 3D camera datasets using Mask R-CNN, Faster R-CNN, and YOLACT in the testing set and validation set

Model	Validation set (%)	Testing set (%)
Faster R-CNN	87.90	87.52
YOLACT	87.07	97.89
Mask R-CNN	95.22	99.45
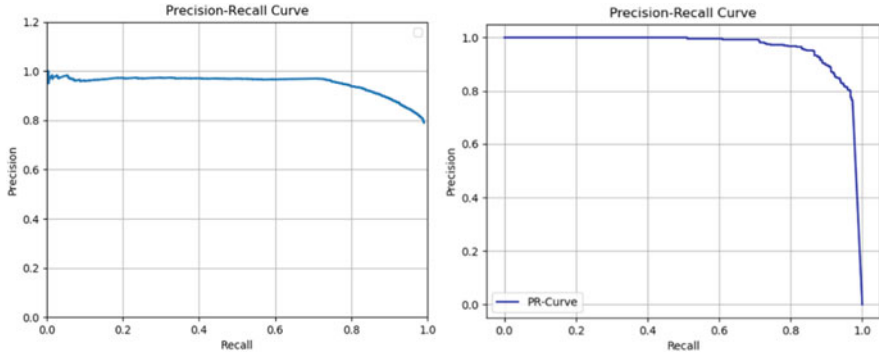
Fig. 12.12 (**a**) Precision-recall curve of Faster R-CNN at learning rate = 0.001 in the testing set and (**b**) precision-recall curve of Mask R-CNN at learning rate = 0.001 in the testing set

The results showed that in the case of independent pear detection, the difference in Mask R-CNN, Faster R-CNN, and YOLACT was that Mask R-CNN and YOLACT generated both masks and bounding boxes, while Faster R-CNN detected pear-only generated bounding boxes. For YOLACT, the detection in independent pear showed the similar results while the performance of masks that generated by YOLACT were lower than Mask R-CNN. The detection accuracy of Mask R-CNN was significantly higher than that of Faster R-CNN and YOLACT under dark light conditions, and the accuracy of the bounding box of Mask R-CNN was higher than that of Faster R-CNN when pears were aggregated, or when the detection target was incomplete pears. And the mask generation of Mask R-CNN was also higher than that of the same instance segmentation algorithm of YOLACT. Under bright light conditions, there was only a slight difference between Mask R-CNN and Faster R-CNN for the accuracy of independent pears. However, in the case of aggregated pears, Mask R-CNN had a higher correct recognition rate than Faster R-CNN. However, the accuracy of YOLACT under strong light conditions, whether in the case of aggregated pears or independent pears, was considerably lower than that of Mask R-CNN.

We also tested the comparison of the recognition of pears in different situations for both after image rotation. When the pears were separated, the accuracy of the two only shows a small difference in the size of the borders. However, when the pears were aggregated, Faster R-CNN failed to recognize individual pears; Mask R-CNN had a higher recognition rate than Faster R-CNN in this case. Since the YOLACT algorithm generated Anchors based on the color of background and pears, YOLACT showed inaccurate recognition of pears in the recognition of rotated images. However, for the case of images with one pear present, the performance was not much different in Mask R-CNN, Faster R-CNN, and YOLACT.
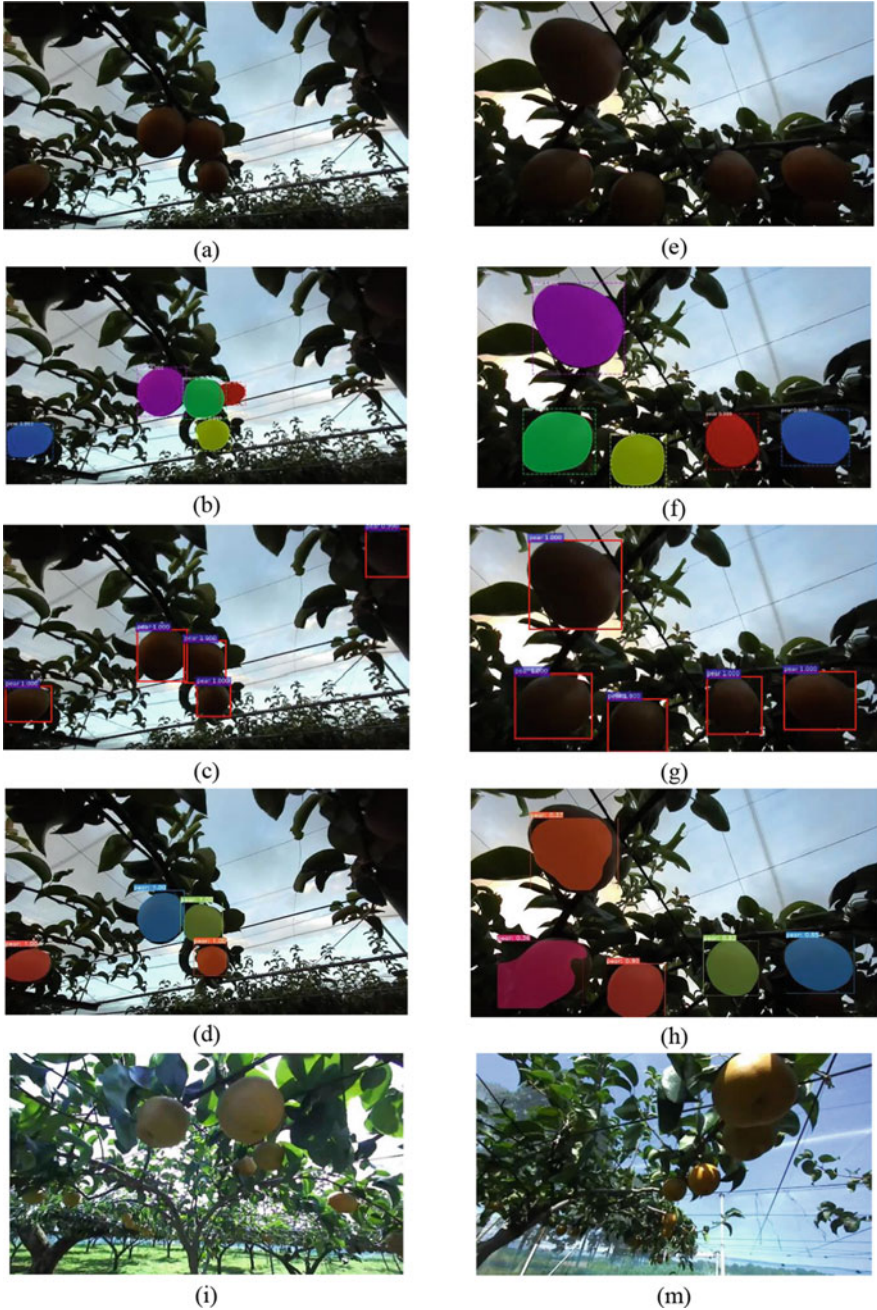
Fig. 12.13 Results of different deep learning algorithms in different situations. Recognition of (**a**–**d**) separated pears in low light; (**e**–**m**) aggregated pears in low light; (**i**–**l**) separated pears in strong light, and (**m**–**p**) aggregated pears in strong light. (**a**, **e**, **i**, **m**) Original images; (**b**, **f**, **j**, **n**) testing images in Mask R-CNN; and (**c**, **j**, **k**, **o**) testing images in Faster R-CNN; (**d**, **h**, **l**, **p**) testing images in YOLACT

Fig. 12.13 (continued)

12.4 Discussion

Machine vision technology has become very popular due to the robust identification of objects and classification in industrial and agricultural technological applications. This approach can be implemented to future fruit picking robots or mechanical devices as machines replace human labor. Thus, the cost of population labor will be significantly reduced; in terms of detection, with the development of vision technology, the detection accuracy will also improve. In our study, we compared this technology with other deep learning models. The average precision of the YOLOv4 model for pear recognition in complex orchards was 93.76% and that of YOLOv4-tiny for pear recognition was 94.09% (Parico & Ahamed, 2021). Moreover, by comparing the datasets using Faster R-CNN between apples, mangoes, and oranges, the mAP of apples was 86.87%, mangoes was 89.36%, and oranges was 87.39 (Wan & Goudos, 2020). The mAPs of this research were decreased when the camera field of view included fruit covered with leaves for the recognition of fruit with the Faster R-CNN algorithm. Even by labeling the fruit with different shading objects as different classes, mAPs for different classes did not significantly improve.
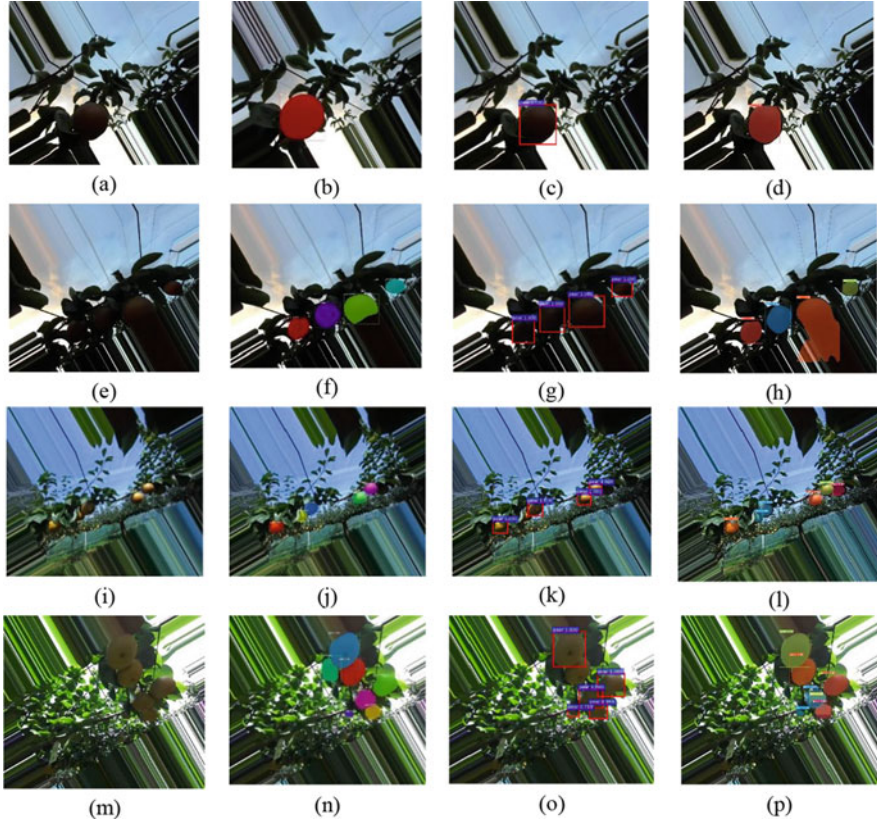
Fig. 12.14 Results of different deep learning algorithms. Recognition of (**a**–**d**) separated pear in low light; (**e**–**h**) aggregated pears in low light; (**i**–**l**) separated pears in strong light, and (**m**–**p**) aggregated pears in strong light (**a**, **e**, **i**, **m**) original images; (**b**, **f**, **j**, **n**) testing images in Mask R-CNN; and (**c**, **g**, **k**, **o**) testing images in Faster R-CNN; (**d**, **h**, **l**, **p**) testing images in YOLACT

The mAPs of leaf-occluded fruit, branch-occluded fruit, non-occluded fruit, and fruit-occluded fruit were 89.90%, 85.80%, 90.90%, and 84.80%, respectively (Gao et al., 2020). By comparing the same dataset, which was taken by the same stereo camera, we knew that the mAPs of Faster R-CNN reached 87.52% on the testing set and 87.90% on the validation set; the mAPs of YOLACT reached 97.89% on the testing set and 87.07% on validation set. We also found that the performance of Faster-RCNN became less accurate when testing aggregated pears compared with Mask R-CNN. The performance of YOLACT also became less accurate when the strong light condition whether in the case of individual pears or aggregated pears in orchard.

Therefore, although Faster R-CNN achieved good results in detection, distinguishing types of objects and individuals, the mAPs of Faster R-CNN, in the case of identifying one type of fruit, hardly improved dramatically, even by data

augmentation. And although YOLACT was the same instance segmentation algorithm as Mask R-CNN, was less accurate than Mask R-CNN in various situations (strong and low light). Due to the complex orchard environment, for example, the color of pears changed when the light intensity was different; it was difficult to detect individual pears when pears and leaves overlapped due to many branches; or when multiple pears were gathered. The accuracy of YOLACT was greatly reduced under the condition that the color distinction was not obvious. And Faster R-CNN had difficulty in improving the detection accuracy to a higher level and was prone to inaccurate detection results when detecting aggregated pears, Therefore, Mask R-CNN solves this problem perfectly, Mask R-CNN used two-stage instance segmentation technology and had advanced improvements in detecting aggregated pears and their accuracy. In this research, by using Mask R-CNN for the 3D stereo dataset, mAP (lr = 0.001) was 99.45% in the testing set and mAP(lr = 0.001) was 95.22%, which was much higher than that of Faster R-CNN. Due to this series of factors, Mask R-CNN has made great progress in detection. The mask generated by the Mask R-CNN network distinguished individual pears well when there were pear clusters in the environment.

We chose to use a 3D stereo camera for data acquisition. Traditional monocular cameras obtained better resolution for images; however, they had limitations in extended measurement. Although the principle of the monocular camera measurement method was simple, its accuracy was much lower than that of the 3D stereo camera. The ZED camera was different from ordinary traditional cameras; it obtained depth information by generating point clouds to calculate the actual distance between the camera and the pear. Using the ZED camera with Mask R-CNN combined instance segmentation and depth ranging to identify each of individual pears in complex orchards and calculate their actual distance.

We compared Faster R-CNN, who belonged to two stage (detection + segmentation) algorithm with Mask R-CNN. Since Faster R-CNN did not generate masks in prediction, it had an advantage in recognition speed as a traditional visual recognition model, while Mask R-CNN had an advantage in accuracy. We also compared YOLACT with Mask R-CNN, which was different from Faster R-CNN, it belonged to one-stage (detection) algorithm, so it was much faster than the previous two methods in terms of processing speed, and it was an instance segmentation algorithm, so it also generated more accurate masks in prediction, but since YOLACT was the real-time processing, it is much better in terms of accuracy.

However, since it is real-time processing, it was much less accurate than Mask R-CNN. Furthermore, Mask R-CNN provided an increase in accuracy based on by adding a new mask branch. However, the disadvantage was also obvious: its detection speed was only 5 fps and there was a delay with the agricultural picking robots running at high speed. However, in slow picking, its effect was obviously remarkable at the development stages with high recognition accuracy. Since the ZED camera's PERFORMANCE format was used, the resolution of the pictures it took was reduced compared to that of a regular camera. Therefore, while detecting, the dark leaves of shadows were detected as dark pears due to the influence of light. In this respect, the dataset accuracy needs to be improved.

However, in this study, the distance measuring module was not added, and only the ZED camera was used to make RGBA datasets. The ZED camera measured the distance through the depth point cloud. In future research, adding the comparison of the ranging module can further contribute to the development of agricultural robots for complex orchard work such as robotic picking machines using an end-effector or manipulator.

12.5   Conclusion

In this research, we developed a method for agricultural picking robots by using Mask R-CNN in an orchard from 3D stereo camera datasets. For the recognition of pears, the Mask R-CNN had the mAPs of 95.22% for validation set and 99.45% was observed for the testing set. On the other hand, mAPs were observed 87.9% in the validation set and 87.52% in the testing set using Faster R-CNN. And mAPs were observed 87.07% in the validation set and 97.89% in the testing set. Furthermore, the 3D stereo camera was used with the Mask R-CNN dataset of pears in complex orchards for two different types of images: original RGBA and depth. Unlike the images taken by a conventional camera, the images taken by the 3D stereo camera could be used in the development of adding a distance measurement module and training a set of depth maps as a dataset in future work. Since we obtained the mask information of the identified pear, instead of obtaining the coordinates of the bounding boxes, so that the center point of the pears was calculated more accurately by the ZED camera, the accuracy of the coordinates of the center point obtained by masks was higher than the method of the center point obtained by the bounding boxes. This was because the positioning of the bounding boxes under different conditions showed some deviations. Therefore, we obtained high recognition accuracy on videos captured with the ZED camera, which was a 3D stereo camera, and organized them into applicable datasets using an advanced target detection model based on the Mask R-CNN with TensorFlow, including the Keras library. We used the ResNet101 and FPN as the backbone layer; And FCN was chosen as the head layer in Mask R-CNN to achieve higher accuracy in detection. The mAP of the dataset taken by the ZED stereo camera with an image size of 1280×720 using the Mask R-CNN at a learning rate of 0.001 reached 0.3099 using the training weight, and the mAP of the validation set reached 0.4637 at a learning rate of 0.001. We also verified the test speed, although we found that YOLACT tested at 0.1132 s per image taken by ZED camera and 1.229 s on Mask R-CNN, the average picking speed of some current picking robots was still around 6.000 s. Therefore the detection speed of Mask R-CNN was seen as exploitable on low-speed fruit picking mechanisms, but was deficient on picking robots operating at high speeds. Therefore, we provided another method to improve efficiency in agricultural production, thus solving the difficulty of previous vision models to recognize fruits because of their overlapping. Further research will be conducted to implement this vision recognition method for

the development of agricultural robots in picking fruits using an end-effector or a manipulator.

Conflicts of Interest   The authors declare no conflicts of interest.

References

Barua, S. (2020). *Understanding coronanomics: The economic implications of the coronavirus (COVID-19)* pandemic. Retrieved April 1, 2020, from https://ssrn.com/abstract=3566477

Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations: Concepts and components. *Biosystems Engineering, 149*, 94–111.

Bodla, N., Singh, B., Chellappa, R., & Davis, L. S. (2017). Soft-NMS—Improving object detection with one line of code. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 5561–5569).

Bolya, D., Zhou, C., Xiao, F., & Lee, Y. J. (2019). Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9157–9166).

Boniecki, P., & Piekarska-Boniecka, H. (2004). The SOFM type neural networks in the process of identification of selected orchard pests. *Journal of Research and Applications in Agricultural Engineering, 49*(4), 5–9.

Boniecki, P., Koszela, K., & Przybylak, A. (2010). Classification of selected apples varieties and dried carrots using neural network type Kohonen. *Journal of Research and Applications in Agricultural Engineering, 55*(1), 11–15.

Cai, Z., & Vasconcelos, N. (2019). Cascade R-CNN: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 43*, 1483–1498.

Dorrer, M. G., & Tolmacheva, A. E. (2020). Comparison of the YOLOv3 and mask R-CNN architectures' efficiency in the smart refrigerator's computer vision. *Journal of Physics: Conference Series, 1679*(4), 042022.

Ertam, F., & Aydın, G. (2017). Data classification with deep learning using Tensorflow. In *2017 International Conference on Computer Science and Engineering (UBMK)* (pp. 755–758). IEEE.

Gao, F., Fu, L., Zhang, X., Majeed, Y., Li, R., Karkee, M., & Zhang, Q. (2020). Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN. *Computers and Electronics in Agriculture, 176*, 105634.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1440–1448).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2015). Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 38*(1), 142–158.

Hannan, M. W., & Burks, T. F. (2004). Current developments in automated citrus harvesting. In *2004 ASAE Annual Meeting* (p. 1). American Society of Agricultural and Biological Engineers.

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

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2961–2969).

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S. & Murphy, K. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 7310–7311).

Jia, W., Tian, Y., Luo, R., Zhang, Z., Lian, J., & Zheng, Y. (2020). Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot. *Computers and Electronics in Agriculture, 172*, 105380.

Jiang, A., Noguchi, R., & Ahamed, T. (2022). Tree trunk recognition in orchard autonomous operations under different light conditions using a thermal camera and faster R-CNN. *Sensors, 22*(5), 2065.

Kirkland, E. J. (2010). Bilinear interpolation. In *Advanced computing in electron microscopy* (pp. 261–263). Springer.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems, 25*(2).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2980–2988).

Ortiz, L. E., Cabrera, E. V., & Gonçalves, L. M. (2018). Depth data error modeling of the ZED 3D vision sensor from stereolabs. *ELCVIA: Electronic Letters on Computer Vision and Image Analysis, 17*(1), 1–15.

Parico, A. I. B., & Ahamed, T. (2021). Real time pear fruit detection and counting using YOLOv4 models and deep SORT. *Sensors, 21*(14), 4803.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 779–788).

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems, 28*, 91–99.

Russell, B. C., Torralba, A., Murphy, K. P., & Freeman, W. T. (2008). LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision, 77*(1), 157–173.

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.

Saito, T. (2016). Advances in Japanese pear breeding in Japan. *Breeding Science, 66*(1), 46–59.

Schrder, C. (2014). *Employment in European Agriculture: Labour costs, flexibility and contractual aspects.* Retrieved from agricultura.gencat.cat/web/.content/de_departament/ de02_estadistiques_observatoris/27_butlletins/02_butlletins_nd/documents_nd/ fitxers_estatics_nd/2017/0193_2017_Ocupacio_Agraria-UE-2014.pdf

Sobol, Z., Jakubowski, T., & Nawara, P. (2020). Application of the CIE L* a* b* method for the evaluation of the color of fried products from potato tubers exposed to C band ultraviolet light. *Sustainability, 12*(8), 3487.

Tran, T. M., Ta, K. D., Hoang, M., Nguyen, T. V., Nguyen, N. D., & Pham, G. N. (2020). A study on determination of simple objects volume using ZED stereo camera based on 3D-points and segmentation images. *International Journal, 8*(5), 1990.

Wan, S., & Goudos, S. (2020). Faster R-CNN for multi-class fruit detection using a robotic vision system. *Computer Networks, 168*, 107036.

Wei, X., Jia, K., Lan, J., Li, Y., Zeng, Y., & Wang, C. (2014). Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot. *Optik, 125*(19), 5684–5689.

Yu, Y., Zhang, K., Yang, L., & Zhang, D. (2019). Fruit detection for strawberry harvesting robot in non-structural environment based on mask-RCNN. *Computers and Electronics in Agriculture, 163*, 104846.

Chapter 13
Thermal Imaging and Deep Learning Object Detection Algorithms for Early Embryo Detection: A Methodology Development Addressed to Quail Precision Hatching

Victor Massaki Nakaguchi and Tofael Ahamed

Abstract Poultry production utilized many available technologies in terms of farm-industry automation and sanitary control. However, there is a lack of robust techniques and affordable equipment for avian embryo detection and sexual segregation at the early stages. In this work, we aimed to evaluate the potential use of thermal microcameras for detecting embryos in quail eggs via thermal images during the first 168 h (7 days) of incubation. We propose a methodology to collect data during incubation period. Additionally, to support the visual analysis, YOLO deep learning object detection algorithms were applied to detect unfertilized eggs; the results showed its potential to distinguish fertilized eggs from unfertilized eggs during the incubation period after filtering radiometric images. We compared YOLOv4, YOLOv5, and SSD-MobileNet V2 trained models. The mAP@0.50 of the YOLOv4, YOLOv5, and SSD-MobileNet V2 was 98.62%, 99.5%, and 91.8%, respectively. We also compared three testing datasets for different intervals of rotation of eggs, as our hypothesis was that fewer turning periods could improve the visualization of fertilized eggs features, and applied three treatments: 1.5, 6, and 12 h. The results showed that turning eggs in different periods did not exhibit a linear relation, as the F1 Score for YOLOv4 of detection for the 12 h period was 0.569, that for the 6 h period was 0.404 and that for the 1.5 h period was 0.384. YOLOv5 F1 Scores for 12, 6, and 1.5 h was 1, 0.545, and 0.386, respectively. SSD-MobileNet V2 performed F1 score of 0.60 for 12 h, 0.22 for 6 h, and 0 for 1.5 h turning periods.

Keywords Quail eggs · Thermal imaging · Precision livestock farming · Embryo detection · YOLO · Deep learning

V. M. Nakaguchi
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

13.1 Introduction

Precision approaches to improve yield are currently based on sophisticated support decision systems that include several types of sensors, powerful processing units for multidimensional data analysis, and machinery automation. For animal production, certain authors considered that combining precision yield techniques with intensive farming systems constitutes the best way to enhance productivity and sustainability (Wathes et al., 2008; Lovarelli et al., 2020). Therefore, the best practices to save natural resources include the mitigation of losses and the systematic use of energy inside the bioproduction ecosystem.

On breeding farms, the critical process of genetic improvement, incubation, and hatching is processed; this type of farm is responsible for providing specialized strains to other farms (meat and egg production) and even for supplying the pharmaceutical industry during vaccine research and manufacturing (Yu et al., 2019). Historically, the insufficiency of efficient, affordable, and robust technologies has driven these farms toward low-precision hatching and culling of undesirable strains. In conventional hatching, there is a lack of rapid, high confidence methods to identify unfertilized eggs and dead embryos in early stages. Due to this low-precision hatching, hidden costs are associated with a waste of energy and physical space during incubation periods (Dong et al., 2019). For culling undesirable genes, there is also a deficiency of real-time methods and industrial equipment for sex segregation at the embryo level.

The most common technique for embryo detection is still candling, which consists of using a light source against the eggshell to view the content inside the eggs; nevertheless, it represents an inefficient, labor-consuming, and subjective technique (Liu & Ngadi, 2013). Moreover, candling is not applied to quail eggs because their small size and different eggshell colors render its utilization difficult. Several nondestructive methods to assess chicken egg content in early stages have been reported, including visible light transmission change detection (Bamelis et al., 2002), acoustic resonance (Coucke et al., 1997), near-infrared (NIR) hyperspectral imaging (Zhang et al., 2014), spectroscopy methods using visible (VIS)/NIR (Qin et al., 2017; Khaliduzzaman et al., 2021), optical sensing using PhotoPlethysmoGraphy (PPG), and deep learning classification (Yu et al., 2019).

Quail farms are well distributed worldwide as chicken farms, although the largest producers of eggs are concentrated in East Asia and Brazil (Bertechini, 2012). The consumption of this product represents approximately 10% of all eggs that are globally consumed (Lukanov, 2019). Furthermore, the production of quail for meat and eggs has been increasing as the global demand for food continues to grow. In developing countries, quail poultry represents a viable alternative to supply animal protein, especially because of the reduced size of birds, its high nutritional value and its resilient avian for raising in "backyard" systems (Shanawany, 1994). However, as quail farms have not yet been industrially established in many countries, breeding programs are not easily identified; therefore, farmers are responsible for breeding their own flocks, which is an arduous task that can lead to genetic

depression of quail flocks caused by consanguinity, resulting in low fertility, low productivity of eggs, and a high rate of mortality (Martins, 2002). Another recurrent problem associated with quail egg production is the low hatchability rate; on average, 40% of all incubated eggs do not hatch. Several factors can contribute to this problem, the most common being long-term storage under bad climate conditions, dead embryos, or unfertilized eggs (Shanawany, 1994).

Recently, reliable methods using noncontact and nondestructive analysis in real time represent the best opportunity for embryo detection in the poultry industry. The advent of big data, powerful processing units, and more efficient algorithms are considered to be responsible for bringing computer vision (CV) methods using the deep neural network (DNN) approach to fruition. There are several types of heuristic algorithms based on neural networks (NNs), such as convolutional neural networks (CNNs). The ability of CNNs to address complex nonlinear problems such as image recognition and classification is responsible for presenting machines with vision sense and mimicking humans' capacity to solve problems. However, one of the greatest challenges for these algorithms is regarding speed and accuracy; sturdy models are those able to generalize predictions of any new data with a high confidence level similar to humans.

The You Only Look Once (YOLO) object detection algorithm is currently one of the fastest and most accurate models for image classification. The YOLO object detection algorithm was released by Redmon and Farhadi (2016). The breakthrough of this algorithm was to use a single CNN to predict classes and bounding box coordinates as a regression problem. This algorithm can also be referred to as a Single Shot Detector in the class of one-stage detectors. Once an image is viewed by dividing it into a grid with a size of $S \times S$, the algorithm predicts the class and bounding boxes for each grid cell. Two-stage object detector models, such as the R-CNN (Girshick et al., 2014) series (including Fast R-CNN (Girshick, 2015) and Faster R-CNN (Ren et al., 2015)), use the region proposal technique to divide the image into regions and then classify each region according to the proposal boxes. Although this technique effectively requires too much time for training, it is not possible for real-time detection.

The main component of CV systems is the camera. Optical cameras capture light wavelengths in the visible light range of the electromagnetic spectrum; nevertheless, limitations arise due to light reflectance dependency. Thermal cameras can overcome light dependency once they capture radiometric information transmitted through the air by measuring the temperature of an object surface and by solving their intensity of infrared spectral wavelengths that reach the camera. The high-cost equipment, low resolution, reduced field of view, and low frame recording represent the main limitations of thermal cameras (Williams et al., 2022). However, the recently increasing demand associated with many applications, including healthy monitoring, is enabling thermal cameras to become popular sensors, which could make them more accessible in the near future.

The aim of this work was to investigate the potential of thermal microcameras for fast visual, early embryo detection in quail eggs as a supportive method to improve the hatching rate and contribute to the further development of automatic incubator

systems that are able to segregate fertilized and unfertilized eggs. Wild birds rotate their eggs several times a day during natural incubation to improve hatchability (Tullett & Deeming, 1987), and the rotation of eggs is essential to ensure normal embryo development in many avian species (Yoshizaki & Saito, 2002; Wilson, 1990; Oliveira et al., 2020). The hypothesis was that longer intervals between turning eggs could make the identification of unfertilized eggs easier using thermal-based visual detection- systems. Since the less rotations could keep the developing embryo static and the temperature associated to it less distributed among the inside content, therefore facilitating the interpretation by thermal cameras.

13.2 Materials and Methods

13.2.1 Thermal Imaging

As a noncontact nondestructive method, thermal imaging can be defined as the sum of the radiance emitted from a material, the environment (other material radiance), and atmospheric transmission (Eq. 13.1). Most of the challenges associated with thermal imaging are to interpret absolute temperatures because thermal radiometric cameras only reads information from opaque materials; moreover, thermal imaging may be affected by the radiometric properties of objects and the medium: transmittance (τ), emissivity (ε) and reflectance (Fig. 13.1), including the body itself and surroundings (Maldague, 2001; Lahiri et al., 2015). Another factor that influences
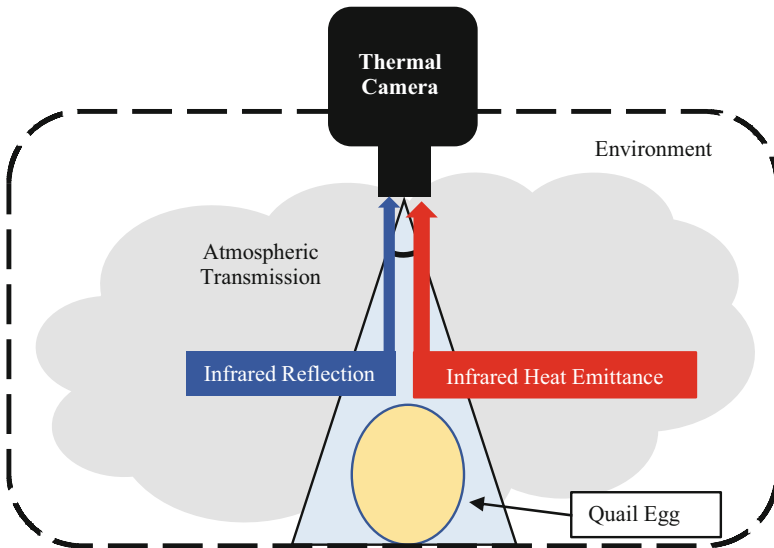
Fig. 13.1 Thermal camera radiance relationship among material, environment, and the atmosphere

the interpretation of radiometric images is the size of the target; small surfaces may make the measurements difficult because the number of pixels describing the surface is diminished.

$$\mathrm{Rcam} = \tau \varepsilon \mathrm{Rmat} + \tau (1 - \varepsilon)\mathrm{Renv} + (1 - \varepsilon)\mathrm{Ratm} \qquad (13.1)$$

Rcam is the radiance read by the camera, Rmat denotes the radiance emitted by the material of interest or body, and Ratm is the radiance from the atmosphere.

13.2.1.1 Transmittance (τ)

Transmittance is the ratio of a radiant flux transmitted (Φ_t) to an incident flux (Φ_i) (Höpe, 2014) in function of the emission wavelength (λ). Atmospheric transmittance represents one of the greatest issues in thermal imaging analyses because it can change the radiometric temperature measurement, thus interfering with the active heat read by the camera (Eq. 13.2) Inside the incubator machines, humidity and temperature influence the transmission of radiance from the eggs, low humidity reduces the transmission of radiance, and the best strategy for enhancing the transmission in this case is to reduce the distance from the target.

$$\tau(\lambda) = \frac{\Phi_t(\lambda)}{\Phi_i(\lambda)} \qquad (13.2)$$

13.2.1.2 Emissivity (ε)

Emissivity is the effectiveness of a material to emit thermal energy compared with a perfect absorbing energy body, a blackbody, at the same temperature. The real values are measured on a scale from 0 to 1. The emissivity depends on material characteristics such as format, temperature, roughness, spectral wavelength, oxidation, and view angle (Höpe, 2014), and the emissivity of the *Galloanserae* species eggs (chicken, quail, turkey, duck, and swan) is higher, approximately 0.98–1 (Björn & Nilsson, 2018). The emissivity (Eq. 13.3) can be defined as the absorption light of a blackbody minus the reflectance of the object (R_e).

$$\varepsilon_0(\lambda) = 1 - R_e(\lambda) \qquad (13.3)$$

13.2.1.3 Reflectance (ρ)

Reflectance (Eq. 13.4) is the relation of radiant lux in watts between the reflected signal (Φ_r) and the incident signal (Φ_i) (Höpe, 2014). The reflectance can directly affect the interpretation of thermal values and is directly related to the angle of view of the target from the camera. Short distances from the target increase the reflectance captured by the camera; on the other hand, long distances may not be enough to obtain information from small objects.

$$\rho(\lambda) = \frac{\Phi_r(\lambda)}{\Phi_i(\lambda)} \qquad (13.4)$$

13.2.2 Experimental Environment

An experiment was conducted in the Department of Life and Environmental Sciences of the University of Tsukuba, Tsukuba, Ibaraki (36°11′19.8″N, 140°10′20.4″E) in the laboratory of Bioproduction and Machinery during the middle of the spring season, with which the average range daily temperature was 11–18 °C.

Japanese quail (*Coturnix japonica*) eggs were collected from a quail farm located in the city of Toyohashi, province of Aichi in Japan. The eggs were aleatory collected and shipped by mail on the same day and transported at room temperature in an appropriate package to avoid dehydration and impacts. After they arrived, the eggs were put in an airy place for approximately 5 h, and no kind of treatment, either washing or wiping, was applied. By performing the procedure, we repeated the same process performed in breeding farms. Next, we put the eggs in numerical order, marking them from 1 to 30, and the opposite face was marked from 1′ to 30′ (for each treatment of 30 eggs). The eggs were then placed directly inside the automatic incubator machine.

The equipment chosen to incubate the eggs was a fully automatic incubator machine (no brand) with 110v and automatic control of temperature, humidity, and rotation. The temperature was set to 37.8 °C with a low variation of +/−0.3 °C, and the humidity was maintained at 60%, varying by +/−10%.

We divided our experiment into two phases. In the first phase, a total of 60 quail eggs were incubated twice, and 30 eggs were incubated at a time. In this step, the objective was to define the methodology to collect data using a thermal microcamera, such as the position of eggs and camera, including the best interval time to collect data. Phase two consisted of the experiment. Here, we incubated 120 eggs in total, divided into 4 groups of 30 eggs each. For three groups, the eggs were rotated in different periods: every 90 min, 6, and 12 h. The fourth groups were composed only of unfertilized eggs bought from grocery stores. We incubated 30 of

these eggs to perform an accuracy assessment of unfertile eggs and applied these data to train the deep learning models.

13.2.3 Thermal Image Acquisition and Radiometric Corrections

The thermal images were collected with a thermal microcamera FLIR® (Teledyne FLIR LLC, Wilsonville, Oregon, U.S.) Model VUE™ 336, 6.8 mm, with a sensor resolution of 336 × 256 pixels and a spectral band range 7.5–13.5 μm, size 2.26″ (5.74 cm) × 1.75″ (4.44 cm). This is a powerful camera especially designed to board unmanned aerial vehicles (UAV) and can be controlled by a smartphone app named FLIR® UAS™ 2, which is provided by the same manufacturer.

The camera was placed in a top-view position, with a distance of 10 cm from the targets (eggs), and the egg-by-egg images were collected inside the incubator, avoiding exposure of eggs to ambient room temperature for a long period (Fig. 13.2). The image resolution provided by the camera was 640 pixels in width by 487 in height. Data were manually collected every 12 h (at 9 AM/9 PM), and the thermal camera was controlled by a SHARP® smartphone (Sharp Corporation, Sakai, Osaka, Japan), AQUOS™ sense4 basic Model A003SH and Operational System ANDROID™ version 11 app connected by Bluetooth. This procedure was performed for 7 days for each group (treatment) of eggs. To collect the data, we separately moved the eggs to the left corner of the incubator, where only one egg at a time could fit in the frame image and moved the eggs by picking them up from the equator borders. Radiometric images were saved in a micro-Secure Digital (SD) Card on the camera in Joint Photographic Experts Group (JPG) format and
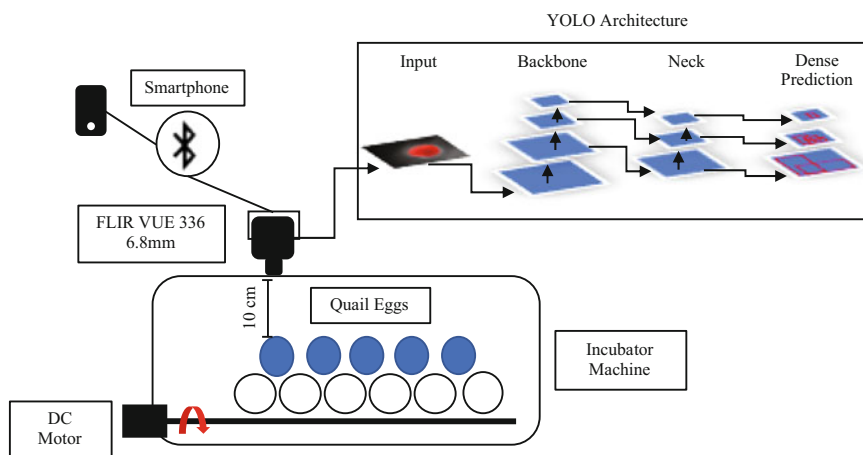
Fig. 13.2 Quail egg incubation process using a thermal camera and a deep learning structure

transferred to a personal computer (PC) for data analysis. On the 8th day, the eggs were broken to assess the embryos inside the eggs.

For rotation of eggs, we employed a DC 12v motor embedded in the incubator. The eggs were turned 180° every period (90 min, 6 and 12 h) for each cluster, and the unfertile eggs were turned every 90 min. The radiometric images were corrected with the software FLIR® Thermal Studio™; the images were filtered with Isotherms and classified for the above temperatures; and manual adjustments of contrast and red saturation were performed to highlight the visible features on the eggs. Isometric transformation is a radiometric pixel classification that highlights temperatures above, in the middle or below a threshold. For egg incubation analysis, we chose the isotherms above the threshold, making it possible to capture features from fertilized and unfertilized eggs.

13.2.4 Deep Learning Algorithms and Analysis Environment

YOLOv4 is embedded in the framework Darknet (neural network framework, open source written in C programming language and CUDA). This supervised learning-based algorithm uses a single CNN to extract features of images and to create a model based on a training dataset to predict objects with a certain level of accuracy and their positions on frames or pictures.

From its release in 2016, YOLO object detection family have been gradually expanded. The fourth generation of YOLO, also referred to as the 4th version or YOLOv4, released by Bochkovskiy et al. (2020), has been one of the fastest and most accurate object detection models (Bochkovskiy et al., 2020). YOLOv4 uses the Cross Stage Partial Darknet-53 (CSPDarknet-53) (Wang et al., 2020), which is a new backbone that is capable of enhancing CNN learning, the path aggregation network (PANet) and spatial pyramid pooling (SPP). These new additions were responsible for enhancing speed by 12% and accuracy by 10% compared with YOLOv3 (previous version of YOLO).

The breakthrough of YOLO was its ability to visualize an entire image at once, dividing it into a grid of $S \times S$ and then to create a map of probabilities for Region of Interest (ROI) by regression (Fig. 13.3). The ROI tells CNN which region has a high chance of finding the object in each frame. These characteristics was improved in YOLOv4, thus, enabling real-time object detection implementation with more accuracy. In addition, YOLOv4 was designed to be efficiently trained using only one graphics processing unit (GPU).

Recently, YOLO series have been evolving and several versions are currently available. Improvements are being done by companies, including YOLOv5 series (YOLOv5, 2021), developed by Ultralytics®. However, no peer reviewed article paper has being released along with them. Besides that, the community of developers engaged on YOLO family started to complain about the usage of YOLO's name by companies when launching new improved versions. YOLOv5, uses the same backbone as YOLOv4 (CSPDarknet-53), the difference is in the neck part, which is composed by feature pyramid network (FPN) (Liu et al., 2018) and pixel aggregation
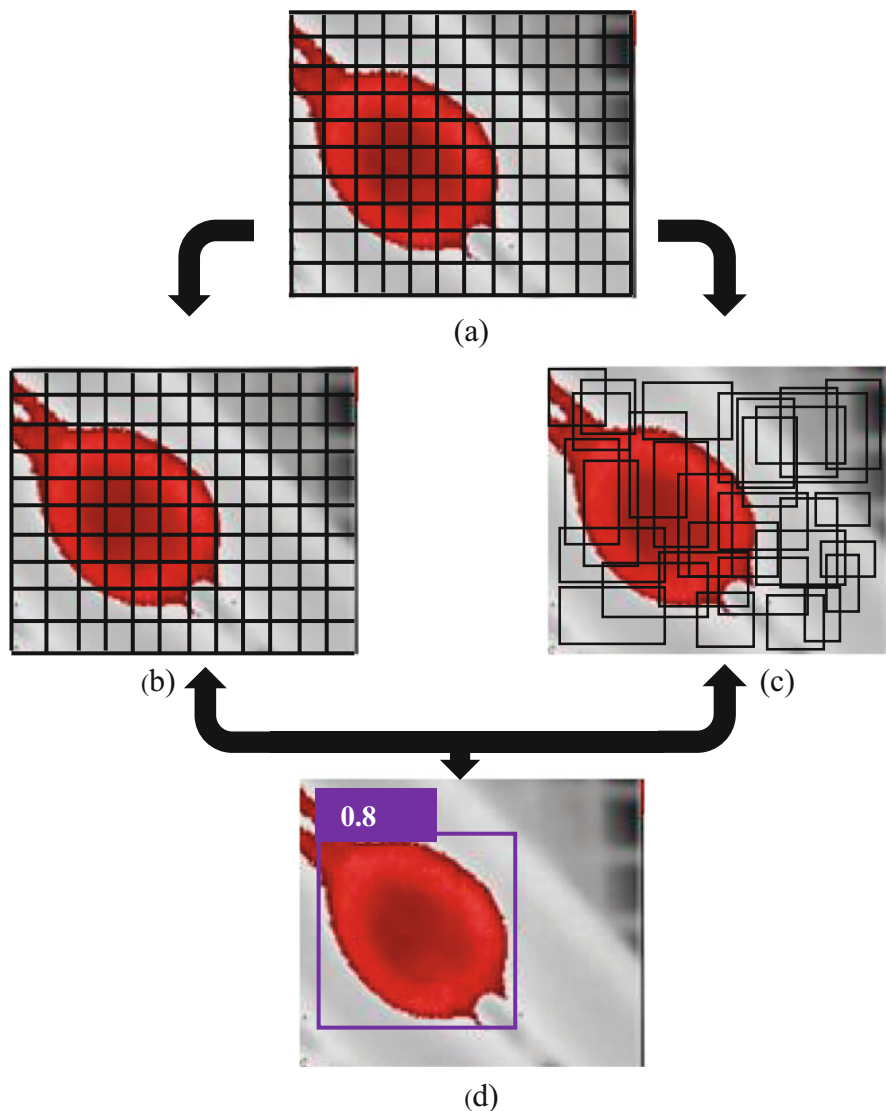
Fig. 13.3 (**a**) Input frame in the deep learning process $S \times S$ grid. (**b**) Class probability map from each grid cell. (**c**) Predicted bounding box and confidence. (**d**) Final object detection

network (PAN) (Lin et al., 2017), these additions are responsible for improvements in accuracy and faster training process. Another modification is that YOLOv5 is embed in PyTorch framework. YOLOv5 is qualified as faster and accurate, besides accomplish light files which makes it suitable for low-end deployment devices.

In this work, we trained an object detection model using YOLOv4 to validate our results for embryo detection as a supportive method to assess the visual observation

of thermal imaging features and to contribute to the future development of automatic classification equipment, in addition we trained a YOLOv5-L6 and an SSD-MobileNet V2 to compare the performance of improvements on the algorithm.

As the YOLOv4 and YOLOv5 share the same base architecture. We also trained an SSD-MobileNet V2 model to compare the performance of different architectures when detecting unfertilized eggs using the methodology proposed in this work to collect thermal images.

SSD-MobileNet V2 (Sandler et al., 2018) is another representant of one-stage detectors architecture that adopts the same single shot detection (SSD) mechanism similarly to YOLO. However, it gained popularity due to its faster performance at low compute devices such as mobile phones (therefore, MobileNet) and low-end computers such as development boards NVIDIA® Jetson series, Raspberry Pi, and Google Coral. The V2 version of this deep learning algorithm introduced depth-wise convolution layer, which reduce the number of parameters and contribute to improve the performance, in the V2 added the expansion-filtering-compression, known as inverted residual structure, which contribute for improvement of performance. In this work we trained a model SSD-MobileNet V2 to compare the performance of two SSD models, keeping in mind the potential application of our methodology on high throughput system.

13.2.4.1 Models Training

We trained all the models using only images of unfertilized eggs and collected a total of 420 images (30 images each period of 12 h). From these images, the same procedure as previously described (FLIR® Thermal Studio) was performed, and then image augmentation was performed to increase the dataset to make the model more predictive.

13.2.4.2 Data Labeling

A short program written in Python programming language was utilized to label the images by using the OpenCV library. We drew the bounding boxes (bbox) and saved them into the YOLO format coordinates (Fig. 13.4). As for SSD-MobileNet V2 we labeled the data using the LabelImg software in format PASCAL VOC XML.

13.2.4.3 Data Augmentation

The images from unfertilized eggs were augmented to enlarge the dataset and made the models more effective in detecting unseen eggs. We applied random spatial/pixel level transformations by rotating and changing the thermal effects of visualization. Thus, datasets were enhanced to 1892 images in total. Image augmentation for spatial transformation was performed in several ways: clockwise rotation 90°,
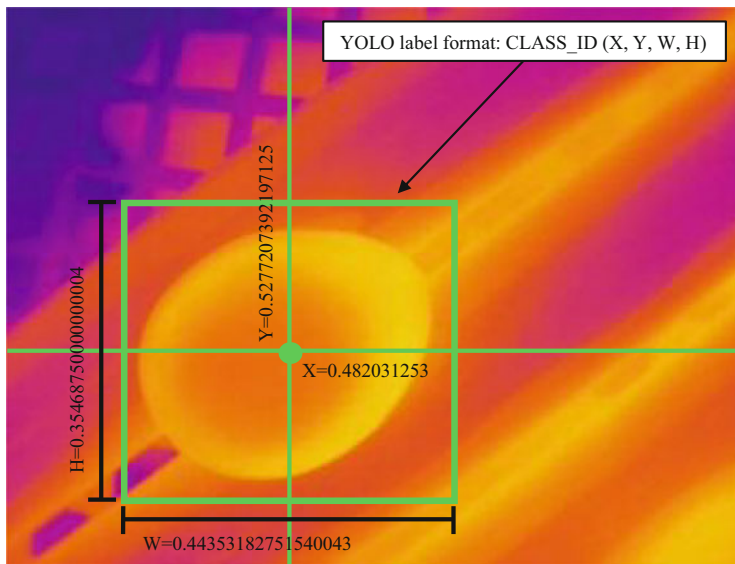
Fig. 13.4 YOLO labeling format coordinates and sizes in pixels
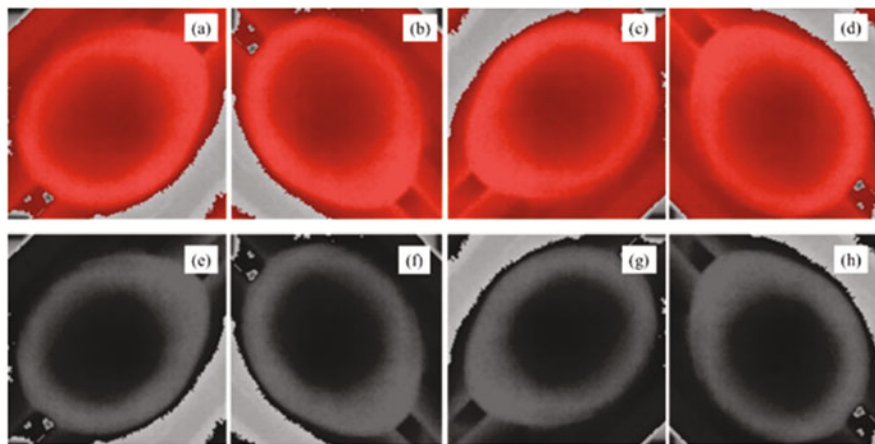
Fig. 13.5 Quail egg image enhancement process and augmentation for dataset preparation during incubation in isotherm processing of original images and monochrome transformation. (**a**) Original images. (**b**–**d**). Augmentation of original images using 90°, 180°, and 270° rotations. (**e**) Original position with pixel augmentation using monochrome transformation. (**f**–**h**) Augmentation of monochrome transformed images for 90°, 180°, and 270° rotations

clockwise rotation 180°, and clockwise rotation 270° (Fig. 13.5). In addition, pixel transformation was performed using grayscale conversion.

The images were divided into two datasets: training dataset and validation dataset. We adopted the proportion of 70:30. In total, 1325 images were selected

Table 13.1 YOLOv4 and YOLOv5 training parameters

	Batch	Input size	Momentum	Decay	Learning rate
YOLOv4	64	416×416	0.949	0.0005	0.0001
YOLOv5	16	416×416	0.937	0.0005	0.0001

for training and 567 images were selected for validation. To test the model, we used the data from the incubation treatment (420 images for each cluster separately tested) and then compared the precision of detection.

The network size for training the YOLOv4 model was set to 416×416, and the number of iterations was set to 4000 steps (Table 13.1). However, we stopped training when the average loss no longer decreased. The training was performed on a PC with 32 GB of RAM memory, an NVIDIA® GTX 1650™ 4 GB GPU and a central processing unit (CPU) Intel® Xeon™ E5-1607, Python version 3.8.5, CUDA 10.1, cuDNN 7.6.5, and OpenCV 4.4.0.

YOLOv5 model was trained in the Google Collab cloud platform. The framework version used was PyTorch 1.11.0 + cu102 and 16 GB GPU Tesla T4. The training parameters are shown in Table 13.1, the epochs of training (number of iterations) were set to 60, the source code was cloned from official Ultralytics® GitHub.

SSD-MobileNet V2 was trained using the same computational resources used for training YOLOv4. However, we trained the model using the TensorFlow API object detection framework, in an environment built with Tensorflow 2.3.1 and Tensorflow-gpu 2.3.1. The input size of this model was 320×320 and 40,000 steps for training.

13.2.4.4 Model Evaluation

Several metrics were applied to evaluate the deep learning models, including precision (P), recall (R), F1 score, and mean average precision (mAP). Object detection evaluations are based on four factors: true positive detections (TP): unfertilized eggs are correctly detected; true negatives (TN): unfertilized eggs are not shown and are not detected; false-positive (FP): object is detected, but it does not correspond to any class, and false negative (FN): model does not label an object but was supposed to perform this task (Fig. 13.6).

P (Eq. 13.5) and R (Eq. 13.6) are measurements that evaluate the relevance of detection. R returns the real relevance of the results, considering false negative detections, while P considers false-positive detections. The F1 score (Eq. 13.7) is a measurement that indicates the relation degree of P and R, as the higher the F1 score is, the higher the values of P and R, the more accurate the detections may be.

To categorize the truthiness of a class, the concept of intersection over union (IoU) was applied (Eq. 13.8). This metric regards ground truth and detection (Fig. 13.7)
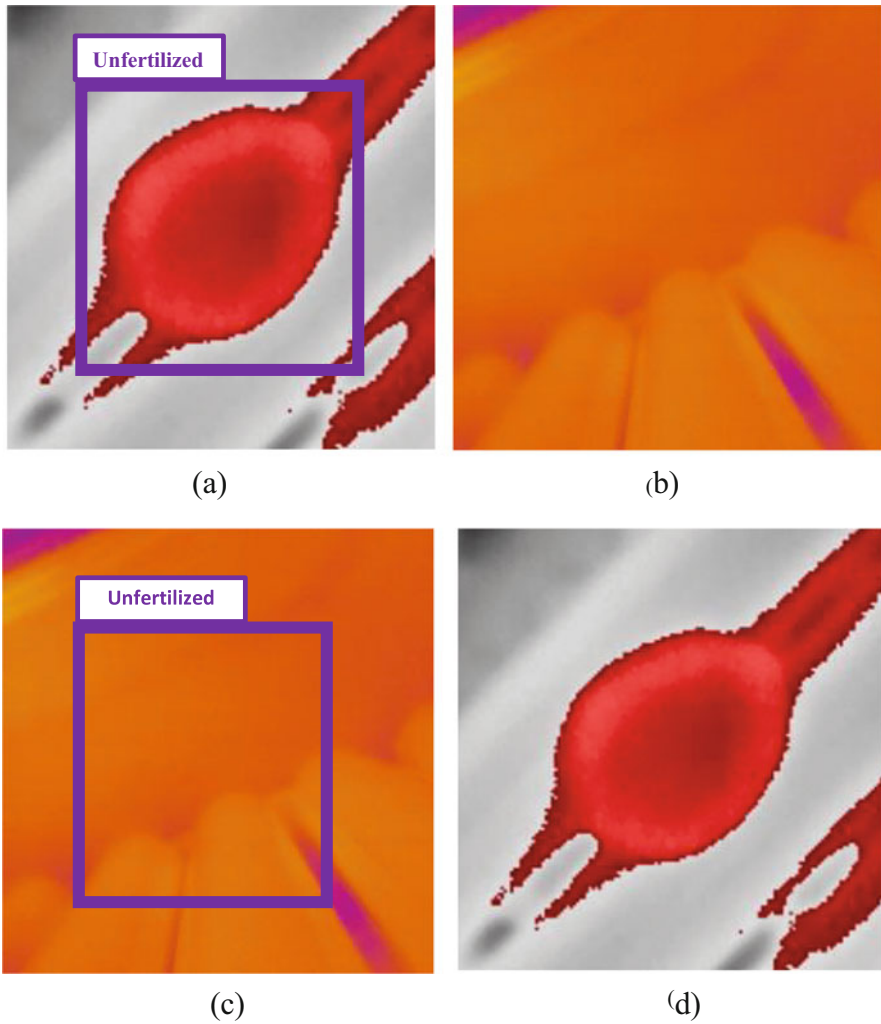
Fig. 13.6 (**a**) True positive; (**b**) true negative (TN); (**c**) false-positive (FP); (**d**) false negative (FN)

$$P = \frac{TP}{TP + FP} \tag{13.5}$$

$$R = \frac{TP}{TP + FN} \tag{13.6}$$

If IoU = 0.5
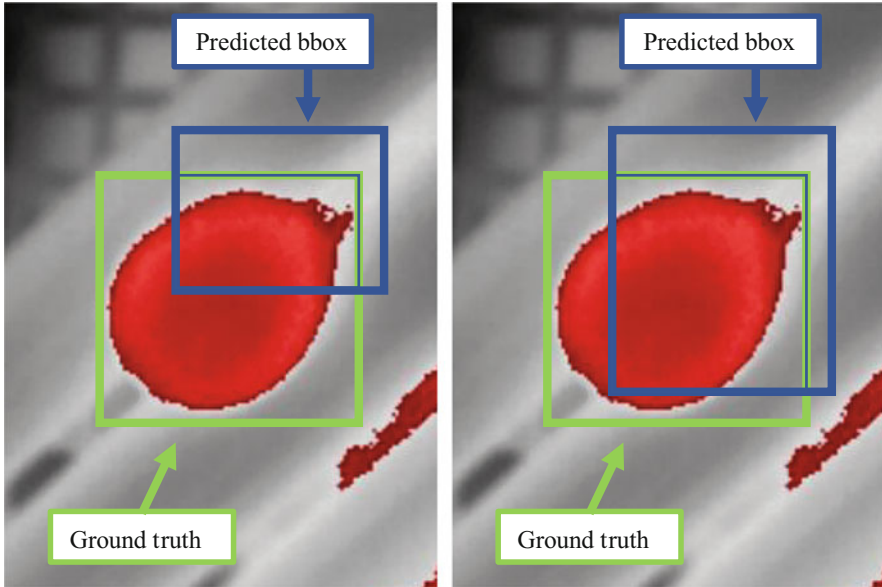
| False Positive (FP) | True Positive (TP) |
| IoU = 0.4 | IoU = 0.7 |

Fig. 13.7 IoU details of false-positive and true positive detection for unfertilized eggs

$$\text{F1 score} = \frac{2PR}{P+R} \tag{13.7}$$

$$\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}} \tag{13.8}$$

The mean average precision (mAP) is the average precision (AP) over the number of classes (Eq. 13.10). The AP is a metric used in the PASCAL VOC challenge (Everingham et al., 2010); it was obtained by calculating the area under the *P-R* curves interpolated at 11 points (Eq. 13.9). However, the interpretation may vary depending on the problem of classification. For instance, in COCO dataset evaluation, the metric AP was considered equivalent to mAP. This metric is important for object detection model evaluation because it considers the arrangement between *P* and *R* and the relation of FP and FN.

$$AP = \frac{1}{11} \sum_{R_i} PR_i \tag{13.9}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{13.10}$$

13.3 Results

The first two trials served the purpose of establishing the best procedure to collect data for the experiment. The results from this trial showed that collecting data from the upside face of eggs was more effective for the instant identification of features than collecting data from the downside face at either the short end or the large end of eggs. From the first trial, we collected images by removing the eggs from the incubator machine. In the second trial, we collected images from the inside of the incubator machine, and the second option suggest being the best for preventing the instantaneous exchange of temperature within the environment and for improving thermal transmission. On the 8th day, the eggs were artificially hatched to assess the presence of embryos (Table 13.2). As expected, the unfertilized eggs presented no embryos.

13.3.1 Thermal Features of Incubating Eggs

Using isotherm filtering, we observed different feature patterns of fertilized eggs to unfertilized eggs (Fig. 13.8).

Different periods of egg turning showed different patterns of thermal images, although similarities were also observed. For example, the common characteristic of most fertilized eggs was the presence of a "chamber" structure in the middle of the dark red spot (which is related to the composition of embryo and egg structures). As the days of incubation were passing, the chamber structure became less apparent (Fig. 13.9) For example, through thermal analysis under isotherm classification, the

Table 13.2 Assessment of embryos on the 8th day of incubation

Period of turning eggs	With embryo	No embryo	Number of egg w/no embryo
12 h	29	1	11
6 h	26	4	1, 11, 20, 28
1.5 h	25	5	2, 8, 14, 22, 27
Unfertilized 1.5 h	0	30	1–30
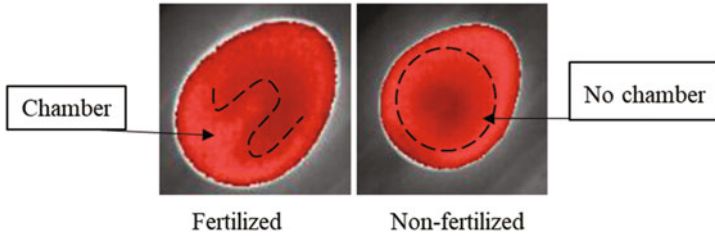
Fig. 13.8 Isotherm filtering for observing different feature patterns of fertilized (No. 10) and unfertilized eggs (No. 11) during the first days of incubation
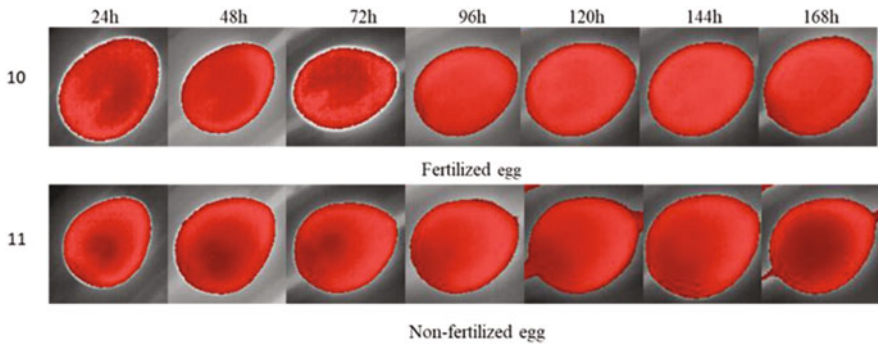
Fig. 13.9 Thermal analysis under isotherm classification for fertilized and unfertilized eggs after turning for a 12-h period over a 7-day incubation period

evolution of 2 eggs over 7 days is shown. The eggs were incubated for a turning period of 12 h. It is possible to observe that after 72 h, the structures became less apparent compared with unfertilized eggs. For unfertilized eggs, the dark spots became larger due to the accumulation of gases from the decomposition of yolk and dehydration.

Eggs turned every 6 h showed constant features during the data collection period (Fig. 13.10). Fertilized eggs had chamber features, while the binary pair did not show the same characteristics as eggs incubated for the 12 h rotation period.

For the eggs incubated during the 1.5 h period of turning, we observed that similar structures were also identified; however, from visual analysis by human eyes, such features were less clear than those of the previous two images shown above (Fig. 13.11).

13.3.2 Training Results

The YOLOv4 model was trained for approximately 20 h, and we stopped the training when the average loss reached less than 0.5 and the mAP was higher than
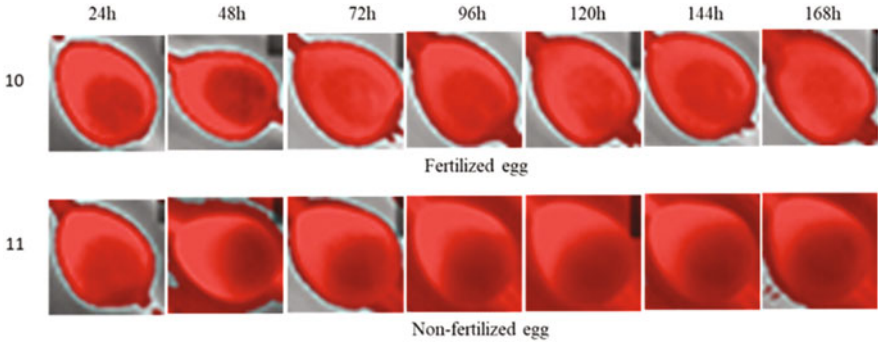
Fig. 13.10 Thermal images or eggs incubated at turning periods of 6 h in a total 7-day incubation period for fertilized and unfertilized eggs
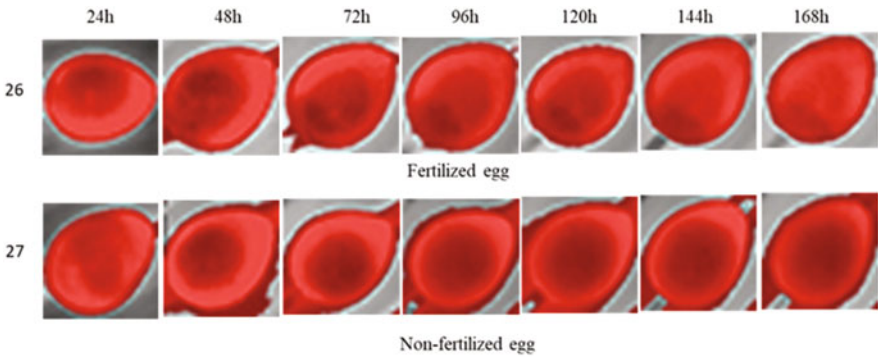
Fig. 13.11 Thermal images from eggs incubated at turning periods of 1.5 h in a total 7-day incubation period for fertilized and unfertilized eggs

90% for the training dataset (Fig. 13.12a). YOLOv5 trained in Google Collab have taken 1 h to be completed (Fig. 13.12b). SSD-MobilenetV2 model was obtained after 4 h of training in the PC environment (Fig. 13.12c).

The evaluation of training was performed by calculating the measurements as described in the results (Table 13.3). To test the model on three experimental treatments, we composed all the data of each cluster in only one image. As shown in the table, each egg picture was cropped to 30 × 30 pixels to reduce the size for comparing images from 12, 6, and 1.5 h (Figs. 13.13, 13.14, and 13.15 shows an example of testing YOLOv4 model).

To compare the images, we numbered the eggs from 1 to 30 in the rows. For the columns, we indicated the periods of data collection, from 12 h of incubation (first day) to 168 h (seventh day). Next, we calculated the precision, recall, and F1-score according to the detection assessment (Table 13.4 and Fig. 13.16). Note that we have discarded the "156 h" data from the 6 and 1.5 h datasets due to errors generated while collecting images. However, the analysis was conducted based only on the presented

Fig. 13.12 (**a**) Average loss during training of datasets using Darknet framework for YOLOv4. (**b**) YOLOv5 Training in Google Collab using PyTorch framework. (**c**) SSD-MobileNet V2 training loss in the TensorFlow framework
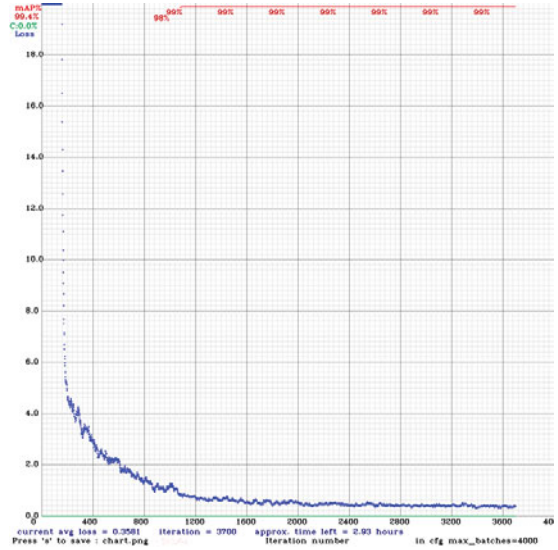
(a)
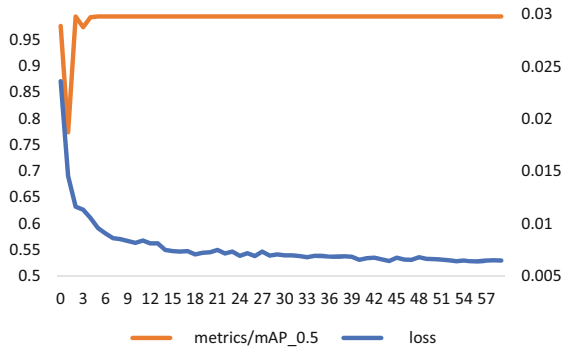
(b)
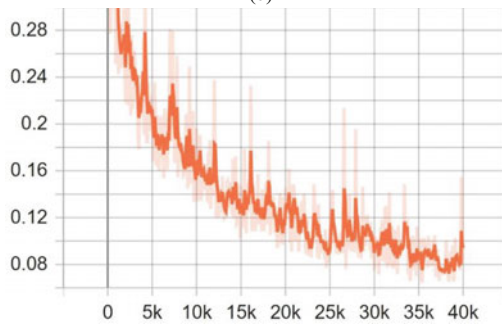
(c)

Table 13.3 YOLOv4, YOLOv5, and SSD-MobileNet V2 training model evaluation using the IoU threshold of 0.5 or 50% for the detection of unfertilized eggs

Model	Dataset	Precision (P)	Recall (R)	F1 score	Average IoU (%)	mAP@0.50 (%)
YOLOv4	Validation	0.97	0.99	0.98	77.39	98.62
YOLOv5	Validation	1	0.99	0.99	–[a]	99.5
SSD-MobileNet V2	Validation	1	0.94	0.96	–[a]	91.8

[a] Information not available

data as follows; Table 13.5 presents the evaluation of metrics according to describe in Sect. 13.2.4.4.

Table 13.3 shows the deep learning algorithms training performance, according to the metrics explained in Sect. 13.2.4.4, the mAP@0.50 is the main evaluation method for training models, it can be interpreted as the area under the interpolation of P and R curves. The value 0.50 is regarding the IoU, that considers an object detection if the overlap section overcome 50% of the ground truth (labeled object). YOLOv5 showed better mAP@0.50, followed by YOLOv4 and SSD-MobileNet V2.

The unfertilized eggs were assessed on the 8th day of incubation as indicated in Table 13.2. The calculation presented in Tables 13.4 and 13.5 are based on that.

In Table 13.4 the object detection resultant from our test dataset were tabulated. The total detections show the number of bounding boxes that were displayed for each model and for each treatment for detection of unfertilized eggs, the 6 h dataset displayed more bounding boxes over the three datasets. The total detection can also be interpreted as the sum of TP and FP detections as well, according to example given in Fig. 13.6, Sect. 13.2.4.4. TP column stands for the bounding boxes displayed around correct unfertilized eggs; TN is the total of eggs that were neither classified as unfertilized eggs nor the probability of being an unfertilized egg falls below the threshold (50% for YOLOv4 and YOLOv5 and 18% for SSD-Mobilenet V2), therefore, not detected as unfertilized eggs. FN is the quantity of unfertilized eggs that were not detected by the object detection algorithms. FP represents the number of bounding boxes displayed around eggs but not correspond to unfertilized eggs indeed.

Table 13.5 results from the calculation of metrics by using the data on Table 13.4. The P value was obtained using Eq. (13.5), it can be understood as the relation of unfertilized eggs that were corrected marked with bounding box over the total bounding boxes. The R is a metric the tells us the proportion of unfertilized eggs that were not detected by the object detector but should be detected. The F1 score stands for the balance between P and R, from this metric we can evaluate the proportion of assertiveness of each deep learning model on unseen dataset, in our case F1 score can tell us the proportion of unfertilized eggs detected by the YOLOv4, YOLOv5, and SSD-MobileNet V2 algorithm.
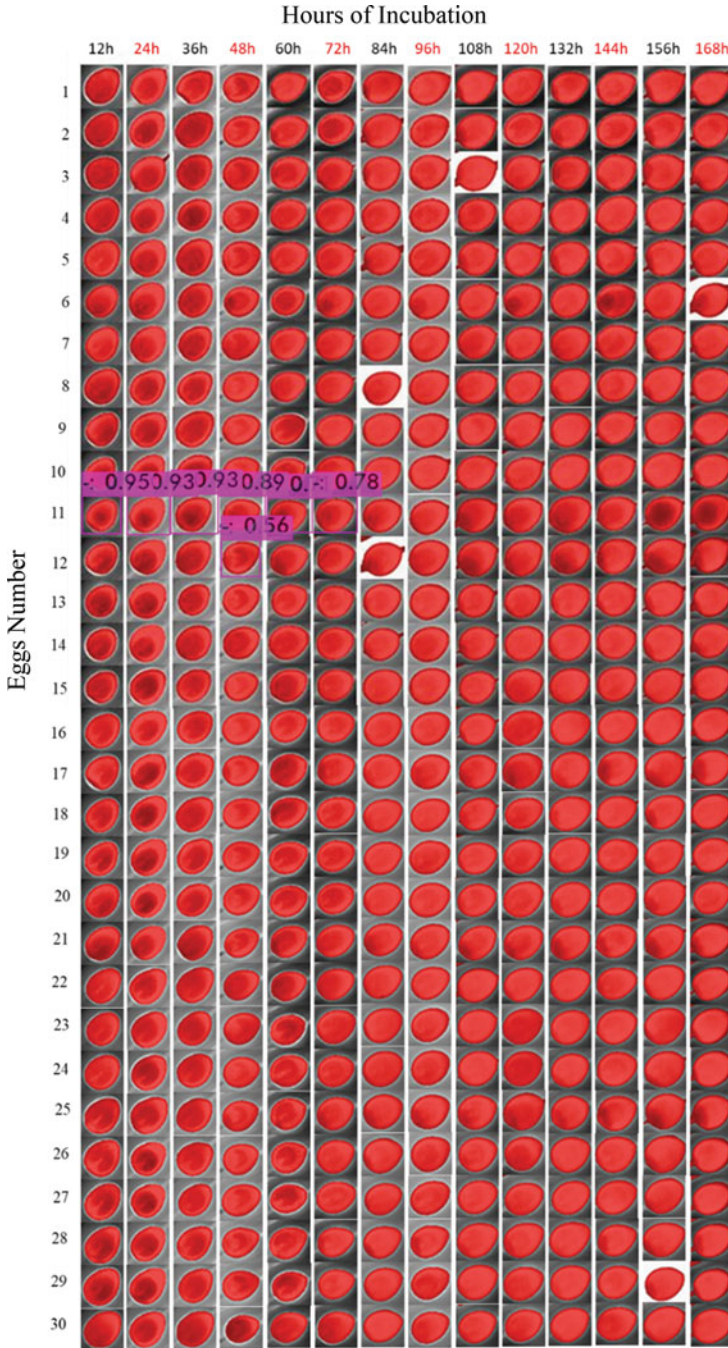
Fig. 13.13 Object detection of eggs turned every 12 h under no threshold
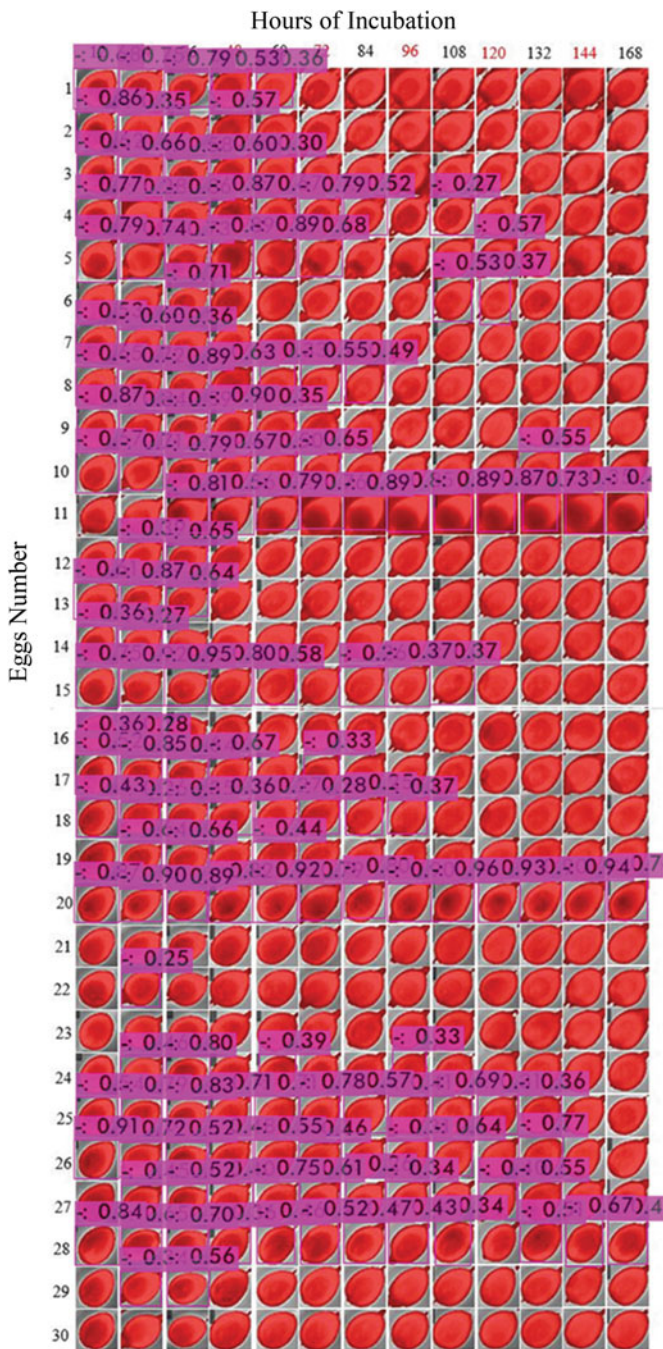
Fig. 13.14 Object detection of eggs turned every 6 h under no threshold
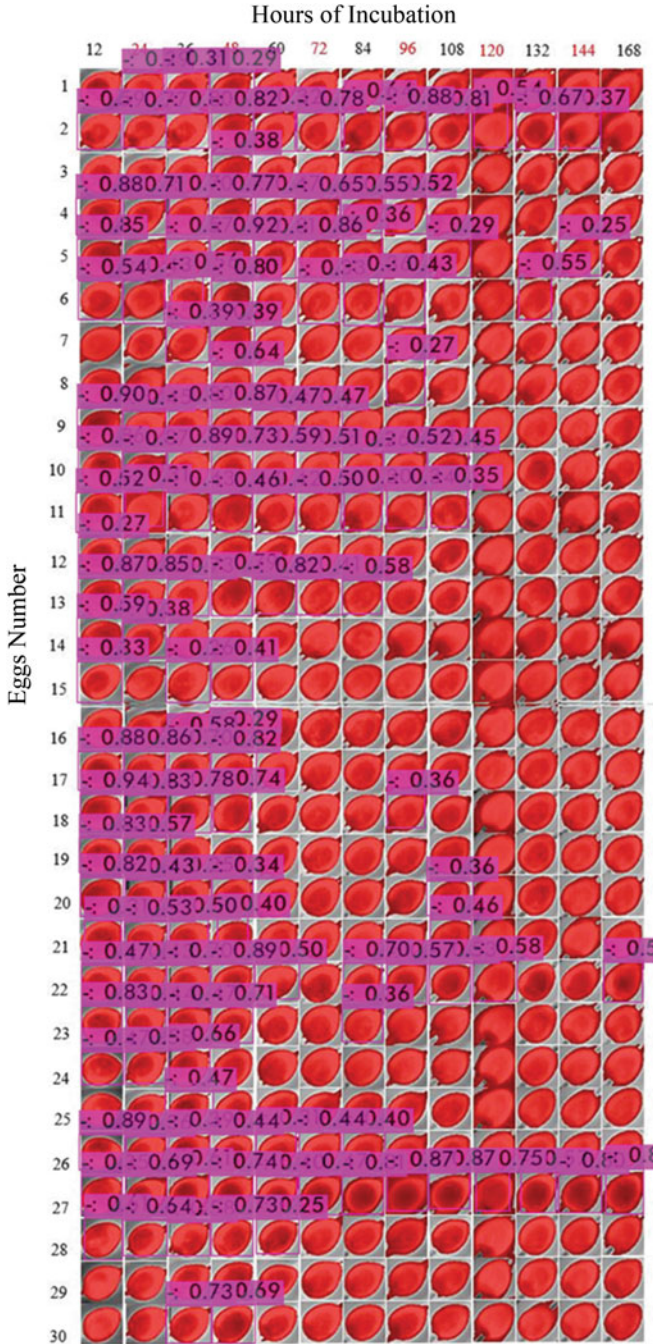
Fig. 13.15 Object Detection of eggs turned every 1.5 h (90 min) under no threshold

Table 13.4 Testing dataset detection evaluation parameters under a threshold of 0.5 in the YOLOv4 and YOLOv5, SSD-MobileNet V2 the threshold at 0.18 models for the detection of unfertilized eggs

Model	Dataset (h)	Total detections	True positive (TP)	True negative (TN)	False negative (FN)	False-positive (FP)
YOLOv4	12	7	6	413	8	1
	6	106	32	314	20	74
	1.5	86	29	334	36	57
YOLOv5	12	11	11	409	0	0
	6	36	24	384	28	12
	1.5	23	17	397	48	6
SSD-MobileNet V2	12	6	6	414	8	0
	6	11	7	409	45	4
	1.5	6	0	414	65	6

13.4 Discussion

The embryo development of quail eggs takes 16.5 days to complete according to Ainsworth et al. (2010). The same authors also affirms that although the incubation period of quail eggs and hen eggs differ (21.5 days for hen's eggs), the embryo development stages are quite similar, making it possible to compare the development process between them. Several studies on the characterization of avian embryo development have been reported (Hamburger & Hamilton, 1951; Sittmann et al., 1966; Graham & Meier, 1975; Ruffins et al., 2007). Most of the studies were focused on understanding the biological functioning of structures and genetics. However, due to the recent availability of sophisticated sensors and computational methods, new approaches for poultry yield improvements have been reported. In this study, we aimed to unleash the potential of thermal cameras to detect features that could differentiate fertilized eggs from unfertilized eggs as a nondestructive and noncontact technique. The overall goal was to contribute to monitoring the hatching process for more efficient quail hatching management.

The method of incubation utilized in this study differed from most common automatic incubator machines in industrial quail farms. For industrial incubators, the eggs are usually allocated in the vertical position and are periodically turned 45°, which is specifically related to facilitating labor operations on a large scale. Van de Ven et al. (2011) compared the position of hen eggs during incubation and concluded that position did not significantly interfere with the hatching rate. Oliveira et al. (2020) compared different turning periods of hen eggs and concluded that reducing the frequency of turning eggs affects the hatchability of chicks. In this study, we did not evaluate the hatchability of eggs or mortality.

Thermal cameras produce images by solving the intensity of infrared wavelengths transmitted to the thermal sensor; the major limitations of this kind of camera are low resolution and high price (Williams et al., 2022). Reflectance, transmittance, and

Fig. 13.16 (**a**) Testing data set precision comparison between three models. (**b**) Recall comparison. (**c**) F1 Score comparison
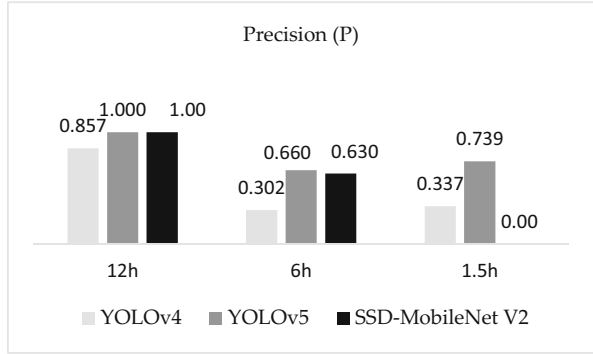
(a)
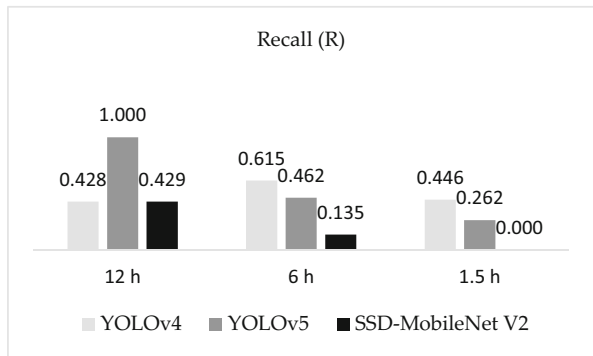
(b)
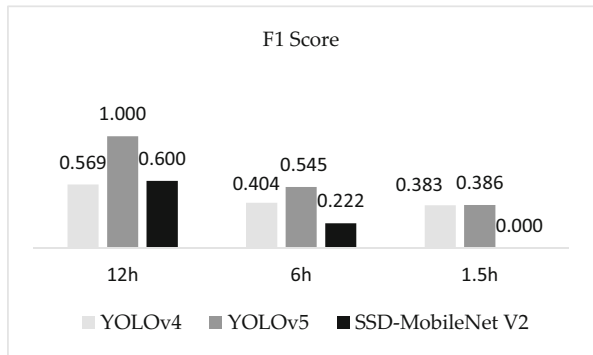
(c)

emissivity are the main factors of the application of thermal cameras, and the main use of this kind of sensor is specifically related to nocturne vision and body temperature monitoring. In this study, we proposed the use of thermal imaging to

Table 13.5 Evaluation of precision (*P*), recall (*R*), and F1-score in the YOLOv4, YOLOv5 and SSD-MobileNet V2 models for the detection of unfertilized eggs

Model	Dataset (h)	Precision (*P*)	Recall (*R*)	F1-score
YOLOv4	12	0.857	0.428	0.569
	6	0.301	0.615	0.404
	1.5	0.337	0.446	0.383
YOLOv5	12	1	1	1
	6	0.66	0.5	0.56
	1.5	0.60	0.26	0.36
SSD-MobileNet V2	12	1	0.42	0.59
	6	0.63	0.13	0.21
	1.5	0	0	0

observe the thermal behavior of quail eggs during incubation stages. The use of isotherm filtering is the main point of this proposed methodology to identify unfertilized eggs, as isotherms can cluster radiometric information under a threshold. The variations in the eggshell due to the development of the embryo and the transformation of the yolk sac content, allantois and air chamber were expected to interfere with the dynamics of gases through the micropores of the shell. Therefore, thermal imaging can capture these changes. However, the lack of a standard configuration of isotherm controls might be the main factor responsible for the poor clarity of features that could facilitate the classification of embryo absence.

The deep learning-based algorithms proved to be a powerful tool for vision systems operation in real time, and the high precision of the model showed that it is possible to extract features of thermal images from unfertilized eggs. However, we observed that more data are necessary to improve the robustness of the model. We realized that classifying images using only one class might be not enough to push the bias of the model for more precise detailed feature extractions inside the eggs. In further research, we will address such limitations to overcome by increasing the classes.

From the results we observed that YOLOv5 over performed the all the other models. For training performance mAP@0.50 was smoothly higher than other two models, and the F1 score in the testing dataset was better, that is because the model presented less FP over all detections. YOLOv4 followed the YOLOv5, and the worst results were obtained from SSD-MobileNet V2. The result of P shows that YOLOv5 was more precise for detecting unfertilized eggs under 50% of confidence.

We also noticed that the fertilized egg thermal profile shows similar structures that are not dependent on the period of egg turning. However, the low resolution of thermal cameras can compromise the recognition of features, that is, the distance and transmittance effects can compromise the detection of features. We observed that errors occurring during the collection of data may be responsible for poor radiometric data, and consequently, the misclassification of eggs. For the testing results, we observed that a few numbers of unfertilized eggs in each treatment were not sufficient to infer that different turning periods can improve the detection of a

class. However, using these datasets, we could test the deep learning model and ensure the potential for detecting embryos (by exclusion of unfertilized eggs) in the early stages of incubation. Most unfertilized eggs were correctly recognized, although the error caused by false-positive detections led the precision of the model to low values (except for the 12 h intervals, which presented a low error but only one fertilized egg).

This work proved that it is possible to use a nondestructive and noninvasive method to identify embryos in the early stages of development. In addition, thermal cameras can contribute to an enhanced hatching rate because unfertilized eggs can be removed before incubation, which represents a gain in the contention of a waste of resources, such as space and energy in quail breeding farms. High throughput applications of the methodology proposed in this work are possible. However, fine studies will be addressed for standardization of isotherms filtering, as the lack of filtering pattern represents the main bottleneck for the methodology presented herewith. The breakthrough methodology proposed to identify embryos in early stages during incubation period can be extended to other avian eggs, besides that, low-cost devices and online inference are possible as well. Further studies will be conducted in this regard by expanding data sets and development of fine-tuning kernels for fertilized eggs thermal imaging.

13.5   Conclusions

The methodology developed in this study was able to identify different features of fertilized quail eggs from unfertilized eggs by using thermal microcamera and deep learning algorithms. The use of isotherm analysis during the incubation period associated with the YOLO object detection algorithm showed immense potential to compose automated systems based on CV for the classification of unfertilized eggs in early stages. Different characteristics for fertilized eggs and unfertilized eggs were identified for all the evaluated treatments, concluding that different periods of egg rotation do not interfere in the identification of embryos in early incubation. We noticed that unfertilized eggs could be identified after 12 h of incubation. To test the model, we reduced the original size and clustered all images into only one image dataset to test the robustness of the model when fitting the model all at once. The results showed that the model did not overfit the training dataset, as we could observe that most of our targets were correctly classified. The results from the evaluation of the training dataset showed high precision for the validation dataset. However, when testing the images under each treatment dataset, the precision was reduced. Several reasons may be responsible for the reduced precision of the testing model, such as the short resolution of testing images and issues associated with errors when collecting data using a thermal microcamera at each period. A further study will be conducted to enhance the precision of detection, including enlargement of the dataset and classification of fertilized eggs.

Conflicts of Interest   The authors declare no conflicts of interest.

References

Ainsworth, S. J., Stanley, R. L., & Evans, D. J. (2010). Developmental stages of the Japanese quail. *Journal of Anatomy, 216*(1), 3–15. https://doi.org/10.1111/j.1469-7580.2009.01173.x

Bamelis, F. R., Tona, K., DeBaerdemaeker, J. G., & Decuypere, E. M. (2002). Detection of early embryonic development in chicken eggs using visible light transmission. *British Poultry Science, 43*, 204–212.

Bertechini, A. G. (2012) The quail production. Area: Poultry welfare and environment. In *Proceedings of the 24th World's Poultry Congress*, Salvador, Bahia, Brazil (pp. 1–4).

Björn, L. O., & Nilsson, A. M. (2018) Thermal emissivity: Basics, measurement, and biological examples. In *Bioinspired engineering of thermal materials* (pp. 159–174). Wiley.

Bochkovskiy, A., Wang, C.-Y., & Mark Liao, H.-Y. (2020) Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

Coucke, P. M., Room, G. M., Decuypere, E. M., & De Baerdemaeker, J. G. (1997). Monitoring embryo development in chicken eggs using acoustic resonance analysis. *Biotechnology Progress, 13*(4), 474–478.

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. *Computers and Electronics in Agriculture, 157*, 471–478. https://doi.org/10.1016/j.compag.2019.01.021

Everingham, M., et al. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision, 88*(2), 303–338.

Girshick, R. (2015). Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1440–1448).

Girshick, R., et al. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Graham, D. L., & Meier, G. W. (1975). Standards of morphological development of the quail, Coturnix coturnix japonica, embryo. *Growth, 39*(3), 389–400.

Hamburger, V., & Hamilton, H. L. (1951). A series of normal stages in the development of the chick embryo. *Journal of Morphology, 88*(1), 49–92.

Höpe, A. (2014). Diffuse reflectance and transmittance. In *Spectrophotometry—Accurate measurement of optical properties of materials* (Experimental methods in the physical sciences) (Vol. 46, pp. 179–219). Academic. https://doi.org/10.1016/B978-0-12-386022-4.00006-6

Khaliduzzaman, A., Kashimori, A., Suzuki, T., Ogawa, Y., & Kondo, N. (2021). Research note: Nondestructive detection of super grade chick embryos or hatchlings using near-infrared spectroscopy. *Poultry Science, 100*(7), 101189. https://doi.org/10.1016/j.psj.2021.101189. ISSN 0032-5791.

Lahiri, B. B., Bagavathiappan, S., Soumya, C., Jayakumar, T., & Philip, J. (2015). Infrared thermography based studies on mobile phone induced heating. *Infrared Physics & Technology, 71*, 242–251. https://doi.org/10.1016/j.infrared.2015.04.010. ISSN 1350-4495.

Lin, T. Y., Dollár, P., Girshick, R., et al. (2017). Feature pyramid networks for object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2117–2125). Retrieved from https://arxiv.org/abs/1612.03144v2

Liu, L., & Ngadi, M. O. (2013). Detecting fertility and early embryo development of chicken eggs using near-infrared hyperspectral imaging. *Food and Bioprocess Technology, 6*, 2503–2513. https://doi.org/10.1007/s11947-012-0933-3

Liu, S., Qi, L., Qin, H., et al. (2018). Path aggregation network for instance segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 8759–8768). Retrieved from https://arxiv.org/abs/1803.01534v4

Lovarelli, D., Bacenetti, J., & Marcella Guarino, A. (2020). Review on dairy cattle farming: Is precision livestock farming the compromise for an environmental, economic and social sustainable production? *Journal of Cleaner Production, 262*, 121409. https://doi.org/10.1016/j.jclepro.2020.121409. ISSN 0959-6526.

Lukanov, H. (2019). Domestic quail (Coturnix japonica domestica), is there such farm animal? *World's Poultry Science Journal, 75*(4), 547–558. https://doi.org/10.1017/S0043933919000631

Maldague, X. (2001). *Theory and practice of infrared technology for nondestructive testing*. Wiley.

Martins, E. N. (2002). Perspectivas do melhoramento genético de codornas no Brasil. *Simpósio internacional de coturnicultura, 1*, 109–112.

Oliveira, G. d. S., dos Santos, V. M., Rodrigues, J. C., & Nascimento, S. T. (2020). Effects of different egg turning frequencies on incubation efficiency parameters. *Poultry Science, 99*(9), 4417–4420. https://doi.org/10.1016/j.psj.2020.05.045. ISSN 0032-5791.

Qin, W. C., Tang, X. Y., Peng, Y. K., & Zhao, X. H. (2017). Identification of fertilized chicken eggs based on visible/near-infrared spectrum during early stage of incubation. *Spectroscopy and Spectral Analysis, 37*(1), 200–204.

Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 21–26 July 2017 (pp. 7263–7271).

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* (*NIPS*) (pp. 91–99).

Ruffins, S. W., et al. (2007). Digital three-dimensional atlas of quail development using high-resolution MRI. *The Scientific World Journal, 7*, 592–604.

Sandler, M., et al. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Shanawany, M. M. (1994). *Quail production systems: a review*. FAO.

Sittmann, K., Abplanalp, H., & Fraser, R. A. (1966). Inbreeding depression in Japanese quail. *Genetics, 54*(2), 371–379. https://doi.org/10.1093/genetics/54.2.371

Tullett, S. G., & Deeming, D. C. (1987). Failure to turn eggs during incubation: Effects on embryo weight, development of the chorioallantois and absorption of albumen. *British Poultry Science, 28*(2), 239–243.

van de Ven, L. J. F., Baller, L., van Wagenberg, A. V., Kemp, B., & van den Brand, H. (2011). Effects of egg position during late incubation on hatching parameters and chick quality. *Poultry Science, 90*(10), 2342–2347. https://doi.org/10.3382/ps.2011-01467. ISSN 0032-5791.

Wang, C.-Y., et al. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In *2020 IEEE CVF Conference on Computer Vision and Pattern Recognition Workshops* (*CVPRW*).

Wathes, C. M., Kristensen, H. H., Aerts, J. M., & Berckmans, D. (2008). Is precision livestock farming an engineer's daydream or nightmare, an animal's friend or foe, and a farmer's panacea or pitfall? *Computers and Electronics in Agriculture, 64*, 2–10. https://doi.org/10.1016/j.compag.2008.05.005

Williams, S. M., Bariselli, S., Palego, C., Holland, R., & Cross, P. (2022). A comparison of machine-learning assisted optical and thermal camera systems for beehive activity counting. *Smart Agricultural Technology, 2*, 100038. https://doi.org/10.1016/j.atech.2022.100038. ISSN 2772-3755.

Wilson, H. R. (1990). Physiological requirements of the developing embryo: Temperature and turning. In *Avian incubation* (pp. 145–156).

YOLOv5. (2021). Retrieved July 2022, from https://github.com/ultralytics/yolov5

Yoshizaki, N., & Saito, H. (2002). Changes in shell membranes during the development of quail embryos. *Poultry Science, 81*, 246–251.

Yu, H., Wang, G., Zhao, Z., Wang, H., & Wang, Z. (2019). Chicken embryo fertility detection based on PPG and convolutional neural network. *Infrared Physics & Technology, 103*, 103075. https://doi.org/10.1016/j.infrared.2019.103075

Zhang, W., Pan, L., Tu, K., Zhang, Q., & Liu, M. (2014). Comparison of spectral and image morphological analysis for egg early hatching property detection based on hyperspectral imaging. *PLoS One, 9*(2), e88659. https://doi.org/10.1371/journal.pone.0088659

Chapter 14
Strategic Short Note: Intelligent Sensing and Robotic Picking of Kiwifruit in Orchard

Longsheng Fu

Abstract Kiwifruits are commercially grown on sturdy support structures such as T-bars and pergolas, which has great desiring to be harvested mechanically as unstable field labor availability and increased labor cost. Intelligent sensing and nondestructive picking of fruit are the two main key technologies for robotic harvesting. Deep learning technologies has been employed to train and detect kiwifruit, which achieved good performance by improving YOLOv3-tiny with two convolutional kernels to learn features of fruit calyx. Fruits were then classified into five classes based on their robotic picking strategy and field occlusions, which succeed to identify those fruits occluded by wires and branches. On the other hand, non-destructive end-effector was developed by simulating hand picking with a bionic mechanism. Drop distance of kiwifruit from the end-effector to container without fruit damage was also investigated. A kiwifruit picking robot was finally integrated with a tracked trolley, robot arm, fruit convey tube, and control system, which can pick a fruit with 2 s.

Keywords Kiwifruit · Intelligent sensing · Deep learning · Non-destructive picking · Robot

14.1 Introduction

China is the largest country producing kiwifruits worldwide. Harvesting kiwifruits in this area mainly depends on manual picking, which is labor-intensive. Therefore, introducing mechanical harvesting is highly desired. Kiwifruits are commercially grown on sturdy support structures such as T-bars and pergolas. Automatic detection of kiwifruit in the orchard is challenging because illumination varies through the day and night and because of color similarity between kiwifruit and the complex

L. Fu (✉)

College of Mechanical and Electronic Engineering, Northwest A&F University, Xianyang, Shaanxi, China
e-mail: fulsh@nwafu.edu.cn

background of leaves, branches and stems (Fu et al., 2021). Also, kiwifruits grow in clusters, which may result in having occluded and touching fruits (Suo et al., 2021). Intelligent sensing and nondestructive picking of fruit are the two main key technologies for robotic harvesting. Therefore, the main purpose of this research was to develop intelligent sensing of kiwifruit and nondestructive picking of fruit using robotic technologies.

14.2   Intelligent Sensing of Kiwifruit

One of the hardest challenges in the orchard automation is the harvesting robot specially for soft and delicate fruits. The fruits detection or object of recognition from leaves, branches, and wires are most toughs works. In this regard, a fast and accurate object detection algorithm was developed to automatically detect kiwifruits in the orchard by improving the YOLOv3-tiny model (Fu et al., 2021). Based on the characteristics of kiwifruit images, two convolutional kernels of 3×3 and 1×1 were added to the fifth and sixth convolution layers of the YOLOv3-tiny model, respectively, to develop a deep YOLOv3-tiny (DY3TNet) model. It takes multiple 1×1 convolutional layers in intermediate layers of the network to reduce the computational complexity. Testing images captured from day and night and compared with other deep learning models, namely, Faster R-CNN with ZFNet, Faster R-CNN with VGG16, YOLOv2, and YOLOv3-tiny with DY3TNet models achieved the highest average precision of 0.9005 with the smallest data weight of 27 MB. Furthermore, it took only 34 ms on average to process an image of a resolution of 2352×1568 pixels. The DY3TNet model, along with the YOLOv3-tiny model, showed better performance on images captured with flash than those without. Moreover, the experiments indicated that the image augmentation process could improve the detection performance, and a simple lighting arrangement could improve the success rate of detection in the orchard. The experimental results demonstrated that the improved DY3TNet model was smaller and efficient and it could increase the applicability of real-time kiwifruit detection in the orchard even through small hardware devices were used.

Deep learning has achieved kiwifruit detection with high accuracy and fast speed. However, all the kiwifruits have been labeled and detected as only one class in most researches for robotic fruit picking, where fruits occluded by branches or wires have been detected as pickable targets. End-effectors or robots may be damaged by the branches or wires when they are forced to pick those fruits. Therefore, kiwifruits are labeled, trained, and detected in multi-classes based on their occlusions to avoid detecting fruits occluded by branches or wires as pickable targets (Suo et al., 2021). Fruits are classified into four classes and five classes according to robotic picking strategy and field occlusions, respectively. Well-known YOLOv3 and recently released YOLOv4 are employed to do transfer learning for multi-classes kiwifruit detection. Results show that mAP (mean average precision) of fruits in the

five-classes was higher than that in the four-classes, while mAP of YOLOv4 was higher than YOLOv3.

14.3 Nondestructive Picking of Fruit

Fruit nondestructive picking is the other the key technologies of developing picking robot. Firstly, based on the artificial way of kiwifruit picking and the biology characteristics of kiwifruit stem, a fruit picking method for robot was proposed, which need to separate the fruit from stem and hold the fruit to prevent it dropping. Then, the picking method was verified by a special designed separation test of fruit and its stem. After that, an end-effector was designed and manufactured based on the fruit picking method, which approached a fruit from the bottom, and enveloped and grabbed the fruit from two sides, and then rotated up to separate the fruit from stem (Fu et al., 2015). In the end, the performance of end-effector prototype was tested on the most common cultivar "Hayward" at the Meixian Kiwifruit Experimental Station of Northwest A&F University. The results showed that the proposed picking method could separate the fruit successfully with the least force of 1.3 N when the angle between fruit and stem was set as 60°, which is not significantly different from the manual picking of which the angle between fruit and stem is approximately to 90° in normal. The end-effector was tested on 68 samples (28 in the morning, 25 in the noon, and 15 in the night). All of them were successfully picked and held by the end-effector. Among them, two samples were picked with stem which might be caused by the reason of that the fruit is not ripe enough to be harvested. All the picked fruit were free of damage until ripen for eating. In all, the end-effector could effectively solve the problems of separating the adjacent fruits, grab a single fruit with an accuracy of 100%, and pick and hold it nondestructively.

14.4 Kiwifruit Picking Robot

The above two key technologies were integrated with a tracked trolley, robot arm, fruit convey tube, and control system to develop a kiwifruit picking robot, which can pick a fruit with 2 s. It can be improved by adding more arms and end-effectors and working long hours, although it is slow than human picking. Another study has been conducted to investigate the drop distance of kiwifruit from the end-effector to container without fruit damage (Wu et al., 2022) (Figs. 14.1 and 14.2).

Fig. 14.1 Kiwifruit detected as five classes by YOLOv4 based on occlusion types

14.5   Conclusions

Open field autonomous robots have been advanced tremendously with the GNSS application for wase multiple satellite operations. In addition to Lidar brings the further application ease for obstacle avoidances. However, the orchard automation still requires significant attention to bring it in the field levels. There are significant

Fig. 14.2 Developed kiwifruit picking robot

scopes opened with the development of deep learning and AI application that over common conventional image processing shortcomings. This research efforts shows deep learning based light model can be used for orchard application picking of harvesting robots.

References

Fu, L., Zhang, F., Gejima, Y., Li, Z., Wang, B., & Cui, Y. (2015). Development and experiment of end-effector for kiwifruit harvesting robot. *Transactions of Chinese Society for Agricultural Machinery, 46*(3), 1–8.

Fu, L., Feng, Y., Jingzhu, W., Liu, Z., Gao, F., Majeed, Y., Al-Mallahi, A., Zhang, Q., Li, R., & Cui, Y. (2021). Fast and accurate detection of kiwifruit in orchard using improved YOLOv3-tiny model. *Precision Agriculture, 22*(3), 754–776.

Suo, R., Gao, F., Zhou, Z., Longsheng, F., Song, Z., Dhupia, J., Li, R., & Cui, Y. (2021). Improved multi-classes kiwifruit detection in orchard to avoid collisions during robotic picking. *Computers and Electronics in Agriculture, 182*, 106052.

Wu, Z., Li, G., Yang, R., Longsheng, F., Li, R., & Wang, S. (2022). Coefficient of restitution of kiwifruit without external interference. *Journal of Food Engineering, 327*, 111060.

Chapter 15
Low-Cost Automatic Machinery Development to Increase Timeliness and Efficiency of Operation for Small-Scale Farmers to Achieve SDGs

Arkar Minn and Tofael Ahamed

Abstract Sustainable development goals (SDGs) aim to increase productivity and utilization of resources in the agricultural sector. However, the agricultural sector is currently confronted with the effects of climate change and workforce shortages. The population has risen such that food security has become vital. Furthermore, the global agriculture industry is struggling with issues such as labor and farmer scarcity as well as rising labor costs. To bring a solution to this current trend, the problems of agricultural labor forces, automation, sensor advancements, artificial intelligence, and IoT that can support more young farmers are highly required. The innovations and implementation of advanced machinery can be stratified based on regional demand and population engagement in the agriculture sector both in developed and developing nations. There are six degrees of mechanization and automation are significant noted: Level 0 refers to no automation, Level 1 is assistance in automation, Level 2 outlines partial automation, Level 3 is conditional automation, Level 4 is high automation, and Level 5 is full automation considering the sensing system lateral and longitudinal control of machinery. Therefore, the purpose of this chapter is to discuss the application levels of mechanization to support labor shortages and increase productivity in developed and developing countries. This article solely describes the current trend in agricultural machinery adoption and the levels of mechanization that can be recommended for appropriateness globally. In addition to level selection, this article introduces sensors and transformation stages with low-cost automation, specifically shifting toward autonomous machinery development in the future spectrum.

A. Minn
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering, Yezin Agricultural University, 15013, Nay Pyi Taw, Myanmar

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

15.1 Introduction

The agricultural sector is currently confronted with the effects of climate change and workforce shortages. Climate change has a direct impact on food productivity. The population has risen such that food security has become vital. Furthermore, the global agriculture industry is struggling with issues such as labor and farmer scarcity as well as rising labor costs. Between 1950 and 2000, the number of family farm laborers in the USA fell by 7.60 to 2.06 million, while the number of hired farm employees fell by 2.33 to 1.13 million (Tassell, 2022). Between 2005 and 2020, the number of agricultural workers in EU-N13 fell by 13 million to 7 million (EC, 2017). In Japan, the total number of people employed in agriculture decreased by 2.42 million to 2.08 million between 2009 and 2015, with the 65+ age group making up 48% of the agricultural labor force (Statistic Bureu of Japan, 2016). The agricultural labor force has considerably declined in other Asian nations, such as China, India, Pakistan, Bangladesh, Thailand, Indonesia, Malaysia, and Myanmar. To bring a solution to this current trend in agricultural labor forces, automation, advancements in sensors, artificial intelligence, and IoT can support and motivate more young farmers and entrepreneurs. Aryal et al. (2021) state that farm mechanization has the potential to increase the income of agricultural households, improve food security, and reduce poverty, thereby contributing to the fulfillment of the SDGs. Automation of farm machinery has become increasingly important. For example, rice transplantation in wetlands has become difficult due to the climate and frequent interruption of rainfall. Labor transformation to industry also makes a difficult point for transplanting, broadcasting, and harvesting at the peak agricultural season. Hence, farmers prefer to use machines with semi automation or partial automation because automation is reliable and precise and increases the timeliness and efficiency of their operations.

Additionally, due to the increase in labor cost and plant disease factors, farmers noticed farm machinery automation, and they wanted to introduce that machine to their business. Furthermore, field sensing explored the significance of enabling precision farming to maximize production efficiency with minimum environmental impact. In this regard, there are four major steps to adopt the machinery in current agricultural trends with innovation and climate adaptation. Through these steps, the mechanization needs to incorporate with Big Data and artificial intelligence, smart IoT, and field sensing. Each nation uses the step differently depending on the availability of technology resources with innovative approaches. In recent trends, small units of low-cost intelligent/automation systems can utilize and further develop locally with their technological resources. Therefore, the recent advancement of low-cost automation can help in the agriculture sector of small-scale farmers.

The innovations and implementation of advanced machinery can be stratified based on regional demand and population engagement in agriculture in both developed and developing nations. In addition, challenges of socioeconomic and safety concerns must be considered. With all these considerations, several potential mechanization levels have been identified as prospective solutions (Sims & Kienzle, 2017) (FAO, sustainable agricultural mechanization, 2017). The core concept of the theory of induced innovation is that technical advancement is critical to enhancing agricultural productivity; thus, it is not exogenous to the development process (Hayami & Ruttan, 1971). There are six degrees of mechanization and automation that are significant and are noted as follows: Level 0 refers to no automation, Level 1 is assistance in automation, Level 2 outlines partial automation, Level 3 is conditional automation, Level 4 is high automation, and Level 5 is full automation considering the sensing system lateral and longitudinal control of machinery. Therefore, the purpose of this chapter is to discuss the application levels of mechanization to support labor shortages and increase productivity in developed and developing countries. Thus, this article solely describes the current trend in agricultural machinery adoption that can be transformed into different levels of mechanization for appropriateness globally.

15.2 Current Agricultural Trends

In the current trend, farm automation has transformed from a higher mechanical to an autonomous system at different levels through contributions from research institutes and industries. The automation of farm machinery systems has been developed with several basic principles to adapt to the implementation.

15.2.1 Control and Navigation System

Navigation systems have been created utilizing navigation sensors. Farm equipment has a position and a heading, and the sensors can be used to determine the equipment's posture. Lateral and longitudinal control of machinery has the major points to control machines in operation. The position information is the key that enables absolute (global navigation satellite system, GNSS) and relative positioning (local sensing). The development of GNSSs with higher accuracy and lower cost makes this score more vibrant across nations.

15.2.2 Vehicle Motion Models

The components of vehicle motion models mainly refer to dead reckoning, kinematic, and dynamic models for forward and backward motion and turning the vehicles. The vehicle motion model has significant roles at the different levels of machinery automation.

15.2.3 Navigation Planner

A navigation planer can be used to track location, follow an implement, and drive straight up on the global or local map. In the planning process, an obstacle avoidance system is also established based on the reference navigation planner using an advanced sensing system. Inertial navigation has also been developed based on landmark-based navigation and line navigation to follow the vehicle on the map.

15.2.4 Steering Controllers

Steering control has significantly improved at different application levels. The steering system control covers the open loop, proportional integral derivative (PID), and fractional proportional integral derivative (FPID) controllers, among others that can be used to steer agricultural vehicles. Trajectory controls have challenges for enabling navigation using Cartesian and polar-based coordinates.

15.2.5 Field Sensing, Recognition, and Sensor Data Fusion

Vision-based sensors, environment ranging sensors, time-of-flight range sensors, and image sensors are the main contributing sensors in automation and intelligent control systems. In vision sensing, line-scan cameras and area-scan cameras, including LiDAR and 3D depth cameras, have significant applications worldwide. A similar concept of sensor fusion involves processing sensor data and controlling vehicle movements based on a changing operating environment around the vehicle and sensor data from a set of sensors with varying degrees of accuracy (Baillie et al., 2018). Increased production is currently required to make automated robotic systems (ARS) economically justifiable and to enable their use at the commercial-scale level (Goense, 2005). Satisfactory performance of ARS within unstructured environments requires that both productivity and work quality are significantly higher than with conventional methods and equipment, including semiautomated systems (Bechar & Vigneault, 2016). Variable-rate technology (VRT) is the system most expected to

reduce agricultural inputs such as chemical fertilizers, pesticides, herbicides, irriga-
tion water, and precision placements of seeds.

15.2.6   Variable-Rate Technologies

VRTs are new technologies for determining nonuniform farm inputs, such as seeds,
fertilizers, and pesticides, to be applied at different rates across a field, without
manually changing rate settings on equipment or having to make multiple passes
over an area. The VRT is one of the three most common components for site-specific
crop management (SSCM). The equipment used to perform variable-rate applica-
tions is commonly called variable-rate technology or VRT (Ahamed et al., 2014).
There are two types of VRT: GPS-based systems and local sensor-based systems. A
GPS-based system is a map-based system and historic approach. A local sensor-
based system is a real-time system, and no maps are needed. A map-based system
(GPS-based) can be adjusted with the application rate based on an electronic map of
field properties. Another type is referred to as local sensor-based systems, which use
real-time sensors to perform on-the-go measurements and then control the input in
the same path. There are advantages and disadvantages between map-based and
local sensor-based systems (Fig. 15.1). Modern farm machinery, such as sprayers
and spreaders, have VRT-capable machinery. Most of the time, it is possible to
modify current machinery and equipment. Most tractor navigation systems can be
integrated with VRA map reading and signal control abilities for site-specific control
of agricultural inputs.
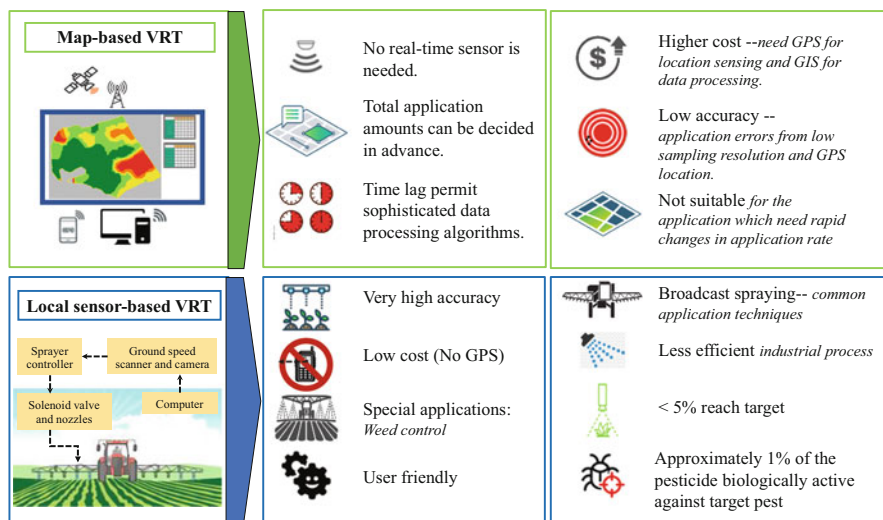
Fig. 15.1 Comparison of map-based and local sensor-based systems in VRT

15.2.7   Communication Protocols

The controller area network (CAN) bus communication protocol based on ISOBUS systems has the advantages of configuring machinery and implements from a wide range of dealers and manufacturers. ISOBUS is an international communication protocol that sets the standard for agriculture electronics. Farm equipment manufacturers decided on ISOBUS as the common protocol for electronic communication between tools, tractors, and computers under the International Standards Organization (ISO). It can communicate between sensors, actuators, and controllers, facilitating this standard to switch data between tractors, implements and onboard controllers. Currently, ISOBUS-compatible implements are available in the farm machinery market. Through standardized ISOBUS components, it is intended to provide plug-and-play functionality for every combination, enabling the machine to control any device or application. The ISOBUS universal protocol has several simplification benefits with a display that interacts with multiple implements and machines. In addition to simplicity, the user-friendly control setting reduces downtime and minimizes installation and interface issues. Easy connections with the standardized connector between different components and automation can be performed with the ISOBUS system to increase the efficiency and reduce fatigue of the operator.

Therefore, farm mechanization has the scope of considering the abovementioned controllers, sensors, and navigation planners for guiding agricultural vehicles to address labor shortages in a different scheme based on the requirements for each country. This scope can be outlined with different levels of mechanization starting from Level 0 to Level 5 (SAE, 2021) (Fig. 15.2).

Level 0	Level 1	Level 2	Level 3	Level 4	Level 5
No Automation	Assisted Automation	Partial Automation	Conditional Automation	High Automation	Full Automation
Vehicle does not have any intervening system.	The rest operations done by the system.	The system has the longitudinal and lateral control for specific application.	The system has both longitudinal and lateral control for specific application.	System is capable to manage all situations automatically for a defined job.	The system is capable to deal with all situations automatically for all jobs.

Fig. 15.2 Levels of automation referring to machinery control, lateral control, and navigation planning

15.3   Levels of Automation in Farm Machinery

The development of Level 0, Level 1, and Level 2 automation in small-scale farm machinery are emphasized considering low-cost autonomous machinery development. Levels 3, 4, and 5 of automation are focused on medium- to large-scale farm machinery automation systems. The following sections discuss levels 0–5 in more detail with the transformation potentials.

15.3.1   Level 0: No Automation

Most farm tasks are carried out by humans at the Level 0 stage. This level is still present in developing countries. Farmers used utility tractors with a simple hydraulic system, a mini-tiller, and a power tiller. Small-scale farm machinery with a small single-cylinder engine is the primary power source. The transformation is also required to automate the small-scale machinery starting with the power tiller, specifically riding types. Sophistication in automation can be performed for the power tiller with some consideration of low-cost actuators, sensors and GNSS systems. One of the examples of transformation and its recommendations to use low-cost equipment for automation on power tillers is described.

15.3.1.1   Transformation of Automation on a Power Tiller

Most developing countries in Asia use power tillers extensively in their food production. The advantages of a power tiller include its low cost, lightweight, ease of maintenance and repair, and simplicity of operation. However, it also has drawbacks, including mass vibration, pollution, and risky operation. When using a power tiller, farmers need to pay close attention. The following suggestions are added to modify the ordinary power tiller (Fig. 15.3) with automation.

Main clutch lever All moving parts of the power tiller are controlled by the main clutch. The operator can control it manually with the main clutch lever. For the automation assistance, it can be replaced by an electric linear actuator or a servo motor with a controller.

Steering mechanism Steering is manually controlled by side clutch levers; it can be replaced by two electric linear actuators.

Engine stop mode The engine stop mode can be transformed with a servo motor. For the petrol engine, a servo motor can be used to operate the engine switch to turn on and off, and for the diesel engine, a servo motor can operate the fuel valve.

Fig. 15.3 Automated power tiller

Implement mode The implement mode can be attached to the power tiller for the tillage and driving operations. An electric linear actuator can raise and lower the implement programmability.

Vision control Low-cost RGB, 3D depth camera or LiDAR can serve as the vision of the power tiller. Object detection and obstacle avoidance can be performed by the program.

Monitoring The radio transmitter module, Wi-Fi module or Bluetooth module has the opportunity to perform as the mobile control unit of the operator. The operator can contact the machine stages via mobile devices such as smartphones, tablets, and computers. Sensors can be used for machine stages such as fuel level and engine temperature.

Machine Control mode The operator can control the power tiller via a joystick controller. A microcontroller-based control system can be used as a remote-control system over a wireless platform.

Power source All electric and electronic devices obtain power from the battery, and the battery can be recharged from the alternator.

15.3.2 Level 1: Assisted Automation

In the Level 1 stage, farmers use utility tractors with hydraulic control systems. In rice cultivation, an ordinary rice transplanter with a mechanical function is used. In harvesting operations, threshers are also widely used. One of the examples of transformation and its recommendations to use the low-cost equipment for automation on the seed broadcaster are referred to in the following section.

15.3.2.1   Transformation of Automation for Seed/Fertilizer Broadcasting Device

Most seed and fertilizer broadcasting operations are performed with tractor-mounted broadcasters/spreaders. Precision broadcasters are now available in the market, but they are not affordable for small-scale farmers. There are two types of broadcasters: centrifugal disk type and pendulum type. The farmers most commonly used centrifugal-type fertilizer spreaders due to their robustness, simplicity, low cost, and large working widths (Antille et al., 2015; Villette et al., 2017). For the development of automation for centrifugal spreaders at the small-scale level, some of the sensors and attachments can be installed for semi-autonomous or autonomous vehicles (Fig. 15.4). The following suggestions are added to set up the spreader with automation.

Embedded system-based design A microcontroller serves as the operator (Harmanda et al., 2019). The programmer can set up the operation procedures for the specific operations. The microcontroller receives data from the sensors and computes and makes a decision using the program. Currently, low-cost high-technology microcontrollers are available, such as Arduino.

Substitution with electric devices Former control systems were made by mechanical control systems such as linkage and lever. Electric devices and attachments can replace the mechanical system, as they have more benefits, such as lightweight, quietness, and precision. Some electric devices are electric linear actuators, servo motors, stepper motors, and rotary encoders.

Adjustable opener design A programmable opening system can reduce the waste of seed/fertilizer, save usage, maintain a constant rate, and select the material dosage. Moreover, it can be programmed according to crop types such as paddy, wheat, maize, and pulses.

Headland control system The programmable control system of the broadcasting mechanism has the scope of effectiveness at the headland in the field. It can save the

(a)   (b)

Fig. 15.4 Small-scale multi-crop seed broadcasting machine (**a**) CAD design (**b**) Prototype fabrication

seed and equal the distribution pattern. This program is arranged by the RTK GPS (real-time kinematic global positioning system) and optical sensor (Camera, LiDAR).

Narrow space adjustment Narrow space adjustment can be arranged by the diffusing mechanism. The diffuser can be attached to the broadcasting outlet of the device so that the seed will not spread away from the spreading pattern. This adjustment is effective, saves the seed, and avoids spreading outside of the pattern.

Environment sensing This arrangement is set up by the sensors and IoT-based programs, such as methodological data from weather stations, for real-time environment operation. In broadcasting operations, the prevailing wind is the main obstacle to operation in the fields. If the program obtains weather data (wind speed and direction), the program can arrange the motor speed of the spreading mechanism. This arrangement has the advantage of saving the seed and equal distribution patterns in the field.

15.3.3 Level-2: Partial Automation

In the Level-2 stage, the prime mover is the utility tractor with an electronic control system. The hydraulic position control system is attached to the rice transplanter. Combine harvesters became popular in this stage. Few examples of transformation and its recommendations to use the low-cost equipment for automation for partial automation are described.

15.3.3.1 Transformation of Partial Automation

Hydraulic systems have been developed in agricultural tractors. Most utility tractors have auxiliary hydraulic hoses to control the rear attached implements in operation. With the assistance of an auxiliary hydraulic system, farmers can handle multiple implements in single operations (for example, tandem/gang arrangement of the implements) and can achieve a wide range of tillage. Additionally, hydraulic pump technology was developed and has resulted in less occupation of pump fitting and more power in controlling implementation.

   Hydrostatic transmission (HST) allows a vehicle to convert mechanical energy into hydraulic power and then back to mechanical energy. HST technology has been developed in agricultural machinery so that farmers can operate their work without worrying about machine slippage and digging in the mud. Crawler tractors, some combine harvesters that work in muddy fields and farm multipurpose carriers are fitted with HST drives (Fig. 15.5).
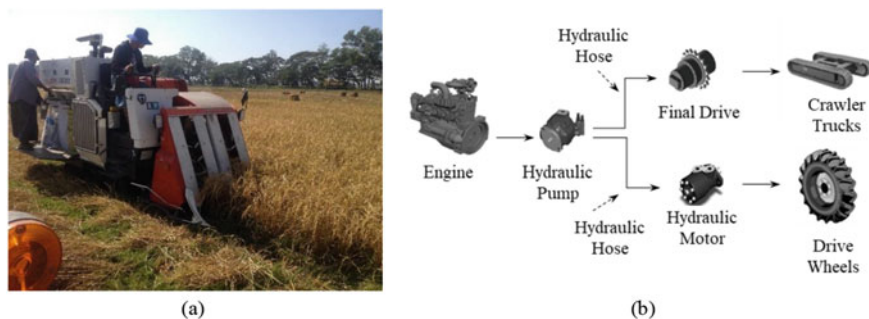
Fig. 15.5 (**a**) HST drive fitted on a combine harvester (**b**) HST drive system for crawler or wheel

15.3.4 Level 3: Conditional Automation

Technology plays an important role in the development of the crop production sector and has the potential key drivers of sustainable agriculture. Recently, new sensors have been developed with less weight and size, low cost, and high performance. Sensors can be applied and combined in production systems, accepting an increase in data and ultimately increases in information. This technology is of great importance for supporting digital transformation, precision agriculture, and smart farming to obtain high productivity and maintain sustainable agriculture. In the Level-3 stage, sensing technology and a global positioning system (GPS) have been developed. Most of the implements are fitted with sensors for controlling and monitoring operations. An autosteering system was developed with the aid of sensors and navigation systems.

15.3.4.1 Transformation of Conditional Automation

A real-time kinematic (RTK) GNSS system was developed for farm vehicle navigation, and RTK-GNSS provides precise accuracy of the vehicle location so that farmers can operate with hands-free steering and take control of the operations tasks. Moreover, machine data such as vehicle location and task information can be shared with others to improve operational performance and can perform precision agriculture. In recent years, low-cost RTK-GNSS has independently developed. Furthermore, a hybrid navigation system with a combination of inertial measurement units (IMUs) and GNSSs has also been introduced (Rinnan et al., 2009). This system is not only easy to switch between agricultural machines, but IMU can also be placed in the vicinity of the GNSS antenna to obtain a more accurate position, and direction calculation can be performed (Ahamed et al., 2014). To obtain high accuracy and high-rate positioning, the azimuth detection method improved the accuracy of agricultural work even in irregular fields and made it a high-speed output navigation sensor. Based on hybrid navigation that takes advantage of GNSS and IMU, by
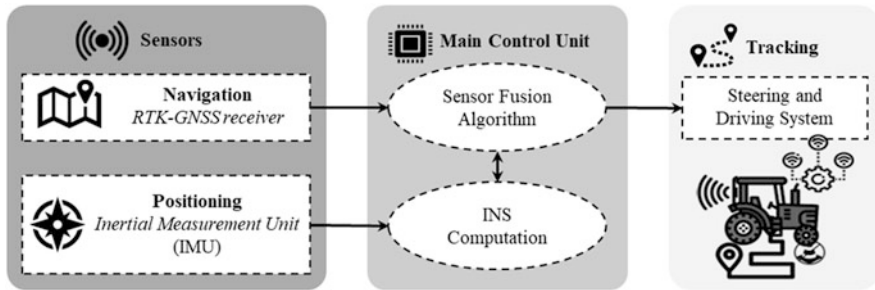
Fig. 15.6 Tractor navigation system with IMU+RTK-GNSS

utilizing machine information (vehicle speed and steering angle of this machine, traveling condition of this machine generated from map information). This operation can perform unique optimum calculations according to each driving condition (Fig. 15.6). Achieving high-rate detection and high accuracy positioning (within 3 cm) and azimuth angle (within 1°) with various operation patterns, including turning, completes the navigation sensor suitable for automatic agricultural machine travel (Ahamed et al., 2014).

15.3.5 Level 4: High Automation

Level 4 is sensing and IoT-based farm machinery development. Farmers can access their operations through mobile devices. Electromechanical drives became popular in this stage. A critical labor shortage threatens the ability of farmers to produce and harvest their crops. Migrant labor availability is subject to political forces outside of the power of the agricultural community. However, farmers are taking proactive steps to address this issue. They are teaming with technology companies to develop new automated harvesting technologies and more. There are three main technologies in future trends of automatic farm machinery: electrification, automation, and artificial intelligence (Fig. 15.7). Some of the examples described include electrification, automation, artificial intelligence and IoT-based components.

15.3.5.1 Transformation of Automation for Machinery

Electrification Most agricultural equipment, such as prime movers, are widely used internal combustion engines as power sources. Currently, fossil fuel has been decreasing in production, and engines release carbon dioxide. Well-known manufacturers have also changed the trend to use other energy sources, such as electrification. In farm machinery operations, electrification uses batteries as a power source. Batteries connect electrical drives such as electric motors that have heavy torque at low speeds, are more powerful, are more reliable, have lower noise, are lightweight,
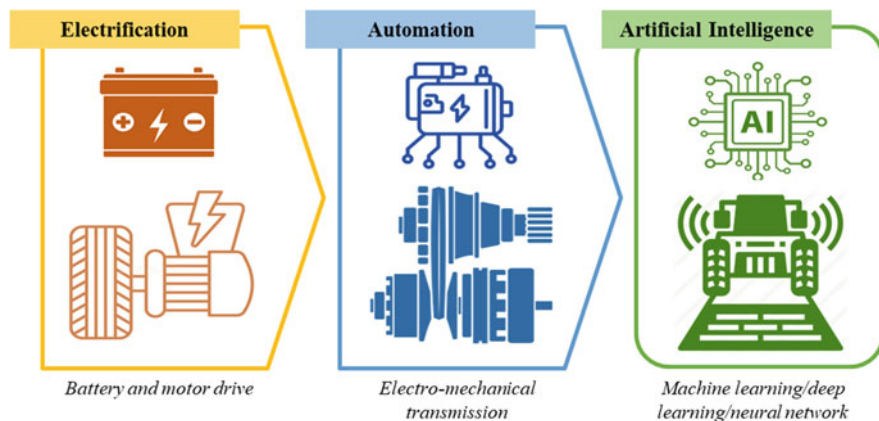
Fig. 15.7 Transformation at the level of high automation

and are compact so that motors can replace engines and hydraulics. Another advantage of using electrification is zero emissions.

Automation Currently, many well-known manufacturers are producing autonomous prime movers. The autonomous prime mover makes real-time automatic adjustments in farming operations. Some autonomous prime movers use electric drives so that they help to reduce soil compaction more than conventional prime movers. An electric drive has no operation emissions, low noise, and low wear and maintenance costs. In power transmission technology, John Deere Co. changed their new tractor in conventional continuously variable transmission (CVT) to electromechanical power split (Deere, n.d.).

Artificial intelligence (AI) Based on sensors and artificial intelligence, the system recognizes the difference between the desired work and environment to operate individual tasks. There are many methods for using artificial intelligence, such as machine learning, deep learning, and neural networks. Currently, this technology is used in farming operations not only in land preparation operations but also in soil management, crop management, plant disease management, and weed management. Moreover, postharvest and food processing operations have also used this technology.

15.3.6 Level 5: Full Automation

In the Level 5 stage, all farm machinery operations consider the full autonomy including remote operations (Fig. 15.8). Autonomous machines and intelligent systems are the core things of the stage.
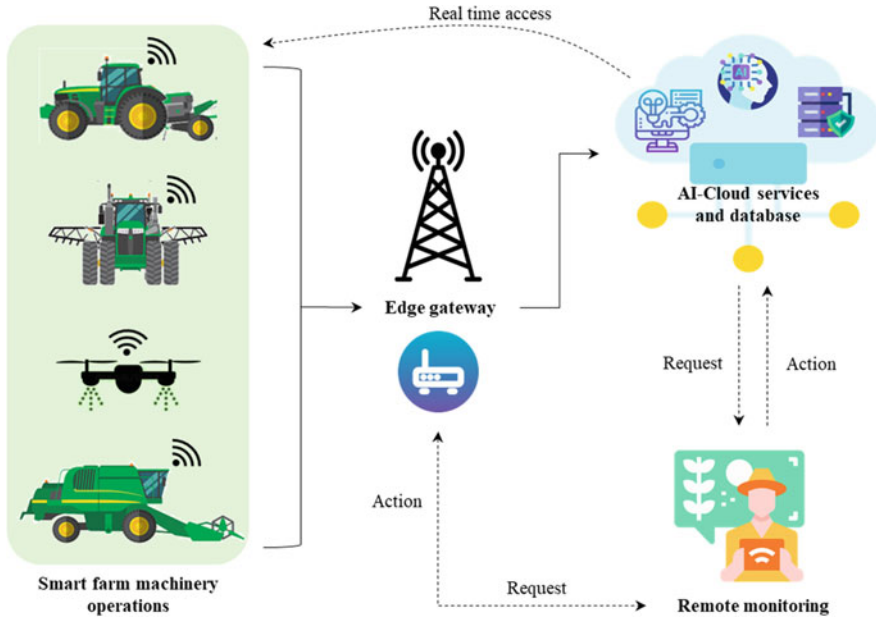
Fig. 15.8 Full automation

15.3.6.1 Transformation of Automation Systems

Some perspectives of small-scale autonomous farming vehicles and equipment are as follows. In the small-scale unit, Fendt from the AGCO company produced the Fendt Xaver sowing robot. That robot can operate precision planting, and it is a fully autonomous vehicle. The Xaver company intends to install a sowing robot with sensors for soil moisture measurements, temperature, humidity, and plant residues in the future (Fendt Xaver, 2020). Global unmanned spray system (GUSS) sprayers can be remotely supervised by a single operator, allowing growers to spray orchards and vineyards more quickly and consistently, using fewer resources and reducing costs by eliminating operator error and downtime (Oitzman, 2022). The machines also control application rates and sprayer speeds across entire fields and in variable terrain with adjustable software parameters (Deere, n.d.). AutoSpraySystems produced an R150 sprayer and attached the JetSprayer™ airflow sprayer system. This machine is small-scale, autonomous, and all-in-electric drives (AutoSpraySystems, n.d.).

Autonomous performance on rice transplanters has been developed by Japanese manufacturers such as Kubota and Yanmar. Yanmar produced the YR8D model, and Kubota produced the Agri Robo Rice Transplanter. Kubota has developed automation on rice transplanters but still needs to be operated under human supervision. Kubota expects to develop a completely unmanned operation of the machine in the future. Yanmar launched the YR8D series rice transplanter commercially in 2019, and the rice transplanter also needs to be performed under the guidance of people.

Table 15.1 Autonomous tractor listed in 2022

Sr.	Manufacturer	Model	Power type	Automation level
1	John Deere	8R Tractor	Engine type	Fully autonomous
2	CASE iH	ACV	Engine type	Fully autonomous
3	New Holland	NH^{Drive} Concept Autonomous Tractor	Engine type	Fully autonomous
4	Massey Ferguson	MF NEXT Concept Tractor	Engine type	Fully autonomous
5	ClASS	AgBot	Engine type	Fully autonomous
6	Fendt	FendtX concept	Engine-electric	Semi-autonomous
	Fendt	Fendt e100 Vario	Electric type	Semi-autonomous
7	Yanmar	Concept tractor YT01	Engine type	Fully autonomous
		Robot Tractor (YT488A/498A/4104A/5113A)	Engine type	Fully autonomous
8	Kubota	Kubota Concept Tractor	Electric type	Fully autonomous
		AgriRobo Tractor	Engine type	Fully autonomous

Yanmar announced that to create sustainable agriculture, they intend to continue to develop autonomous technology and incorporate it into an array of products and services.

Self-propelled multi-function machine is a complete set of an implementation of the operations. It can perform tillage, seeding, inter-weeding, spraying, and harvesting operations in a single unit. However, this type of machine has a very complicated system and high cost. This type of machine has significant potential for future development. Autonomous tractors are operator-less tractors that use sensors and perception technology to provide higher effectiveness and reduce human interference. Future research and development need to align with performance monitoring that extends to general-purpose tractor operations (Baillie et al., 2018). In many parts of the world's agricultural sector, there is a constant challenge of skill labor finding during the peak use seasons. Thus, automation and sensing technology can solve this problem. Large manufacturers are researching autonomous tractors and their operations. There are two types of autonomous tractors in agriculture: upgrading automation on existing tractors (engine type) and electric vehicles (electric type). Some of the autonomous tractors and their manufacturers series up to 2022 (Table 15.1). Furthermore, according to the land preparation to harvest task referring to Level 0 to Level 5, summarized information provides present trends and future development potentials (Table 15.2).

Table 15.2 Summary of the level of automation in farm machinery

Task	Level 0	Level 1	Level 2	Level 3	Level 4	Level 5
	No Automation	Assisted Automation	Partial Automation	Conditional Automation	High Automation	Full Automation
Land preparation	Utility Tractor – Manual hydraulic system	Utility Tractor – Hydraulic control system	Utility Tractor – Electronic control system – GPS – Auto-hinged system	Utility Tractor – Sensor-aided – Semi-auto steering – Cab control	Utility Tractor – Sensor-aided – IoT-based control – Electro–mechanical drive	Autonomous Tractor
Planting/Sowing/Fertilizer application	Seed drill – Manual control (Tractor-drawn)	Seed drill – Mechanical function (Tractor-drawn)	Seed drill – Hydraulic/Pneumatic Control (Tractor-drawn)	Seed drill – Sensor-aided – Auto-select function (Tractor-drawn)	Seed drill – Computerized – IoT based (Tractor-drawn)	Autonomous seeder
	Planter	Planter – Mechanical function (Tractor-drawn)	Planter – Hydraulic/Pneumatic Control (Tractor-drawn)	Planter – Sensor-aided – Auto-select function (Tractor-drawn)	Planter – Computerized – IoT based (Tractor-drawn)	Autonomous Planter
	Broadcaster/spreader	Broadcaster – Mechanical function (Tractor-drawn)	Broadcaster – Hydraulic/Pneumatic Control (Tractor-drawn)	Broadcaster – Sensor-aided	Broadcaster – Computerized	Autonomous Broadcaster

Operation				– Auto-select function (Tractor-drawn)	– IoT based (Tractor-drawn)	
				Broadcasting Drone	Broadcasting Drone	Broadcasting Drone
				– Operator-assisted	– Path-planning	– Intelligence system
						– IoT based
	Manual Transplanting	Rice Transplanter	Rice Transplanter	Rice Transplanter	Rice Transplanter	Autonomous Rice Transplanter
		– Mechanical control	– Hydraulic Position control	– Sensor-aided	– GPS	
					– Auto-steering	
Weeding/ Interculture	Utility Tractor	Utility Tractor	Utility Tractor	Utility Tractor	Semi-Autonomous Weeder	Autonomous Weeder
Crop Protection	Boom Sprayer (Tractor-drawn)	Utility Tractor	Utility Tractor	Utility Tractor	Utility Tractor	Autonomous Tractor
	– Mechanical control					
	Speed Sprayer	Speed Sprayer	Speed Sprayer	Speed Sprayer	Speed Sprayer	Autonomous Speed Sprayer
	– Mechanical control	– Electrical control	– Electronics control	– Sensor assisted	– Semi-autonomous	
				Drone Sprayer	Drone Sprayer	Drone Sprayer
				– Operator-assisted	– Path-planning	– Intelligence system
						– IoT based
Harvesting/ Threshing	– Reaper	– Baler	Combine harvester	Combine harvester	Semi-Autonomous Combine harvester	Autonomous Combine harvester
	– Thresher	– Combine harvester	– GPS	– Sensor-aided	– IoT based control	
	– Mower		– Sensor-aided	– Data Access via network		
	– Cutter/Chopper					

15.4 Discussion

The levels of mechanization are solely dependent on different countries' present needs and the social status of labor shortages. In referring to the several levels, smart farming came up from Level 2 or onward with the application of different sensors. *Smart* denotes sustainable mechanization and adaptation through resilient techniques. In smart farming, farms that are managed by information and communication technologies are used to increase the quantity and quality of products while optimizing the human labor needed. In the conversion of conventional farming to smart farming, we need to convert analogical or mechanical technology to digital data and information acquisition. In addition, smart farming provides benefits in the focus areas for site-specific management to ensure productivity. The productivity of agricultural machinery significantly increases through automation due to increased efficiency, reliability, and precision and a reduced need for human mediation (Burks et al., 2005; Schueller, 2014). Therefore, efficiency adopts new ideas to farm smarter. Performance is measured by capturing farm data and investing in precision agriculture (right source, right rate, right place, and right time) (Baillie et al., 2017). Finally, profitability can be achieved by strategically choosing crops and markets.

The potential for efficiency advantages from more precise tractor control as well as the time-consuming, systematic nature of machine operations are two factors driving the relatively advanced state of autonomous vehicle technology in agriculture (Bechar and Vigneault, 2016). With automation, manufacturers and farmers want to address many issues. The majority of products on the market are large-scale sizes since manufacturers desire to produce their goods in enormous quantities (Baillie et al., 2018). However, small-scale farmers, particularly those in developing nations, require automated small-scale machinery. Currently, sensor and sensing technology has improved rapidly, and low-cost high-precision sensors are available on the market. In farm machinery operations, farmers can utilize sensors in various types of applications. There are machine control applications, crop and soil monitoring applications, seeding, and harvesting applications. Sensor utilization in the agricultural vehicle automation system is shown in Fig. 15.9.

Particularly in developing countries, most farmers work on small plots of land, and they rely heavily on automation development in small-scale farming equipment. Commonly, they utilized small-scale farming equipment such as power tillers and garden tractors. Power tillers are among the machines that are commonly employed for both land preparation and prime moving tasks such as transport, threshing, and water lifting in many developing countries.

Small-scale farmers, especially in developing countries, face challenges in transforming from conventional farming to smart farming. They do not have experience in assessing components, device design, sensor calibrations, and expertise in the case of data transmission. Most of the farmers are in poor connectivity environments with high transmission costs, making it difficult to obtain equal data access and processing measurements for feedback in data storage and analysis. Furthermore, there is a lack of feedback and implementation: there is a lack of assessment in
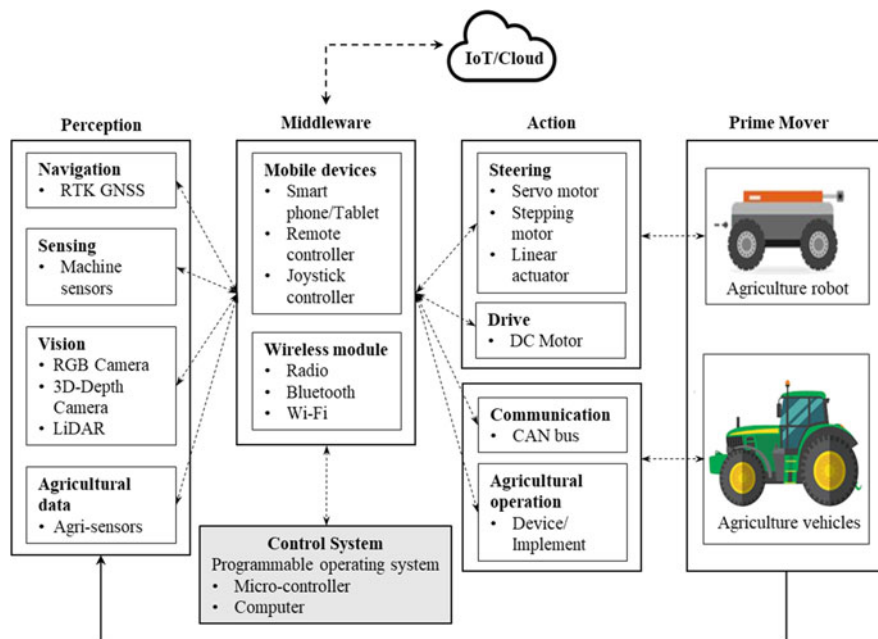
Fig. 15.9 Agricultural vehicle automation system

smartphones, remote locations with mobile devices, and IoT generation of their countries' situations.

15.5 Conclusion

The development of the automation of farm machinery for small-scale farmers is very important because the agriculture sector faces problems with cultivation areas, increasing populations, and addressing the challenges of climate change and labor shortages. Therefore, the sustainability of automation related to farm machinery and its level of implication is a prominent point for increasing productivity. Most manufacturers are intended not only to develop the automation of agriculture but also to practice sustainable agriculture. Automation machinery has advantages and can outline through Level 0–5 mechanization stages for transformation. Additionally, future automation has the chance to shift manual labor that currently necessitates a large number of workers in different production field and food industries. Automating farm equipment relieves laborious tasks and paves the way for more sustainable food production.

In conclusion, developing countries should start transforming automation on their small-scale farm machinery to Level 2 because most of the developing countries are still at Level 0 and/or Level 1. Transforming the automation of their existing

small-scale farm machinery will increase the timeliness and efficiency of their farming operations. Furthermore, transforming automation enhances the benefits: increased productivity, reduced injury, time and cost savings, solved labor shortage problems, and improved sustainability and food safety. The transformation of low-cost automation systems on small-scale farm machinery can significantly ensure SDG goal-2: end hunger, achieve food security and improve nutrition and promote sustainable agriculture, goal-9: build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation, and goal-10: reduce inequality within and among countries.

References

Ahamed, T., Takigawa, T., Noguchi, R., & Tian, L. (2014). *Bioproduction engineering: A road map for sustainable agricultural practices*. Nova Science Publishers. ISBN: 978-1-62618-122-9.

Antille, D. L., Gallar, L., Miller, P. C., & Godwin, R. J. (2015). An investigation into the fertilizerparticle dynamics off-the-disc. *Applied Engineering in Agriculture, 31*(1), 49–60. https://doi.org/10.13031/aea.31.10729

Aryal, J. P., Thapa, G., & Simtowe, F. (2021). Mechanisation of small-scale farms in South Asia: Empirical evidence derived from farm households survey. *Technology in Society, 65*, 101591. https://doi.org/10.1016/j.techsoc.2021.101591

Autospraysystems. (n.d.). *R150 robot*. Retrieved from https://autospraysystems.com/robot-1

Baillie, C., et al. (2017). *Developments in autonomous tractors*. Retrieved from https://grdc.com.au/resources-and-publications /grdc-update-papers/tab-content/grdc-update-papers/2017/07/developments-in-autonomous-tractors

Baillie, C., et al. (2018). A review of the state of the art in agricultural automation. A review of the state of the art in agricultural automation. Part III: Agricultural machinery navigation systems. In *2018 ASABE annual international meeting (p. 1)*. American society of agricultural and biological engineers.

Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations: Concepts and components. *Biosystems Engineering, 149*, 94–111. https://doi.org/10.1016/j.biosystemseng.2016.06.014

Burks, T., Villegas, F., Hannan, M., Flood, S., Sivaraman, B., Subramanian, V., & Sikes, J. (2005). Engineering and horticultural aspects of robotic fruit harvesting: Opportunities and constraints. *HortTechnology, 15*(1), 79–87. https://doi.org/10.21273/HORTTECH.15.1.0079

Deere, J. (n.d.). *Future of farming*. Retrieved from John Deere: https://www.deere.co.uk/en/agriculture/future-of-farming/

EC (2017). *EU agricultural outlook for the agricultural markets and income 2017–2030*. European commission. Retrieved from https://agriculture.ec.europa.eu/data-and-analysis/markets/outlook/medium- term_en

FAO. (2017). Retrieved from https://www.fao.org/sustainable-agricultural-mechanization/overview/why-mechanization-isimportant/en/

Fendt Xaver. (2020). *Latest generation of seed sowing robots: The Fendt Saver comes of age*. Retrieved from https://www.fendt.com/int/2-fendt-xaver

Goense, D. (2005). The economics of autonomous vehicles in agriculture. In *2005 ASAE annual meeting* (p. 1). American society of agricultural and biological engineers.

Harmanda, T. T., Hazim, M., Akbar, A. R., Wahjuni, S., & Priandana, K. (2019, July). Development of a seed-planter wheeled robot prototype. In IOP conference series: Earth and environmental science (Vol. 299, No. 1, p. 012055). IOP Publishing.

Hayami Y., & Ruttan, V. (1971). Induced innovation in agricultural development (Vol. Discussion Paper No.3). Minneapolis, Minnesota 55455. Retrieved from https://conservancy.umn.edu/bitstream/handle/11299/54243/1971-03.pdf;sequence=1

Oitzman, M. (2022, April 22). *John Deere forms joint venture with GUSS Automation*. Retrieved from The Robot Report: https://www.therobotreport.com/john-deere-forms-joint-venture-with-guss-automation/

Rinnan, A., Sigmond, M. E., Robertsen, A., & Gundersen, N. (2009). *Qualification of a Hybrid GNSS and IMU Solution*. Kongsberg Seatex AS, Trondheim, Norway, DP Conference Houston, p. 19. Retrieved from http://dynamic-positioning.com/proceedings/dp2009/sensors_rinnan.pdf

SAE. (2021). SAE levels of driving automation™ refined for clarity and international audience. Society of automatics engineers international. Retrieved from https://www.sae.org/blog/sae-j3016-update

Schueller, J. K. (2014). Engineering advancements. In *automation:The future of weed control in cropping systems* (pp. 35–49). Springer. https://doi.org/10.1007/978-94-007-7512-1_3

Sims, B., & Kienzle, J. (2017). Sustainable agricultural mechanization for smallholders: what is it and how can we implement it? *Agriculture, 7*(6), 50. https://doi.org/10.3390/agriculture7060050

Statistics Bureau of Japan. (2016). *Japan Statistical Yearbook*. Retrieved from https://www.stat.go.jp/english/data/nenkan/65nenkan/index.html

Tassell, L. V. (2022, July 6). *Agricultural Economics - University of Nebraska–Lincoln*. Retrieved from University of Nebraska–Lincoln. https://agecon.unl.edu/2022-07-06%20Cornhusker%20Econ%20-%20Van%20Tassell%20-%203.pdf

Villette, S., Piron, E., & Miclet, D. (2017). Hybrid centrifugal spreading model to study the fertiliser spatial distribution and its assessment using the transverse coefficient of variation. *Computers and Electronics in Agriculture, 137*, 115–129. https://doi.org/10.1016/j.compag.2017.03.023

Chapter 16
Vision-Based Leader Vehicle Trajectory Tracking for Multiple Agricultural Vehicles

Linhuan Zhang, Tofael Ahamed, Yan Zhang, Pengbo Gao, and Tomohiro Takigawa

Abstract The aim of this study was to design a navigation system composed of a human-controlled leader vehicle and a follower vehicle. The follower vehicle automatically tracks the leader vehicle. With such a system, a human driver can control two vehicles efficiently in agricultural operations. The tracking system was developed for the leader and the follower vehicle, and control of the follower was performed using a camera vision system. A stable and accurate monocular vision-based sensing system was designed, consisting of a camera and rectangular markers. Noise in the data acquisition was reduced by using the least-squares method. A feedback control algorithm was used to allow the follower vehicle to track the trajectory of the leader vehicle. A proportional–integral–derivative (PID) controller was introduced to maintain the required distance between the leader and the follower vehicle. Field experiments were conducted to evaluate the sensing and tracking performances of the leader–follower system while the leader vehicle was driven at an average speed of 0.3 m/s. In the case of linear trajectory tracking, the RMS errors were 6.5, 8.9, and 16.4 cm for straight, turning, and zigzag paths, respectively. Again, for parallel trajectory tracking, the root mean square (RMS) errors were found to be 7.1, 14.6, and 14.0 cm for straight, turning and zigzag paths, respectively. The navigation performances indicated that the autonomous follower vehicle was able to follow the leader vehicle, and the tracking accuracy was found to be satisfactory. Therefore, the developed leader–follower system can be implemented for the harvesting of grains, using a combine as the leader and an unloader as the autonomous follower vehicle.

Keywords Multiple vehicles · Monocular vision · Quadratic curve fitting · Trajectory tracking

L. Zhang · Y. Zhang · P. Gao
Graduate School of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan

T. Ahamed (✉) · T. Takigawa
Faculty of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

16.1   Introduction

Multiple autonomous vehicles can improve the efficiency of agricultural operations by performing labor-intensive tasks such as transporting, plowing, sowing, fertilizing, spraying, and harvesting (Iida et al., 1999; Noguchi & Barawid Jr, 2011). The simultaneous control of multiple robotic vehicles has received attention from several researchers. For example, multiple moss-harvesting robotic tractors were commanded and monitored by a human driver, who also functioned as the leader (Johnson et al., 2009). To enable navigation under complex road conditions, an autonomous follower tractor could change formation with the human-driven leader tractor to avoid obstacles based on commands from the leader (Zhang et al., 2010). The FOLLOW and GOTO algorithms were developed to control multiple vehicles in a flexible way, both in formation and independently (Noguchi et al., 2004). When considering the farming task style, a common operational method of multiple autonomous vehicles should be effective when an autonomous or a human-driven leader vehicle can lead one or more follower vehicles. Furthermore, following a trajectory identical or parallel to that of the leader is important in outdoor farm conditions. For example, while driving along a narrow road with obstacles, followers can adopt an in-line formation with the leader for safety, whereas while working on farmland, followers in a parallel formation with the leader could follow trajectories parallel to that of the leader, allowing the farming task to be performed without overlap or missed areas. For such a target, integrity model involving leader motion information, such as steering and the speed of the leader, could allow for precise tracking (Morin et al., 2008; Ou et al., 2013; Peng et al., 2014). However, the transmission of the leader motion information to the follower through a wireless device creates the risk of wireless distribution or failure. Aiming to solve this problem, the leader's position and velocity were estimated based on local sensors (Kise et al., 2004), and a neural network (NN)-based extended Kalman filter (EKF) was designed to estimate leader speed and accommodate modeling errors (Johnson et al., 2004). By cooperating with GPS location, a time-delayed leader tracking model was established and showed excellent tracking performance (Goi et al., 2010).

In any event, the follower vehicle needs to continually update its relative position with respect to the leader to fulfill the tracking task. Regarding safety, absolute sensors such as those employing GPS are not suitable for the tracking task because they may lose the satellite signal and are subject to multipath interference. Local sensors, such as cameras and laser range finders (LRF), are considered to be better approaches and have been successfully applied for tracking under both indoor and outdoor conditions (Abe et al., 2005; Ahamed et al., 2006, 2009; Espinosa et al., 2011). Compared with LRFs, camera vision can provide more information than data obtained via LRF scanning with less cost and has thus been wildly utilized for navigation, mapping, and tracking (Caballero et al., 2009; Courbon et al., 2010; Han et al., 2004). For tracking control of multiple robotic vehicles, a camera vision-based leader–follower relative position estimating method has been designed, recognizing a leader vehicle using features of the leader vehicle (Hasegawa et al., 2000; Kannan

et al., 2011). However, this method was sensitive to lighting conditions and was distance-limited as well as time-intensive. A common and effective method for solving those problems was to use an artificial marker to identify the leader and estimate the leader–follower relative position using pre-known geometry or color information of the markers (Krajník et al., 2014). The advantages of the marker-based method are that it could support stable recognition, accurate position estimation, and fast calculation. Moreover, it ensures the tracking accuracy and tracking safety for a leader–follower trajectory tracking system. In this research, to avoid using GPS and wireless devices, the designed control law of the follower vehicle for leader trajectory tracking only relied on the relative leader–follower position, which was obtained from the camera vision. This study aimed to develop a vision-based feedback controller designed to track a leader vehicle trajectory while maintaining an in-line or parallel formation. Thus, the objectives of this research were as follows:

1. To establish an autonomous vehicle as a follower vehicle able to conduct tracking tasks.
2. To construct a robust and accurate monocular vision system able to estimate the relative position between a leader and a follower.
3. To develop a control algorithm able to realize accurate leader vehicle trajectory-tracking for multiple agricultural machinery combinations, with a human-driven leader and an autonomous follower.

16.2   Materials and Methods

An electronic vehicle (CHIKUSUI EJ-20, CANYCOM, Tokyo, Japan) was modified into an autonomous vehicle, i.e., the follower vehicle. Both the leader and the follower vehicle had a 60 cm wheel base length and 49 cm drawbar length. Major subsystems of the autonomous vehicle included steering control, speed control, power, and sensor units. The sensory data and control status were transmitted to an upper level controller through parallel communication. The basic instrumentation system for the autonomous vehicle is depicted in Fig. 16.1a. A Pro 9000 Web camera (Logitech, Lausanne, Switzerland) with two million pixels and a 70 ˝ view angle, was mounted on the autonomous follower vehicle to provide vision information. A LMS 511 LRF (SICK, Waldkirch, Germany) was utilized as an assist device to provide reference data and recorded trajectories during tracking. Steering control of the autonomous vehicle was conducted using an electronic cylinder (LPF040L2.0VK2J, TSUBAKIMOTO CHAIN, Osaka, Japan). The length of the piston rod was 200 mm and the maximum speed was 40 mm/s. It could provide stable thrust power up to 400 N. Figure 16.1b shows components of the robot vehicle.
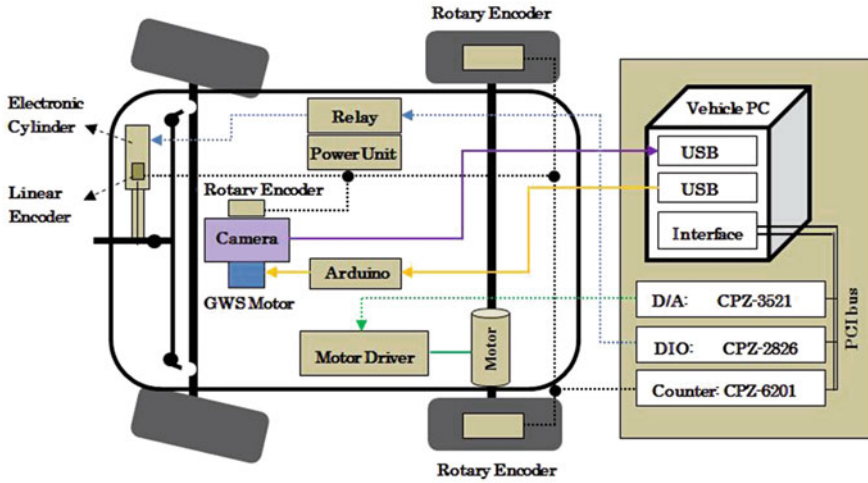
(a)
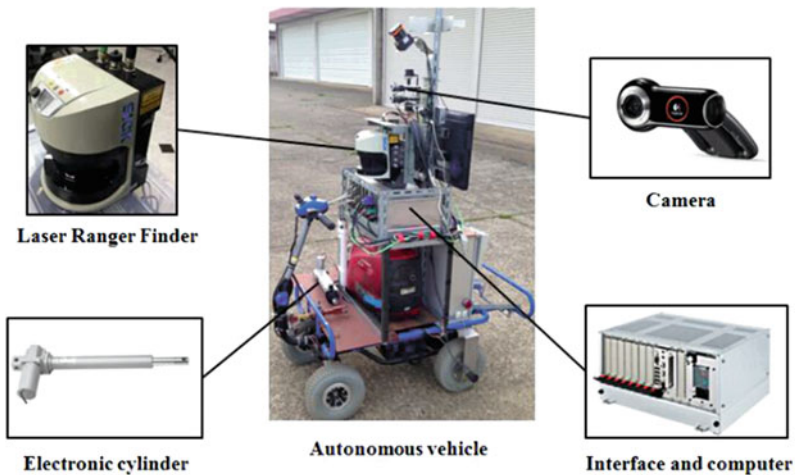
(b)

Fig. 16.1 The autonomous follower in the leader–follower system. (**a**) Sensors arrangements in the autonomous unit; (**b**) Hardware components of the autonomous follower tracking system

16.2.1 Leader–Follower Relative Position and Camera–Marker Sensing System

Figure 16.2a describes the relative position between the leader and follower. By identifying the relative heading angle β, relative distance D, and orientation angle α

(a)

(b)

(c)

Fig. 16.2 Geometrical disposition between the leader and the follower (**a**) Leader–follower relative position. (**b**) Relative position between camera and marker plane. (**c**) Servo motor implemented with the camera–marker system

of the leader relative to the follower, the follower vehicle could identify the leader position. The leader–follower relative position was obtained from the camera–marker system (Fig. 16.2b), in which the camera was mounted on the rear wheel center point P_1 of the follower vehicle, the mar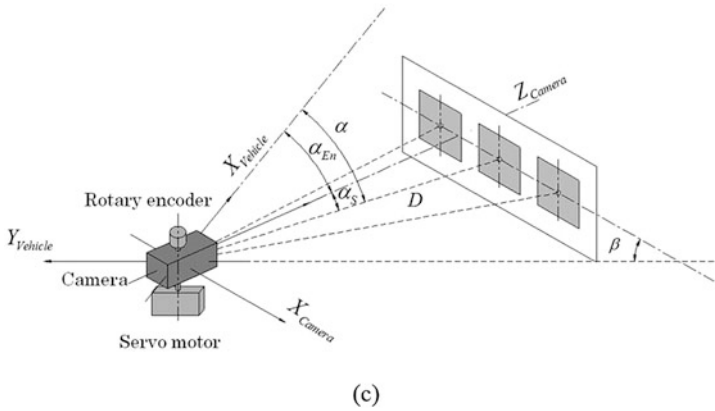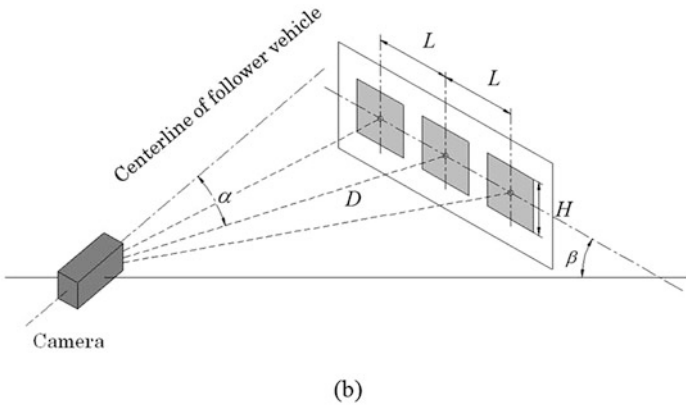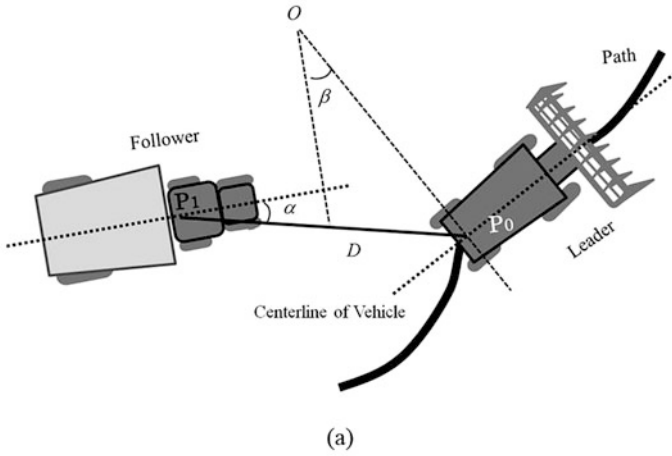ker was installed perpendicular to the centerline of the leader vehicle and the position of the middle square of the marker was at the rear wheel center point P_0. The side length of each square H and the interval between square centers L were 0.2 and 0.4 m, respectively. In the leader–follower system, the following steps were followed to develop the relative positioning system: camera servo systems, marker detection, marker positioning, and estimation of offset of the roll angle between camera and marker.

16.2.1.1 Camera Servo System

Losing the target was a severe problem during the tracking of the leader vehicle; it potentially occurred owing to the limitations of the camera view field, especially on a large-curvature path. To overcome this problem, a camera servo system was designed to keep the marker in the center of the camera view field. The camera servo system comprised a GWS servo motor and a rotary encoder with a camera (Fig. 16.2c). By responding to the angle α_s from the middle square center to the camera optical axis, the servo motor could rotate the camera directly to the marker center. The rotation angle α_{En} between the optical axis and the centerline of the follower vehicle could be monitored by a rotary encoder installed above the camera. The existing follow relationship can be expressed as:

$$\alpha = \alpha_s + \alpha_{En} \tag{16.1}$$

16.2.1.2 Marker Detection

The marker was detected based on its pre-known geometry information, including its square shape features and relative spatial relationship between squares in the marker plane. The image processing flow comprises the following four steps: transforming an original RGB image into a grayscale image and then enhancing the contrast ratio, extracting contours, finding rectangles from the contour images, and determining the marker (Fig. 16.3).

Pre-known geometry information could reduce the computational cost and benefit real-time detection. Additionally, the high contrast ratio between the black squares and the white background enabled the generation of acutance contours and created stability for the detection of the marker. However, contour extraction was still influenced by illumination conditions. Low illumination conditions or strong sunlight under an outdoor environment would reduce the contrast ratio of the image and corrode the contour of squares, causing failure of marker detection. To expand the scope to adapt to various illumination conditions, a commonly used normal

Fig. 16.3 Image processing for marker detection. (**a**) Contour image; (**b**) Detected marker

distribution of the image histogram method was utilized to enhance image contrast. Affected by posture changes of the vehicles, squares projected on the image plane would show the shapes of rectangles. Thus, rectangles were recognized and selected in the contour image. Relying on the relative spatial relationship between the three squares, false targets with rectangular shapes, such as rooms and windows, could be filtered, and only squares formed by the marker could be extracted.

16.2.1.3   Marker Positioning

Given that the vision data were obtained from a single camera and the relative position between the marker and the camera was estimated based on the known side length of the marker squares, the position of each square in the marker plane could be described by its center point. The pitch angle of the vehicle body was neglected, meaning that the sides of squares in the vertical direction would not be affected by the posture changes of the leader and follower vehicles when projected onto the image plane. For this reason, the centerline of the squares in the vertical direction could be used to estimate the relative position between the camera and marker plane. Utilizing the geometric relationship between similar triangles under a perspective model (Fig. 16.4), the position of the square center P_C in camera-based coordinates could be estimated as follows:

$$X_C = \frac{x - c_x}{f_x} Z_C \tag{16.2}$$

$$Z_C = \frac{H}{h} f \tag{16.3}$$

$$\alpha = \text{arc tan} \left(\frac{X_c}{Z_c} \right) \tag{16.4}$$
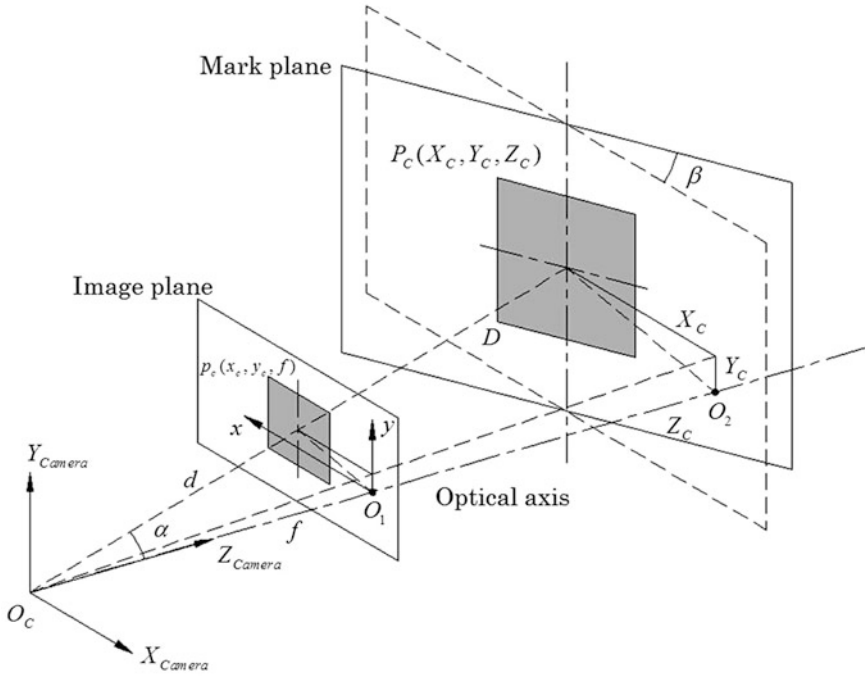
Fig. 16.4 Camera perspective model

where the coordinates of the square center under the image coordinate and camera-based coordinate systems could be written as $p_c(x_c, y_c, f)$ and $P_C(X_C, Y_C, Z_C)$, respectively. f and f_x represent the focal length, and c_x is the shift of the optical axis obtained from camera calibration; h is the height of squares in the image plane.

16.2.1.4   Offset of Roll Angle between Camera and Marker

On uneven farm ground, rolling of the camera or the marker plane would occur and affect the leader–follower relative position observation accuracy. The calculation of the leader–follower relative position should offset the rolling effect of the camera or the marker plane. For example, suppose the leader vehicle is driven on a horizontal surface, while the follower vehicle forms a roll angle γ around its optical axis from the horizontal surface (Fig. 16.5). $P_{CN}(X_{CN}, Y_{CN}, Z_{CN})$ are the coordinates of square centers based on the camera coordinate system and $P_{HN}(X_{HN}, Y_{HN}, Z_{HN})$ are the coordinates of square centers with respect to the horizontal surface (Fig. 16.5). Clearly, the position of P_{CN} represents the relative position between the camera and the marker plane, and the position P_{HN} represents the relative position between the follower and the leader vehicles. Thus, the relationship between P_{CN} and P_{HN} could be written as
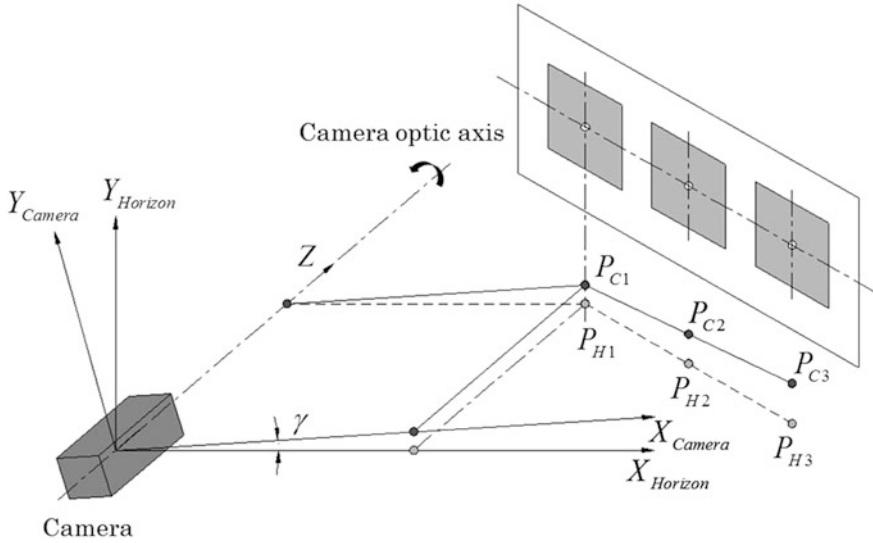
Fig. 16.5 Model for offsetting vehicle roll effect

$$X_{CN} = X_{HN} \cos \gamma - Y_{HN} \sin \gamma \tag{16.5}$$

$$Z_{CN} = X_{HN} \tag{16.6}$$

Because the relative position between the leader and the follower vehicle only corresponds to the X-Z coordinates, the square centers can be assumed to lie on the horizontal surface. Then, Eq. (16.5) can be rewritten as

$$X_{HN} = \frac{X_{CN}}{\cos \gamma} \tag{16.7}$$

$$\gamma = \tan^{-1} \left(\frac{3 \sum_{n=1}^{3} x_{cn} y_{cn} - \left(\sum_{n=1}^{3} x_{cn} \right) \left(\sum_{n=1}^{3} y_{cn} \right)}{3 \sum_{n=1}^{3} x^2{}_{cn} - \left(\sum_{n=1}^{3} x^2{}_{cn} \right)^2} \right) \tag{16.8}$$

where (x_{cn}, y_{cn}) represent coordinates of the square centers $p_c(x_c, y_c, f)$ in the plane.

16.2.1.5   Transformation of Coordinates and Relative Positioning of the Marker

The transformation of coordinates between the camera and the follower vehicle could be expressed as follows (Fig. 16.2c):

$$X_{VN} = \frac{X_{CN}}{cos\gamma} \sin \alpha_{En} + Z_{CN} \cos \alpha_{En} \qquad (16.9)$$

$$Y_{VN} = \frac{X_{CN}}{cos\gamma} \cos \alpha_{En} + Z_{CN} \sin \alpha_{En} \qquad (16.10)$$

where $P_{VN}(X_{VN}, Y_{VN})$ are the coordinates of the square centers in the follower-based local coordinates. The relative distance D and relative angle β between the leader and the follower vehicle could be calculated as:

$$D\sqrt{X^2_{V2} + Y^2_{V2}} \qquad (16.11)$$

$$\beta = \tan^{-1} \left(\frac{3 \sum\limits_{N=1}^{3} X_{VN} Y_{VN} - \left(\sum\limits_{N=1}^{3} X_{VN} \right) \left(\sum\limits_{N=1}^{3} Y_{VN} \right)}{3 \sum\limits_{N=1}^{3} X^2_{VN} - \left(\sum\limits_{N=1}^{3} X_{VN} \right)^2} \right) \qquad (16.12)$$

Then, the relative position between the leader and the follower vehicle could be written as

$$x_{l\_F} = X_{V2} \qquad (16.13)$$

$$y_{l\_F} = Y_{V2} \qquad (16.14)$$

$$\theta_{l\_F} = \beta \qquad (16.15)$$

where x_{l_F}, y_{l_F} represents the local position of the leader based on the follower and θ_{l_F} is the local heading angle of the leader based on the follower.

16.2.2   Camera Vision Data Estimation and Smoothing

Limited by the monocular vision method, the observed leader–follower relative position was noisy under the worst farm conditions. In some cases, large observed errors would occur or there was even a failure to detect the marker plane. The estimation and smoothing of the observation data were necessary to ensure the accurate tracking of the leader vehicle and also to improve the motion stability of the follower vehicle. Because the motion of the two vehicles was continuous, the variation of relative distance and angle between the leader and the follower vehicle was also continuous. The commonly used method of least-squares was introduced to estimate and smooth the relative distance D and the relative heading angle β between the leader and the follower vehicle by fitting a quadratic curve separately. During the process of data estimation and smoothing, estimated data could be obtained by fitting

the stored latest n points of observation data to a quadratic curve using the least-squares method. In this study, the quadratic curve could be written as:

$$q(n) = an^2 + bn + c \tag{16.16}$$

where n denotes observation times used to store and fit the data and $q(n)$ is the vector of the stored observation data sequence, including the relative distance and the relative heading angle. $q(n)$ is defined as

$$q(n) = \begin{pmatrix} D(n) \\ \beta(n) \end{pmatrix} \tag{16.17}$$

To ensure the fitting effect, avoid collapse of the least-squares method and maintain the original transfer tendency of the leader–follower relative position, the data stored for fitting required appropriate handling. The estimation and smoothing process was realized through two steps: first, once a new camera observation was available, the sequence of the stored observation data would be updated and the latest stored data after updating was temporarily determined as follows:

$$q(i) = q(i + 1) \; i\epsilon(0, 1 \ldots .n - 1) \tag{16.18}$$

$$q(n) = \begin{cases} q_{C\_obs} & q_{E\_1} > q_{Th} \\ q(n-1) & q_{E\_1} < q_{Th} \end{cases} \tag{16.19}$$

$$q_{E\_1} = |q(n - 1) - q_{C\_obs}| \tag{16.20}$$

Second, after fitting to the quadratic curve using the least-squares method, the latest stored data and the current leader–follower relative position could be determined as follows:

$$q(n) = \begin{cases} q_{C\_obs} & q_{E\_2} > q_{Th} \\ q_{Fit} & q_{E\_2} < q_{Th} \end{cases} \tag{16.21}$$

$$q_{E\_2} = |q_{C\_obs} - q_{Fit}| \tag{16.22}$$

$$q_{Est} = q(n) \tag{16.23}$$

Where q_{C_obs} is the vector of the current camera observed data, q_{Fit} is the vector of the fitted current relative distance and relative angle using the stored n times of observation data, q_{Est} is the vector defining the current relative distance and relative angle, $q(i)$ is the vector of the stored i_{th} observation, q_{E_1} is the vector of the distance between the current observation and last observation, q_{E_2} is the vector of the distance between the current observation and fitted observation, and q_{Th} is the vector of the threshold values, set as (1 m 40°).

16.2.3   Design of Control Law for the Leader Trajectory Tracking of Follower Vehicle

In this study, only the leader–follower relative position information was used by the follower vehicle to track the leader trajectory. The absence of information exchange and absolute reference positions made the leader trajectory thoroughly uncertain for the follower vehicle, and the tracking position for the follower vehicle was ambiguous. A feedback control method based on the leader–follower relative position was proposed to track the trajectory of the leader.

As described in Fig. 16.6, the required position of the follower vehicle is set at P_2, with a distance d_{01} from the leader vehicle rear axis and an angle Φ_{01} with the leader vehicle rear axis. Assuming the leader vehicle is driven with a straight trajectory, the position of P_2 in the leader-based local coordinates could be written as:
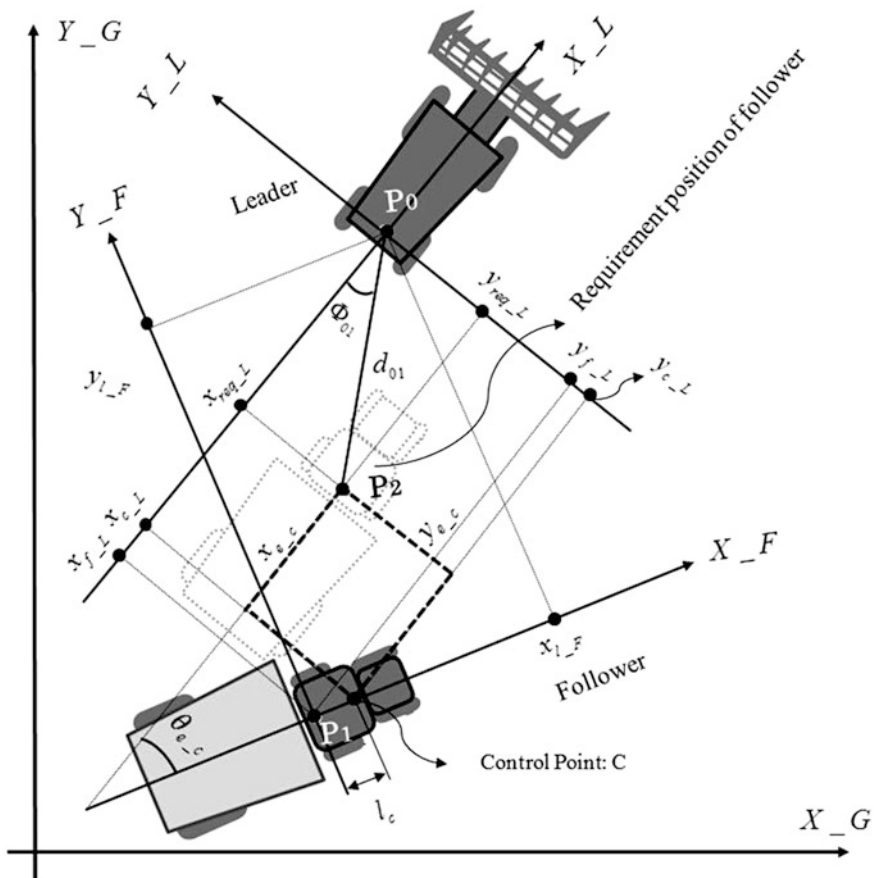
Fig. 16.6 Relationship and coordinate transformation between the leader and the follower vehicles

$$\begin{bmatrix} x_{req\_L} \\ y_{req\_L} \\ \theta_{req\_L} \end{bmatrix} = \begin{bmatrix} d_{01} \cos \Phi_{01} \\ d_{01} \sin \Phi_{01} \\ 0 \end{bmatrix} \tag{16.24}$$

To improve the control freedom of the follower vehicle to realize the tracking of the uncertain leader vehicle trajectory, a control point C, located on the centerline of the follower vehicle, was introduced. Moreover, the distance from the rear wheel axial center to the control point C was defined as

$$l_C = k_0 l \tag{16.25}$$

where l the length from the front wheel axial center to the rear wheel axial center; l_C is the length from the rear wheel axial center to the control point C; the parameter k_0 is used to determine the location of control point C. The position of the control point C in leader-based local coordinates could be written as:

$$\begin{bmatrix} x_{c\_L} \\ y_{c\_L} \\ \theta_{c\_L} \end{bmatrix} = \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{l\_F} \\ y_{l\_F} \\ \theta_{l\_F} \end{bmatrix} + \begin{bmatrix} l_c \cos \beta \\ -l_c \sin \beta \\ 0 \end{bmatrix} \tag{16.26}$$

Combining with Eqs. (16.13), (16.14), (16.15), (16.24), (16.25), and (16.26), the control point C-based position tracking error between the follower vehicle and its requirement position could be calculated as:

$$\begin{bmatrix} x_{e\_c} \\ y_{e\_c} \\ \theta_{e\_c} \end{bmatrix} = \begin{bmatrix} x_{c\_L} \\ y_{c\_L} \\ \theta_{c\_L} \end{bmatrix} - \begin{bmatrix} x_{reqL} \\ y_{reqL} \\ \theta_{reqL} \end{bmatrix} = - \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{V2} \\ Y_{V2} \\ \beta \end{bmatrix}$$
$$+ \begin{bmatrix} l_c \cos \beta \\ -l_c \sin \beta \\ 0 \end{bmatrix} - \begin{bmatrix} d_{01} \cos \Phi_{01} \\ d_{01} \sin \Phi_{01} \\ 0 \end{bmatrix} \tag{16.27}$$

A simple steering strategy for responding to longitudinal and heading tracking error is given as

$$\delta = k_1 y_{e\_c} + k_2 \theta_{e\_c} + k_3 \sin \left(\theta_{e\_c} \right) \tag{16.28}$$

A PID controller was designed to maintain the required distance between the leader and the follower vehicle; control of the follower velocity could be given as

$$v_t = v_{t-1} + k_D (e_t - e_{t-1}) + k_I e_t + k_p (e_t - 2e_{t-1} + e_{t-2}) \tag{16.29}$$

$$e = (D - d_{01}) \tag{16.30}$$

where k_1, k_2, k_3 are control parameters corresponding to required distance d_{01} and angle Φ_{01}, k_D, k_I, k_p are parameter of the PID controller adjusted during field

experiments. Notice that once the required values of distance d_{01} and angle Φ_{01} were altered, the control parameters also needed to be adjusted.

16.3 Field Experiments

Experiments for verifying the stability and accuracy of the camera–marker sensing system and leader trajectory tracking accuracy were conducted at the Agricultural and Forestry Research Center, University of Tsukuba (Ibaraki, Japan). The camera–marker sensing system evaluation experiments included both a static and a dynamic evaluation experiment. The static evaluation experiment was intended to verify the stability and accuracy of the designed observation method and optimize the camera coefficients. The dynamic evaluation experiment was designed to determine the threshold values for data estimation and smoothing, analyze the observation stability and accuracy, and verify the effectiveness of the least-squares method-based data estimation and smoothing solution. A SICK LMS 511 LRF was used to provide reference data, and the relative position from the LRF to the marker plane was used as reference data to evaluate the camera observation accuracy (Fig. 16.7a).

In the tracking accuracy evaluation experiments, linear and parallel tracking experiments were conducted on straight, turning, and zigzag paths. Cylindrical markers were mounted above the rear wheel centers of the leader and follower vehicles (Fig. 16.7b) to facilitate the LMS 511 LRF in recording their trajectories at a frequency of 25 Hz. The leader vehicle was driven at a velocity of 0.3 m/s. The required distance d01 between the leader and the follower vehicle was 4 m in linear tracking. In parallel tracking, the required lateral and longitudinal offsets of the follower vehicle were set at 4 m and 2 m from the leader vehicle so that the trajectory of the follower vehicle could parallel that of the leader vehicle at a 2 m interval.

(a)   (b)

Fig. 16.7 Field experiments of the leader–follower system. (**a**) Evaluation of the camera–marker system; (**b**) tracking of a trajectory of the leader vehicle

16.4   Results and Discussion

16.4.1   Evaluation of Camera–Marker Observation System

In the static evaluation experiment, the maximum distance from the camera to the marker was approximately 6 m and the relative angle that formed between the marker and the camera axis ranged from −40° to 40° (Fig. 16.8). Using the LRF data as reference, the accuracy of the leader–follower relative position obtained from the camera–marker system could be evaluated (Fig. 16.9). Linear regression analysis showed that the orientation angle and distance between the leader and follower vehicles obtained from the camera–marker system were stable and had high accuracy (Fig. 16.9a, e). Meanwhile, the leader–follower relative angle obtained from the camera–marker system was unstable (Fig. 16.9c). Compared with the accuracy of the orientation angle (Fig. 16.9f), the accuracy of the distance and relative angle obtained from the camera–marker system degraded as the relative distance from the camera to the marker increased (Fig. 16.9b, d). This phenomenon was mainly caused by the limitation of the camera; the pitch angle of the vehicle also potentially caused an observation error of the uneven ground.

The RMS errors of the leader–follower relative distance, relative angle, and orientation angle observation were calculated. When the distance between the camera and the marker was 6 m, the RMS errors of the leader–follower relative distance, relative angle and orientation angle observation were 5.8 cm, 5.07°, and 0.228°, respectively. At 4 m, the RMS errors of the leader–follower relative distance, relative angle, and orientation angle observation were 3.63 cm, 3.01°, and 0.239°, respectively. Considering that the orientation angle obtained from the camera–marker system was stable and had high accuracy, data estimation and smoothing was only conducted for the distance and relative angle observed. In the dynamic evaluation experiment, the leader vehicle was driven along a zigzag path and the follower vehicle was controlled in remote mode to follow the leader. The camera
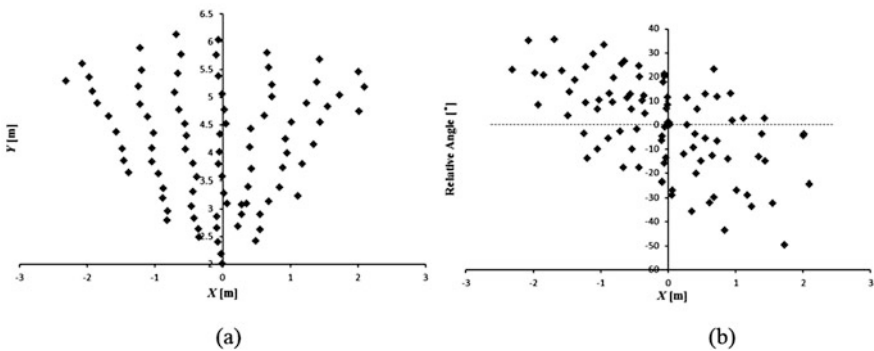
Fig. 16.8 Position of the marker. (**a**) Location of the marker. (**b**) Relative angle between the marker and the x-axis
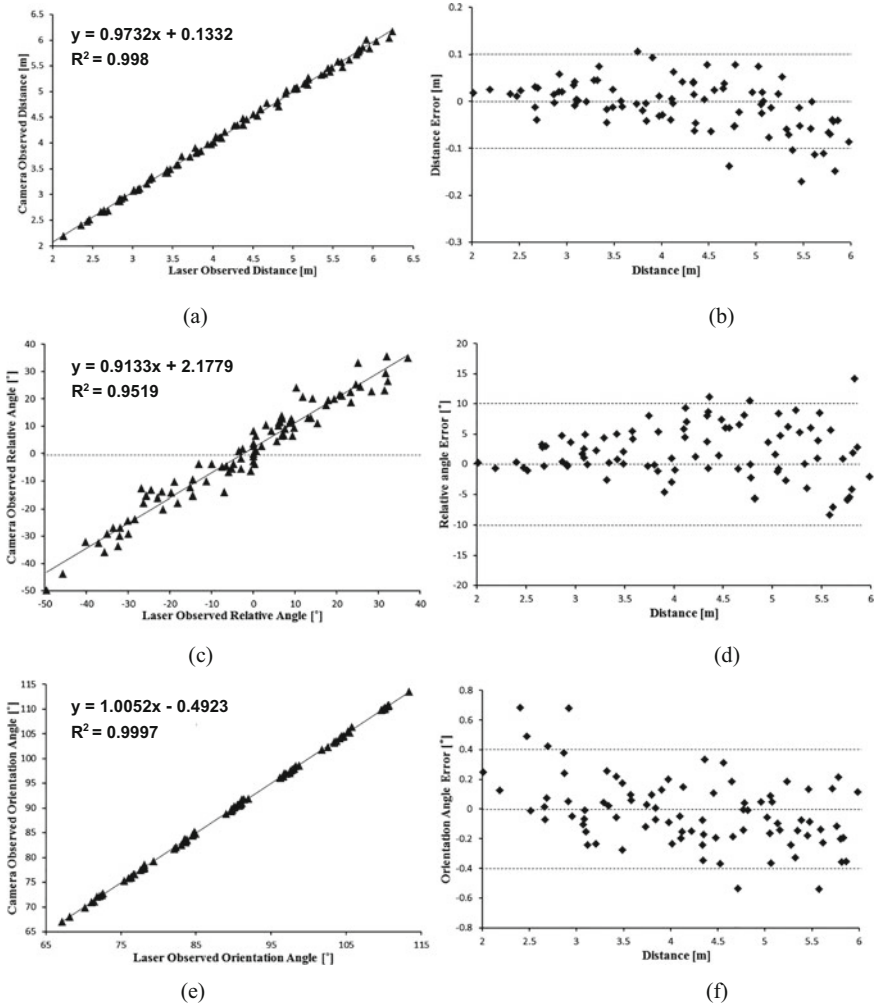
Fig. 16.9 Linear Regression and Accuracy analysis of the camera observation referenced with the laser observation. (**a**) Distance; (**b**) Relative angle; (**c**) Orientation angle; (**d**) Distance error; (**e**) Relative angle error; (**f**) Orientation angle error

observation data before estimation and smoothing, the estimated and smoothed data obtained through least-squares-based curve fitting, and the LRF observation data were recorded during driving.

The results showed that both the camera data before estimation and smoothing and the estimated and smoothed camera data closely matched the LRF data (Fig. 16.10). The RMS errors of the camera observation before estimation and smoothing were 4.7 cm and 3.15° for the relative distance and relative angle, respectively. These coincided with the results under static conditions, meaning that
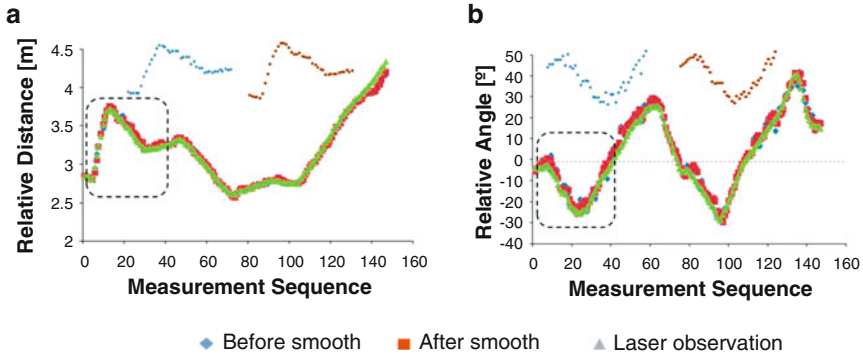
Fig. 16.10 Relative position between the camera and marker before smooth, smoothed, and LRF data. (**a**) Relative distance; (**b**) Relative angle
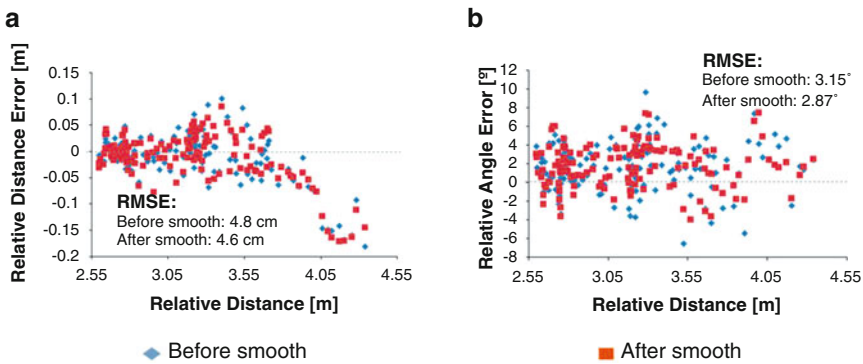
Fig. 16.11 Relative position error for camera observation. (**a**) Relative distance error; (**b**) Relative angle error

the motion of the marker and the camera had little effect on the observation accuracy. During the experiment, the camera-observed data were smoothed by fitting a curve using the least-squares method. After data estimation and smoothing, the camera observation data were observably smoothed, as shown in the dotted rectangle (Fig. 16.10). Furthermore, the accuracy of the leader–follower relative position observation was improved after data estimation and smoothing, and the RMS errors of the relative distance and relative angle were reduced to 4.6 cm and 2.87°, respectively (Fig. 16.11). Compared with the camera observation data before estimation and smoothing, the dispersion of the estimated and smoothed data were also reduced, with the standard deviations of the relative distance and relative angle reduced from 4.9 to 4.2 cm and 3.74 to 2.55°, respectively (Fig. 16.12). Those performances showed the potential for stable and accurate observation when applied to real-sized tractors, being clearly insensitive to the uneven ground and having stable motion characteristics compared with the small-sized vehicles.
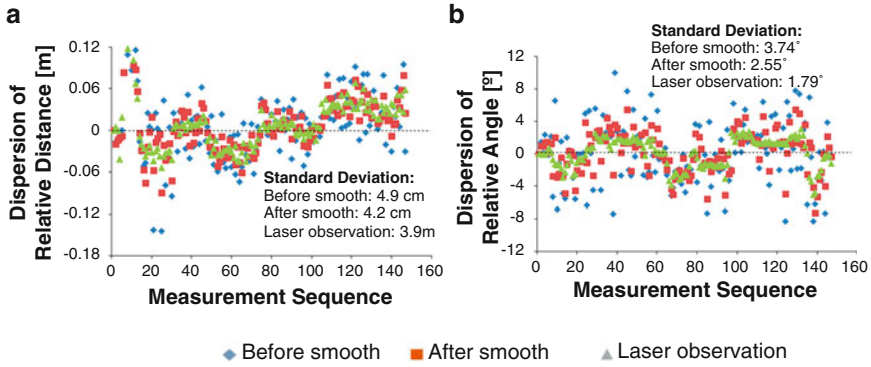
Fig. 16.12 Dispersion of camera observation data. (**a**) Dispersion of relative distance; (**b**) Dispersion of relative angle
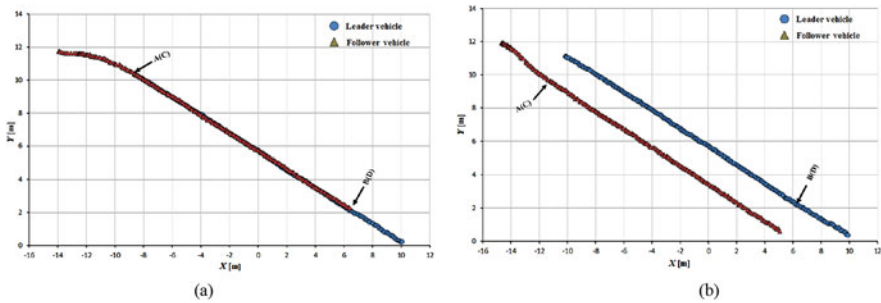
Fig. 16.13 Leader trajectory tracking on a straight path. (**a**) Linear tracking; (**b**) Parallel tracking

16.4.2 Tracking Performance

Tracking accuracy was evaluated using the interval space between the leader and follower vehicle trajectories; the trajectory segments AB and CD were used to calculate this interval space. The follower vehicle could adjust its state and arrive at its required position relative to the leader rapidly and smoothly (Figs. 16.13, 16.14 and 16.15).

The tracking error between the leader and follower vehicle trajectories is shown in Figs. 16.16, 16.17 and 16.18. During tracking on a straight path, a very low tracking error between the trajectories of the leader and follower vehicles was observed; the maximum and RMS tracking errors between these trajectories were 12.5 and 6.5 cm for linear tracking and 14.1 and 7.1 cm for parallel tracking, respectively (Fig. 16.16). During tracking on a turning path, the maximum and RMS tracking errors between the trajectories were 18.2 cm and 8.9 cm for linear and 29.0 cm and 14.6 cm for parallel tracking, respectively (Fig. 16.17). During tracking on a zigzag path, the maximum and RMS tracking errors between the trajectories were 35.0 cm and 16.4 cm for linear and 24.5 cm and 14.0 cm for
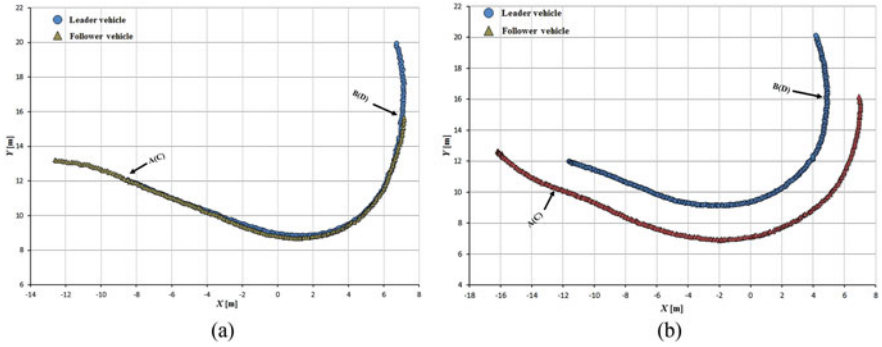
Fig. 16.14 Leader trajectory tracking on a turning path. (**a**) Linear tracking; (**b**) Parallel tracking
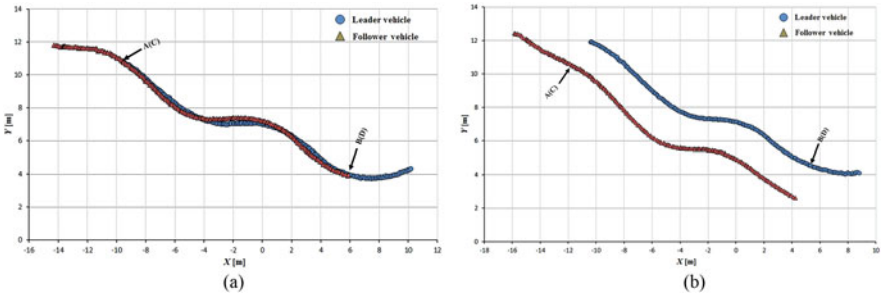
Fig. 16.15 Leader trajectory tracking on a zigzag path. (**a**) Linear tracking; (**b**) Parallel tracking
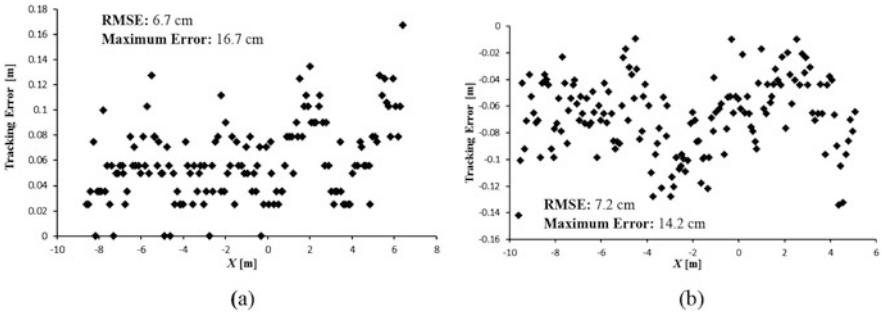
Fig. 16.16 Tracking error between leader and the follower trajectories during tracking on a straight path. (**a**) Linear tracking; (**b**) Parallel tracking

parallel tracking, respectively (Fig. 16.18). In comparison with the straight path, the turning and zigzag path tracking showed higher error. From trajectories of the leader and follower vehicles, it can be observed that a larger variation of the direction of the leader vehicle would result in a larger tracking error (Figs. 16.14 and 16.15). This error remained at a low level when the leader vehicle was driven on a
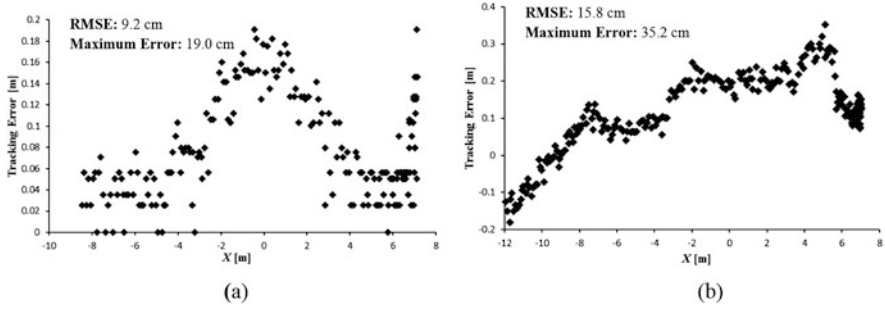
Fig. 16.17 Tracking error between the leader and the follower trajectories during tracking on a turning path. (**a**) Linear tracking; (**b**) Parallel tracking
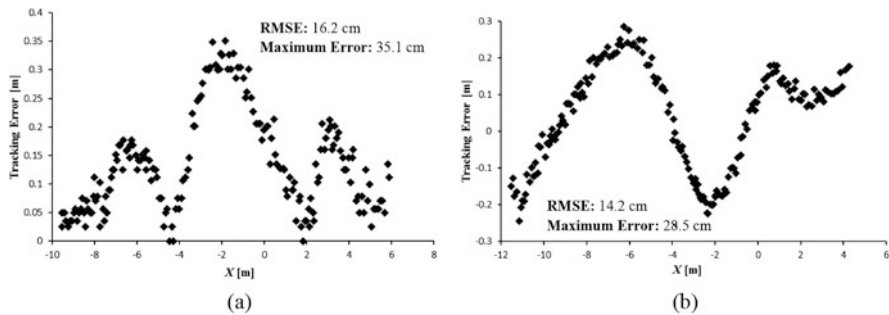
Fig. 16.18 Tracking error between the leader and the follower trajectory during tracking on a zigzag path. (**a**) Linear tracking; (**b**) Parallel tracking

constant-curvature path. Considering road space and agricultural operations, the tracking accuracy was sufficient to ensure safe tracking and precision operation.

16.5   Discussion

The driverless follower is the key of this research, which confirmed the high accuracy in following the human-driven leader and the performance of the control system. Experiments were conducted using a specially built robot as follower to confirm the accuracy of tracking and develop a control system without any built-in communication between the leader and the follower. The traveling courses were chosen according to standard agricultural operations, such as straight, turning, and zigzag paths. The tracking performance between the leader and the follower was satisfactory under regular field conditions. Undulating terrain and adverse climatic conditions were ignored in the field experiments. The experiments were conducted mostly under daytime conditions. The camera marker system was assisted with the LRF for cross checking the accuracy of the marker positions both in static and

dynamic conditions. The LRF was utilized as an assist device to provide reference data and recorded the trajectories during tracking. The contour extraction was influenced by illumination conditions. Low illumination conditions or strong sunlight under an outdoor environment would reduce the contrast ratio of the image and affect the contour of squares, causing marker detection failure. To expand the scope to adapt to various illumination conditions, the image contrast was enhanced using a histogram method that ensured stable observation under various light conditions while conducting experiments during the daytime. The vertical vehicle's movement or pitch angle of the vehicle body was not considered, as there was not much effect of posture changes of the leader and follower vehicles when projected onto the image plane while traveling on regular ground. This was one of the limitations of this research. However, to overcome such limitations, the centerline of the squares in the vertical direction was used to estimate the relative position between the camera and marker plane. The experiments were conducted with the prototype robot to confirm the accuracy and develop suitable control systems. The validation was done with a human-driven small vehicle as leader and the autonomous prototype robot as follower. Definitely in an agricultural environment an actual size autonomous unit can be used as follower by implementing the proposed camera marker sensing and control systems. The productivity would definitely be higher, by reducing labor through enabling the human-driven leader and autonomous follower system.

16.6   Conclusions

In this study, a human-driven leader and automatic follower trajectory-tracking system was developed. A low-cost camera servo system, comprising a web camera, encoder, and a servomotor, was implemented. An effective camera–marker detection method was developed to follow the leader, which was controlled by an operator. A solution for enhancing image contrast that involved using the histogram method, offsetting vehicle roll angle, and estimating and smoothing the camera observation using the least-squares method ensured a stable and accurate monocular vision system that was able to estimate the relative position between the leader and the follower vehicles with high accuracy. A feedback control rule and a PID controller were also developed and exhibited good performance for linear and parallel leader trajectory tracking. The estimation and smoothing of the camera observation data reduced camera noise and yielded relative positional information between the leader and the follower vehicle with high accuracy. As a result, a stable velocity and steering angle of the follower vehicle and high accuracy of the trajectory tracking was established. Thus, a low-cost, reliable navigation system for a leader and follower vehicle tracking system was demonstrated. In further research, the leader should be converted to a remote control unit to make it unnecessary for the operator to be on-board the leader vehicle. Additionally, to overcome the limitations of the prototype leader–follower system, such as guidance in the adverse climatic conditions are required to consider for agricultural operations.

References

Abe, G., Mizushima, A., & Noguchi, N. (2005). Study on a straight follower control algorithm based on a laser scanner. *Journal of the Japanese Society of Agricultural Machinery (Japan), 67*(3), 65–71.

Ahamed, T., Takigawa, T., Koike, M., Honma, T., Hasegawa, H., & Zhang, Q. (2006). Navigation using a laser range finder for autonomous tractor (part 1) positioning of implement. *Journal of the Japanese Society of Agricultural Machinery (Japan), 68*(1), 68–77.

Ahamed, T., Tian, L., Takigawa, T., & Zhang, Y. (2009). Development of auto-hitching navigation system for farm implements using laser range finder. *Transactions of the ASABE, 52*(5), 1793–1803. https://doi.org/10.13031/2013.29120

Caballero, F., Merino, L., Ferruz, J., & Ollero, A. J. J. (2009). Vision-based odometry and SLAM for medium and high altitude flying UAVs. *Journal of Intelligent and Robotic Systems, 54*(1), 137–161.

Courbon, J., Mezouar, Y., Guénard, N., & Martinet, P. (2010). Vision-based navigation of unmanned aerial vehicles. *Control Engineering Practice, 18*(7), 789–799.

Espinosa, F., Santos, C., Marrón-Romera, M., Pizarro, D., Valdés, F., & Dongil, J. J. S. (2011). Odometry and laser scanner fusion based on a discrete extended Kalman filter for robotic platooning guidance. *Sensors, 11*(9), 8339–8357.

Goi, H. K., Giesbrecht, J. L., Barfoot, T. D., & Francis, B. A. (2010). Vision-based autonomous convoying with constant time delay. *Journal of Field Robotics, 27*(4), 430–449.

Han, S., Zhang, Q., Ni, B., & Reid, J. (2004). A guidance directrix approach to vision-based vehicle guidance systems. *Computers and Electronics in Agriculture, 43*(3), 179–195.

Hasegawa, H., Takigawa, T., Koike, M., Yoda, A., & Sakai, N. (2000). Studies on visual recognition of an agricultural autonomous tractor detection of the field state by image processing. *Japanese Journal of Farm Work Research, 35*(3), 141–147.

Iida, M., Maekawa, T., & Umeda, M. (1999). Automatic follow-up vehicle system for agriculture (Part 1). *Journal of the Japanese Society of Agricultural Machinery, 61*(1), 99–106.

Johnson, D. A., Naffin, D. J., Puhalla, J. S., Sanchez, J., & Wellington, C. K. (2009). Development and implementation of a team of robotic tractors for autonomous peat moss harvesting. *Journal of Field Robotics, 26*(6–7), 549–571.

Johnson, E. N., Calise, A. J., Sattigeri, R., Watanabe, Y., & Madyastha, V. (2004). *Approaches to vision-based formation control.* Paper presented at the 2004 43rd IEEE conference on decision and control (CDC) (IEEE Cat. No. 04CH37601).

Kannan, S. K., Johnson, E. N., Watanabe, Y., & Sattigeri, R. (2011). Vision-based tracking of uncooperative targets. *International Journal of Aerospace Engineering, 2011*, 243268.

Kise, M., Noguchi, N., Ishii, K., & Terao, H. (2004). Laser scanner-based obstacle detection system for autonomous tractor movement and shape detection targeting at agricultural vehicle. *Journal of the Japanese Society of Agricultural Machinery (Japan), 66*(2), 97–104.

Krajník, T., Nitsche, M., Faigl, J., Vaněk, P., Saska, M., Přeučil, L., Duckett, T., & Mejail, M. (2014). A practical multirobot localization system. *Journal of Intelligent & Robotic Systems, 76*(3), 539–562.

Morin, P., Samson, C. J. S., & h. o. r. (2008). Motion control of wheeled mobile robots. In B. Siciliano & O. Khatib (Eds.), *Springer handbook of robotics* (pp. 799–826). Springer.

Noguchi, N., & Barawid, O. C., Jr. (2011). Robot farming system using multiple robot tractors in Japan agriculture. *IFAC Proceedings Volumes, 44*(1), 633–637.

Noguchi, N., Will, J., Reid, J., & Zhang, Q. (2004). Development of a master–slave robot system for farm operations. *Computers and Electronics in Agriculture, 44*(1), 1–19.

Ou, M., Li, S., & Wang, C. (2013). Finite-time tracking control for multiple non-holonomic mobile robots based on visual servoing. *International Journal of Control, 86*(12), 2175–2188.

Peng, Z., Wang, D., Liu, H. H., & Sun, G. (2014). Neural adaptive control for leader–follower flocking of networked nonholonomic agents with unknown nonlinear dynamics. *Adaptive Control and Signal Processing, 28*(6), 479–495.

Zhang, X., Geimer, M., Noack, P. O., & Grandl, L. (2010). A semi-autonomous tractor in an intelligent master–slave vehicle system. *Intelligent Service Robotics, 3*(4), 263–269.

Chapter 17
Autonomous Robots in Orchard Management: Present Status and Future Trends

R. M. Rasika D. Abeyrathna and Tofael Ahamed

Abstract This chapter aims to explore the opportunities and challenges in orchard management regarding the automation of orchard automation from preharvest to postharvest systems. Labor shortages and dependency on expert growers have become challenging in developed and developing countries. Furthermore, extensive dependence on seasonal labor, which is scarce, and rapidly increasing labor costs have sparked interest in the use of automated machines in orchard operations, which in turn is crucial for improving the quality of high-value fruit crops. Reasons for commercial implementation failure include high initial and maintenance costs. Architectural differences in the fields render trajectory planning and manipulator operations impractical. Newly introduced orchard architectures enable robotic platforms to successfully achieve the desired robotic manipulation. Task-specific robotic manipulators and grippers with few degrees of freedom robotic arms in combination with developed neural networks and classification methods are innovations for achieving the commercialization of orchard automation and low-cost roboticization of medium-scale orchard crops.

Keywords Orchard architectures · Trajectory planning · Manipulators · Grippers · Neural networks

R. M. R. D. Abeyrathna
Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Ibaraki, Japan

Department of Agricultural Engineering, Faculty of Agriculture, University of Peradeniya, Peradeniya, Sri Lanka

T. Ahamed (✉)
Faculty of Life & Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

17.1 Introduction

Fruit cultivation is one of the key subsectors of horticulture and plays a vital role in human nutrition. Fruits are highly nutritious, making them essential for human health. A large variety of fruit crops are grown worldwide. Tree fruits are one of the important subcategories in the fruit industry and contribute significantly to global fruit production. Of the 887 million tons of estimated fruit production worldwide in 2020, tree fruit crops (i.e., apples, grapes, oranges, mangoes, pears, etc.) accounted for 50% of the total production (FAO, 2020).

Tree fruit crops are usually referred to as orchard crops. An area that is often enclosed and devoted to fruit cultivation is usually denoted as an orchard. Since orchard fruit production is highly competitive, growers are pursuing more effective methods to manage their orchard operations precisely and automatically. However, the orchard fruit market also remains competitive in the global market due to its uniformity, taste, lack of injuries, and optimum harvesting time and methods. Furthermore, insufficient labor availability coupled with rapidly increasing labor costs and, more importantly, the risk of dependence on seasonal human labor has sparked interest in the use of automated machines and the scope of the utilization of robots in orchard operations, which in turn is crucial for improving the quality of high-value fruit crops.

During the past few decades, substantial improvements in field robotic systems have been reported due to technological advancements in high-tech computers and sensors with high processing power. Furthermore, the availability of cheap electronics provides more practical situations worldwide. Therefore, current methods of production of most field crops, such as corn, wheat, and soybean, mainly rely on machinery and automated systems with the support of the Global Navigation Satellite System (GNSS). However, GNSS has limitations due to interruptions and complex canopies. The positioning of sensors for vision and perception have more reliability in orchard-based automation. In principle, these developed technologies might be adapted to orchard fruit production. However, due to cultivation and morphological differences between fruit crops and field crops, some management options that are unique in orchard fruit production (i.e., pruning, thinning, training, and harvesting) have made the task more difficult. For instance, automating fruit harvesting requires unique applications due to the irregular distribution of fruit on trees, uneven maturity of the fruit and the requirement for high-quality harvested fruits. Meanwhile, the implementation of several operations in orchard fruit production, such as fertilizer and pesticide application and irrigation, is similar to that of field crops. However, these operations require specific mechanisms that comply with the morphological features and management practices. Moreover, the natural morphological features of orchard crops have increased the difficulty of applying manipulator kinematics in orchard operations.

Amidst these complications in adaptation and implementation, during the last few decades, several researchers have attempted to mechanize orchard operations, among which most of them are dedicated to one specific task (i.e., harvesting or

spraying), and several robot prototypes have been developed and deployed. However, the mechanization in tree fruit production is still far from the expectation and orchard management relies primarily on human labor. Few or no commercially viable automation options are available for orchard crops at present.

In fact, automated robots that emulate human labor are essential for automating different operations in orchard management. The essential components that should be incorporated into developed robotics to accomplish automation of these tasks include a vision system, a moving mechanism (i.e., autonomous vehicles) and a special mechanism to achieve the designated tasks (manipulators and grippers).

Previously, scientists attempted to achieve orchard automation that would coordinate with the existing tree canopy structure. However, they later realized that the most efficient and economical method of achieving orchard automation is to change the tree structure to be fitted into the robotic platform. This approach is the only method that can reduce the number of degrees of freedom in the robotic arms, which will lead to simple, low-cost, and quick manipulations. Thus, the attention of the researchers has been diverted toward changing the orchard architecture to facilitate simple manipulator operations, and thus the development of financially affordable automation options for tree crops seems to be the next goal to ensure increased efficiency in orchard operations.

Within the context discussed above, the focus of this chapter is to first introduce vision systems for orchard robotic operations, followed by an account of the existing orchard operation robots and their operating principles based on orchard pruning, thinning, spraying, and harvesting orchard operations. The chapter proposes constructive suggestions for improving orchard automation through simplified robotic operations.

17.2   Vision Systems for Autonomous Orchard Robots

One of the most important components not only in the agricultural robot but also in any type of working robot is the vision system that detects the autonomous movements of the other two main components of a robot; the manipulator and the robotic platform rely on the data generated from the vision system. More importantly, without obtaining accurate information from the sensors, which are basically cameras, neither manipulators nor robotic platforms perform predetermined tasks accurately. Data acquired from the sensors are then processed by a controller (i.e., microcontroller, programmable logic controller or computer) before being fed to the actuators to perform the predetermined task.

The vision technology has been used for different orchard operations, such as analyzing the growth of orchard trees; identifying pests, diseases, and weeds; applying chemical or biofertilizers; harvesting fruits; pruning branches; removing excess flowers and fruits; and navigating autonomous vehicles inside the orchard.

For robotic applications, either individual or different combinations of cameras have been used to improve the accuracy by obtaining more details. State-of-the-art
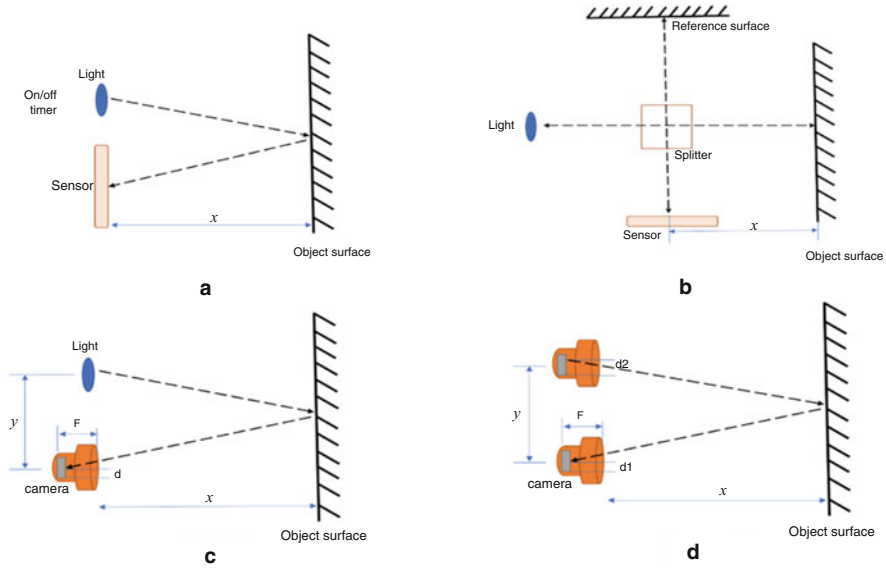
Fig. 17.1 Illustration of 3D vision techniques used in agricultural robotics: (**a**) time-of-flight (ToF) technique, (**b**) Michelson interferometer principle, (**c**) light beam triangulation technique, and (**d**) stereo vision triangulation technique

2D and 3D vision systems are commonly used to identify objects under real-world conditions. Spectral waves such as ultrasound, light, infrared, and light have been used to obtain depth information, but optical techniques have become more popular because of fast data acquisition, safety, and the ability to obtain a high lateral resolution (Büttgen et al., 2005). However, 3D images have x, y, and z coordinates and thereby generate more details than 2D images, which consist of x and y coordinates. Therefore, the use of 3D images has become popular as it is more useful in manipulating robotic arms to achieve the desired tasks.

Detection of the target fruit in the 3D orientation has the potential to determine the efficiency of the whole robotic platform. In 3D imaging, different basic techniques have been used for acquiring depth values, as illustrated in Fig. 17.1. The time-of-flight (ToF) technique (Fig. 17.1a) measures the wave receiving time compared to the wave generated time when the speed of light is known. The different wave modulation methods that have been applied include pseudo noise modulation, pulse modulation and continuous wave modulation. In this technique, the distance to object x is obtained from Eq. (17.1), where t is the traveling time of the light pulse. Figure 17.1b illustrates the interferometry principle, which is used to obtain the depth at the nanometer level accuracy. Here, the distance or depth x is proportional to n number of fringes and wavelength of the light (λ), as calculated using Eq. (17.2) (Lange, 2000). Figure 17.1c shows the triangulation technique using a light beam, and Fig. 17.1d shows the same technique using passive illumination. In these two techniques, the distance x is obtained by measuring the location that hits

Fig. 17.2 Typical apple orchard spacing between rows for autonomous vehicle navigation

the receiving image sensor (d, $d1$, or $d2$) or obtaining the angle of the incoming and reflecting light beams, which are calculated using Eqs. (17.3) and (17.4) (F is the focal length, y is the baseline) (Schwarte et al., 1999; Jähne et al., 1999; Vázquez-Arellano et al., 2016).

$$x = f(t) \tag{17.1}$$

$$x = \frac{n\lambda}{2} \tag{17.2}$$

$$x = \frac{yF}{d} \tag{17.3}$$

$$x = \frac{yF}{d2 - d1} \tag{17.4}$$

The sensor readings may change depending on the light conditions received by the camera sensors. The lighting conditions basically depend on environmental factors such as weather variability and time, especially in temperate countries. Retinex principle and wavelet transform techniques are used to enhance the image qualities and overcome the effect of varying light conditions (Wang et al., 2017). The effect of obstacles such as branches and fruits is another factor that hinders the sensor reading. Figure 17.2 shows the typical inter row navigation pathway between apple rows, and it further illustrates how hard it is to reach apples even by hand without collision with other fruits and branches.

Therefore, the vision systems should be sufficiently accurate to detect the obstacles and proceed with the robotic arm operations. New canopy structures were proposed to have only one fruit on one stem by improving fruit tree genetics to

avoid the hindrances generated by extra fruits. In this case, the machine vision system easily detects and localizes fruits. With the introduction of v-terraces concepts, the effect of obstacles created by branches and reaching the fruits in the deep canopy and fruit clusters can be avoided in orchard operations. Different studies have used different methods, such as their color, shape, texture, size, and edge features, to detect fruits. Tabb et al. (2006a, b) and Silwal et al. (2017) used a Bayesian classifier along with neural networks to improve detection, which requires different classification techniques and thresholding levels. The convex hull method was introduced to identify single fruits and to detect the center point of the citrus fruits (Changhui et al., 2017).

Manipulation of the end effector based on camera coordinates may vary. One method is to use the traveling salesman problem (TSP) technique, which finds the minimum distance to the target location. However, this method requires the help of other algorithms (Silwal et al., 2017).

The other robotic arm manipulation method is the global camera system, where the camera is mounted at a fixed position and captures images at the beginning of one harvesting cycle. After fruit detection and localization, the manipulator moves to all the fruits detected in the images by starting at the first one in the sequence; the order of reaching toward the fruits is determined by the manipulator parameters, and the inverse kinematic is used to calculate the manipulator position each time or the manipulator moves to homing positions each time and recalculate the next target. Once the manipulator finishes reaching all the targets in the frame, the camera captures images for the second round. Previous studies suggest that the camera must coordinate with a precisely calibrated end effector or manipulator to achieve this task, and the error between two coordinates may result in problems for the next cycle as well (Tabb et al., 2006a, b).

The autonomous movement of robotic platforms and manipulators depends on the data generated and processed by the vision system. The use of the Global Navigation Satellite System (GNSS) is one of the commonly applied methods. However, this method is not useful when disturbances appear in the orchard while the vehicle is moving through orchards. Similarly, the global positioning system (GPS) is unable to identify obstacles in the field. Real-time kinematic (RTK) technology has been used to overcome these problems. However, the use of vision systems in combination with artificial and machine vision is more reliable and has become more technologically advanced. Obstacles, artificial markers, and field structures have been used as field markers and identified by vision systems. In machine vision, recent image processing techniques include instance segmentation and object detection through training, validation and testing of large datasets using neural networks, convolutional neural networks (CNNs), and deep neural networks (DNNs) to increase the accuracy of detection.

The research and applications help produce different CNN-based neural networks to increase the accuracy of object detection and semantic segmentation. For example, a study has been conducted to localize apples by integrating CNNs (Häni et al., 2018), similar to mango identification and localization using deep learning (Stein

et al., 2016). Furthermore, a deep learning pipeline was used to count apples and oranges after detection based on a blob detector (Chen et al., 2017).

Integration of vector machines was developed to distinguish and classify apple fruits, leaves, and branches with the support of a genetic algorithm (Yongting & Jun, 2017). The calculation of the time required for one robotic cycle is essential in terms of productivity (Silwal et al., 2017); when the weight of the components increases, the inertia is high, and energy usage also increases, resulting in low robotic arm performance. Most robotic arms use steppers and servo motors, which use DC power from batteries or power from generators. Hydraulic motors and actuators might also represent a novel approach; the difficulty lies in sensing angles and distances accurately, where more precision is needed, and the weight of the parts is greater.

17.3   Autonomous Robotic Pruning in the Orchards

Physical removal of branches and limbs is known as pruning, which is one of the orchard management operations that is second to harvesting in terms of labor requirements, while it requires 20% or more of the total production cost per year (Gallardo et al., 2011; Hansen, 2011). Additionally, it is a cumulative process that requires skilled workers to perform it.

Penetration of sunlight inside the canopy creates high productivity in each branch, resulting in high-quality fruits. In orchard management for robotic operations, the most important goal is to produce fruits of equal size without pests and diseases, which facilitates easy robotic manipulations and picking, to achieve equal treatments for each part of the tree. Pruning in combination with thinning helps to achieve the balance between vegetation growth and reproductive growth. Pruning is an essential practice and always follows some protocols, depending on the plant species, to achieve the desired shape.

In fruit orchard management, the final goal is to obtain continuous production during each cropping season while improving the yield and maintaining the economic balance each harvesting season; the horticultural balance of the orchard should be maintained to achieve this goal. The reproductive and vegetative growth of trees depends on the canopy balance. Hence, maintaining the canopy balance is an important task for both maintaining a good healthy tree and continuous production in terms of fruit size and quality (Schupp et al., 2017). Training and pruning management practices are essential to achieve strong and healthy orchard conditions (Feucht & Larsen, 2009), which is particularly important for orchards that use robotic operations. Figure 17.3 shows the well-maintained fruit orchard systems in Tsukuba-Plant Innovation Research Center (T-PIRC Farm) at the University of Tsukuba.

Maintaining a safe and healthy workforce is a challenging task in orchard operations to ensure continuous maintenance and production because repetitive motion causes muscle pains, back pains, injuries caused by tools, slips from ladders and eye injuries in laborers (Fetzer, 2017). Replacing skilled labor with a machine is
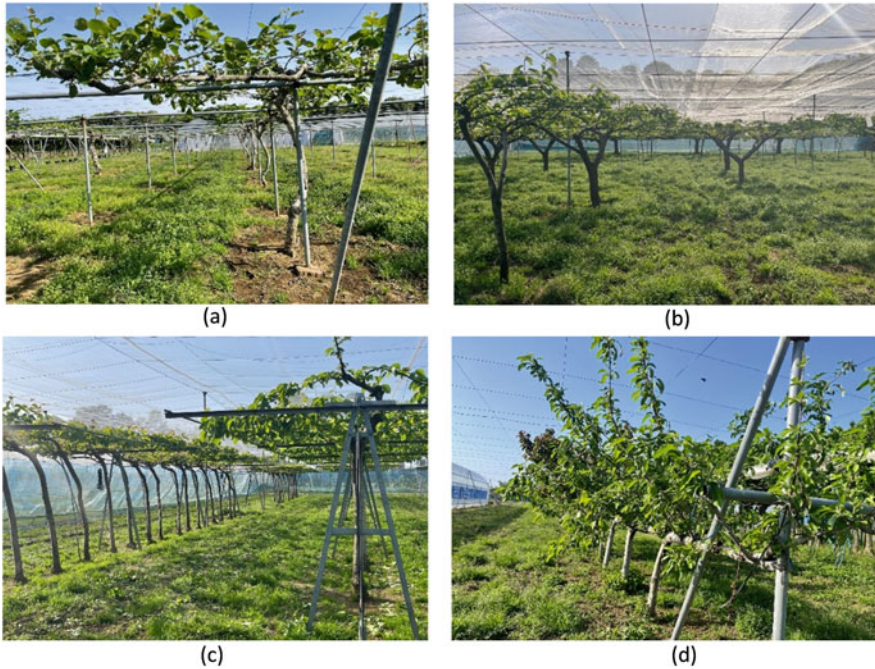
Fig. 17.3 (**a**) "Two-arm" orchard tree training system, (**b**) hedgerow orchard tree training system, (**c**) single curtain and double curtain orchard training system, and (**d**) Guyot orchard fruit training system

a challenging task for researchers and orchard owners, and impressive developments have been achieved in the past few years. Pruning is an inspiring task, and robotic pruning operations are even more impressive. The rapid development of sensors, cameras, and neural networks has helped researchers replace laborers with pruning robots.

Mechanical pruning is one of the techniques used by orchard owners to reduce the canopy size and maintain the shape of trees. Hedging is the mass removal of biomass from the trees that requires hand pruning again as a follow-up process to remove unwanted dead branches, which is sometimes integrated with chemical pruning. Several factors determine the proper method for pruning, including the plant age, species, canopy shape, height, cutting time, and planting density (Sansavini, 1976) . Dormant pruning is conducted to remove unproductive tree parts and to obtain more space for other growing parts and maintain plant health, and powered shears and manual loppers are used for this operation (He & Schupp, 2018). Mechanical pruning is already automated, but nonselective pruning is performed.

Pruning is performed in the winter and summer seasons, and robotic pruning has been mainly chosen for winter because the vision systems can easily identify the pruning points. The implementation of orchard systems such as those shown in Figs. 17.4, 17.5, and 17.6 is a labor-intensive task, and the architecture has the ability
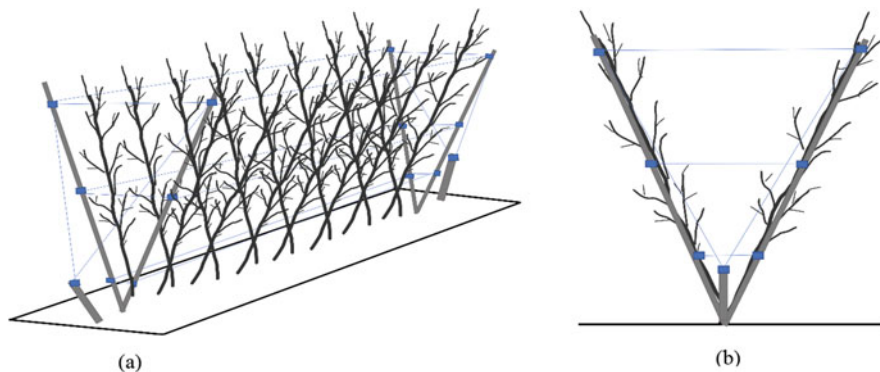
Fig. 17.4 (**a**) Side view and (**b**) front view of a V-trellis fruit tree training orchard
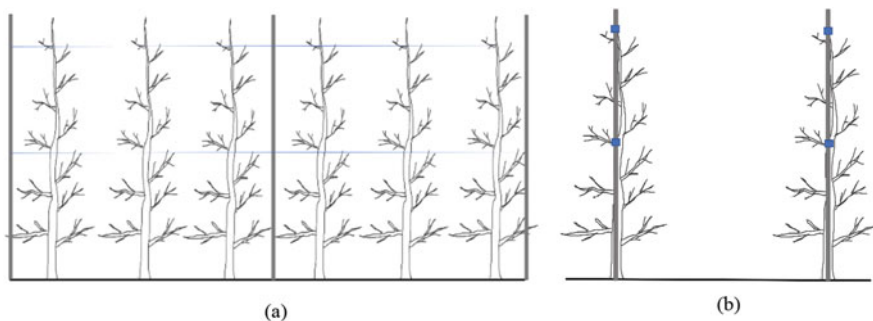
Fig. 17.5 (**a**) Side view and (**b**) front view of a vertical axis fruit orchard system
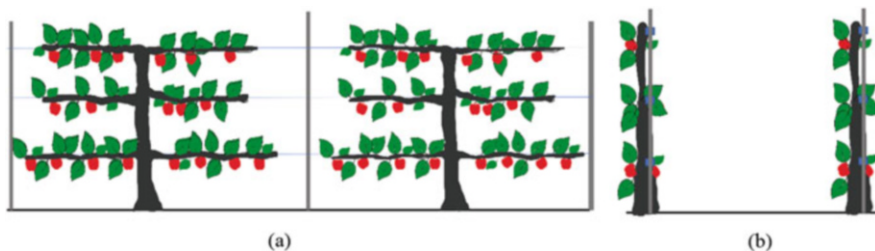
Fig. 17.6 (**a**) Side view and (**b**) front view of espaliers training of apple trees

to facilitate easy autonomous vehicle movements between tree rows, the easy identification of target fruits by the vision system and easy manipulator movements to the canopy with fewer collisions. This system creates a getaway for researchers to develop manipulators with a smaller number of degrees of freedom for easy orchard operations. As illustrated in Fig. 17.7a, a normal orchard pruning robot for normal orchard pruning requires a manipulator with a higher number of degrees of freedom to reach the target locations. Figure 17.7b shows the pruning robot in a well-
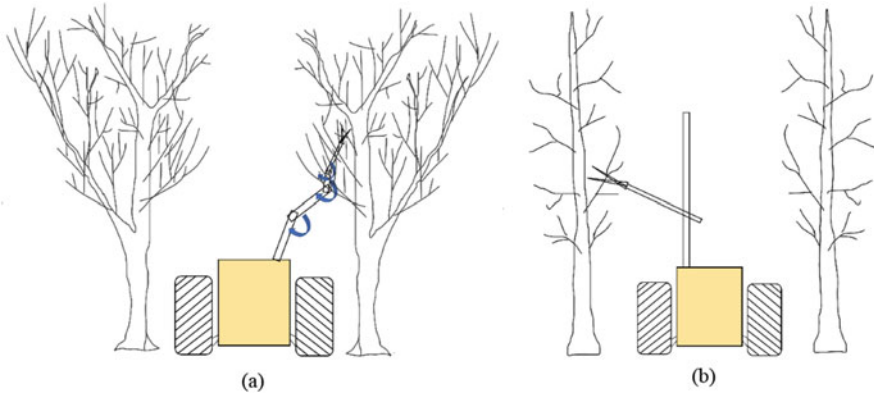
Fig. 17.7 Scopes of robotic pruning: (**a**) higher DoF robotic arms and (**b**) simple robotic arms

maintained vertical axis orchard system; the robotic manipulator can perform the desired task using a smaller number of movements.

However, pruning is frequently a challenging task, as the gripper must generate a higher force for pruning and the robotic arm should withstand the counterforce. Under real-world conditions, the collision of other branches with the robotic manipulator, fruits and flowers becomes a problem, restricting the gripper from moving to the target position. Additionally, environmental factors such as varying light conditions cause some errors in vision systems, and wind sometimes shakes the whole tree, which affects manipulator movements.

Normal orchard trees do not grow in the same shape, and the inconsistency in the size of crop rows and trees prevents the easy application of precision techniques and automation. Well-maintained orchard systems, which are more unique and simplified, with tree training help machine vision systems to identify and detect the fruits and branches easily using developed neural networks and uses simple robotic kinematics to reach the target locations (Karkee et al., 2014). Training of orchard trees for robotic operations is becoming popular, but the main aim of this management practice is to maintain a uniform tree structure year-round.

Grape vine branch detection has become popular because of its uniform canopy structure. A 3D camera was used to determine the exact cutting site for pruning on the branches by measuring the branch length, and the algorithm accuracy was 96.8% (Mahdi & Abdolabbas, 2017).

A robotic manipulator for pruning should have a specific end effector; it may be a cutting blade or blades with a scissor motion (Lehnert, 2012). Pruning robots are more complex than harvesting robots because after detecting the branches, the algorithms should calculate the specific points for pruning based on pruning rules, and then the manipulator should move to specific locations while placing the end effector cutting point perpendicular to the branch; the cutting angle is not always perpendicular. LiDAR (Light Detection and Ranging) sensors have been used to overcome GPS inaccuracy and as a support method for object detection. LiDAR uses

Table 17.1 Some of robotic orchard pruning applications/3D model developments

Orchard application	Sensors and algorithms
Apple tree dormant pruning (Medeiros et al., 2017)	Time-of-flight camera, Laser scanner split-and-merge clustering algorithm, the robust fitting algorithm
Fruit (Apple) tree traits measurements (Tabb & Medeiros, 2017)	Color cameras, Robotic System for Tree Shape Estimation (RoTSE), outdoor environment
Apple tree dormant pruning (Akbar et al., 2016)	Kinect2 sensor, 3D reconstruction, and modeling-based algorithm (Akbar et al., 2016)
Apple tree dormant pruning (Chattopadhyay et al., 2016)	Kinect2 sensor (KinectFusion software), semicircle fit model created
Apple tree shape sensing (Karkee & Adhikari, 2015)	Time-of-Flight (ToF) camera
3D modeling of apple tree (Elfiky et al., 2015)	Skeleton-based Geometric (SbG) features based, model accuracy 96.0%, algorithm—(circle-based-layer-aware modeling)
Apple pruning branches identification (Karkee et al., 2014)	ToF 3D camera, based on developed pruning rules

pulsed light waves to assess the surroundings, which can be used for many orchard and agricultural applications.

The use of LiDAR sensors for biomass mapping in forests is a popular method. 3D reconstruction is required to perform this task, and the "Simple Tree" tool was developed as a cloud-based 3D tree reconstruction method for forest inventories using laser-scanned trees, with a 0.965 coefficient of determination, indicating its effectiveness (Brandtberg et al., 2003).

Another study was conducted to identify primary branches and diameters for dormant pruning. The system identified the branches with 98% accuracy and the diameters with 0.6 error, and a laser sensor was used for this study (Medeiros et al., 2017). However, the LiDAR sensor combined with the reconstruction algorithm was used to generate a 3D model of the canopy structure by achieving a 0.78 correlation with branch length and a 0.99 correlation with branch diameter (Bucksch & Fleck, 2011). Meanwhile, 3D cameras were also used to develop 3D models for orchard tree thinning and pruning (Cheng et al., 2007).

LiDAR, laser scanner sensors, and 3D camera images have been used to develop 3D models and then make the decision based on that model using artificial intelligence. The decisions are always dependent on predetermined pruning principles for the target tree. For the grape vines, a similar technique was used to determine the pruning points. Table 17.1 illustrates some of the developed 3D models and the use of 3D models for orchard pruning.

The development of 3D models for pruning applications has become popular because these pruning models have been used for the identification of pruning points in tress and robotic manipulations. A 3D model was developed for existing tree structures to operate passive robots (Edan et al., 1991). When developing tree models in 3D space, a simulation procedure for the geometric features based on the L-Systems method was introduced (Prusinkiewicz et al., 1990). For Buckeye

Gala apple tree pruning, a pruning severity index was introduced, the limb-to-trunk ratio (LTR) index, which was used for 3 consecutive years and achieved promising results (Schupp et al., 2017). Similarly, the geometric features of the fruit trees are described with the help of geometric transformation. The Markov chain method was introduced to analyze fruit tree architectures (Costes et al., 2006). The development of this model is based on three methods: 3D scanning (Méndez et al., 2014), visual images (Santos & Ueda, 2013), and mechanical sampling of points (Edan et al., 1991).

Meanwhile, thinning is important as pruning to balance the reproductive growth of fruit orchard systems. Maintaining an equal fruit size is much more important for robotic harvesting during the harvesting period and for maintaining fruit quality, appearance and preferability. The removal of extra flowers to facilitate the desired size and yield is known as thinning, and manual labor and chemicals are used for this purpose. The use of robotic platforms is possible with the existing developed neural networks by detecting and counting flowers. This process is completed with high accuracy in developed, well-trained orchard systems. 3D stereo cameras have been used to determine the 3D locations of the flowers after detecting flowers based on color and shape features, followed by robotic manipulation to the target location to spray chemicals or remove the flowers using finger grippers.

A mechanical string thinner was developed to reduce the flower density in peach orchards and tested for more than two seasons in different orchard systems with promising results (Schupp & Baugher, 2011). The mechanical thinning efficiency depends on the angle of the string thinner and the speed of rotation. Table 17.2 illustrates some of the research conducted in orchard flower identification and thinning.

Based on the evidence available, mechanical string thinning is a promising solution for reducing flower density. Meanwhile, it can be developed for area-specific mechanical string thinning, as some of the studies have proven its success (Fig. 17.8). The developed neural networks, sensors, and developed orchard structures will lead to manipulator-based flower thinning, which has been reported by some scientists.

17.4 Pollination of Orchards Crops Using Autonomous Robots

Mechanical pollination is another important application of orchard robotics that may reduce labor costs and reduce the time required for intensive operation. The detection of flowers is the main objective of a bud simulation. With the help of neural network detection models such as YOLO (You Only Look Once) and Mask-RCNN (Region-Based Convolutional Neural Network), this task becomes achievable, but real-time detection and movement of the manipulator or bud simulation mechanism is challenging.

Table 17.2 Orchard flower detection and thinning applications

Orchard application	Application
Apple orchard (Dias et al., 2018)	Novel CNN-flower detection, precision and recall more than 90%
Peach orchard (Horton et al., 2017)	Unmanned aerial vehicles (UAV)—multispectral camera, average detection 84.3%/peach flowers emerged before leaves
Almond orchard (Underwood et al., 2016)	LiDAR, color camera, based on flower and fruit distribution-yield prediction
Pear orchard (Wouters et al., 2015)	Multispectral camera-detect flower buds, use of near-infrared wave bands, canonical correlation analysis—spectral discriminant model, detection around 87%
Peach orchard (Lyons et al., 2015)	Thee brush end effector, selective thinning
Apple orchard (Hočevar et al., 2014)	Still and industrial color camera, (hue, saturation, luminance)-based flower cluster identification, hypothetical based spraying
Apple orchard (Gebbers et al., 2013)	Stereo camera, real-time flower density based thinner adjustment
Peach orchard (Yang, 2012)	Table-top thinning robotic arm
Peach orchard (Nielsen et al., 2012)	Correlation-based stereo algorithm, nighttime-flash illumination operations, outdoor mapping of flower locations
Peach orchard (Aasted et al., 2011)	Mechanical thinning, perpendicular V-orchard system-LiDAR sensor for autonomous navigation rotating thinner
Apple orchard (Aggelopoulou et al., 2011)	Chemical thinning, predicting yield based on images at tree full bloom
Peach orchard (Emery et al., 2010)	Laser-based 3D images, dynamical scanning approach, (indoor), robotic guide thinning
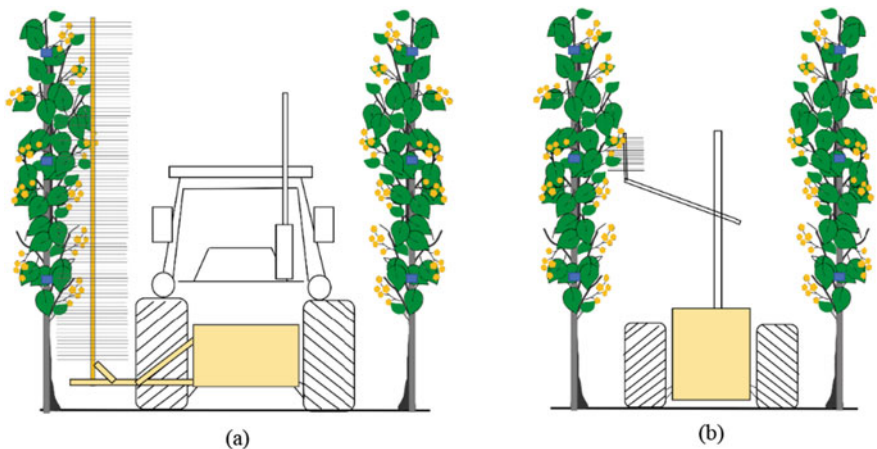
Fig 17.8 (**a**) Mechanical string thinning of flowers. (**b**) Robotic string thinning of identified parts of the orchard trees by the robotic manipulators with a lower number of degrees of freedom

However, after UAVs (unmanned aerial vehicles) were integrated in robotic operations, spraying liquids and granules became easy operations. Compared with the ground operation pollination vehicles, UAVs have more advantages in terms of canopy coverage. A novel pollination method has been established by developing robotic bees for pollination to simulate environmental conditions and assess real orchard conditions (Abutalipov et al., 2016). For spraying, pollen must be mixed with another liquid that helps to deposit it on the flowers, and it should be used in a powdered form. Mechanical spraying of liquid and powders is a challenging task for many reasons, while precision spraying is even more thought-provoking and requires a precision nozzle system, pressure control system for liquid and powders to have the desired application pressure at various places, boom length that is automatically adjusted according to the canopy structure, and a consideration of environmental factors (ex-wind speed) and vehicle running speed.

Meanwhile, several studies have been performed to develop a kiwifruit autonomous pollination system with a manipulator operation. For example, in harvesting robots, the manipulator is attached to an autonomous platform, and a 70% flower detection is achieved using a CNN-based image recognition model. The results reveal that the pollination percentage is 80% (Barnett et al., 2017). Parallel work was conducted by Washington State University to examine the effect and productivity of pollination by suspending pollen in sucrose and boron solutions. Automated pollination was compared with natural pollination, and the results showed that this novel method is three times more effective than pollination by bees (Whiting, 2016). Electrostatic chemical spraying is an achievable task, and more studies of electrostatic pollination are needed to determine a reliable method.

17.5   Use of Fertilizer and Liquid Chemical Application Autonomous Robots in Orchards

Fertilizers and pesticides become very critical when they are applied at larger concentrations on the plants; spending extra money on excess chemicals is not profitable or economical. Most of the time, the farmers do not follow the instructions on the label when applying chemicals manually, and they do not even care about the physical parameters such as the canopy density of the plants. Pesticides and herbicides are harmful to human health.

When applying these chemicals to orchards, workers should wear protective equipment for a long time, which is a tedious task. Moreover, an excessive dosage may also be harmful to plants and the environment (Fig. 17.9). With the development of new mechanical and robotics systems, this pesticide spraying work can now be performed using robots. Nevertheless, the dose adjustment of chemicals based on the canopy structure and environmental conditions is a critical factor to consider in autonomous application. Site-specific chemical application is one of the processes that evolves after the mass application method, and thus this method can reduce the
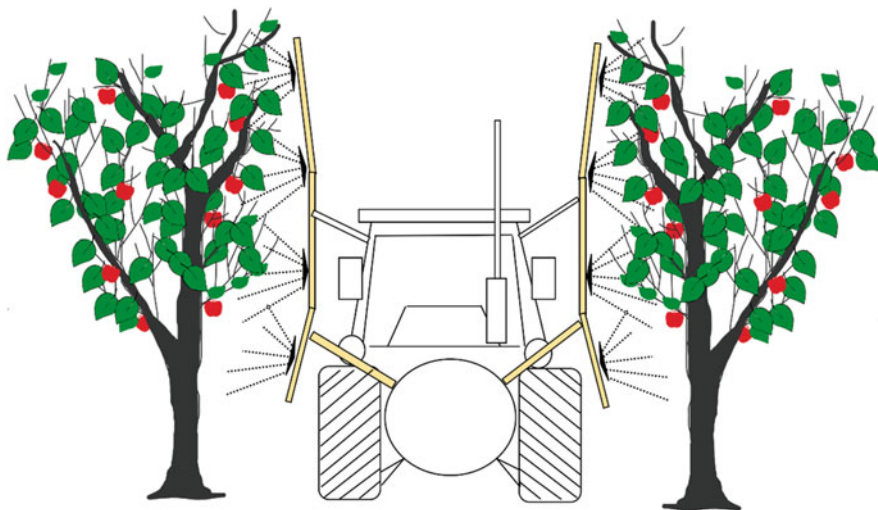
Fig. 17.9 Mass chemical application in an apple orchard using autonomous navigation sprayers under normal orchard conditions

amount of chemicals used and the chemical cost at the same time. Furthermore, it reduces the harmful effect on the environment.

A vision system based on a neural network model is essential to detect the target location for site-specific spraying, and a grape cluster detection algorithm (GDA) was developed. This GDA identifies and distinguishes grape clusters from other parts of the grape vine. Similarly, the Foliage Detection Algorithm (FDA) was developed to detect grape leaves, and with these developments, the identification of specific locations of grape vine became a possibility. Real-time detection was performed using an RGB camera, and the spraying distances were calculated using distance sensors. The decision was made based on machine vision, and the spray nozzles were operated as actuators. The results ensure that the system saves 45% of pesticide use (Berenstein & Edan, 2018).

Hyperspectral images have been used to analyze plant health. This method is used to apply fertilizer by determining plant health, and this technology was used in the RIPPA (Bogue, 2016) and Ladybird robots (Underwood et al., 2015). The same technology may be applied to identify fruit health during the development of fruit growth, which helps to improve yield by applying different treatments. Scientists realize that when farmers spray chemicals, they should also consider the physical nature of each tree because if all those trees are at the same maturity level, dimensional and structural aspects may change. Based on these findings, researchers concluded that humans cannot identify those indices on their own. A developed machine vision system must also identify those indices when spraying chemicals, and the method of spraying is also not controlled by humans. Therefore, automatic adjustments must be incorporated into the nozzle to achieve precision spraying. The
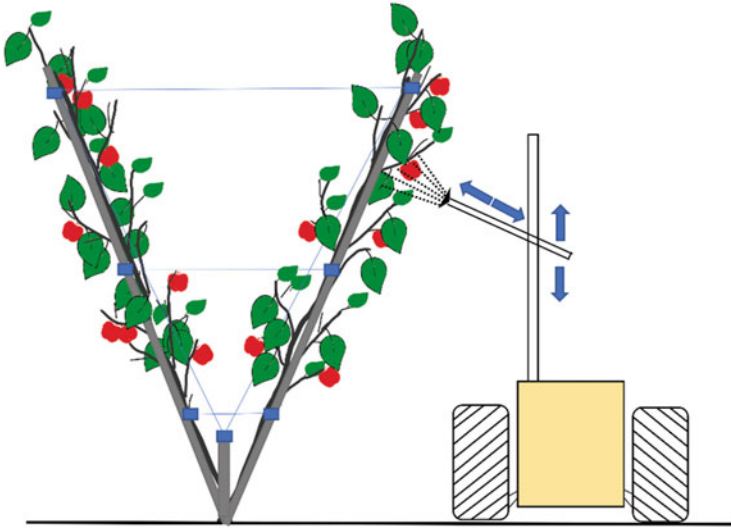
Fig. 17.10 Area-specific chemical application with a robotic arm with smaller degrees freedom in the V-trellis apple orchard system

area-specific chemical spraying robot with a minimum number of degrees of freedom for robotic applications can be further developed to provide a solution for site-specific chemical application (Fig. 17.10). Spot spraying through UAVs has achieved significant progress in reducing chemical application to fruit orchards.

The use of unmanned aerial vehicles (UAVs) has become popular in farm management (Fig. 17.11); for large orchard management, UAVs are an essential tool, and UAV multispectral cameras effectively calculate the normalized difference vegetation index (NDVI). The models have been used as a reference to identify uneven orchard parameters, such as irrigation conditions and soil conditions. Then, precautions can be taken to apply fertilizer and irrigate water and pesticides. Earlier satellite images were used to analyze orchard conditions, but UAVs now fly at low altitudes without any interference from clouds because if clouds are present, the satellite images become unproductive (Kim et al., 2019).

A study was conducted to understand the dimensional parameters of vineyards in terms of chemical applications. Chemical penetration to the canopy depth and losses may depend on the dimensional parameters of the tree (Maccarrone & Scienza, 1998). A similar study has been performed to analyze the nozzle dose adjustments based on size parameters of the target tree parts, but under real conditions, it fails to achieve desired levels because of the effect of environmental parameters (e.g., wind) (Walklate et al., 2004).

Initially, chemical spraying robots were designed for the orchard to obtain an understanding of the distances from autonomous vehicles to orchard trees. Ultrasonic sensors were used to measure the distances and feed to the main controller, while hydraulic spraying was performed based on the signals sent from the main
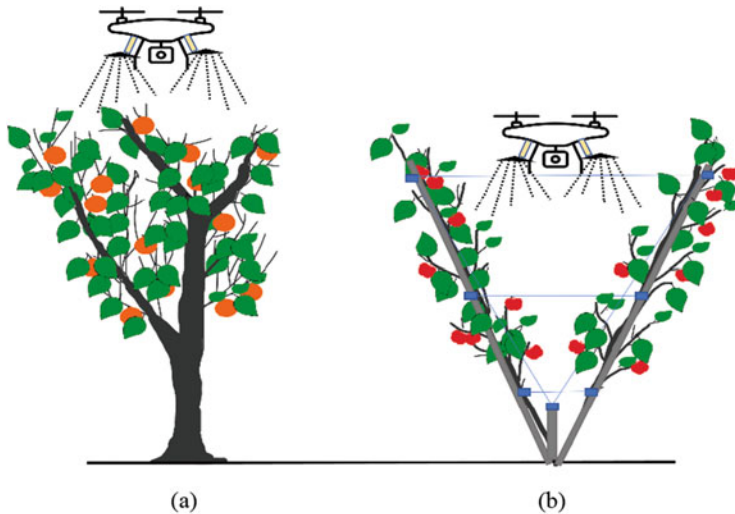
Fig. 17.11 (**a**) Unmanned aerial vehicles (UAV) mass chemical spraying of a normal orange orchard and (**b**) a V-trellis apple orchard system

controller (Giles et al., 1987, 1988). As an innovation in orchard chemical spraying robots, an air-assisted chemical method (Balsari & Tamagnone, 1998) and an automatic flow adjustment spraying method (Martn et al., 2001) were developed. Although research and development in automated orchard chemical and liquid fertilizer application have been achieved to a certain extent, other areas still require attention.

The most used non-imaging sensors for characterizing the structural properties of the canopy are ultrasonic sensors and LIDAR sensors. Infrared sensors were also used to obtain the canopy structure, and these sensors analyze the canopy volume and depth (Jeon et al., 2011). Laser scanners are also being used to analyze canopy structures, and the results have been obtained with a 6.8% coefficient of variance (Liu & Zhu, 2016).

Several autonomous systems have been developed and introduced for spraying chemicals in orchards. Some were able to achieve the purpose of minimizing the wastage of high-cost agrochemicals (i.e., herbicides and pesticides), as these machines are deployed to spray chemicals only to the targeted areas while avoiding nontarget areas with the use of a well-developed vision-based system (Berenstein et al., 2010). For instance, autonomous vehicles that were developed to apply pesticides to control pests in grapevines save 92% of chemicals compared to a manual broadcasting sprayer (Kang et al., 2012). Similarly, the results of another study that aimed to apply chemicals to control grape canopy powdery mildew disease using autonomous vehicles showed that the mechanism reduces chemical wastage by 35% at an accuracy level of 85–100% (Oberti et al., 2016). Table 17.3 shows the robotic spraying of chemicals/fertilizers in orchard management.

Table 17.3 Existing orchard robots for fertilizer and liquid chemical application

Robot	Orchard application	Vision system	Navigation
Bly-c-agri (Sarri et al., 2019)-Hexacopter	Grape orchard—chemical weeding	Multispectral	IMU and GPS
		Camera, normalized difference vegetation index (NDVI)	
Robotic sprayer (Berenstein & Edan, 2018)-4WD	Grape orchard—chemical weeding	RGB camera and laser (Foliage Detection Algorithm—(FDA) and Grape clusters Detection Algorithms—(GDA) algorithms)	–
SAVSAR (Adamides et al., 2017)-4WD autonomous vehicle	Grape orchard—chemical weeding	RGB camera and LiDAR (Light Detecting and Ranging) (FDA and GDA algorithms)	–
AgriRobot (Adamides et al., 2017)-4WD autonomous vehicle	Grape orchard—chemical weeding	RGB camera and LiDAR (FDA and GDA algorithms)	–
Bly-c-agri (Turner et al., 2011)-UAV-Octocopter	Grape orchard—chemical weeding	–	GNSS (Global Navigation Satellite System)
Cäsar (Anon., n.d.)-4WD autonomous vehicle	Vineyard—multipurpose	Obstacle detection—Ultrasonic sensor	(Real-Time Kinematic)-RTK GNSS GPS
Ted (Anon., n.d.)-4WS autonomous vehicle	Grape orchard—mechanical weeding	RGB camera	RTK/GPS (Centimetric precision)

With the progress in new research, machine vision and robotic technologies are slowly becoming more reliable and human friendly. With all precautions, robotic systems with deep neural networks using a combination of machine vision will replace humans and generate promising results by considering the factors discussed above.

17.6 Autonomous Robots for Harvesting Fruits in Orchards

Manual fruit picking is a tedious task in regard to large-scale orchard harvesting. Successive repetition of the same task causes musculoskeletal pain, disorders, and back pain in labors. If the canopies are high, laborers must use ladders; then, the risk

of accidents is high because they must come down the ladder each time the bucket is full to ensure that the fruits remain in the collector (Fathallah, 2010).

Currently, semiautomated vehicles have been used for orchard fruit harvesting, which facilitate a platform for human laborers to easily harvest the fruits by shaking and collection. The other type is an autonomous human-driven vehicle with a robotic platform to harvest the fruits. The third type is a fully autonomous vehicle and robotic platform. For juice making, the ideal method is to shake the tree and collect the fruits that are already commercialized (Zhang, 2017). The robotic harvesting of fruits without damage is more suitable for the fresh market.

Robotic harvesting of orchard fruits is a challenging task, and the identification and recognition of fruits in canopies are very important. Localization of individual fruits helps manipulators easily pick fruits; fruits, especially apples, grow in clusters under real environmental conditions (Baeten et al., 2008; Bulanon et al., 2002).

The primary task of robotic harvesting is to detect the target fruit under real conditions. For this task, different features have been used, for example, color, texture, shape, size, and edge detection. These parameters are then analyzed using neural networks and classification methods (Silwal et al., 2014). However, the main problems have occurred in the real-time detection of fruits and branches. Designing grippers for a single fruit is a common practice based on the physical shape and the properties of the fruit, but the problem is how to address fruit clusters.

The convex hull technique was introduced to identify individual citrus fruits from fruit clusters and individual fruit center points as a method to overcome fruit clustering (Changhui et al., 2017) . In addition to fruit clustering, the other main challenge is the operation of the robotic arm under variable lighting conditions. As a form of remediation, different techniques have been introduced for image segmentation and classification, and Retinex theory is an innovative method to minimize lightning problems (McCann, 2014). Using this approach, the image enhancement method combines the Retinex principle and wavelet transform (Wang et al., 2017). Table 17.4 illustrates some of the fruit detection methods and their results in terms of specific applications and techniques.

The fruit canopy is a complex structure, and occlusion is another point at which robotic manipulators fail, although the identification is accurate. Figure 17.12 shows the apple harvesting robot performing the manipulation task under normal orchard conditions with a robotic arm with a high degree of freedom. Reaching the detected fruits inside the canopy is challenging, and the robotic arm either misses the target fruit or fails to complete the manipulator operation because of the branches and fruit clusters. As a solution, pruning is the practice of maintaining a simple standard orchard structure, which helps manipulators easily reach the detected target fruit.

The reduction of several degrees of freedom and the complexity of the robotic structure may lead to an easy operation, low cost, and easy maintenance. Figure 17.13 illustrates the harvesting robot with a smaller number of degree freedom robots that operated in a well-maintained orchard with specially trained trees.

Meanwhile, trained orchard systems help with manual labor as well. Y-trellis tall spindle orchard training systems have been developed to improve the productivity of labor and yield. These orchard architectures help laborers perform all management

Table 17.4 Some of the fruit detection and identification methods and their accuracy

Application	Machine vision system	Accuracy of identification
Apple detection ("Gala" and "Blondee") (Chu et al., 2021)	Novel suppression Mask R-CNN	F1—0.905 detection time 0.25 s per frame
Identification of cutting points of grape clusters (Luo et al., 2018)	Segmentation algorithm, geometric model, profile analysis, computational method-for identify the cutting points	Average recognition accuracy—88.33% Cutting points detection (double overlapping clusters)—81.66%
Detection of cheery branch shaking locations (Amatya et al., 2017)	Bumblebee® XB3 camera, (ToF) based 3D camera—depth values, branch pixel, and cherry pixel-based detection of branches	Three shakings of target branches—Y-trellis system—efficiency of 92.3%
		Vertical trellis—efficiency 86.4%
Detection of pears and apples (Font et al., 2014)	Minoru 3D USB Webcam, laser pointed red cross method	Average distance error—4% to 5%
		30% average diameter error
Red apple separation from the image (Ji et al., 2012)	Image segmentation based on region growth and color feature (classification algorithm—support vector machine)	Recognition—89%
		Average recognition time—352 ms
Identification of green apples (Linker et al., 2012)	Color and smoothness-based model	Identification—85%
Identify and count green citrus (Kurtulmus et al., 2011)	Eigenface algorithm (Sirovich & Kirby, 1987) (color intensity, saturation combine with circular Gabor texture) Blob analysis	Citrus fruit detection—75.3%
Outdoor apple detection in canopy images (Stajnko et al., 2009)	Circular Hough Transform (CHT)	Apple detection—89%

tasks in the field while providing high-quality fruits (Robinson et al., 2007; Bergerman et al., 2012).

Accurate and precisely determining harvesting indices is important to harvest fruits with less damage, which in turn assures the economic stability of the orchard, as high-quality fruits sell for higher prices in the market. However, most of the existing fruit harvesters (i.e., target fruit harvesters) fail to achieve this purpose. As a remediation method, scientists have developed yield estimation robots to identify the yield before harvesting (Table 17.5).

An autonomous robotic harvester is usually comprised of a vision system for detecting and locating the fruit; a manipulator that moves the gripper to the identified target; and an end effector, usually a gripper, which is designed for detaching the fruit. To date, several researchers have attempted to develop autonomous harvesters
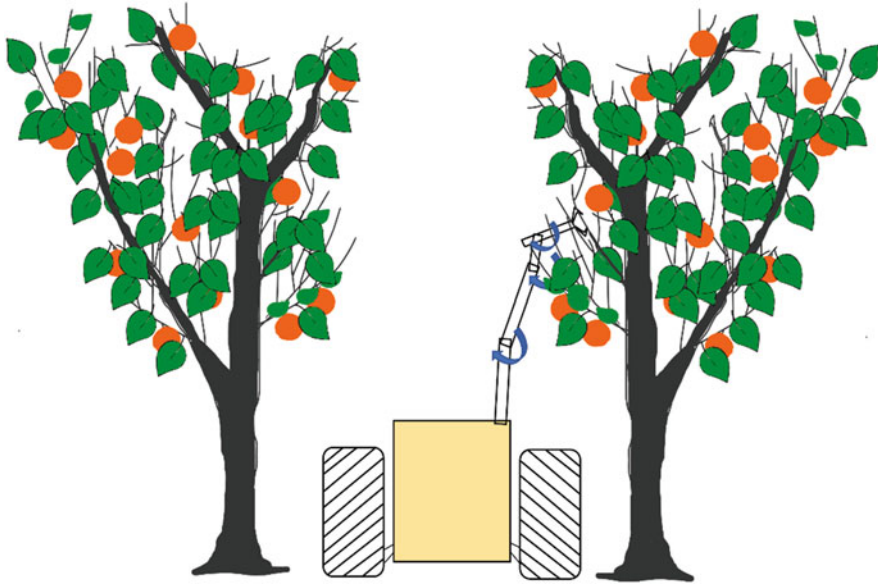
Fig. 17.12 Orange harvesting using multiple robotic arms with a higher number of DoFs

with different grippers and manipulators using vision systems. These applications have their limits. Despite the considerable efforts made by both the government and private sector in developing commercially successful harvesters for orchard crops, research and development in this area are gradually but steadily progressing.

Once the manipulator successfully reaches the target fruit based on 3D coordinates from the sensors, the gripper or the end effector should detach the fruit without causing any damage. The fruits are not always equal in size, and the skin may be fragile; therefore, the key features of the gripper should be minimizing the damage and detaching the fruit from the stem. Silicon rubber bionic concept-based soft robotics have become popular because of their adaptive ability and infinite freedom. Soft gripping of fruit is a key task because the final goal of robotic harvesting is to obtain undamaged fruits, providing support for newly emerging 3D printing combined with super-elastic silica gel to create new soft grippers (Pi et al., 2021).

Another novel apple gripping technique was introduced by the Abundant Robotics Company, which used a vacuum suction gripper. The suction gripper sucks the apples with high pressure; however, due to the high suction pressure, it also sucks the leaves that touch the fruits (Simonite, 2017).

It is essential to identify the target fruits for harvest must be identified before the use of robotic arms for harvesting. Table 17.6 shows some of the fruit identification applications and their success rates. If the detection point is readable, *x*, *y*, and *z* coordinates are shown, then the robotic arm with the gripper is moved to that target point. A citrus harvesting robot has been developed after several attempts (Rabatel
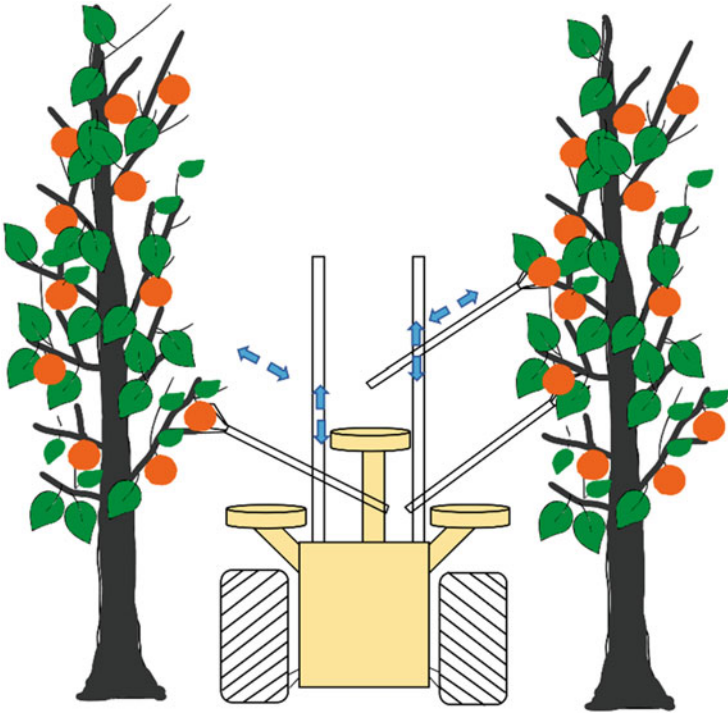
Fig. 17.13 Possible methods for harvesting oranges using multiple robotic arms with fewer DoFs

et al., 1995). Table 17.7 illustrates some orchard fruit harvesting robots and their applications.

17.7 Discussion

Different climatic conditions and management options in different regions are some of the critical factors that hinder the development of common harvesters. Moreover, robotic applications for biological materials are far more complicated, as biological materials do not act as we expected. For example, the analysis of fruit size by the robotic platform developer may deviate under real conditions due to the presence of obstacles such as branches, overlapping fruits, and fruit orientation. Therefore, recent developments in machine vision and artificial intelligence programs have focused more on identifying and distinguishing obstacles in robotic harvesting. The combination of machine vision and artificial intelligence is an innovative approach to address the labor shortage while training robots as skilled laborers.

If the manipulator operation can be simplified by reducing the number of degrees of freedom, then the operation time for one cycle will be reduced, the maintenance

Table 17.5 Existing yield estimation robots

Robot	Orchard application	Vision system	Navigation
Kubota farm vehicle (Lowe et al., 2021)	Grape-yield estimation	LiDAR	GPS and IMU
		Continuous-Time Simultaneous Localization and Mapping (SLAM)	
Agrob V16 (Reis et al., 2018)	Grape-yield estimation	RGB-D and RGB cameras	RTK/GPS/(Inertial Navigation System)-INS and LiDAR
		Local Binary Pattern-histogram (hLBP) and Support Vector Machine (SVM)-algorithms	
Shrimp (Bargoti & Underwood, 2017)	Apple-yield estimation	RGB camera (multilayer perceptron)-MLP and CNN algorithms	–
Agrob V14 (Santos et al., 2016)	Grape-yield estimation	RGB camera	LiDAR
		Support Vector Machine (SVM)-algorithm	
VINBOT (Lopes et al., 2016)	Grape-yield estimation	RGB and (near-infrared) NIR cameras-NDVI algorithm	RTK, DGPS, and LiDAR
AgriBOT (Abrahão et al., 2011)	Orange-yield estimation	RGB camera	GPS/INS
VineRobot (Anon., n.d.)	Grape-yield estimation	RGB camera	–
		Ultrasonic sensors	
		FA-Sense LEAF, FA-Sense ANTH	

cost is decreased, and the safe operation for both people working in the orchard and for robots will be safer. Therefore, the overall efficiency, effectiveness, and robustness will be increased and economically feasible for commercial manufacturing, as indicated in the 3D view shown in Fig. 17.14. The combination of 3D vision and newly introduced horticultural architectural training methods, such as trellis systems, allows smooth guidance of the machine vision and robotic manipulators to the target fruits.

Specifying robotic arms for a specific task is the next task, as discussed in the preceding sections. Of course, the robotic arms should be specific to pruning, thinning, spraying, and harvesting, but here, the focus is the possibility of specifying robotic manipulators in each individual section where the canopy is fully maintained under standard orchard architectural conditions.

17.8 Conclusions

Orchard management practices extensively depend on seasonal labor, which is costly and scarce. Improvement in task-specific robots, which are robust, economically, and technically achievable for at least medium-scale orchard farmers, is

Table 17.6 Examples of the end effectors developed or orchard fruit grasping

Gripper/end effector	Orchard application
Gripper to analyze the forces (bending, cutting resistant) (Bu et al., 2020)	Apple Harvesting
Soft finger-based gripper (Kang et al., 2020)	Apple Harvesting
Bite mode end effector, pneumatic operation (Wang et al., 2019)	Orange Harvesting
Gripper with four fingers (Onishi et al., 2019)	Apple Harvesting
Pneumatic operated, 3D-printed soft end effector (Hohimer et al., 2019)	Apple Harvesting
Vacuum Suction gripper (Simonite, 2017)	Apple Harvesting
Bionic gripper, operated using stepper motor (Longtao et al., 2017)	Kiwifruit Harvesting
Opening span type, tendon-driven fingers, three fingers, soft polyurethane pads, and springs in the joints for return action (Silwal et al., 2017)	Apple Harvesting
Three fingers (Davidson et al., 2016)	Apple Harvesting
Spooned shaped gripper, two fingers, shear cutting, three sensors—collision, pressure, position sensors (De-An et al., 2011)	Apple Harvesting
Pendule holder with fingers/Fuji apples (Bulanon & Kataoka, 2010)	Apple Harvesting
Flexible gripper, silicone funnel shaped (10.5 cm diameter), vacuum suction (Baeten et al., 2008)	Apple Harvesting

essential for replacing manual labor. Adapting robotic platforms that may replace the use of human laborers will avoid this problem while achieving greater efficiency and productivity in orchard operations. Recently, robotic applications for orchard management have increased rapidly along with the development of new, robust technologies. Both the government and organizations in the private sector are trying to develop low-cost robotic applications that can be adopted at the field level and are compatible with farmers. However, difficulties encountered in simplifying the robotic operations to the level that is easily managed by the farmers have hindered the commercial success of automated orchard management. As discussed in the preceding sections, pruning, thinning, spraying, and harvesting are major management practices involved in orchard management. The manipulator and the end effector are the critical operational units that should be specific for orchard operations. The existing technologies may lead the robotic manipulators to the unstructured thick canopy to reach the target locations, but the robotic system may become complex with an additional degree of freedom and sensors. Meanwhile, those systems require huge computational capacities for processing data in real time, which would not be a method to commercialize them for farmers or medium-scale operations. The identification of target locations has now become easy with the

Table 17.7 Existing orchard fruit harvesting robots

Robot	Orchard application	Vision system	Manipulator
Apple harvesting robot (Kang et al., 2020)	Apple harvesting	RGB-D camera, Dasnet, 3D-SHT, and Octree	6-DoF
	Operation time for one cycle—7s	F1: 0.81	
Kiwi fruit harvesting—multiple robotic arms (Williams et al., 2019; Barnett et al., 2020)	Kiwi fruit harvesting	Baslar ac1920-40uc USB 3.0 stereo cameras, fully convolutional network (FCN), semantic segmentation, blob detector, developed novel dynamic scheduling algorithm	Four, 3-axis articulated robotic arms
	Harvesting success 51.0%		
	Average cycle time of 5.5s		
Apple harvesting robot (Silwal et al., 2017)	Apple harvesting—the success rate of 84%	Circular Hough Transformation (CHT), Blob Analysis (BA)	6-DoF with prismatic base
	Average localization time 1.5 s	Apple prioritization—traveling salesman problem (TSP) with the Nearest Neighbor algorithm. Global camera system. Circular Hough Transformation (CHT). CCD (Charged Couple Device) and color camera (Prosilica GC1290C, AVT Technologies)	
	Average picking time 6 s		
Citrus harvesting robot (Mehta & Burks, 2014)	Citrus harvesting	Two color CCD cameras (KT & C, KPCS20-CP1), Lyapunov-based stability analysis-manipulator control	7-DoF
Apple harvesting robot (De-An et al., 2011)	Apple harvesting	Image-based Vision Servo (IBVS), CCD camera, vector median filter, Hue Intensity and Saturation (HIS) model, feature extraction—Rotation, Scale, and Translation (RST)	5-DoF structure
	Success rate 77%		
	Average cycle time 15 s		
Autonomous Fruit Picking Machine (AFPM) (Baeten et al., 2008)	Apple harvesting	Camera mounted center of the gripper, image processing—Halcon software	Industrial robot (Panasonic VR006L), mounted on four-wheel tractor
	Success rate 80%		

Fig. 17.14 3D views of a robotic arm with (**a**) multiple degrees of freedom and (**b**) a lower number of degrees of freedom

development of 3D sensor systems and the integration neural networks and classification methods. In the case of harvesting, few robotic arms are attached to a single autonomous platform, and each robotic arm covers different areas or covers specific tasks with a smaller number of degrees of freedom, which has immense potential to be adopted with the prime mover as attachments with lower costs and less complexity. The total system should be available as an attachment to a four-wheel tractor or an autonomous platform with task-specific interchangeable manipulators. Therefore, simple robots with a smaller number of degrees of freedom seem to be the most promising solution.

References

Aasted, M. M., Dise, R. J., Baugher, T. A., Schupp, J. R., Heinemann, P. H., & Singh, S. (2011). Autonomous mechanical thinning using scanning LIDAR. In *ASABE Annual International Meeting* (p. 1111792). https://doi.org/10.13031/2013.38170

Abrahão, G. S., Megda, P. T., Guerrero, H. B., & Becker, M. (2011). AgriBOT project: Comparison between the D* and focussed D* navigation algorithms. In *International Congress of Mechanical Engineering—COBEM*, Natal, Brazil (pp. 24–28).

Abutalipov, R. N., Bolgov, Y. V., & Senov, H. M. (2016). Flowering plants pollination robotic system for greenhouses by means of nano copter (drone aircraft). In *2016 IEEE Conference on Quality Management, Transport and Information Security, Information Technologies (IT & MQ & IS)* (pp. 7–9). https://doi.org/10.1109/ITMQIS.2016.7751907

Adamides, G., Katsanos, C., Constantinou, I., Christou, G., Xenos, M., Hadzilacos, T., & Edan, Y. (2017). Design and development of a semi-autonomous agricultural vineyard sprayer:

Human–robot interaction aspects. *Journal of Field Robotics, 34*(8), 1407–1426. https://doi.org/10.1002/rob.21721

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

Akbar, S. A., Chattopadhyay, S., Elfiky, N. M., & Kak, A. (2016). A novel benchmark RGBD dataset for dormant apple trees and its application to automatic pruning. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 347–354). https://doi.org/10.1109/CVPRW.2016.50

Amatya, S., Karkee, M., Zhang, Q., & Whiting, M. D. (2017). Automated detection of branch shaking locations for robotic cherry harvesting using machine vision. *Robotics, 6*(4), 31. https://doi.org/10.3390/robotics6040031

Baeten, J., Donné, K., Boedrij, S., Beckers, W., & Claesen, E. (2008). Autonomous fruit picking machine: A robotic apple harvester. In *Field and service robotics: Springer tracts in advanced robotics* (pp. 531–539). Springer. https://doi.org/10.1007/978-3-540-75404-6_51

Balsari, P., & Tamagnone, M. (1998). An ultrasonic airblast sprayer. In *Proceedings of EurAgEng International Conference of Agricultural Engineering* (pp. 585–586). EurAgEng.

Bargoti, S., & Underwood, J. P. (2017). Image segmentation for fruit detection and yield estimation in apple orchards. *Journal of Field Robotics, 34*(6), 1039–1060. https://doi.org/10.1002/rob.21699

Barnett, J., Seabright, M., Williams, H., Nejati, M., Scarfe, A., Bell, J., Jones, M. H., Martinsen, P., Schaare, P., & Duke, M. (2017). Robotic Pollination-Targeting kiwifruit flowers for commercial application. In *PA17 International Tri-Conference for Precision Agriculture,* Hamilton, New Zealand.

Barnett, J., Duke, M., Au, C. K., & Lim, S. H. (2020). Work distribution of multiple Cartesian robot arms for kiwifruit harvesting. *Computers and Electronics in Agriculture, 169*, 105202. https://doi.org/10.1016/j.compag.2019.105202

Berenstein, R., & Edan, Y. (2018). Automatic adjustable spraying device for site-specific agricultural application. *IEEE Transactions on Automation Science and Engineering, 15*(2), 641–650. https://doi.org/10.1109/TASE.2017.2656143

Berenstein, R., Shahar, O. B., Shapiro, A., & Edan, Y. (2010). Grape clusters and foliage detection algorithms for autonomous selective vineyard sprayer. *Intelligent Service Robotics, 3*, 233–243. https://doi.org/10.1007/s11370-010-0078-z

Bergerman, M., Singh, S., & Hamner, B. (2012). Results with autonomous vehicles operating in specialty crops. In *2012 IEEE International Conference on Robotics and Automation* (pp. 1829–1835). https://doi.org/10.1109/ICRA.2012.6225150

Bogue, R. (2016). Robots poised to revolutionise agriculture. *Industrial Robot: An International Journal, 43*(5), 450–456.

Brandtberg, T., Warner, T. A., Landenberger, R. E., & McGraw, J. B. (2003). Detection and analysis of individual leaf-off tree crowns in small footprint, high sampling density LIDAR data from the eastern deciduous forest in North America. *Remote Sensing of Environment, 85*, 290–303.

Bu, L., Hu, G., Chen, C., Sugirbay, A., & Chen, J. (2020). Experimental and simulation analysis of optimum picking patterns for robotic apple harvesting. *Scientia Horticulturae, 261*, 108937. https://doi.org/10.1016/j.scienta.2019.108937

Bucksch, A., & Fleck, S. (2011). Automated detection of branch dimensions in woody skeletons of fruit tree canopies. *Photogrammetric Engineering and Remote Sensing, 77*, 229–240. https://doi.org/10.14358/PERS.77.3.229

Bulanon, D. M., & Kataoka, T. (2010). Fruit detection system and an end effector for robotic harvesting of Fuji apples. *Agricultural Engineering International: CIGR Journal, 12*(1), 203–210.

Bulanon, D. M., Kataoka, T., Ota, Y., & Hiroma, T. (2002). AE—Automation and Emerging Technologies: Segmentation algorithm for the automatic recognition of Fuji apples. *Biosystems Engineering, 83*(4), 405–412.

Büttgen, B., Oggier, T., Lehmann, M., Kaufmann, R., & Lustenberger, F. (2005). CCD/CMOS lock-in pixel for range imaging: Challenges, limitations and state-of-the-art. In *Proceedings of 1st Range Imaging Research Day* (pp. 21–32).

Changhui, Y., Youcheng, H., Lin, H., Sa , L., & Yanping, L. (2017). Overlapped fruit recognition for citrus harvesting robot in natural scenes. In *2nd International Conference on Robotics and Automation Engineering (ICRAE)* (pp. 398–402). https://doi.org/10.1109/ICRAE.2017.8291418

Chattopadhyay, S., Akbar, S. A., Elfiky, N. M., 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)* (pp. 1–9). https://doi.org/10.1109/WACV.2016.7477596

Chen, S. W., Shivakumar, S. S., Dcunha, S., Das, J., Okon, E., Qu, C., Taylor, C. J., & Kumar, V. (2017). Counting apples and oranges with deep learning. *IEEE Robotics and Automation Letters, 2*(2), 781–788. https://doi.org/10.1109/LRA.2017.2651944

Cheng, Z.-L., Zhang, X.-P., & Chen, B.-Q. (2007). Simple reconstruction of tree branches from a single range image. *Journal of Computer Science and Technology, 22*, 846–858. https://doi.org/10.1007/s11390-007-9095-6

Chu, P., Li, V., Lammers, K., Lu, R., & Liu, X. (2021). DeepApple: Deep learning-based apple detection using a suppression mask R-CNN. *Pattern Recognition Letters, 147*, 206–211. https://doi.org/10.1016/j.patrec.2021.04.022

Costes, E., Lauri, P. E., & Regnard, J. L. (2006). Analyzing fruit tree architecture: Implications for tree management and fruit production. *Horticultural Reviews, 32*, 1–61.

Davidson, J. R., Silwal, A., Hohimer, C. J., Karkee, M., Mo, C., & Zhang, Q. (2016). Proof-of-concept of a robotic apple harvester. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 634–639). https://doi.org/10.1109/IROS.2016.7759119

De-An, Z., Jidong, L., Wei, J., Ying, Z., & Yu, C. (2011). Design and control of an apple harvesting robot. *Biosystems Engineering, 110*(2), 112–122. https://doi.org/10.1016/j.biosystemseng.2011.07.005

Dias, P. A., Tabb, A., & Medeiros, H. (2018). Apple flower detection using deep convolutional networks. *Computers in Industry, 99*, 17–28. https://doi.org/10.1016/j.compind.2018.03.010

Edan, Y., Flash, T., Peiper, U. M., Shmulevich, I., & Sarig, Y. (1991). Near-minimum-time task planning for fruit-picking robots. *IEEE Transactions on Robotics and Automation, 7*(1), 48–56.

Elfiky, N. M., Akbar, S. A., 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 *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 65–73). https://doi.org/10.1109/CVPRW.2015.7301298

Emery, K. G., Faubion, D. M., Walsh, C. S., & Tao, Y. (2010). Development of 3-D range imaging system to scan peach branches for selective robotic blossom thinning. In *American Society of Agricultural and Biological Engineers Annual International Meeting,* Pittsburgh, Pennsylvania: ASABE 2010. https://doi.org/10.13031/2013.29828

FAO. (2020). *Statistical year book*. FAO.

Fathallah, F. A. (2010). Musculoskeletal disorders in labor-intensive agriculture. *Applied Ergonomics, 41*(6), 738–743. https://doi.org/10.1016/j.apergo.2010.03.003

Fetzer, L. (2017). *Penn State Extension-orchard safety: Tips for safe pruning*. Retrieved April 15, 2022, from https://extension.psu.edu/orchard-pruning-safety-tips

Feucht, J. R., & Larsen, H. (2009). *Training and pruning fruit trees.* Colorado State University: In Colorado State University Extension; Fact Sheet No. 7.003.

Font, D., Pallejà, T., Tresanchez, M., Runcan, D., Moreno, J., Martínez, D., Teixidó, M., & Palacín, J. (2014). A proposal for automatic fruit harvesting by combining a low cost stereovision camera and a robotic arm. *Sensors, 14*(7), 11557–11579. https://doi.org/10.3390/s140711557

Gallardo, K., Taylor, M., & Hinman, H. (2011). *Cost estimates of establishing and producing Gala apples in Washington*. Washington State University. Retrieved April 15, 2022, from https://pubs.extension.wsu.edu/2009-cost-estimates-of-establishing-and-producing-gala-apples-in-washington

Gebbers, R., Pflanz, M., Betz, A., Hille, B., Mattner, J., Autrum, T. R., Özyurtlu, M., Schischmanow, A., Scheele, M., Schrenk, J., & Schrenk, L. (2013). OptiThin—Implementation of precision horticulture by tree-specific mechanical thinning. In G.-J. i. Potsdam (Ed.), *Massendatenmanagement in Der Agrar- Und Ern{ä}hrungswirtschaft* (pp. 95–98).

Giles, D. K., Delwiche, M. J., & Dodd, R. B. (1987). Control of orchard spraying based on electronic sensing of target characteristics. *Transactions of the ASAE, 30*(6), 1624–1630.

Giles, D. K., Delwiche, M. J., & Dodd, R. B. (1988). Electronic measurement of tree canopy volume. *Transactions of the ASAE, 31*(1), 264–272.

Häni, N., Roy, P., & Isler, V. (2018). Apple counting using convolutional neural networks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 2559–2565). https://doi.org/10.1109/IROS.2018.8594304

Hansen, M. (2011). *Good fruit grower*. Retrieved April 5, 2022, from https://www.goodfruit.com/fruiting-walls-suit-machinery/

He, L., & Schupp, J. (2018). Sensing and automation in pruning of apple trees: A review. *Agronomy, 8*(10), 211. https://doi.org/10.3390/agronomy8100211

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

Hohimer, C. J., 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. *Transactions of the ASABE, 62*(2), 405–414. https://doi.org/10.13031/trans.12986

Horton, R., Cano, E., Bulanon, D., & Fallahi, E. (2017). Peach flower monitoring using aerial multispectral imaging. *Journal of Imaging, 3*(1), 2. https://doi.org/10.3390/jimaging3010002

Jähne, B., Haußecker, H., & Geißler, P. (1999). In B. Jähne, H. Haußecker, & P. Geißler (Eds.), *Handbook of computer vision and applications* (Vol. 1–2). Academic Press.

Jeon, H. Y., Zhu, H., Derksen, R., Ozkan, E., & Krause, C. (2011). Evaluation of ultrasonic sensor for variable-rate spray applications. *Computers and Electronics in Agriculture, 75*(1), 213–221. https://doi.org/10.1016/j.compag.2010.11.007

Ji, W., Zhao, D., Cheng, B. X., Zhang, Y., & Wang, J. (2012). Automatic recognition vision system guided for apple harvesting robot. *Computers & Electrical Engineering, 38*(5), 1186–1195. https://doi.org/10.1016/j.compeleceng.2011.11.005

Kang, F., Wang, H., Pierce, F. J., Zhang, Q., & Wang, S. (2012). Sucker detection of grapevines for targeted spray using optical sensors. *Transactions of the ASABE, 55*(5), 2007–2014.

Kang, H., Zhou, H., & Chen, C. (2020). Visual perception and modeling for autonomous apple harvesting. *IEEE Access, 8*, 62151–62163. https://doi.org/10.1109/ACCESS.2020.2984556

Karkee, M., & Adhikari, B. (2015). A method for three-dimensional reconstruction of apple trees for automated pruning. *Transactions of the ASABE, 58*(3), 565–574. https://doi.org/10.13031/trans.58.10799

Karkee, M., Adhikari, B., Amatya, S., & Zhang, Q. (2014). Identification of pruning branches in tall spindle apple trees for automated pruning. *Computers and Electronics in Agriculture, 103*, 127–135. https://doi.org/10.1016/j.compag.2014.02.013

Kim, J., Kim, S., Ju, C., & Son, H. I. (2019). Unmanned aerial vehicles in agriculture: A review of perspective of platform, control, and applications. *IEEE Access, 7*, 105100–105115. https://doi.org/10.1109/ACCESS.2019.2932119

Kurtulmus, F., Lee, W. S., & Vardar, A. (2011). Green citrus detection using 'eigenfruit', color and circular Gabor texture features under natural outdoor conditions. *Computers and Electronics in Agriculture, 78*(2), 140–149. https://doi.org/10.1016/j.compag.2011.07.001

Lange, R. (2000). *Time-of-flight distance measurement with solid-state image sensors in CMOS/CCD-technology*. University of Siegen.

Lehnert, R. (2012). *Good fruit grower*. Retrieved April 13, 2022, from https://www.goodfruit.com/robotic-pruning/

Linker, R., Cohen, O., & Naor, A. (2012). Determination of the number of green apples in RGB images recorded in orchards. *Computers and Electronics in Agriculture, 81*, 45–57. https://doi.org/10.1016/j.compag.2011.11.007

Liu, H., & Zhu, H. (2016). Evaluation of a laser scanning sensor in detection of complex-shaped targets for variable-rate sprayer development. *Transactions of the ASABE, 59*(5), 1181–1192.

Longtao, M., Yadong, L., Yongjie, C., Haibin, L., Lipeng, C., Longsheng , F., & Yoshinori, G. (2017). Design of end-effector for Kiwifruit Harvesting Robot Experiment. In *An ASABE Meeting Presentation.* https://doi.org/10.13031/aim.201700666

Lopes, C., Graça, J., Sastre, J., Reyes, M., Guzman, R., Braga, R., Monteiro, A., & Pinto, P. (2016). Vineyard yield estimation by Vinbot robot-preliminary results with the white variety Viosinho. In *11th International Terroir Congress*, McMinnville, Oregon. https://doi.org/10.13140/RG.2.1.3912.0886

Lowe, T., Moghadam, P., Edwards, E., & Williams, J. (2021). Canopy density estimation in perennial horticulture crops using 3D spinning lidar SLAM. *Journal of Field Robotics, 38*(4), 598–618. https://doi.org/10.1002/rob.22006

Luo, L., Tang, Y., Lu, Q., Chen, X., Zhang, P., & Zou, X. (2018). A vision methodology for harvesting robot to detect cutting points on peduncles of double overlapping grape clusters in a vineyard. *Computers in Industry, 99*, 130–139. https://doi.org/10.1016/j.compind.2018.03.017

Lyons, D. J., Heinemann, P. H., Schupp, J. R., Baugher, T. A., & Liu, J. (2015). Development of a selective automated blossom thinning system for peaches. *Transactions of the ASABE, 58*(6), 1447–1457. https://doi.org/10.13031/trans.58.11138

Maccarrone, G., & Scienza, A. (1998). Valutazione della struttura fogliare delle forma di allevamento della vite per ottimizzare i trattamenti. *Vignevini, 5*, 46–48.

Mahdi, H. S., & Abdolabbas, J. (2017). Designing an algorithm for pruning grapevine based on 3D image processing. *Iranian Journal of Biosystems Engineering (Iranian Journal of Agricultural Sciences), 48*(3), 289–297.

Martn, B., Gutiérrez, A., & Molto, E. (2001). Pesticide loss reduction by automatic adaptation of spraying on globular trees. *Journal of Agricultural Engineering Research, 78*, 35–41.

McCann, J. (2014). *Retinex theory*. Encyclopedia of Color Science and Technology.

Medeiros, H., Kim, D., Sun, J., Seshadri, H., Akbar, S. A., Elfiky, N. M., & Park, J. (2017). Modeling dormant fruit trees for agricultural automation. *Journal of Field Robotics, 34*(7), 1203–1224. https://doi.org/10.1002/rob.21679

Mehta, S. S., & Burks, T. F. (2014). Vision-based control of robotic manipulator for citrus harvesting. *Computers and Electronics in Agriculture, 102*, 146–158. https://doi.org/10.1016/j.compag.2014.01.003

Méndez, V., Rosell-Polo, J. R., Sanz, R., Escolà, A., & Catalán, H. (2014). Deciduous tree reconstruction algorithm based on cylinder fitting from mobile terrestrial laser scanned point clouds. *Biosystems Engineering, 124*, 78–88. https://doi.org/10.1016/j.biosystemseng.2014.06.001

Nielsen, M., Slaughter, D. C., & Gliever, C. (2012). Vision-based 3D peach tree reconstruction for automated blossom thinning. *IEEE Transactions on Industrial Informatics, 8*(1), 188–196. https://doi.org/10.1109/TII.2011.2166780

Oberti, R., Marchi, M., Tirelli, P., Calcante, A., Iriti, M., Tona, E., Hočevar, M., Baur, J., Pfaff, J., Schütz, C., & Ulbrich, H. (2016). Selective spraying of grapevines for disease control using a modular agricultural robot. *Biosystems Engineering, 146*, 203–215. https://doi.org/10.1016/j.biosystemseng.2015.12.004

Onishi, Y., Yoshida, T., Kurita, H., Fukao, T., Arihara, H., & Iwai, A. (2019). An automated fruit harvesting robot by using deep learning. *ROBOMECH Journal, 6*, 13. https://doi.org/10.1186/s40648-019-0141-2

Pi, J., Liu, J., Zhou, K., & Qian, M. (2021). An octopus-inspired bionic flexible gripper for apple grasping. *Agriculture, 11*(10), 1014. https://doi.org/10.3390/agriculture11101014

Prusinkiewicz, P., Hammel, M., Hanan, J., & Mech, R. (1990). L-systems: From the theory to visual models of plants. In *Proceedings of the 2nd CSIRO Symposium on Computational Challenges in Life Sciences* (Vol. 3, pp. 1–32). Citeseer.

Rabatel, G., Bourely, A., Sevila, F., & Juste, F. (1995). Robotic harvesting of citrus: State-of-art and development of the French Spanish EUREKA Project. In *International conference Harvest and Postharvest Technologies for Fresh Fruits and Vegetables*, Guanajuato, Mexico (pp. 232–239).

Reis, R., Mendes, J., Santos, F. N., Morais, R., Ferraz, N., Santos, L., & Sousa, A. (2018). Redundant robot localization system based in wireless sensor network. In *Proceedings of the 2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, Torres Vedras, Portugal (pp. 154–159).

Robinson, T. L., DeMarree, A. M., & Hoying, S. A. (2007). An economic comparison of five high density apple planting systems. In *VIII International Symposium on Canopy, Rootstocks and Environmental Physiology in Orchard Systems* (pp. 481–490). ISHS Acta Horticulturae 732. https://doi.org/10.17660/ActaHortic.2007.732.73

Sansavini, S. (1976). Mechanical pruning of fruit trees. *Symposium on High Density Planting, 65*, 183–198. https://doi.org/10.17660/ActaHortic.1978.65.28

Santos, T., & Ueda, J. (2013). Automatic 3D plant reconstruction from photographies, segmentation and classification of leaves and internodes using clustering. In *Embrapa Informática Agropecuária-Resumo em anais de congresso (ALICE)*. Finnish Society of Forest Science.

Santos, F. N., Sobreira, H., Campos, D., Morais, R., Moreira, A. P., & Contente, O. (2016). Towards a reliable robot for steep slope vineyards monitoring. *Journal of Intelligent & Robotic Systems, 83*, 429–444. https://doi.org/10.1007/s10846-016-0340-5

Sarri, D., Martelloni, L., Rimediotti, M., Lisci, R., Lombardo, S., & Vieri, M. (2019). Testing a multi-rotor unmanned aerial vehicle for spray application in high slope terraced vineyard. *Journal of Agricultural Engineering, 50*(1), 38–47. https://doi.org/10.4081/jae.2019.853

Schupp, J. R., & Baugher, T. A. (2011). Peach blossom string thinner performance improved with selective pruning. *HortScience, 46*(11), 1486–1492. https://doi.org/10.21273/HORTSCI.46.11.1486

Schupp, J. R., Winzeler, H. E., Kon, T. M., Marini, R. P., Baugher, T. A., Kime, L. F., & Schupp, M. A. (2017). A method for quantifying whole-tree pruning severity in mature tall spindle apple plantings. *HortScience, 52*(9), 1233–1240.

Schwarte, R., Heinol, H., Buxbaum, B., & Ringbeck, T. (1999). Principles of three-dimensional imaging techniques. In B. Jähne, H. Haußecker, & P. Greißler (Eds.), *Handbook of computer vision and applications* (pp. 464–482). Academic Press.

Silwal, A., Gongal, A., & Karkee, M. (2014). Identification of red apples in field environment with over the row machine vision system. *Agricultural Engineering International: CIGR Journal, 16*(4), 66–75.

Silwal, A., Davidson, J. R., Karkee, M., Mo, C., Zhang, Q., & Lewis, K. (2017). Design, integration, and field evaluation of a robotic apple harvester. *Journal of Field Robotics, 34*(6), 1140–1159. https://doi.org/10.1002/rob.21715

Simonite, T. (2017). *MIT Technology Review*. Retrieved May 23, 2022, from https://www.technologyreview.com/2017/05/03/152012/apple-picking-robot-prepares-to-compete-for-farm-jobs/

Sirovich, L., & Kirby, L. (1987). Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A: Optics, Image Science, and Vision, 4*, 519–524.

Stajnko, D., Rakun, J., & Blanke, M. (2009). Modelling apple fruit yield using image analysis for fruit colour, shape and texture. *European Journal of Horticultural Science, 74*(6), 260–267.

Stein, M., Bargoti, S., & Underwood, J. (2016). Image based mango fruit detection, localisation and yield estimation using multiple view geometry. *Sensors, 16*(11), 1915. https://doi.org/10.3390/s16111915

Tabb, A., & Medeiros, H. (2017). A robotic vision system to measure tree traits. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2017* (pp. 6005–6012). https://doi.org/10.1109/IROS.2017.8206497

Tabb, A. L., Peterson, D. L., & Park, J. (2006a). Segmentation of apple fruit from video via background modeling. In *2006 ASABE Annual International Meeting*. Retrieved from https://elibrary.asabe.org/abstract.asp?aid=20873

Tabb, A. L., Peterson, D. L., & Park, J. (2006b). Segmentation of apple fruit from video via background modeling. In *ASABE Annual Meeting*. https://doi.org/10.13031/2013.20873

Turner, D., Lucieer, A., & Watson, C. (2011). Development of an Unmanned Aerial Vehicle (UAV) for hyper-resolution vineyard mapping based on visible, multispectral and thermal imagery. In *The GEOSS Era: Towards Operational Environmental Monitoring* (Vol. 1). Retrieved from https://www.isprs.org/proceedings/2011/isrse-34/211104015Final00547.pdf

Underwood, J. P., Calleija, M., Taylor, Z., Hung, C., Nieto, J. M., Fitch, R., & Sukkarieh, S. (2015). Real-time target detection and steerable spray for vegetable crops. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA (pp. 26–30).

Underwood, J. P., Hung, C., Whelan, B., & Sukkarieh, S. (2016). Mapping almond orchard canopy volume, flowers, fruit and yield using lidar and vision sensors. *Computers and Electronics in Agriculture, 130*, 83–96. https://doi.org/10.1016/j.compag.2016.09.014

Vázquez-Arellano, M., Griepentrog, H. W., Reiser, D., & Paraforos, D. S. (2016). 3-D imaging systems for agricultural applications—A review. *Sensors, 16*(5), 618. https://doi.org/10.3390/s16050618

Walklate, P. J., Hunt, J. C., Higson, H. L., & Sweet, J. B. (2004). A model of pollen–mediated gene flow for oilseed rape. *Proceedings of the Royal Society of London Series, 271*, 441–449. https://doi.org/10.1098/rspb.2003.2578

Wang, C., Tang, Y., Zou, X., SiTu, W., & Feng, W. (2017). A robust fruit image segmentation algorithm against varying illumination for vision system of fruit harvesting robot. *Optik, 131*, 626–631.

Wang, Y., Yang, Y., Yang, C., Zhao, H., Chen, G., Zhang, Z., Fu, S., Zhang, M., & Xu, H. (2019). End-effector with a bite mode for harvesting citrus fruit in random stalk orientation environment. *Computers and Electronics in Agriculture, 157*, 454–470. https://doi.org/10.1016/j.compag.2019.01.015

Whiting, M. (2016). Mechanical pollination for yield security. Project Report to Washington Tree Fruit. Prosser/WA/99350: WSU.

Williams, H. A., Jones, M. H., Nejati, M., Seabright, M. J., Bell, J., Penhall, N. D., Barnett, J., Duke, M., Scarfe, A. J., Ahn, H., Lim, J., & MacDonald, B. A. (2019). Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. *Biosystems Engineering, 181*, 140–156. https://doi.org/10.1016/j.biosystemseng.2019.03.007

Wouters, N., Ketelaere, B. D., Deckers, T., Baerdemaeker, J. D., & Saeys, W. (2015). Multispectral detection of floral buds for automated thinning of pear. *Computers and Electronics in Agriculture, 113*, 93–103. https://doi.org/10.1016/j.compag.2015.01.015

Yang, F. O. (2012). *Development of a table-top robot model for thinning of fruit* [Master's thesis, University of Illinois Urbana-Champlain, USA].

Yongting, T., & Jun, Z. (2017). Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking. *Computers and Electronics in Agriculture, 142*, 388–396.

Zhang, Q. (2017). *Automation in tree fruit production: Principles and practice*. CABI.

Chapter 18
Strategic Short Note: Comparing Soil Moisture Retrieval from Water Cloud Model and Neural Network Using PALSAR-2 for Oil Palm Estates

Veena Shashikant, Abdul Rashid Mohamed Shariff, Aimrun Wayayok, Md Rowshon Kamal, Yang Ping Lee, and Wataru Takeuchi

Abstract It is essential to have the soil moisture retention at the optimal level in order to maintain high yields in oil palm estates. Furthermore, conventional methods for determining soil moisture are difficult, time-consuming, and challenging in the rural estate areas. In this study, synthetic aperture radar (SAR), L-band images, and in situ observations were conducted at an oil palm plantation to employ water cloud model (WCM) inversion for retrieving soil moisture from HH (Horizontal-Horizontal) and HV (Horizontal-Vertical) polarized data. WCM was evaluated by comparing leaf area index (LAI), leaf water area index (LWAI), and normalized plant water content (NPWC), to understand the effects of vegetation on backscattering coefficients. Adding on, neural network (NN) technique was employed to understand capabilities of soil moisture retrievals using mentioned data. Effects of vegetation in the WCM and NN models were then investigated using the k-fold cross validation method to understand the difference in the in situ observations and

V. Shashikant · M. R. Kamal
Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia

A. R. M. Shariff (✉) · A. Wayayok
Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia

SMART Farming Technology Research Center, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia

Laboratory of Plantation System Technology and Mechanization (PSTM), Institute of Plantation Studies (IKP), Universiti Putra Malaysia, Serdang, Malaysia
e-mail: rashidpls@upm.edu.my

Y. P. Lee
FGV R & D Sdn Bhd, Level 9, Wisma FGV, Kuala Lumpur, Malaysia

W. Takeuchi
Department of Human and Social Systems, Institute of Industrial Science, The University of Tokyo, Tokyo, Japan

modelled results. The results demonstrated that HV polarization efficiently approximated the backscatter coefficient compared to HH polarization, while the best fit was achieved by using the LAI as a vegetation descriptor in the WCM model with an accuracy of at least $R^2 = 0.9460$ with RMSE of 0.036 m^3/m^3 whereas the NN model was able to improvise soil moisture content with $R^2 = 0.9638$ and RMSE of 0.012 m^3/m^3.

Keywords Backscatter coefficient · Soil moisture retrieval · LAI · NN · HH and HV polarization

18.1 Introduction

Oil palm trees requires year-round high temperatures and rainfall where prolonged dry periods of 2–3 months can affect yield output (Oettli et al., 2018). Gravimetry to evaluate soil moisture is a highly reliable method but only reflect a small region that changes around the sampling sites. Active remote sensing using SAR sensors, on the other hand, are employed in this study, to measure soil moisture over large areas and due to its ability to penetrate to a depth of approximately 5 cm below the surface as seen in Fig. 18.1. WCM is preferred for its simple technique where it defines the overall backscatter coefficient obtained by the PALSAR-2 over vegetated surfaces as the incoherent sum of the effects of vegetation and soil; thus, enabling retrieval of soil moisture in the oil palm estates (Shashikant et al., 2021). Adding on, single-layer NN approach was built to understand the capabilities of artificial intelligence in retrieving soil moisture in the oil palm estates using the HH and HV polarization with the incidence of PALSAR-2 as seen in the Fig. 18.2.

This study's objective was to retrieve soil moisture from oil palm fields and evaluating the suitable vegetation indicator to be considered in WCM implementing
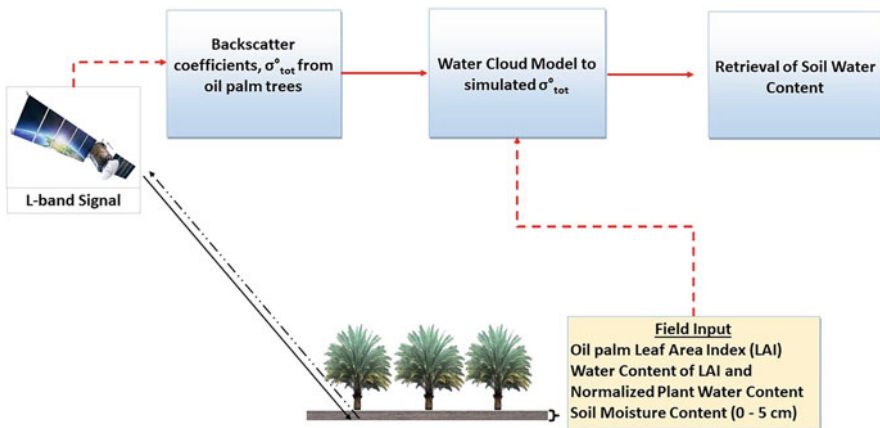
Fig. 18.1 The concept of retrieving soil moisture from oil palm estates using WCM approach
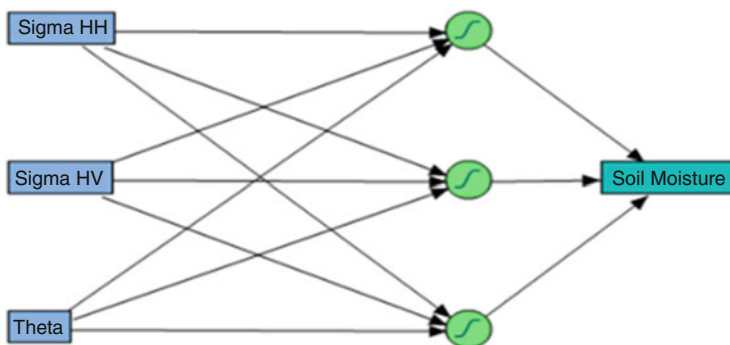
Fig. 18.2 The concept of retrieving soil moisture from oil palm estates using Neural Network (NN) approach

the k-fold cross validation. Secondly, the NN was evaluated with similar cross validation k-fold technique to enable a comparison of the soil moisture retrievals.

18.2 Retrieval of Soil Moisture Content in Oil Palm Fields

PALSAR-2 signals can strongly penetrate the cloud layer as well as the vegetation layer therefore important as oil palm crop have a large canopy to bypass to understand the amount of water present in the soil. As a result, estimating the water content of the soil at the size of an entire field using radar remote sensing is of utmost significance of this study. Numerous studies have shown that the vegetation type, the geometric structure of its cover (including height, branch and leaf shapes, and density distribution), and the water content of the plant canopy all influence radar backscattering and radar wave transmittance in the plant canopy (Liu et al., 2019). In this study, multitemporal SAR data inversion was utilized to assist eliminate the effects of the plant layer on the radar backscatter to decrease mistakes in the soil moisture content data. Overall, HV polarization, utilizing LWAI as vegetation descriptor yielded an R^2 value of 0.951 with an RMSE of 0.046 m^3/m^3, while LAI yielded a smaller error with an RMSE of 0.036 m^3/m^3 and an R^2 value of 0.946 in the testing dataset using k-fold technique as seen in Table 18.1. In contrast, the highest RMSE was obtained in the scenarios utilizing LAI with constant 1, where the RMSE was 0.059 m^3/m^3 in the testing dataset. For the NN model results as shown in Fig. 18.3 the soil moisture retrievals were more accurate with RMSE = 0.012 m^3/m^3.

Table 18.1 The retrievals of soil moisture using SAR vs. field soil moisture in the testing data set using proposed V_1 and V_2 as the vegetation indicator in WCM

Image Polarization	Description (n=19)	V1 =1 & V2 = LAI	V_1 =LAI & V_2= 1	V_1 = V_2= LAI	V_1 = V_2 = LWAI	V_1 = V_2 = NPWC
	R^2	0.942	0.985	0.949	0.964	0.985
HH	RMSE (m³/m³)	0.086	0.073	0.095	0.083	0.077
	MAE	0.078	0.072	0.088	0.077	0.077
	R^2	0.885	0.906	0.946	0.951	0.944
HV	RMSE (m³/m³)	0.059	0.059	0.046*	0.036*	0.047*
	MAE	0.058	0.055	0.044*	0.035*	0.047*

Note: *According to the Global Monitoring for Environment and Security (GMES) standards and requirement of RMSE less then 0.05 m³/m³ is accepted for soil moisture retrievals using remote sensing approach.
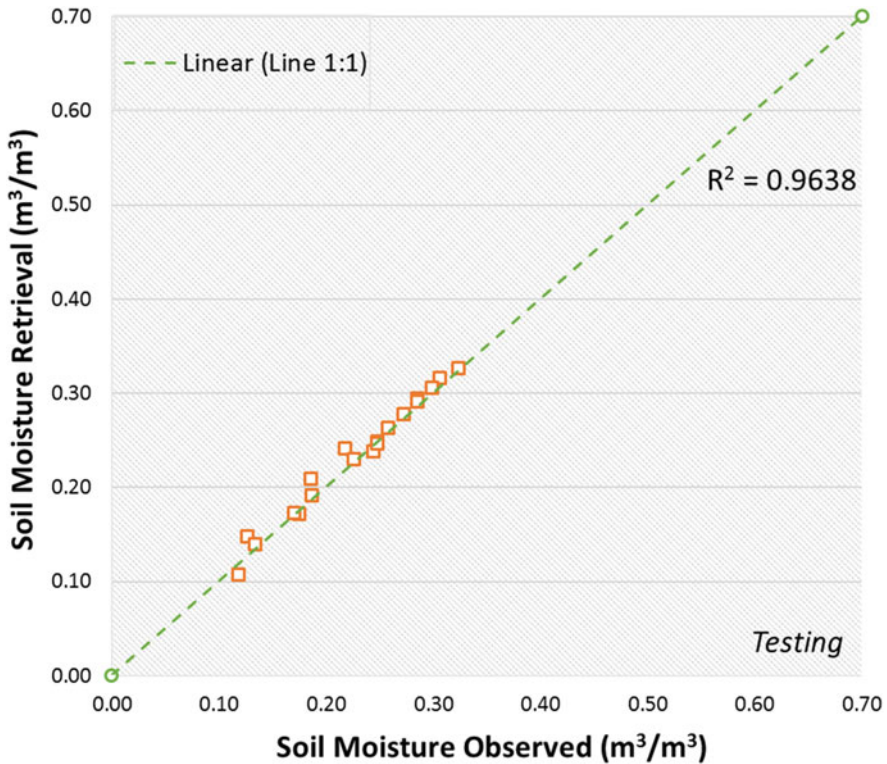
Fig. 18.3 The retrievals of soil moisture using NN vs. field soil moisture in the testing data set

18.3 Conclusion

In summary, the cross-polarized HV backscatter coefficient is revealed to be more vulnerable than the co-polarized backscatter HH in utilizing the WCM model. On the other hand, NN has been able to predict soil moisture content directly using PALSAR-2 images in oil palm estates. This research gives a useful and novel contribution on soil moisture for oil palm cultivation using PALSAR-2 sensor data. This research will help oil palm estates to prepare for climate-related droughts and optimizing soil moisture by showcasing the capabilities of the WCM and NN. This study demonstrated the Global Monitoring for Environment and Security (GMES) criterion was met in the LAI, LWAI, and NPWC in the HV polarization. In addition, NN technique has able to fulfill the GMES criteria for soil moisture retrievals.

References

Liu, C., Chen, Z., Shao, Y., Chen, J., Hasi, T., & Pan, H. (2019). Research advances of SAR remote sensing for agriculture applications: A review. *Journal of Integrative Agriculture, 18*(3), 506–525. https://doi.org/10.1016/S2095-3119(18)62016-7

Oettli, P., Behera, S. K., & Yamagata, T. (2018). Climate based predictability of oil palm tree yield in Malaysia. *Scientific Reports, 8*(1), 2271. https://doi.org/10.1038/s41598-018-20298-0

Shashikant, V., et al. (2021). Vegetation effects on soil moisture retrieval from water cloud model using PALSAR-2 for oil palm trees. *Remote Sensing, 13*(20), 4023.

Chapter 19
Development of a Recognition System for Spraying Areas from Unmanned Aerial Vehicles Using a Machine Learning Approach

Pengbo Gao, Yan Zhang, Linhuan Zhang, Ryozo Noguchi, and Tofael Ahamed

Abstract Unmanned aerial vehicle (UAV)-based spraying systems have recently become important for the precision application of pesticides using machine learning approaches. Therefore, the objective of this research was to develop a machine learning system that has advantages of high computational speed and good accuracy for recognizing spray and non-spray areas for UAV-based sprayers. A machine learning system was developed using the mutual subspace method for images collected from a UAV. Two target lands: agricultural croplands and orchard areas were considered in building two classifiers for distinguishing spray and non-spray areas. The field experiments were conducted in target areas to train and test the system using a commercial UAV (DJI Phantom 3 Pro) with an onboard 4K camera. The images were collected from low (5 m) and high (15 m) altitudes for croplands and orchards, respectively. The recognition system was divided into offline and online systems. In the offline recognition system, 74.4% accuracy was obtained for the classifiers in recognizing spray and non-spray areas for croplands. In the case of orchards, the average classifier recognition accuracy of spray and non-spray areas was 77.0%. On the other hand, the online recognition system performance had an average accuracy of 65.1% for croplands and 75.1% for orchards. The computational time for the online recognition system was minimal, with an average of 0.0031 s for classifier recognition. The developed machine learning system had an average recognition accuracy of 70%, which can be implemented in an autonomous UAV spray system for recognizing spray and non-spray areas for real-time applications.

P. Gao · Y. Zhang · L. Zhang
Graduate School of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan

R. Noguchi · T. Ahamed (✉)
Faculty of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

Keywords Recognition system · Image classifiers · Machine learning system ·
Mutual subspace method

19.1 Introduction

With the development of unmanned aerial vehicle (UAV) technologies, the use of
UAVs has rapidly expanded to different applications such as aerial photography to
monitor vegetation, survey mapping, and scouting with wireless networking (Zhang
et al., 2018a). UAVs have potential for use in agricultural applications and are ideal
for precision agriculture compared to aerial mapping and satellite remote sensing.
The use of UAVs is not only more efficient but also more cost-effective than areal or
high-resolution commercial satellite datasets (Zhang et al., 2018a; Peteinatos et al.,
2014; Manfreda et al., 2018; Bandini et al., 2017; Zarco-Tejada et al., 2012). They
can help monitor crops in real time and provide high-resolution images of the field
and canopy for crop growth and production. High-resolution and machine vision
images are used for the identification of weeds and non-weed areas using ground-
based conventional sprayers (Hung et al., 2014; Rebetez et al., 2016). In recent
advancements, sprayers have been attached to UAV systems to deliver spray in the
field. However, as the payload of a UAV with a sprayer makes it heavier, it becomes
difficult to fly in the field while carrying large quantities of liquid chemicals. The
process of spraying agricultural crops with liquids needs to be very efficient to avoid
spraying non-crop areas. Similarly, the orchard spray system needs to fly at a high
altitude to spray chemicals on the top of the canopy. High payloads of chemicals in
the tank also cause problems. Large tank sizes require more power and generate
more safety concerns while flying. It is very important to recognize the spray area
above orchard and non-orchard areas to ensure precise application of spray
chemicals. For autonomy of UAV-based spraying systems, the ability to recognize
crop and orchard areas is significantly important. Most autonomous and artificial
intelligence systems need to be trained on data prior to application. Training pro-
vides confidence in operations. Machine learning systems have the potential for
training and testing on datasets to add artificial intelligence for different agricultural
operations. In the spraying system, a machine learning system is required to discern
spraying spots and non-spraying spots in operational environments of UAVs prior to
implementing an autonomous spraying system. Ground-based vehicles can function
as image processing systems with advantages for housing onboard sprayers
(Peteinatos et al., 2014; Lee et al., 2012). However, ground vehicles have local
mapping systems without predetermined field coverage. UAVs have the advantage
of identifying the field coverage in advance with good trained datasets and machine
learning systems.

Most studies using UAV have employed limited dataset collection and machine
learning systems. Some research has reported that aerial applications result in only
50% of targets being sprayed from altitudes less than 1 m (Pimentel & Burgess,
2012). To the best of our knowledge, UAV-based sprayers were introduced to the
market and largely implemented in mountains and crop areas to enable spraying with

precision. Most commercial UAVs with sprayers are operated under regulations in many countries. As the technology is tending to develop toward autonomous systems, it is likely that UAV spray systems will have a high potential for autonomous spraying applications. To increase the flight time, UAV manufacturers have improved the endurance of systems by increasing the battery capacity and reducing the total weight of UAVs. In this regard, the application efficiency of chemicals to the spray area from a high altitude must be improved. The height of operation greatly influences how orchard areas can be covered in the minimum amount of time. A similar consideration in selecting the height of operation pertains to croplands.

Unmanned aerial systems (UASs) have mounted sensors that offer an extraordinary opportunity to bridge the existing gap between field observations and traditional air and space-borne remote sensing by providing high spatial detail over relatively large areas in a cost-effective way. UAV systems also have enhanced temporal retrieval for natural and agricultural ecosystem monitoring in order to identify future directions, applications, developments, and challenges (Manfreda et al., 2018). Small UAVs (approximately 1.5 kg) are capable of measuring the range to a water surface using a global navigation satellite system (GNSS) receiver with radar, sonar, and an in-house developed camera-based laser distance sensor (CLDS) for determining the water level (orthometric height) (Bandini et al., 2017). The remote detection of water stress in a citrus orchard was also investigated using leaf-level measurements of chlorophyll fluorescence and photochemical reflectance index (PRI) data, seasonal time-series of crown temperature and PRI, and high-resolution airborne imagery (Zarco-Tejada et al., 2012). These researches have the significant impacts on the application of UAV in agriculture and hydrology. Also opened further avenues to introduce machine learning systems from remote sensing applications for UAV-based sprayers.

How to generate quantitative remote sensing products using rotating-wing and fixed-wing UAVs equipped with commercial off-the-shell (COTS) thermal and narrowband multispectral imaging sensors was also evaluated for vegetation monitoring. Radiometric calibration, atmospheric correction, and photogrammetric methods were employed, which are required to obtain accurate remote sensing products that are useful for vegetation monitoring (Berni et al., 2009a, b). The growing research community comprises tech-enthusiastic hydrologists that aim to design and develop their own sensing systems and adopt a multi-disciplinary perspective to tackling complex observations, often using low-cost equipment intended for other applications to build innovative sensors for measurement (Tauro et al., 2018). A previously reported UAV system was integrated with optical sensing to allow quantitative characterization of surface flow phenomena to yield accurate surface flow maps of sub-meter water bodies (Tauro et al., 2016). Vegetation monitoring and surface flow phenomena help in further confirmation of UAV application for recognition of classifiers. These researches significantly increase the scope of using UAV applications in the research community.

UAVs equipped with inexpensive thermal and narrowband multispectral imaging sensors have been used for agricultural applications and yielded comparable estimations (Berni et al., 2009a, b). Relying on UAV-based remote sensing and imaging

techniques, high-throughput field phenotyping (HTFP) was conducted using thermal imaging for field phenomics of poplar and other tree species for accelerating forest tree genetic improvement against abiotic stress (Ludovisi et al., 2017). HFTP and genetic improvement against abiotic stress also increase further application of machine learning systems. Several machine learning systems have been introduced in ground-based sprayers using deep learning, neural networks and Bayesian classifiers (Carrio et al., 2017; Majidi & Bab-Hadiashar, 2005; Tellaeche et al., 2008). Most machine learning systems have high complexity data training and large time requirements for real-time application. In our previous research, we found that the kernel mutual subspace method (KMSM) has a high potential to recognize features and actions of tracking with accuracy greater than 80% in real time (Zhang et al., 2018b). Furthermore, the KMSM along with the Hankel matrix were used for action recognition of machinery operator with a processing time of 0.07 (Zhang et al., 2018b). The subspace method has been used for the recognition of human faces and objects (Sakano et al., 2005; Fukui & Yamaguchi, 2005; Fukui & Yamaguchi, 2007). Therefore, the subspace method has a highly promising capability in machine learning systems for recognizing features of agricultural environments. In on-board spraying applications using UAVs, the recognition of features with minimum time and high accuracy can be performed using mutual subspace method (MSM).

Therefore, the objective of this research is to develop a machine learning system for recognizing the features of spraying and non-spraying areas for applying UAV-based sprayers in agricultural croplands and orchards. It is expected that MSM machine learning systems can be employed, offering advantages of low computational complexity and good accuracy in feature recognition systems for real-time applications.

19.2   Materials and Methods

19.2.1   Mutual Subspace Method (MSM)

The MSM was introduced to the field of pattern recognition, a well-known method for object recognition based on image set (Maeda & Watanabe, 1985). MSM is an extension of the Subspace Method (SM) that classifies a set of input pattern vectors into several classes based on multiple canonical angles between the input subspace and the class subspaces (Fig. 19.1). The input subspace is generated from a set of input patterns as a class (Maeda & Watanabe, 1985; Watanabe & Pakvasa, 1973). The SM has high performance in pattern recognition and was developed independently as CLAFIC (class-SELFIC method: the original version of the subspace model) (Iijima et al., 1974) and the multiple similarity method (Yamaguchi et al., 1998). It classifies an input pattern vector into several classes based on the minimum distance or angle between the input pattern vector and each class subspace, where a class subspace corresponds to the distribution of pattern vectors of the class in high-dimensional vector space (Fukui & Yamaguchi, 2014).

Fig. 19.1 Subspace method
(SM)
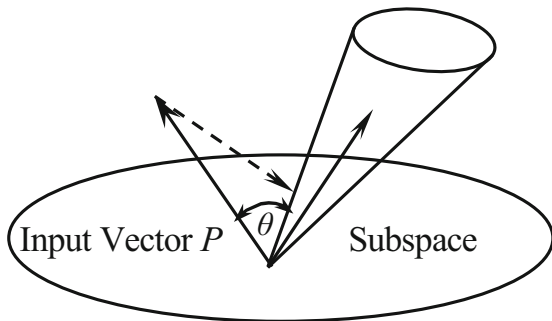
We considered that the input vector P and m class subspaces belong to a k-dimensional vector space; the similarity is defined by the length or the minimum angle between the input vector P and the ith class subspace, where the length of P is often normalized to 1.0. The angle-based similarity can be derived as follows:

$$\cos^2\theta = \sum_{i=1}^{d} \frac{(P \cdot \varphi_i)^2}{\|P\|^2} \tag{19.1}$$

where d is the dimension of the class subspace, and φ_i is the ith k-dimensional orthogonal normal vector (PCA). First, the conventional PCA operates by diagonalizing the covariance matrix C from k feature vectors $\vec{x_j}$ ($a = 1, 2, \ldots, k$) in an n-dimensional feature space, $\mathcal{R}^n$,

$$C = \frac{1}{k}\sum_{j=1}^{k}\left(\vec{x_j} \cdot \vec{x_j}^T\right) \tag{19.2}$$

It gives an eigen decomposition of the covariance matrix by PCA to obtain the principal components $\vec{\nu_i}$ ($i = 1, 2, \ldots, k$) of the distribution:

$$\lambda \vec{\nu} = C \vec{\nu} \tag{19.3}$$

However, we assume that all data here were calculated from the data centroid. This principal component describes the direction of the largest data variation under a linear approximation. The above characteristic equation can be transformed as follows:

$$\lambda \vec{x} = \left[\frac{1}{k}\sum_{j=1}^{k}\left(\vec{x_j} \cdot \vec{x_j}^T\right)\right]\vec{\nu} \tag{19.4}$$
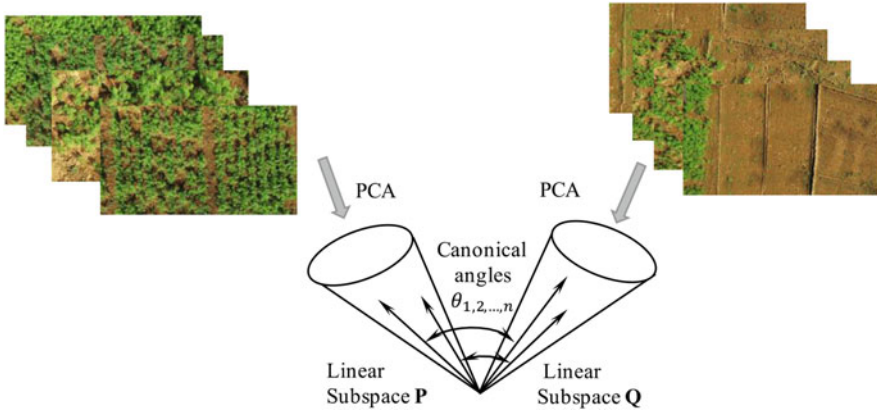
Fig. 19.2 Comparison between two sets of images using MSM

$$= \frac{1}{k} \sum\nolimits_{j=1}^{k} \left(\vec{x_j} \cdot \vec{x_j}^T \right) \vec{\nu} = \frac{1}{k} \sum\nolimits_{j=1}^{k} \overrightarrow{(x_j \cdot \vec{\nu})} \vec{x_j} \tag{19.5}$$

Because $\vec{\nu}$ is in $\{x_1 \ldots, x_k\}$, we obtain

$$\lambda \left(\vec{x_a} \cdot \vec{\nu} \right) = \vec{x_a} \cdot C \vec{\nu} \tag{19.6}$$

where $\vec{xa}$ is a feature vector.

The MSM has been used to compare small variations in training data and recognition target data and results in a powerful recognition technique when the data distribution can be linearly approximated, which occurs when multiple data can be used as recognition target image inputs. In the subspace method, a subspace that has d-dimensional vectors is selected according to a criterion such as the cumulative contribution rate from the eigenvectors, which are obtained using PCA on the entered images (Fukui & Yamaguchi, 2007; Yamaguchi et al., 1998; Schölkopf et al., 1998). Then, the similarity between subspaces is defined according to the angle θ between eigenvectors $P = \{\vec{\mu i}\}$ (registered as a dictionary) and the eigenvectors $Q = \{\vec{\nu j}\}$ (obtained from the input data) (Fig. 19.2).

According to eq. (1), the angle θ between subspaces is given as the maximum eigenvalue (Fukui & Yamaguchi, 2005; Fukui & Yamaguchi, 2007).

$$\cos \theta = \max_{\vec{\mu_i} \in P} \max_{\vec{\nu_i} \in Q} \vec{\mu}^T \vec{\nu} \tag{19.7}$$

where $\vec{\mu_i}^T \vec{\mu_i} = \vec{\nu_j}^T \vec{\nu_j} = 1$, $\vec{\mu_i}^T \vec{\mu_j} = \vec{\nu_i}^T \vec{\nu_j} = 0$, $i \neq j$, $0 < i, j \leq d$, and d is the dimensionality of the subspace used for recognition.

19.2.2 Research Design for Classifiers and MSM

The classifiers are required to be established before the MSM application. The MSM research approach involves two steps: offline recognition and online recognition systems. The offline recognition system was used to validate the model and the accuracy of the recognition of classifiers (Fig. 19.3). The online recognition system was proposed to understand the computational times to enable in the real-time system. In offline recognition, videos must be captured using the UAV and converted through a JPG converter. For offline recognition, selected images were taken for training and testing the classifiers from different datasets of crops and orchards. For online recognition, a new video stream was captured. From the stream video, 1 frame was chosen out of 20 frames from a new video stream. Considering the restricted computational time required by a real-time system, Red-Green-Blue (RGB) images were converted to the gray scale. While testing using the online recognition system, a sliding window was used to obtain four consecutive frames, and principal components analysis (PCA) was applied using the subspace method. In the subspace method, multiples images were required, and we noted that four frames were optimal for use in the subspace method.

In the following sections, details of field experiments for training and testing different datasets and the offline and online recognition systems are described.
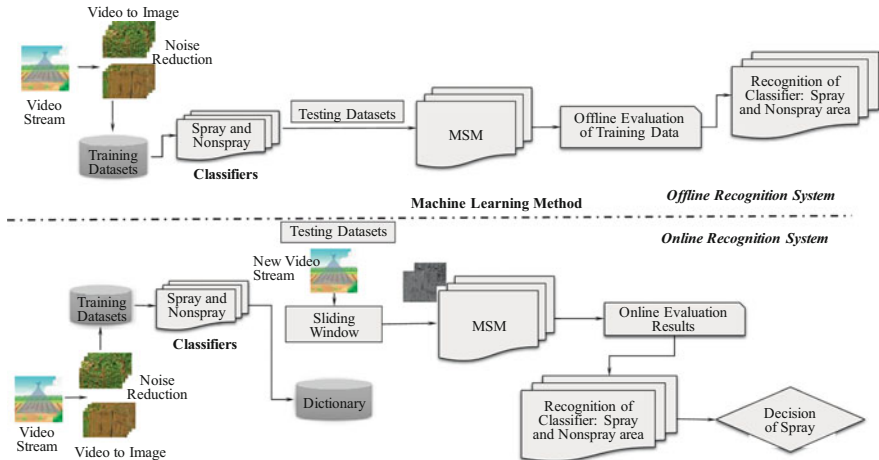
Fig. 19.3 Research framework establishing the classifiers and the MSM

19.2.3 Field Experiment for Training and Testing with Datasets

In order to implement the MSM for feature recognition, different crops, orchards, and fields are required for training the subspace patterns and verifying the recognition accuracy. While selecting the datasets for training, image acquisition at close range is preferable for agricultural croplands. On the other hand, for orchards, a high altitude allows the canopy to be covered in a minimum time. Generally, close-range spraying can effectively reduce the drift and waste of chemicals.

However, UAV sprayer payload and battery operational time are major concerns in enabling autonomous spraying. In this study, two working patterns are defined depending on the flying height. The corresponding work areas are described as follows: for cropland (i.e., carrot, cabbage, and onions), plant height was less than 5 m, and image acquisition was performed using a UAV from a height of 5 m. In the case of orchards or plantations (i.e., chestnut, persimmons, and tall trees), we considered the height of orchards to be less than 15 m, and thus, the acquisition of images was conducted from a height of 15 m from the ground (Table 19.1). Two classifier datasets were collected for cropland spray area recognition: one dataset for spray areas (carrot, cabbage, onions) and one dataset for non-spray areas (inner farm roads, ridges, bare soil). Similarly, two classifier datasets (spray and non-spray areas) were also collected for orchards: one dataset for orchard areas (chestnuts and persimmon) and another dataset for trees that included structured areas (farm houses, green house structure, farm buildings). The classifier datasets were captured using a commercial UAV (DJI Phantom 3 Pro) with an onboard 4K camera with 1/2.3″ CMOS and FOV 94° 20 mm f/2.8 lens.

Table 19.1 Training and testing with datasets classified into two categories for offline and online recognition systems

| Targets | Data sets | | Training image numbers | | Testing image numbers | |
	Spray	Non-spray	Offline (spray + non-spray)	Online (spray + non-spray)	Offline (spray + non-spray)	Online
Carrot	120	120	First half (60 + 60)	All (120 + 120)	Last half (60 + 60)	New video (89)
Cabbage	198	198	First half (99 + 99)	All (198 + 198)	Last half (99 + 99)	New video (298)
Onion	107	107	First half (53 + 53)	All (107 + 107)	Last half (54 + 54)	New video (204)
Chestnut	97	97	First half (48 + 48)	All (97 + 97)	Last half (49 + 49)	New video (180)
Persimmon	94	94	First half (47 + 47)	All (94 + 94)	Last half (47 + 47)	New video (210)
Trees and structure	118	118	First half (59 + 59)	All (118 + 118)	Last half (59 + 59)	New video (141)

The 4K videos were collected and converted to images using a JPG converter at the preprocessing stage. The images were collected in the morning from 10 am to 12 pm to ensure uniform lighting while the UAV flew over the croplands and orchards. Days with clear skies were generally chosen for collecting the videos by the flying UAV. The classifiers were segmented from the videos according to flight heights for croplands and orchards (Table 19.1). Three field experiments were conducted with the UAV in three randomly selected zones; a rural farm with a combination of structures and orchards (L1), a farm with different croplands with orchards (L2) and a research farm with croplands and orchards (L3) (Fig. 19.4). MATLAB 2015a® (MathWorks, California) was used to develop the user interface and training and testing datasets for offline and online recognition systems.

19.2.4 Offline Recognition System

The offline recognition system consisted of learning and recognition phases. The learning phase was started by collecting training image datasets of each class $m \in \{1,...,M\}$ and inputting them into the system. For offline experiments of each land type, we used one of the videos with the first half for training and the last half for testing. The recognition phase was confirmed to begin once the learning phases of the classifiers using scene sequences were completed (Fig. 19.5). Then, PCA was applied to establish the linear subspace as a reference subspace for each class. The training phase was completed in three stages. First, all the collected testing images of $I_j \in \{1,...,J\}$ were input into the system, and each I had frames of $\{f\_1,...,f\_n\}$. Second, the PCA was applied to establish the linear subspace for testing the subspace for each class I_j. Finally, the canonical angles between the current testing subspace and each reference subspace were calculated. The current image was assigned to the class with whom it shared the smallest canonical angles, which indicated that it had the highest similarity when referenced to the training datasets. In an offline experiment setting, the UAV was flown 5 m above the cropland.

The first half of images for training (99 images, spray and 99 images, non-spray) and the last half of images for testing (99 images, spray and 99 images, non-spray) were selected for cabbage fields (Table 19.1). The first half of images for training (53 images, spray and 53 images, non-spray) and the last half of images for testing (54 images, spray and 54 images, non-spray) were selected for onion fields. Similarly, the first half of images for training (60 images, spray and 60 images, non-spray) and the last half of images (60 images, spray and 60 images, non-spray) were selected for testing carrot fields. A height of 15 m was chosen for flying over the orchard areas to collect the first half of images for training (48 images, spray and 48 images, non-spray) and the last half of images for testing (49 images, spray and 49 images, non-spray) for chestnut trees.

Again, the first half of images for training (47 images, spray and 47 images, non-spray) and the last half of images for testing (47 images, spray and 47 images, non-spray) were used in the case of persimmon fields. Finally, the first half of images

Fig. 19.4 Training and testing datasets for building the classifiers for recognizing spray areas and non-spray areas. (**a**) L1: Cropland: Carrot, Spray Area. (**b**) L1: Cropland: Non-spray Area. (**c**) L1: Orchard: Persimmon, Spray Area. (**d**) L1: Orchard: Non-spray Area. (**e**) L2: Cropland: Cabbage, Spray Area. (**f**) L2: Cropland: Non-spray Area. (**g**) L2: Orchard: Chestnut, Spray Area. (**h**) L2:

for training (59 images, spray and 59 images, non-spray) and the last half of images for testing (59 images, spray and 59 images, non-spray) were used for trees and structures. The accuracy analysis of offline recognition system was compared with the true positive and true negative values (Table 19.2). For further confirmation, the extended datasets were considered to check the recognition accuracy of classifiers using MSM.

19.2.5 Online Recognition System

The subspace patterns were trained during the offline recognition process. These patterns were used for the online recognition development of classifiers. A sliding window was used to select four images that were converted to four vectors through resizing and reshaping. The gray scale images were resized to 8×8 and reshaped to one column vector using MATLAB®. A test subspace was generated using PCA for creating a matrix from the vectors. The online recognition progress was completed in the following stages. First, each video from each target crop or orchard was preprocessed, and one image was extracted from every 20 frames. Among the extracted images, there were several frames captured that did not belong to either class during takeoff and landing or that included other plants during entry and exit. Such images were marked as noise images and removed to improve recognition accuracy. In the experiment, two datasets were collected for each target land. For the online experiment, we used all of the frames (removed noise) from one of the videos as training, and we used another video for testing (the video was not directly read; rather, the video was extracted to image frames, and noise was removed). In the second step, we classified the set of sequential images using the MSM classifier. Finally, the spray areas were recognized based on the training datasets (Fig. 19.6). In the datasets, 198 images (spray) and 198 images (non-spray) were collected from a 5 m height for training, and a reference subspace was built for use in the online experiment for cabbage. In case of testing, a new video was taken where one frame was selected out of 20 frames. There were a total of 298 frames used for testing for cabbage. Similarly, 107 images (spray) and 107 images (non-spray) were selected for training in online experiments, and the new video stream was used with a total of 204 images for onion. In the case of carrot, 120 images (spray) and 120 images (non-spray) were used for training, and a new video stream with 89 images was used for testing the datasets. For orchard categories from a height of 15 m, two classifiers were trained using 97 images (spray) and 97 images (non-spray); 94 images (spray) and 94 images (non-spray); 118 images (spray) and 118 images (non-spray) for chestnut, persimmon and trees, respectively. For testing the datasets of each target, a

Fig. 19.4 (continued) Orchard: Non-spray Area. (**i**) L3: Cropland: Onion, Spray Area. (**j**) L3: Cropland: Non-spray Area. (**k**) L3: Orchard: Trees, Spray Area. (**l**) L3: Orchard: Non-spray Area
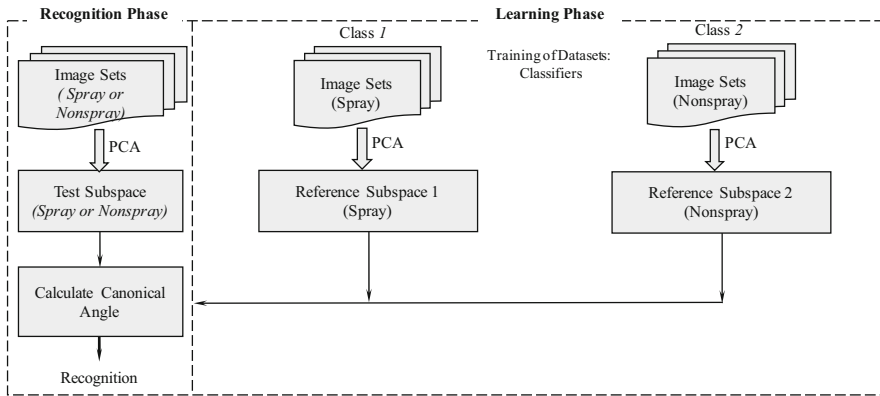
Fig. 19.5 Image sets in classifier recognition in the learning and recognition phases for MSM application

Table 19.2 Accuracy analysis for the offline recognition system

		True condition (offline recognition)		
		Spray	Non-spray	∑Total
Predicted condition (tested by recognition phase)	Spray	True positive	False positive	Total positive
	Non-spray	False negative	True negative	Total negative
	Accuracy	$\text{Accuracy} = \dfrac{\sum \text{True Positive} + \sum \text{True Negative}}{\sum \text{Total}}$		

new video stream was taken with a total of 180 images extracted for chestnut, 210 images for persimmon and 141 images for trees.

19.3 Results

19.3.1 Offline Recognition Performance

In the offline recognition system, the accuracy was 80.5% in the cropland classifiers for spray and non-spray area recognition in the first experimental areas (L1). In the case of orchards, the spray and non-spray area recognition was 75.0% (Table 19.3). In the second experimental area (L2), the recognition accuracy was 70.4% and 86.1% for croplands and orchards, respectively. Finally, mixed crop and orchard areas (L3) were chosen for offline recognition by classifiers. The recognition accuracy was 72.3% and 70.0% for croplands and orchards, respectively. The overall
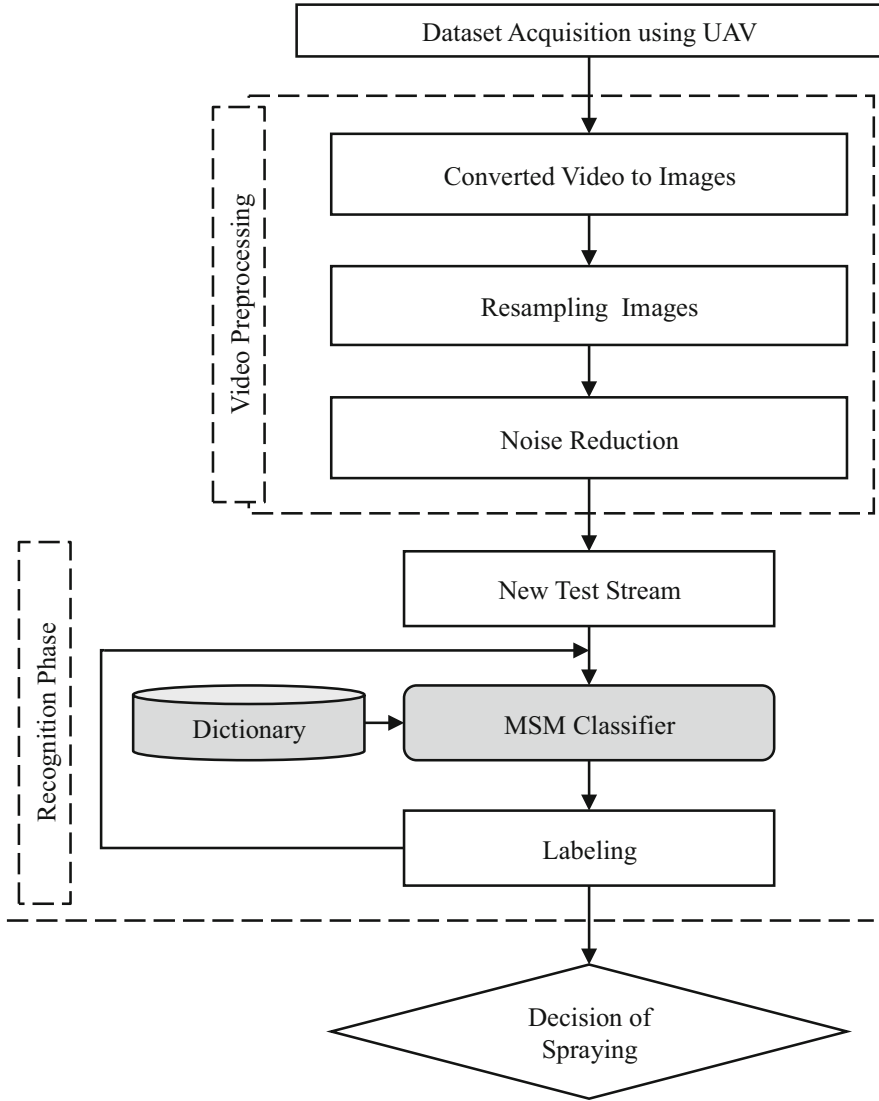
Fig. 19.6 Online recognition system for classification of spraying based on MSM classifiers

accuracy was 74.3% (croplands) and 77.0% (orchards) for the L1, L2, and L3 locations, which had a combination of croplands and orchards (Table 19.3). Wide crop canopy or orchards had the advantage of higher recognition by classifiers. The high accuracy of the recognition system was obtained using the MSM for training and testing using datasets from the three different types of experimental fields.

Table 19.3 Offline classifier recognition and accuracy analysis

			True condition (offline recognition)			
	Location (croplands, Orchards)	Work patterns	Cropland		Orchard	
		Classifiers	Spray	Non-spray	Spray	Non-spray
Predicted condition (tested by the recognition phase)	L1	Spray	74	21	35	9
		Non-spray	16	79	13	31
		Accuracy	80.5%		75%	
	L2	Spray	38	11	41	2
		Non-spray	18	31	10	33
		Accuracy	70.4%		86.1%	
	L3	Spray	56	0	37	18
		Non-spray	31	25	15	40
		Accuracy	72.3%		70.0%	

Table 19.4 Extended datasets for training and testing of classifiers using offline recognition system

| | Data sets | | Training image numbers | Testing image numbers | |
Croplands and Orchards	Spray	Non-spray	Offline	Offline	Accuracy (%)
Carrot	256	256	First half (128 + 128)	Last half (128 + 128)	73.79
Cabbage	440	440	First half (220 + 220)	Last half (220 + 220)	81.25
Onion	210	210	First half (105 + 105)	Last half (105 + 105)	66.32
Chestnut	224	224	First half (112 + 112)	Last half (112 + 112)	77.31
Persimmon	248	248	First half (124 + 124)	Last half (124 + 124)	70.94
Trees and structures	216	216	First half (108 + 108)	Last half (108 + 108)	64.58

Further confirmation, frame numbers were increased for training and testing of datasets, whether there were significant differences in recognition accuracy of classifiers. Extended datasets confirmed the accuracy of MSM method did not change much even if the frames were increased to double for testing and training of datasets in offline recognition system (Table 19.4).

19.3.2   Online Recognition Performance

The developed user interface had the advantage of real-time online information that included the current cropped image, the tested image sets using a sliding window, the predicted category, the recognition rate (the correct classifications were known during the test), the computational time, and the similarity plot. For the cropland classifiers, the UAV was flown at a 5 m height, and the recognition rate was observed to be 65.5% for L1 experimental areas. The computational time was only 0.0031 s for classifier recognition (Fig. 19.7a–c). The flying height was 15 m for orchard classification, and recognition was observed at 69.1% with a computation time of 0.0031 s for each classifier. In the second experimental flying areas (L2), the recognition accuracy of classifiers for noted spray and non-spray areas was 60.8% and 82.2% for croplands and orchards, respectively. The computational time was only 0.0031 s for recognition by the classifiers, and orchard classifier recognition also required only 0.0031 s for each classifier (Fig. 19.8a–c). In the third experimental location (L3), the online recognition rate by classifiers reached 69% in 0.0048 s for each classifier and 71.7% in 0.0031 s for each classifier in croplands and orchards, respectively. The online recognition system had an average accuracy of 65.1% and 75.1% for croplands and orchards, respectively, with a recognition time of 0.0031 s (Table 19.5).

19.4   Discussion

The field experiments were conducted in different types of fields to increase dataset variety for the selection of spray and non-spray areas inside the croplands and orchards. The offline recognition system shows the MSM effectiveness for training and testing the datasets for croplands and orchards. The classifiers were used for croplands and orchards and were limited to being trained and tested on datasets acquired in the late fall season. The MSM has the flexibility to increase the number of classifiers, which may increase the computational time requirement. As UAV spraying is performed at higher speeds, we tend to focus on minimizing the computation time to reduce the computational burden for decision making to recognize the spray and non-spray areas in croplands and orchards. UAVs operating at high speed with limited battery life and a small payload of liquid chemicals demand high computational speed and fast operation with good recognition accuracy. With this consideration, the online recognition system provided some advantages, although its accuracy was not as high as that of the offline recognition system. The system needs further training data to increase accuracy, especially for the identification of croplands less than 5 m high and orchard areas from 15 m high.
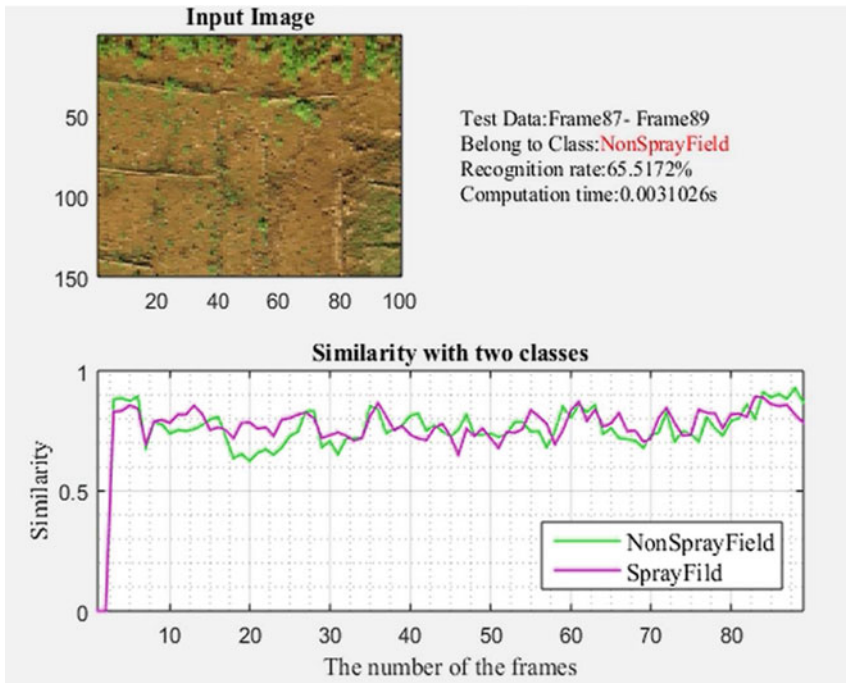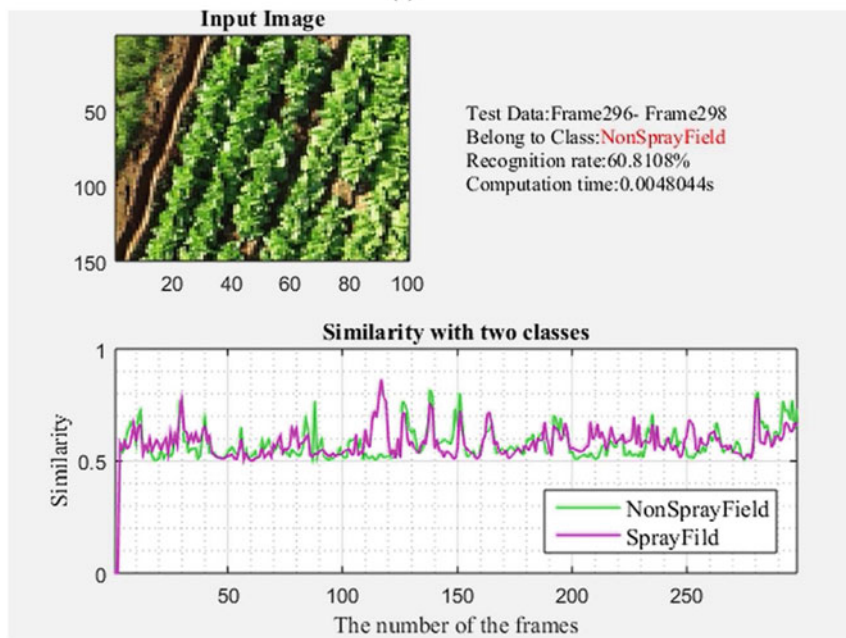
Fig. 19.7 Online recognition performance of a classifier of croplands from a 5 m height. (**a**) Cropland: Carrot. (**b**) Crop field: Cabbage. (**c**) Cropland: Onions
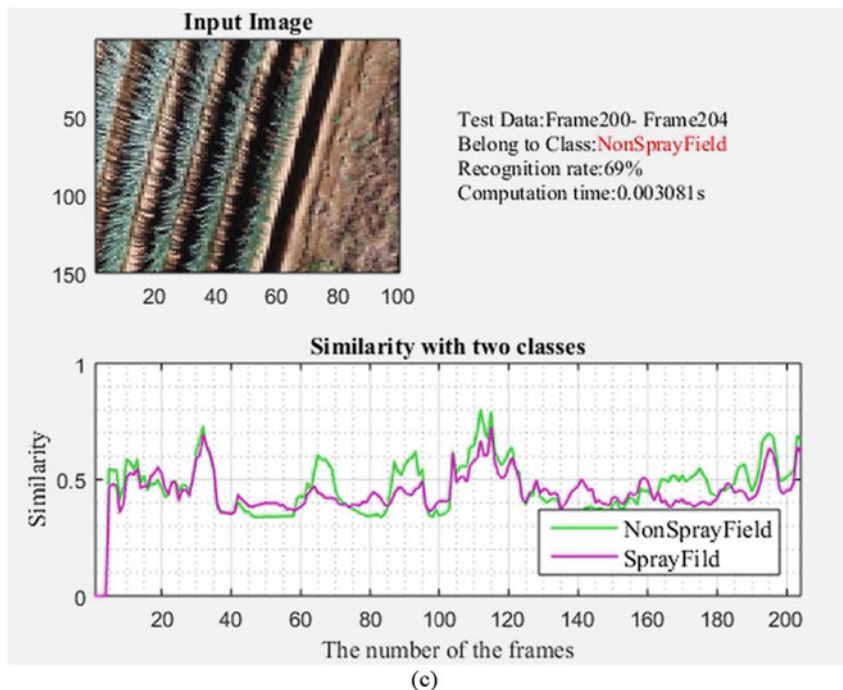
Fig. 19.7 (continued)

In the online experiment setting, similar environments resulted in increased recognition, while adding different categories of orchards reduced recognition. It was very challenging to test the datasets from a fast UAV operating speed at a high altitude. Classifiers were trained and tested on datasets acquired from three different locations to confirm the recognition accuracy. However, complex canopy systems were not present in the features. This MSM system had a limitation in recognizing classifiers in complex canopies of crops or orchards. We could not collect images of complex crop canopy, and we assume that in such canopy systems, upward and downward image acquisition is required to identify the spray and non-spray areas under different lighting conditions. Lighting is a key point that needs to be carefully considered, especially interception through the canopy. It would be ideal to train the UAV features of spray and non-spray areas on a large field to obtain higher accuracy in precision applications ranging from usual to complex canopies of crops. Further studies are required to deal with such complexity of canopies, very large datasets in different lighting conditions, the processing of images to remove noise using extended Kalman filters in onboard UAV systems.
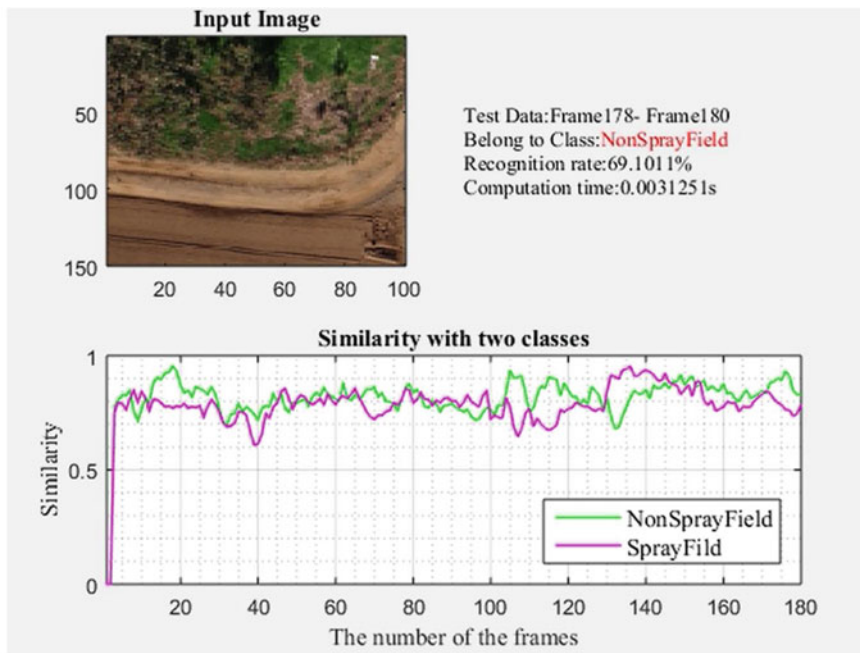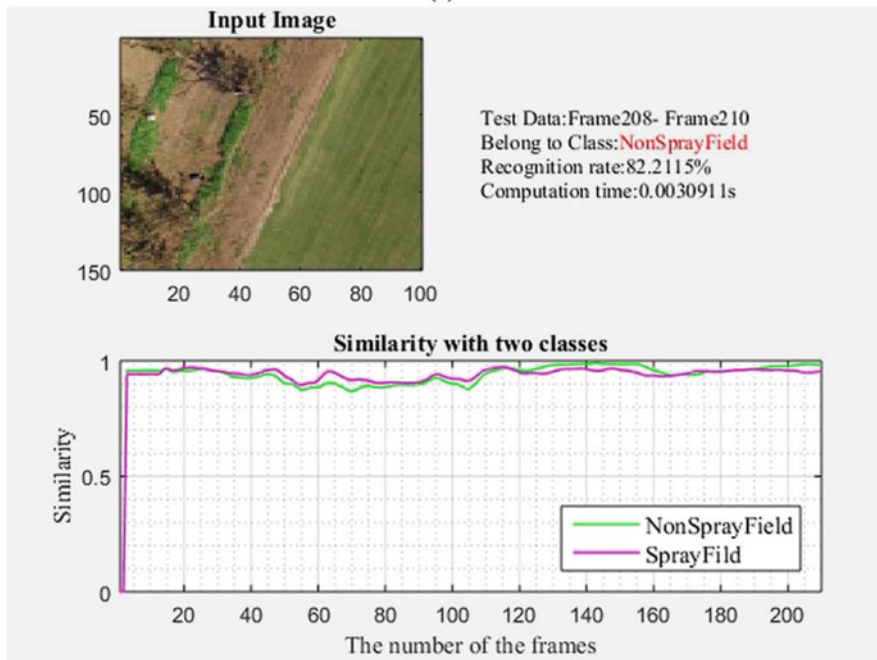
Fig. 19.8 Online recognition performance of a classifier of orchards from a 15 m height. (**a**) Orchard: Chestnut. (**b**) Orchard: Persimmon. (**c**) Orchard: Trees and Structures

Input Image

Test Data:Frame139- Frame141
Belong to Class:NonSprayField
Recognition rate:74.1007%
Computation time:0.0030639s

Similarity with two classes
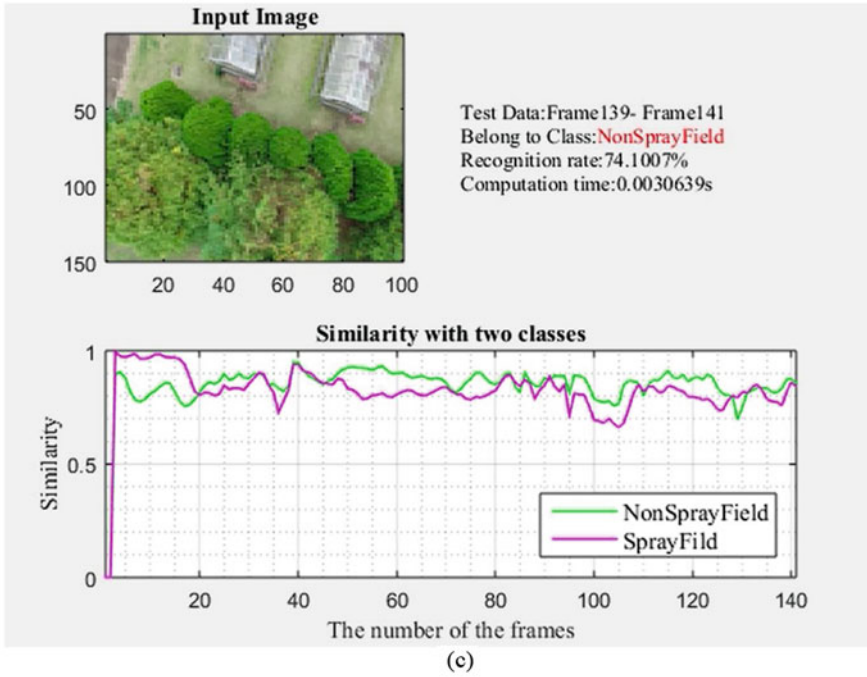
Fig. 19.8 (continued)

Table 19.5 Online classifier recognition and accuracy analysis

Crop/land	Flying height (m)	Accuracy (%)	Recognition time of classifier (s)
Carrot	5	65.51	0.0031
Cabbage	5	60.88	0.0048
Onion	5	69.00	0.0031
Chestnut	15	69.10	0.0031
Persimmon	15	82.21	0.0031
Trees and structures	15	74.10	0.0031

19.5 Conclusion

A machine learning system was developed using MSM for images collected by a UAV in different types of farm fields and orchards. The machine learning system was developed to train and test two classifiers, one for agricultural croplands and one for orchard areas, on different datasets to distinguish spray and non-spray areas for the development of autonomous spraying systems in the future. Images were collected from low (5 m) and high altitude (15 m). The accuracy of the offline recognition system was found to be 74.4% and 77.0% for low and high altitude systems, respectively. On the other hand, the online recognition system performance

had an average accuracy of 65.1% and 75.1% for low altitude and high altitude image acquisition systems, respectively. The computation time for online recognition systems was observed to have a minimum of 0.0031 s (on average) for reporting classifier recognition for the frames in the sliding window. The developed machine learning system for recognizing by classifiers can be implemented in the autonomous UAV spray system for recognizing spray and non-spray areas with minimum computation in real time. In our future experiments, we will improve the training and testing system by incorporating an artificial neural network (ANN) and deep learning to develop a UAV-based autonomous spraying unit for croplands and orchards.

Conflicts of Interest   The authors declare no conflicts of interest.

References

Bandini, F., Jakobsen, J., Olesen, D., Reyna-Gutierrez, J. A., & Bauer-Gottwein, P. (2017). Measuring water level in rivers and lakes from lightweight Unmanned Aerial Vehicles. *Journal of Hydrology, 548*, 237–250.

Berni, J. A., Zarco-Tejada, P. J., Suárez, L., & Fereres, E. (2009a). Thermal and narrowband multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle. *IEEE Transactions on Geoscience and Remote Sensing, 47*(3), 722–738.

Berni, J. A. J., Zarco-Tejada, P. J., Suarez, L., González-Dugo, V., & Fereres, E. (2009b). Remote sensing of vegetation from UAV platforms using lightweight multispectral and thermal imaging sensors. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 38*(6), 6.

Carrio, A., Sampedro, C., Rodriguez-Ramos, A., & Campoy, P. (2017). A review of deep learning methods and applications for unmanned aerial vehicles. *Journal of Sensors, 2017*, 3296874.

Fukui, K., & Yamaguchi, O. (2005). Face recognition using multi-viewpoint patterns for robot vision. In *Robotics research. The eleventh international symposium* (pp. 192–201). Springer.

Fukui, K., & Yamaguchi, O. (2007, November). The kernel orthogonal mutual subspace method and its application to 3D object recognition. In *Asian conference on computer vision* (pp. 467–476). Springer.

Fukui, K., & Yamaguchi, O. (2014). Subspace methods. In *Computer vision* (pp. 777–781). Springer.

Hung, C., Xu, Z., & Sukkarieh, S. (2014). Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a UAV. *Remote Sensing, 6*(12), 12037–12054.

Iijima, T., Genchi, H., & Mori, K. I. (1974). A theory of character recognition by pattern matching method. In *Learning systems and intelligent robots* (pp. 437–450). Springer.

Lee, D. H., Lee, K. S., Cho, Y. J., Lee, J. Y., & Chung, S. O. (2012). Development of autonomous sprayer considering tracking performance on geometrical complexity of ground in greenhouse. *Journal of Biosystems Engineering, 37*(5), 287–295.

Ludovisi, R., Tauro, F., Salvati, R., Khoury, S., Mugnozza Scarascia, G., & Harfouche, A. (2017). UAV-based thermal imaging for high-throughput field phenotyping of black poplar response to drought. *Frontiers in Plant Science, 8*, 1681.

Maeda, K., & Watanabe, S. (1985). Pattern matching method with local structure. *Trans IEIC, J68-D*, 345–352.

Majidi, B., & Bab-Hadiashar, A. (2005, December). Real time aerial natural image interpretation for autonomous ranger drone navigation. In *Digital image computing: Techniques and applications (DICTA'05)* (pp. 65–65). IEEE.

Manfreda, S., McCabe, M. F., Miller, P. E., Lucas, R., Pajuelo Madrigal, V., Mallinis, G., & Toth, B. (2018). On the use of unmanned aerial systems for environmental monitoring. *Remote Sensing, 10*(4), 641.

Peteinatos, G. G., Weis, M., Andújar, D., Rueda Ayala, V., & Gerhards, R. (2014). Potential use of ground-based sensor technologies for weed detection. *Pest Management Science, 70*(2), 190–199.

Pimentel, D., & Burgess, M. (2012). Small amounts of pesticides reaching target insects. *Environment, Development and Sustainability, 14*(1), 1–2.

Rebetez, J., Satizábal, H. F., Mota, M., Noll, D., Büchi, L., Wendling, M., & Burgos, S. (2016, April). Augmenting a convolutional neural network with local histograms-A case study in crop classification from high-resolution UAV imagery. In *ESANN*.

Sakano, H., Mukawa, N., & Nakamura, T. (2005). Kernel mutual subspace method and its application for object recognition. *Electronics and Communications in Japan (Part II: Electronics), 88*(6), 45–53.

Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation, 10*(5), 1299–1319.

Tauro, F., Porfiri, M., & Grimaldi, S. (2016). Surface flow measurements from drones. *Journal of Hydrology, 540*, 240–245.

Tauro, F., Selker, J., Van De Giesen, N., Abrate, T., Uijlenhoet, R., Porfiri, M., & Grimaldi, S. (2018). Measurements and observations in the XXI century (MOXXI): Innovation and multi-disciplinarity to sense the hydrological cycle. *Hydrological Sciences Journal, 63*(2), 169–196.

Tellaeche, A., Burgos-Artizzu, X. P., Pajares, G., & Ribeiro, A. (2008). A vision-based method for weeds identification through the Bayesian decision theory. *Pattern Recognition, 41*(2), 521–530.

Watanabe, S., & Pakvasa, N. (1973, February). Subspace method of pattern recognition. In *Proceedings of the 1st IJCPR* (pp. 25–32).

Yamaguchi, O., Fukui, K., & Maeda, K. I. (1998, April). Face recognition using temporal image sequence. In *Proceedings third IEEE international conference on automatic face and gesture recognition* (pp. 318–323). IEEE.

Zarco-Tejada, P. J., González-Dugo, V., & Berni, J. A. (2012). Fluorescence, temperature and narrow-band indices acquired from a UAV platform for water stress detection using a micro-hyperspectral imager and a thermal camera. *Remote Sensing of Environment, 117*, 322–337.

Zhang, Y., Chen, D., Wang, S., & Tian, L. (2018a). A promising trend for field information collection: An air-ground multi-sensor monitoring system. *Information Processing in Agriculture, 5*(2), 224–233.

Zhang, Y., Gao, P., & Ahamed, T. (2018b). Development of a rescue system for agricultural machinery operators using machine vision. *Biosystems Engineering, 169*, 149–164.

Chapter 20
Basal Stem Rot Disease Classification by Machine Learning Using Thermal Images and an Imbalanced Data Approach

Izrahayu Che Hashim, Abdul Rashid Mohamed Shariff, Siti Khairunniza Bejo, Farrah Melissa Muharam, and Khairulmazmi Ahmad

Abstract Oil palm has become a commodity of global strategic importance due to its rapid expansion. Palm oil is widely utilised in food and as a biodiesel precursor. The oil boosts several countries' economies, especially Malaysia's. However, *Ganoderma boninense* causes basal stem rot (BSR), the most severe disease of oil palms. BSR management controls remain to be ineffective at the moment. There is currently no cure for BSR disease, and the only practical option is to extend the life

---

I. C. Hashim
Centre of Studies for Surveying Sciences and Geomatics, Faculty of Architecture, Planning and Surveying, Seri Iskandar Campus, Universiti Teknologi MARA, Seri Iskandar, Perak, Malaysia
e-mail: izrah696@uitm.edu.my

A. R. M. Shariff (✉) · S. K. Bejo
Department of Biological and Agricultural Engineering, Level 3, Faculty of Engineering, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia

Smart Farming Technology Research Centre, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia

Laboratory of Plantation System Technology and Mechanization (PSTM), Institute of Plantation Studies, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia
e-mail: rashidpls@upm.edu.my; skbejo@upm.edu.my

F. M. Muharam
Laboratory of Plantation System Technology and Mechanization (PSTM), Institute of Plantation Studies, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia

Department of Agriculture Technology, Faculty of Agriculture, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia
e-mail: farrahm@upm.edu.my

K. Ahmad
Laboratory of Plantation System Technology and Mechanization (PSTM), Institute of Plantation Studies, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia

Department of Plant Pathology, Faculty of Agriculture, Universiti Putra Malaysia (UPM), Serdang, Selangor, Malaysia
e-mail: khairulmazmi@upm.edu.my

of the oil palm tree. Thus, we demonstrate how a thermal image technique can be used to distinguish between healthy and BSR-infected trees. We assessed the feasibility of using WEKA standard machine learning algorithms (ML) such as Naive Bayes (NB), multilayer perceptron (MLP), and random forest (RF) to classify healthy and BSR-infected trees. Additionally, we emphasise the data imbalance technique in this study because, in reality, the number of healthy and BSR-infected is not uniform. Therefore, imbalanced data approaches such as random under-sampling (RUS), random over-sampling (ROS), and synthetic minority oversampling (SMOTE) are employed in this classification. In order to evaluate and compare various algorithms and imbalanced approaches, we described the receiver operating characteristic (ROC) curve region (AUC), the precision-recall curve (PRC), and the confusion matrix as an alternative in terms of the success rate of the non-infected and BSR-infected tree. We expect that our technique will assist non-expert users in identifying appropriate machine learning algorithms, resulting in improved performance for accurately predicting BSR disease.

20.1   Introduction

Diseases and pests are an essential concern to the oil palm industry at the moment. Crop yields can suffer significantly while afflicted with diseases and pests that compete for nutrients with palm oil and cause tree destruction. This is critical to avoid considerable crop damage, mainly owing to severe diseases such as basal stem rot (BSR) disease caused by *Ganoderma boninense* (*G. boninense*).

   Various control strategies or approaches have been used or produced to limit the disease's economic effect, including removing or eradicating diseased palms, treating infected palms, and protecting young or healthy palms that have not yet been afflicted (Chung, 2011). However, *G. boninense* disease currently lacks an effective cure (Siddiqui et al., 2021). The majority of control strategies are only effective in reducing disease occurrence and extending the productive life of diseased palms, not entirely curing the disease.

   In what follows, we demonstrate how BSR can be viewed as a problem that results in a drop in oil palm yields and negatively impacts Malaysia's economy (Sect. 20.2). Additionally, we demonstrate that based on the formulation of this problem, a thermal data approach (Sect. 20.3) utilising machine learning (Sect. 20.4) and imbalance data approach (Sect. 20.5) can provide an accurate classification of the healthiness of oil palm trees in a reasonable of time. We next identified healthy and BSR-affected trees using three algorithms plus an imbalance technique from the WEKA open-source package (Sect. 20.6), demonstrating that combining algorithms and imbalance approaches yielded models that performed better than conventional algorithm selection alone (Sect. 20.7). More precisely, we demonstrate that combining the best thermal selection feature with an algorithm and an imbalance technique

makes it possible to differentiate healthy palm trees and those affected with BSR (Sect. 20.8).

This chapter is based previous paper published in the Agronomy, an Open Access Journal by MDPI in 2021 (Hashim et al., 2021a).

20.2   Basal Stem Rot (BSR)

Basal stem rot (BSR) is an oil palm root disease that attacks the basal stem. BSR in oil palm typically expresses itself externally, as visible in the foliage, which occurs when at least 50% of the cross-sectional area of the stem base has been damaged. The rot would hinder nutrients and water transport to the aerial parts, causing symptoms similar to the nutritional deficit and water stress (Rebitanim et al., 2020). Thus, visual signs alone are insufficient to identify BSR, as other water stress causes, including dryness, higher soil water salinity, or hyperacid soils, can produce close foliage signs (Saeidi et al., 2019).

Internal tissues at the stem base or root bole get drily rotted due to the infection. The affected palm runs the possibility of its root or stem bole breaking at any stage of BSR, finally resulting in the palm collapsing. Internal signs include dry and brown rot lesions on the hole and stem, generally with visible mycelial development, and tissues with black bands separating the light brown lesions. Outside the major border of the lesion region, the uninfected tissues are often yellow-pigmented, indicating a host-resistance response to infection (Ariffin et al., 1989). Even before signs of stem or foliar lesion are apparent, the root tissues will be infected, with several saprophytic microbes colonising them. After a significant infection period, BSR's basidiomata will form at the base of the stem or will arise from infected roots. The growing stem's soft tissues gradually decay in younger palms, and it is uncommon for the BSR basidiomata to form before the palm dies (Singh, 1991).

BSR is a severe challenge to the sustainability of oil palm production, particularly in Southeast Asia, and has resulted in economic losses for the worldwide oil palm business (Rebitanim et al., 2020). This disease has a detrimental effect on yields and completely destroys palm stands. This results in income loss and puts the oil palm industry in Southeast Asian nations such as Indonesia and Malaysia under significant pressure (Murphy et al., 2021). Between 2009 and 2011, BSR was detected on approximately 3.71 percent of Malaysia's land area (59,148 hectares out of 1,594,286 hectares) (Kamarudin et al., 2016). *G. boninense* disease has ruined 221,000 hectares of oil palm plantations in Malaysia, resulting in an annual loss of around RM1.5 billion (New Straits Times, 2021). Thus, information on *Ganoderma* prevalence in oil palms is crucial for disease prevention and treatment.

20.3   Detection of BSR Disease

Due to the disease's difficulties in being diagnosed early, it spreads rapidly during the field's earliest stages of oil palm production. Generally, methods for detecting and controlling BSR in its early phases are ineffectual and inaccurate (Chong et al., 2017; Fowotade et al., 2019). Early identification is difficult because of the absence of visible symptoms, which had become a significant impediment to controlling BSR disease. While BSR is incurable, having an early identification tool for infected palms is crucial for economically maintaining the condition.

Numerous researches have been done to develop methods for early detection of the disease, including the *Ganoderma* Selective Medium (GSM) (Ariffin & Idris, 1993), the Polymerase Chain Reaction-DNA (PCR-DNA) method (Idris et al., 2003), the Enzyme-Linked Immunosorbent Assay-Polyclonal Antibody (ELISA-PAb) method (Idris & Rafidah, 2008), and the Headspace Solid-Phase Microextraction (HS-SPME) method integrated with Gas Chromatography-Mass Spectrometry (Zainol Hilmi et al., 2019). Nonetheless, the techniques outlined above are tedious, costly, and inefficient for vast planting zones. A system that is optimum for infection diagnosis involves meticulous planning, swift execution, and non-destructive methods.

Apart from laboratory-based techniques, many remote sensing (RS) approaches have been developed for detecting *G. boninense* in the field. The application of RS technology was a non-destructive technique that may potentially save time. RS includes many platforms and sources, some of which are invisible to the naked sight.

20.3.1   *Remote Sensing Techniques for* G. boninense *Disease Detection*

RS is a non-contact method of obtaining information about an object by measuring the electromagnetic radiation reflected/backscattered or emitted by the Earth's surface (Gomarasca, 2010). Currently, agriculture industry device system development, such as an RS, is being used to monitor real-time disease. RS is a type of art and science that collects information about an object without requiring direct physical touch. Despite the time-consuming nature of the methodologies, BSR was measured and detected utilising RS methods (Khosrokhani et al., 2018). The applications of RS methods are classified as non-imaging or imaging sensor-based as shown in Fig. 20.1. Radiometers-spectroradiometers (positioned on high platforms, tractors, or portable) and fluorescence radiometers are non-imaging sensors. Hyperspectral (narrowband) imaging sensors, multispectral (broadband) imaging sensors, terrestrial laser scanning, thermal infrared imaging sensors, and microwave synthetic aperture radar (SAR) are all variations of imaging sensors.

Early detection and non-destructive disease identification methods, particularly for perennial plants, are critical for smart agriculture and sustainable agricultural
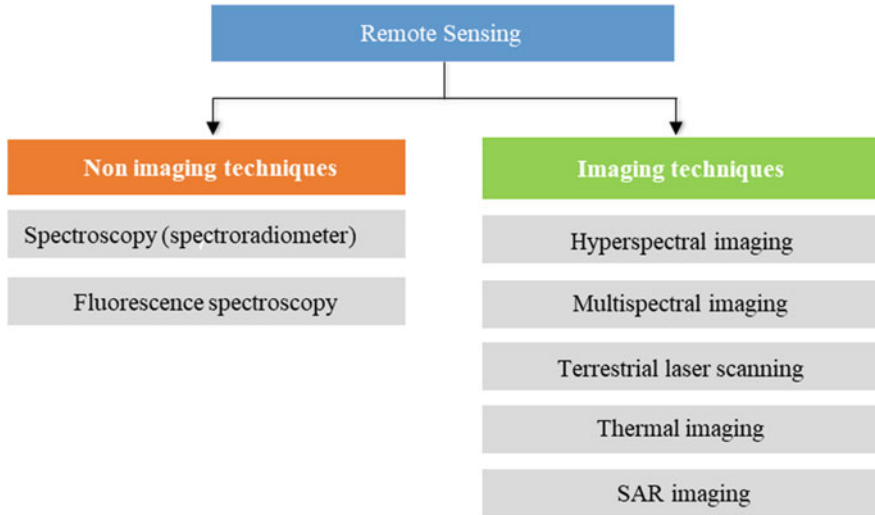
Fig. 20.1 Detection of *Ganoderma* disease utilising RS methods

management (Balasundram et al., 2020). Plant diseases and stress may be monitored using RS approaches (Gogoi et al., 2018; Mee et al., 2017; Naidu et al., 2009; Yang, 2020; Zhang et al., 2019). Recent research has expanded this technology's use to agricultural diseases on a large scale and in real time in field conditions (Bock et al., 2020; Huang et al., 2007; Liu & Wang, 2021; Sighicelli et al., 2009). Additionally, RS approaches may be developed for identifying BSR infections on both the local and major scales within oil palm plantation regions. This approach was chosen because it provides dependable, quick, and real-time surveillance for control and management. Early diagnosis of BSR infection is viewed as a critical source of information for promptly controlling and curtailing it and preventing its spread in plantation regions.

20.3.2 Detection of G. boninense *Using Thermal Imaging*

Thermal imaging detects temperature variations in plants due to water stress. The temperature of the plants reduces owing to the latent heat of evaporation when transpiration occurs and water is evaporated from the plants. Transpiration is a thermodynamic endo-energetic process. Plants' stomatal conductance drops when water-stressed in the soil, resulting in reduced transpiration. This suggests that plants growing on soils with limited water availability will experience a rise in canopy temperature. Thus, by monitoring the canopy temperature in the infrared band, thermal imaging may be utilised to calculate stomatal conductance and plant evapotranspiration (Jones, 1999, 2004; Merlot et al., 2002). According to research on

various plants, the potential for thermal imaging in BSR detection may be investigated and utilised further, as BSR damages the roots of the trees, affecting water transportation and leading to water stress in the oil palm trees.

Although a previous study has been undertaken to detect BSR-infected trees using a thermal technique (Bejo et al., 2015; Johari et al., 2021), our research is unique in terms of camera parameters used to capture and process tree images. This study is novel in terms of camera parameters used to capture tree images since it considers the impacts of various radiation sources, such as emissivity, reflection temperature, and other environmental variables (atmospheric temperature, environmental humidity, and camera proximity), in contrast to previous studies that used the default camera's thermal camera parameters. The default camera settings utilised in the previous investigation were emissivity values of 0.98, the atmospheric temperature of 20 °C, reflected apparent temperature (RAT) of 20 °C, and relative humidity of 50%. While we retained the emissivity value of 0.98 for this study, the RAT value was changed to match the value reflected by the reflector. The reflector is placed within the infrared camera's field of view, and the reflector's emissivity determines its temperature. The reflector temperature is used to determine the RAT value. Meanwhile, environmental temperature and relative humidity were monitored every 30 min and found to be between 24–30 °C and 67–92%, respectively. Additionally, earlier research standardised the image temperature scale between 24 and 34 °C to ensure that the pixel intensity matched the actual temperature representation, in contrast to this study, which evaluated each thermal image by concentrating on temperature variance. In this regard, we are enhancing current techniques and developing novel methods for detecting BSR disease in oil palm fields.

20.4 Machine Learning in Crop Disease

Smart farming is a new concept that uses high-precision algorithms to make agriculture more efficient and effective. It is driven by ML, the scientific field that allows machines to learn without being explicitly programmed. It provides new ways to measure, resolve, and comprehend data-intensive processes in agricultural operational contexts and high-performance computing and big data technology.

Instead of relying on equations defined as models, ML algorithms compute information directly from data (Chang et al., 2018). ML algorithms have been employed in a variety of applications in the recent decade, including land cover mapping (Abdi, 2020; Jamali, 2020; Pan et al., 2020), forest monitoring (Lee et al., 2018; Li et al., 2013; Liu et al., 2018), and agricultural monitoring (Mirani et al., 2021; Liakos et al., 2018; More & Singla, 2019; Rakhra et al., 2022). Precision farming, also known as digital farming (Liakos et al., 2018), has benefited from ML technologies. Controlling pests and diseases are a significant problem in digital agriculture. ML algorithms have also been employed to classify remote sensing data and identify plant diseases (Sharma et al., 2021).

20.4.1 Machine Learning in BSR Disease Detection

Due to its ability to forecast, predict, and classify phenomena related to biology, the ML algorithm has become an accepted technique for agricultural researchers. ML has emerged in facilitating strategic management processes by monitoring and observing early information on plant health. Several researchers have investigated BSR disease detection using ML.

Table 20.1 summarises the ML algorithms used to detect and categorise the degree of BSR infection in oil palms. Numerous input data sets were employed, and several ML approaches were used to achieve various accuracy levels.

20.5 Imbalanced Data Approach

In machine learning, the derivation of classifiers is in minimising misclassified errors and maximising predictive performance (Wang & O'Boyle, 2018). The underlying assumption in these classification methods is that the dataset under investigation comprises a generally balanced number of examples for each specified class. Thus, prior probabilities are expected to be comparable (Guo et al., 2008). Nonetheless, in a number of real-world fields, such as precision medicine, most classification data tends to tilt towards negative class values. The data is considered imbalanced if one

Table 20.1 List of ML algorithms employed for BSR detection

Type of data used	ML algorithm	Capability for severity level classification	Accuracy (%)	Reference
Odour	LDA	No	100	Abdullah et al. (2012)
Spectral image	kNN	Yes	97	Liaghat et al. (2014)
	ANN	Yes	100	Ahmadi et al. (2017)
	RF	No	91	Santoso et al. (2017)
	SVM	Yes	54	Santoso et al. (2019)
Electrical properties	QDA	Yes	80.79	Khaled et al. (2018)
Thermal image	SVM	No	89.2	Bejo et al. (2018)
SAR image	MLP	Yes	77	Hashim et al. (2021b)
Points cloud	KNB	Yes	85	Husin et al. (2020)

of the target variable values has a significantly lesser number of instances than the others.

Skewness was used to describe imbalanced class problems (Longadge et al., 2013). The authors stated that the issue occurs when a dataset is skewed severely, leading to a high false negatives (FN) rate. Class imbalance is a fundamental problem that affects a wide variety of real-world applications, including fraud detection (Warghade et al., 2020), text classification (Padurariu & Breaban, 2019), image and facial recognition (Huang et al., 2020), and clinical diagnosis (Fotouhi et al., 2019). In addition, a study highlighted the uneven class impacts of oil pollution monitoring systems employing SAR to detect and prevent the consequences of oil spills on the environment and prevent illegal dumping (Ouyang et al., 2017). Nevertheless, the database has far fewer oil spill images than those without oil spills, based on only 10% of the spills originating from the sea beds. This further creates challenges linked to data processing results interpretation (Guo et al., 2008).

20.5.1   Data-Level Approaches

Data-level techniques that deal with imbalanced datasets can maintain balanced distribution through the performance of some alteration or modification in the number of minority and majority class instances (Spelmen & Porkodi, 2018). Resampling reduces imbalance by changing the number of instances of each class type which often uses sampling methods like under-sampling, oversampling, or a combination of both (Ali et al., 2013). Resampling approaches are more flexible since they are independent of the classifier employed (López et al., 2013).

20.5.1.1   Under-Sampling

In the case of unbalanced data, under-sampling is one of the most straightforward solutions. To balance the dataset, the principal under-sampling approach randomly excludes cases from the majority class (Tahir et al., 2009) Under-sampling the majority class, most usually done as random under-sampling (RUS), is the simplest yet most effective approach.

The RUS randomly discards the majority class instances until a more balanced distribution is achieved (Hoens & Chawla, 2013). Consider the following scenario: you have a dataset with 50 samples of the majority group and 20 samples of the minority group. By removing 30 majority group samples at random, the RUS attempts to generate an equal group distribution. The final dataset will have 40 samples: 20 minority and 20 majority groups.

20.5.1.2 Oversampling

When dealing with an imbalanced class problem, another typical sample technique is oversampling. Random oversampling (ROS) and synthetic sampling are some of the oversampling approaches available (Chawla et al., 2002; Estabrooks et al., 2004).

Minority group samples are duplicated and reproduced in the dataset by ROS until an equal distribution is achieved. As a result, if there are 50 samples of the majority group and two samples of the minority group, typical oversampling might duplicate the two minority group samples 24 times. There will be 100 samples in the created dataset: 50 samples from the majority group and 50 samples from the minority group (i.e., 25 from the two minority group samples each). Only those minority group values with samples that fall on the border between the majority and minority group values are resampled in targeted oversampling.

20.5.1.3 Synthetic Minority Oversampling Technique (SMOTE)

SMOTE is a sampling technique in which synthetic examples are used to generate more samples from the minority group (Chawla et al., 2002). These new artificial minority class instances were created by interpolating between samples taken from tight proximity to the minority population. SMOTE generates a random minority class representative and finds its nearest minority class neighbour. The synthetic instance is created using a random selection of one of the k-next neighbours, B, and joining A and B in space attributes to make a line segment. Two samples, A and B, are convexly combined to form the synthetic samples (He & Garcia, 2009). Finally, new instances of minority classes are generated.

The study's key novelty and contribution is an investigation into the link between the degree of class imbalance and the performance of the related ML algorithm. In agriculture, however, this is not thoroughly researched. In spite of the fact that many individuals are aware that class imbalance causes problems, no in-depth studies into the precise effects of class imbalance have been carried out.

20.6 Experimental Methodology

The research method is a crucial part of any study. It indicates the study approaches used to collect, pre-process, and analyse data to achieve a suitable result. In this study, the research method consists of four different processes. This process refers to field data collection for thermal images, image processing for thermal data, statistical analysis of data, and finally, the classification employed to discriminate between the non-infected and BSR-infected trees.

20.6.1   Thermal Data Acquisition

Data were collected using a FLIR T620 IR thermal imaging camera (FLIR Systems, Inc., Wilsonville, OR). Each tree section's trunk images were randomly captured from three distinct angles. The tree was 13 years old and was more than 4 m high. The thermal camera is positioned at the height of 1 m above the ground and a distance of 1 m from the tree's trunk. There were two separate morning and afternoon sessions to gather images of trunk sections. The morning session ran from 7:30 to 10:00 AM. Meanwhile, the images from the afternoon session were taken between 4:30 and 7:00 PM. This selection process is based on the progressive solar thermal energy absorption by crop plants during daytime hours. Furthermore, when the environmental temperature increases throughout the day, these objects become less distinguishable from other warm objects detected and highlighted by the camera's sensor. Figure 20.2 illustrates the camera position for trunk image acquisition.

We selected 92 oil palm tree samples at random for this study. According to their health status, the samples were divided into two groups: non-infected (55 samples) and BSR-infected (37 samples). The BSR-infected trees were assessed by an expert from the Malaysian Oil Palm Board according to the visual signs.

It is possible to derive the object's temperature (T_{obj}) using Eq. (20.1). Temperature measurements are performed by various camera manufacturers using similar

Fig. 20.2 Position of the thermal camera when capturing the image trunk
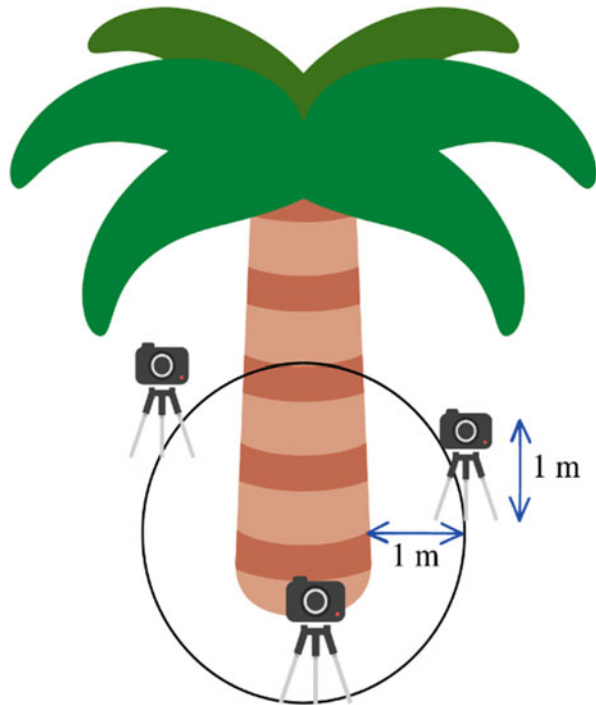
equations (Frank E. Liebmann, 2010). The camera, or software, needs several variable inputs to properly determine the object's temperature to solve Eq. (20.1).

$$T_{\text{obj}} = \sqrt[4]{\frac{W_{\text{tot}} - (1 - \varepsilon_{\text{obj}}) \cdot \tau_{\text{atm}} . \sigma \cdot (T_{\text{ref}})^4 - (1 - \tau_{\text{atm}}) \cdot \sigma \cdot (T_{\text{atm}})^4}{\varepsilon_{\text{obj}} \cdot \tau_{\text{atm}} \cdot \sigma}} \qquad (20.1)$$

where ε_{obj} denotes the emissivity of the object, T_{ref} denotes the temperature of the reflected, τ_{atm} denotes the transmittance of the atmosphere, and T_{atm} denotes the temperature of the surrounding atmosphere. The transmittance of the atmosphere is often determined by the distance between the object and the camera as well as the relative humidity. Commonly, the value is close to one. Due to the fact that the emittance of the atmosphere is near zero ($1 - \tau$atm), this feature has a minor influence on temperature readings. The emissivity of the object and the reflected temperature, on the other hand, have a considerable influence on temperature measurement and must be precisely determined.

20.6.1.1 Emissivity Measurement

An emissivity coating approach was used to assess the emissivity of the oil palm tree's surface in this experimental investigation (Usamentiaga et al., 2014). If a part of the surface under study can be coated with a black paint with a known emissivity, the emissivity of the surface may be determined by adjusting the emissivity setting on the device until the surface temperatures measured on the coated and uncoated surfaces are comparable (Buchlin, 2010). Black electrical tape has also been used instead of black paint in several past studies (Avdelidis & Moropoulou, 2003; Bazilian et al., 2002; Cerdeira et al., 2011; Fokaides & Kalogirou, 2011). The configuration of the emissivity is then changed until the actual temperature is measured. The emissivity configuration is then modified until the actual temperature is determined. The final configurable emissivity is equal to the object's emissivity. As an outcome, the temperature of the oil palm tree and the temperature of the tape recorded by the thermal camera were the same at emissivity of 0.98.

20.6.1.2 Reflected Apparent Temperature (RAT)

It is important to calibrate the reflected apparent temperature to get an accurate measurement. Aluminium foil crumpled and re-flattened is a frequently used alternative (Usamentiaga et al., 2014). To calculate the temperature of the reflector, we use an emissivity of one and a distance of 0 in conjunction with the infrared camera's viewfinder. Repeat the test using the reflector's temperature as the reflector. The resulting temperature value is the final reflected temperature.

20.6.1.3   Atmospheric Temperature and Humidity

It is also possible for a camera to consider the influence of environmental temperatures. The camera's design indicates that transmittance is a function of the relative humidity in the surrounding environment. Every half hour, the temperature and humidity of the environment were measured with a TFA Dostmann Digital Thermo-Hygrometer (30.5002). (TFA-Dostmann.de., Wertheim-Reicholzheim, Germany).

20.6.1.4   Object-to-Camera Distance

In order to capture the fruiting body of *G. boninense* on the basal trunk of a palm tree, we fixed the distance between the object and the camera's front lens to 1 m and focussed the camera at the height of 1 m.

20.6.2   Pre-processing of Thermal Images

The variation of temperature for each of the thermal images was analysed using the camera manufacturer's software, FLIR ResearchIR Max (FLIR Systems, Inc., Oregon, United States). This study utilises two primary image processing processes: image enhancement and identifying the region of interest (ROI).

Image enhancement aims to increase the perceived benefits of images for human viewers or aid in the computer processing of other image-based approaches. Scale Limits From Image and Plateau Equalisation were employed in this study (PE). Scale Limits From Image's purpose is to examine the complete image in order to identify the scale's minimum and maximum values; in contrast, PE allows for good contrast in virtually all settings. The PE slider will enable users to adjust the algorithm's aggressiveness and the intensity of the image improvements. Figure 20.3
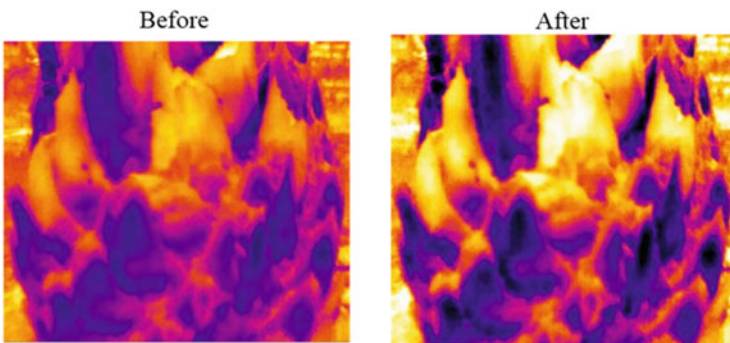
Fig. 20.3 The difference between the images before and after the image-enhancing technique
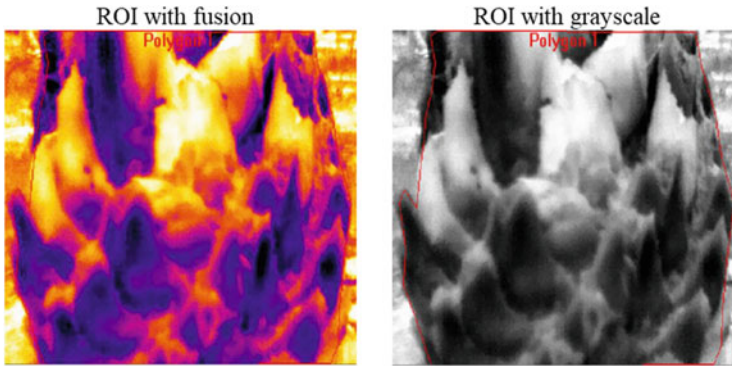
Fig. 20.4 The ROI ith fusion and grayscale temperature polygon for oil palm trees

illustrates the difference between the images before and after the image-enhancing technique.

The next consideration is that the oil palm trunk region must be segregated from its surrounding area. The procedure begins by identifying areas that are most likely to contain foreground objects within the image. The current software defines these regions using the box, ellipse, line, bendable line, polygon, freehand, spot cursor, and measurement cursor forms. This study represented the region of interest (ROI) by a polygon due to the oil palm tree's irregular trunk surfaces. The ROI temperature polygon for oil palm trees is shown in Fig. 20.4.

20.6.3 Thermal Image Feature Extraction

Feature extraction is a process for reducing the dimension of an image by efficiently representing its most notable components as a compact feature vector. FLIR Tools in the FLIR ResearcherIR environment software were used to extract features from each thermal image. The following features were extracted from the ROI of the thermal images that represent the oil palm trees: maximum temperature of the oil palm trunk (T_{max}), minimum temperature of the oil palm trunk (T_{min}), the centre temperature of the oil palm trunk (T_{center}), mean temperature of the oil palm trunk (T_{mean}), and standard deviation temperature of the oil palm trunk (T_{sd}). Each characteristic retrieved from three images was averaged. The statistical Analysis of Variance (ANOVA) was performed on each temperature feature to discover which one was the most significant in distinguishing between non-infected and BSR-infected trees.

20.6.4 Statistical Analysis

The following ANOVA tests were used in this study; (1) to determine the temperature characteristics of non-infected and BSR-infected trees during the morning and evening sessions; and (2) to determine the relationship between non-infected and BSR-infected trees and the thermal camera's characteristic temperature.

20.6.5 Machine Learning Approach

The Waikato Environment for Knowledge Analysis (WEKA) version 3.8.5 was utilised to distinguish between non-infected and BSR-infected trees throughout the process of classification. Three ML approaches were used:

1. Naive Bayes (NB) is a generative probabilistic model based on the notion of predictor feature conditional independence (Frank et al., 2000). It enables the estimation of class-conditional probabilities for the sample data, which may be determined straight from the training data instead of evaluating all feature possibilities (Christopher M. Bishop, 2006).
2. Multilayer perceptron (MLP) is a feed-forward artificial neural network model that maps input datasets to a collection of acceptable outputs. MLPs are formed when numerous layers of nodes are joined (Marius et al., 2009). Each node, with the exception of the input nodes, is a neuron (or processing element) with a nonlinear activation function. The MLP employs a supervised learning approach termed backpropagation to train the network (Stańczyk, 2013). MLP is a variation of conventional linear perception that is capable of discriminating between non-linearly separable inputs.
3. Random forest (RF) algorithm is a classification method that provides a collection of stochastic decision trees. Each tree is trained using a unique bootstrap sample drawn from the original datasets. A majority vote derived from an ensemble of trees generated using the RF approach is used to assign the dataset (Immitzer et al., 2012). Significant benefits include excellent predictive performance, resistance to noise, and effectiveness when dealing with an imbalanced sample (Breiman, 2001; Guo et al., 2008; Rodriguez-Galiano et al., 2012).

20.6.6 Imbalance Data Approach

We are concentrating on the minority group. This analysis was hampered by a data imbalance caused by the lower number of BSR-infected samples compared to non-infected ones. A data-level method will be used to overcome this problem. In terms of data-level imbalance management, the incidence of the two classes is algorithmically equalised to increase the imbalance ratio.

Table 20.2 A description of the parameters that were utilised in the imbalanced methods

Imbalanced approach	Technique	Parameter
Data sampling	RUS	distributionSpread = 1
	ROS	biasToUniformClass = 1
		noReplacement = false
	SMOTE	classValue = 0
		nearestNeighbors = 5
		percentage = 50

Table 20.3 A pre-processed dataset for non-infected and BSR-infected trees using an imbalanced approach

	Training		Testing	
Imbalanced approach	Non-infected (majority)	BSR-infected (minority)	Non-infected (majority)	BSR-infected (minority)
Single (without data imbalance approach	38	25	17	12
RUS	25	25	12	12
ROS	31	31	14	14
SMOTE	38	38	17	17

RUS, ROS, and SMOTE resampling techniques were used in this investigation. For resampling, WEKA, an open-source machine learning application, was used. The resampling parameters were obtained prior to classification and are summarised in Table 20.2.

For testing reasons, we partitioned the dataset into 70% training data and 30% test data. The technique utilised to pre-process the dataset for non-infected and BSR-infected trees is illustrated in Table 20.3.

20.7 Experimental Evaluation

In order to identify between healthy and BSR-affected palm trees, we analysed the outcomes of integrating algorithms and imbalance methodologies.

20.7.1 Time Session Selection

The two-way ANOVA was used to assess the main effects of status (non-infected and infected with BSR) and session (morning and evening) on the mean temperature of the oil palm trees and their interaction effects.

The effects of status and session were statistically significant at $p < 0.005$., as shown in Table 20.4. The main effect for the status generated an F ratio of

Table 20.4 ANOVA summary table for mean temperature extracted from the thermal images

Source	df	Mean square	F	Sig.	Partial eta squared
Status	1	20.382	9.700	0.002	0.051
Session	1	598.529	284.851	0.000	0.613
Status * Time	1	39.073	18.596	0.000	0.094
Error	180	2.101			
Total	183				

*Factor interactions, *df* degrees of freedom, *F* variance ratio, *Sig.* significant

F (1,180) = 9.70, $p < 0.002$, demonstrating a substantial difference between non-infected and BSR-infected oil palm trees. The main effect for the session resulted from an F ratio of F (1,180) = 284.851, $p < 0.001$, showing a significant difference between the morning and evening sessions. The interaction effect was statistically significant F (1,180) = 18.596, $p < 0.001$. Based on these results, it can be inferred that data collection for the thermal image acquisition process in the trunk area can take place in the morning and evening.

Figure 20.5a depicts the relationship between feature temperature (T_{mean}) and the health of oil palm trees; Fig. 20.5b depicts the relationship between feature temperature (T_{mean}) and the session of a thermal image; and Fig. 20.5c depicts the interaction effect of feature temperature (T_{mean}) with the health of oil palm and session.

The microclimate of insects, parasites, fungi, and other organisms is influenced by the temperature of a tree's trunk. For a plant's survival and to help regulate its temperature, water is taken up from the soil by its roots and transported to its leaves via the xylem. Cooling and maintaining the plant's temperature and transferring nutrients to the leaves for photosynthesis are two of the most significant tasks of transpiration (Sterling, 2005). The trunk's surface temperature changes almost instantly in response to the surrounding environment. In contrast, the temperature may take some time to follow the surface at a large depth in the trunk. Oil palm trees infected with BSR have lower transpiration rates as a result of the damaged basal tissue. The infected BSR tree's temperature is higher than that of the non-infected tree because the plant is unable to cool itself down (Harun & Noor, 2006).

Due to the contradicting conclusion that the evening temperature of BSR-infected trees is lower than that of non-infected trees, this phenomenon can be elaborated by referencing Omran (2017), who found peanut leaf spots. Because leaf tissue that is dead or seriously injured has a lower thermal capacity than normal tissue, it can be heated more easily and emits more light. However, dead leaf tissue can be cooled more quickly than healthy tissue. This explains the increased brilliance of the diseased plant in the afternoon, which corresponds to the fall in air temperature.

As stated previously, the properties of non-infected and infected BSR trees vary significantly between the morning and evening sessions. The morning (7:30–10:00 AM) and evening (6:00–10:00 PM) thermal imaging sessions are deemed optimal for determining the health state of oil palms (4.30–7:00 PM). In this study, however,
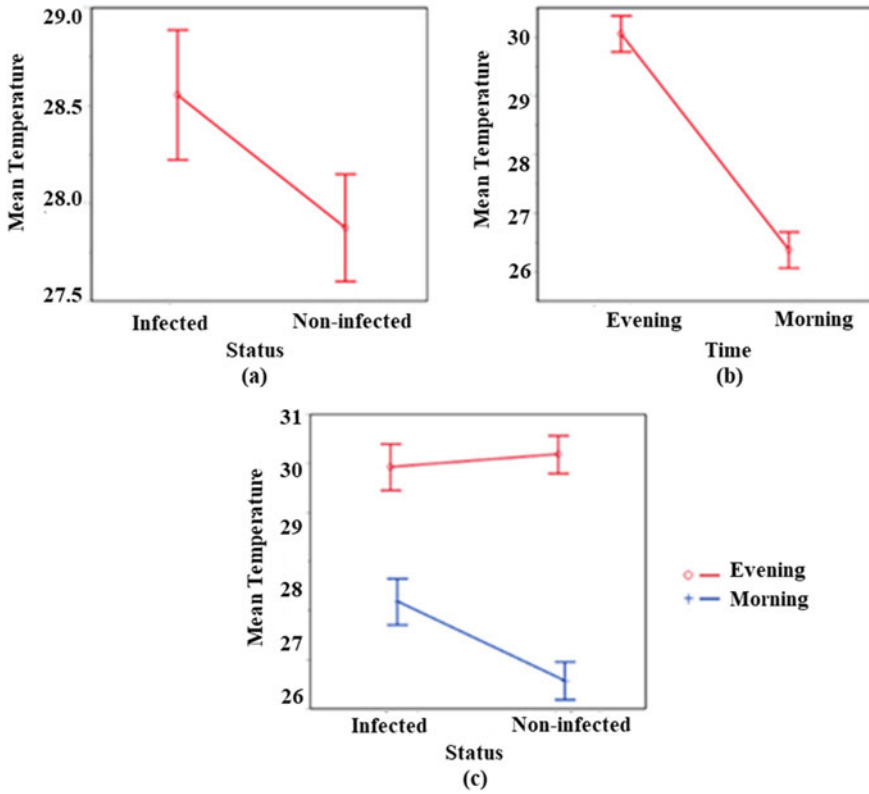
Fig. 20.5 (**a**) The relationship between feature temperature (T_{mean}) and the status of the oil palm trees. (**b**) The relationship between feature temperature (T_{mean}) and the session captured in thermal image. (**c**) The interaction effect of feature temperature (T_{mean}) with the healthiness of oil palm and session

we only analysed data from the morning session, as thermal cameras are often more effective in the morning than in the afternoon.

20.7.2 Selection of Feature Temperature

Table 20.5 contains a summary of the ANOVA test findings. The significance of the ANOVA result can be observed in all of the feature temperatures, T_{mean}, T_{sd}, T_{center}, T_{max}, and T_{min}. The feature temperatures are then deemed to be suitable for further categorisation of the oil palm trees.

Table 20.5 The results of an ANOVA that compared the mean values of T_{mean}, T_{sd}, T_{center}, T_{max}, and T_{min} for non-infected and BSR-infected oil palm trees

Feature	P value	Significance
T_{mean}	<0.0001	Yes
T_{sd}	<0.0001	Yes
T_{center}	<0.0001	Yes
T_{max}	<0.0001	Yes
T_{min}	0.0392	Yes

Feature	NB				MLP				RF			
	Single	RUS	ROS	SMOTE	Single	RUS	ROS	SMOTE	Single	RUS	ROS	SMOTE
T_{mean}	0.787	0.722	0.797	0.801	0.806	0.766	0.785	0.823	0.665	0.710	0.800	0.734
T_{sd}	0.588	0.555	0.649	0.573	0.629	0.522	0.628	0.627	0.529	0.435	0.754	0.512
T_{center}	0.754	0.692	0.758	0.793	0.791	0.718	0.745	0.801	0.682	0.658	0.838	0.760
T_{max}	0.765	0.638	0.762	0.796	0.833	0.752	0.797	0.846	0.881	0.798	0.921	0.899
T_{min}	0.811	0.780	0.808	0.835	0.799	0.762	0.780	0.815	0.696	0.680	0.838	0.744
T_{mean}, T_{sd}	0.738	0.721	0.802	0.767	0.811	0.755	0.759	0.815	0.677	0.674	0.855	0.748
T_{max}, T_{min}	0.823	0.774	0.827	0.846	0.806	0.718	0.810	0.816	0.801	0.750	0.907	0.844
T_{mean}, T_{sd}, T_{center}, T_{max}, T_{min}	0.796	0.736	0.826	0.807	0.789	0.739	0.801	0.811	0.766	0.706	0.920	0.845

FAIL   POOR   ACCEPTABLE   EXCELLENT   OUTSTANDING

Fig. 20.6 The area under the curve (AUC) for the NB, MLP, and RF classifiers with respect to the temperature feature

Feature	NB				MLP				RF			
	Single	RUS	ROS	SMOTE	Single	RUS	ROS	SMOTE	Single	RUS	ROS	SMOTE
T_{mean}	0.784	0.697	0.760	0.771	0.777	0.721	0.737	0.791	0.648	0.659	0.78	0.728
T_{sd}	0.632	0.596	0.657	0.600	0.685	0.557	0.623	0.637	0.556	0.489	0.739	0.558
T_{center}	0.749	0.692	0.769	0.776	0.758	0.667	0.718	0.767	0.673	0.634	0.811	0.711
T_{max}	0.782	0.634	0.736	0.783	0.814	0.724	0.764	0.827	0.864	0.758	0.902	0.877
T_{min}	0.798	0.75	0.776	0.822	0.765	0.709	0.726	0.775	0.680	0.646	0.816	0.729
T_{mean}, T_{sd}	0.748	0.704	0.782	0.748	0.806	0.712	0.716	0.776	0.671	0.638	0.828	0.727
T_{max}, T_{min}	0.802	0.751	0.782	0.831	0.766	0.679	0.772	0.791	0.765	0.705	0.885	0.811
T_{mean}, T_{sd}, T_{center}, T_{max}, T_{min}	0.770	0.710	0.772	0.785	0.764	0.702	0.771	0.784	0.724	0.652	0.902	0.806

FAIL   POOR   ACCEPTABLE   EXCELLENT   OUTSTANDING

Fig. 20.7 The area Precision-Recall Curve (PRC) for the NB, MLP, and RF classifier with respect to feature temperature

20.7.3 Classification Analysis of Feature Temperature

The thermal image features were classified into non-infected and BSR-infected trees using the imbalanced data methods RUS, ROS, and SMOTE, as well as without an imbalanced data method using numerous classification techniques, including NB, MLP, and RF. Figures 20.6, 20.7 and 20.8 show the AUC, PRC, and success rate (%) of non-infected and BSR-infected trees, respectively.

| Feature | IA | Classification model | | | | | |
| | | NB | | MLP | | RF | |
		N	I	N	I	N	I
T_{mean}	Single	92.11	44.00	84.21	72.00	73.68	36.00
	RUS	88.00	44.00	84.00	72.00	72.00	52.00
	ROS	90.32	48.39	77.42	74.19	70.97	90.32
	SMOTE	92.11	55.26	84.21	71.05	65.79	52.63
T_{sd}	Single	94.74	32.00	94.74	24.00	60.53	52.00
	RUS	92.00	24.00	88.00	24.00	44.00	48.00
	ROS	93.55	29.03	90.32	35.48	70.97	80.65
	SMOTE	94.74	23.68	86.84	39.47	57.89	50.00
T_{center}	Single	92.11	48.00	89.47	56.00	73.68	52.00
	RUS	92.00	48.00	80.00	56.00	72.00	52.00
	ROS	90.32	48.39	77.42	67.74	83.87	90.32
	SMOTE	92.11	55.26	86.84	65.79	76.32	71.05
T_{max}	Single	92.11	36.00	84.21	72.00	86.84	80.00
	RUS	92.00	36.00	84.00	60.00	80.00	72.00
	ROS	93.55	48.39	90.32	74.19	87.10	100.00
	SMOTE	94.74	42.11	81.58	76.32	84.21	81.58
T_{min}	Single	94.74	63.63	86.84	64.00	73.68	56.00
	RUS	84.00	64.00	76.00	64.00	64.00	60.00
	ROS	87.10	67.74	80.65	70.97	74.19	90.32
	SMOTE	89.47	65.79	84.21	65.79	71.05	65.79
T_{mean}, T_{sd}	Single	94.74	44.00	81.58	64.00	73.68	56.00
	RUS	92.00	36.00	84.00	68.00	72.00	72.00
	ROS	93.55	41.94	74.19	77.42	77.42	93.55
	SMOTE	94.74	42.11	84.21	71.05	76.32	68.42
T_{max}, T_{min}	Single	89.47	44.00	86.84	60.00	84.21	76.00
	RUS	88.00	44.00	76.00	64.00	76.00	80.00
	ROS	90.32	61.29	83.87	74.19	87.10	100.00
	SMOTE	89.47	57.89	84.21	63.16	84.21	81.58
$T_{mean}, T_{sd},$ $T_{center}, T_{max},$ T_{min}	Single	92.11	44.00	84.21	60.00	84.21	68.00
	RUS	88.00	40.00	80.00	64.00	76.00	68.00
	ROS	90.32	48.39	83.87	74.19	83.87	96.77
	SMOTE	92.11	52.63	84.21	65.79	84.21	81.58

| POOR | MODERATE | ROBUST |

Fig. 20.8 Non-infected (N) and BSR-infected (I) feature temperature success rates (%)

Table 20.6 ANOVA for the effect of features, imbalanced approaches, and classifiers on AUC and PRC across non-infected and BSR-infected trees

Source	AUC				PRC			
	DF	Sum of squares	F Ratio	Prob > F	DF	Sum of squares	F Ratio	Prob > F
Feature	7	0.451	139.611	<0.0001[a]	7	0.254	95.663	<0.0001[a]
Imbalance Approach	3	0.135	97.435	<0.0001[a]	3	0.135	118.367	<0.0001[a]
Classifier	2	0.003	3.394	0.0430[a]	2	0.002	2.935	0.0641
Feature* Imbalance Approach	21	0.026	2.691	0.0031[a]	21	0.018	2.279	0.0114[a]
Feature* Classifier	14	0.068	10.496	<0.0001[a]	14	0.056	10.610	<0.0001[a]
Imbalance Approach* Classifier	6	0.067	24.031	<0.0001[a]	6	0.080	34.975	<0.0001[a]
Error	42	0.019			42	0.016		
C. Total	95	0.769		<0.0001	95	0.561		<0.0001

[a]The mean difference is significant at the 0.05 level

The results of the AUC and PRC are fairly consistent (Fig. 20.6 and 20.7). The T_{max} feature with the RF classifier and ROS method had the best AUC and PRC (outstanding) values at 0.921 and 0.902. The T_{max} features increase AUC and PRC over a single (without an imbalanced data technique) by combining the RF classifier with the ROS approach. The AUC and PRC are the lowest when using the RUS technique on all features and classifiers.

Figure 20.8 depicts the success rates of each feature for the three classifiers NB, MLP, and RF. In contrast to the other two classifiers, NB is less effective at classifying BSR-infected trees. The T_{max} feature has the highest overall success rate across the three classifiers, whereas the RF classifier has the highest success rate for categorising non-infected and BSR-infected trees. Meanwhile, the ROS technique has a maximum success rate of 87.10% for non-infected trees and 100.00% for BSR-infected trees in terms of the imbalanced approach employing RF classifier.

The findings of the ANOVA model for the effect of features, imbalanced methods, classifiers, and two-way interaction (Feature*Imbalance Approach, Feature*Classifier, and Imbalance Approach*Classifier) on AUC and PRC for non-infected and BSR-infected trees are presented in Table 20.6. At the $\alpha = 0.05$ level, two major factors (feature and imbalance approach) and their two-way interaction are statistically significant for the AUC and PRC response variables. One major factor, "Classifier," was determined not to be significant for PRC.

Considering that each of the major components is statistically significant, Tukey's HSD test was run to discover which levels of these factors result in significantly different performances than the other levels of that factor. The first section of Table 20.7 presents the HSD test results for the major factors characteristic. This factor has eight levels (the eight temperature features used for classification), and

Table 20.7 A comparison of the mean values of AUC and PRC obtained from Tukey's HSD test based on features, imbalanced approaches, classifiers, and the interaction between imbalance approach and classifier

AUC	Mean	PRC	Mean
Feature			
T_{max}, T_{min}	0.810^a	T_{max}	0.789^a
T_{max}	0.807^a	T_{max}, T_{min}	0.778^{ab}
T_{mean}, T_{sd}, T_{center}, T_{max}, T_{min}	0.795^{ab}	T_{mean}, T_{sd}, T_{center}, T_{max}, T_{min}	0.762^{bc}
T_{min}	0.779^{bc}	T_{min}	0.749^{cd}
T_{mean}	0.766^{cd}	T_{mean}, T_{sd}	0.738^{cd}
T_{mean}, T_{sd}	0.760^{cd}	T_{mean}	0.737^{cd}
T_{center}	0.749^d	T_{center}	0.727^d
T_{sd}	0.583^e	T_{sd}	0.611^e
Imbalanced approach			
ROS	0.799^a	ROS	0.772^a
SMOTE	0.777^b	SMOTE	0.759^a
Single	0.751^c	Single	0.741^b
RUS	0.698^d	RUS	0.674^c
Classifier			
MLP	0.764^a	NB	0.742^a
NB	0.754^{ab}	MLP	0.737^a
RF	0.751^b	RF	0.730^a
Imbalance approach*Classifier			
ROS-RF	0.854^a	ROS-RF	0.833^a
SMOTE-MLP	0.794^b	SMOTE-MLP	0.769^b
Single-MLP	0.783^b	Single-MLP	0.767^b
ROS-NB	0.779^b	SMOTE-NB	0.765^b
SMOTE-NB	0.777^b	Single-NB	0.758^{bc}
ROS-MLP	0.763^b	ROS-NB	0.754^{bc}
SMOTE-RF	0.761^b	SMOTE-RF	0.743^{bc}
Single-NB	0.758^b	ROS-MLP	0.728^{cd}
RUS-MLP	0.717^c	Single-RF	0.698^{de}
Single-RF	0.712^{cd}	RUS-NB	0.692^e

Means that do not share a letter are significantly different. Each of these factor levels is associated with a grouping letter. If any factor levels have the same letter, then the multiple comparison method did not determine a significant difference between the mean response. For any factor level that does not share a letter, a significant mean difference was identified. For example, from Imbalanced Approach factor level, we see each of Imbalance Approach has a different letter, i.e., no groups share a letter. Therefore, we can conclude that all four Imbalanced Approach resulted in statistically significant different mean in AUC. Furthermore, with the order of the means also provided from highest to lowest, we can say that ROS resulted in the highest mean in AUC followed by SMOTE, then Single, then Feed RUS

[*] Means in the same column with different letters according to group types are substantially different at $p < 0.05$

each is allocated a letter based on its average performance (across all the other factors). The results of all response variables demonstrate a generally consistent result for AUC and PRC. The T_{max}, T_{min} combination feature exceeds all other features in the AUC, whereas T_{max} surpasses all other features in the PRC. Tukey's HSD result demonstrates that the combination of temperature features T_{max}, T_{min} did not differ statistically from the T_{max} feature and the combination T_{mean}, T_{sd}, T_{center}, T_{max}, T_{min}, but it did differ substantially from the other temperature feature in AUC. In PRC, however, the T_{max} feature was not statistically distinct from the T_{max}, T_{min} combination feature, but it was significantly distinct from the other temperature features. The T_{sd} feature, on the other hand, is shown to be lower than the statistically significant margin of all other temperature features.

The major factor levels of the four imbalance techniques perform significantly differently, with the ROS approach surpassing the three other ways in AUC and PRC. Nonetheless, the RUS approach is lower than all other imbalance approaches by a statistically significant margin.

In terms of AUC, MLP did not differ statistically from NB; however, it did differ significantly from RF. There is no statistically significant difference between the three classifiers in PRC's major factor.

The mean AUC and PRC values for the imbalance technique and classifier interaction are also shown in Table 20.7. ROS-RF had a higher mean value for the imbalance approach and classifier and significantly deviated from other interactions. RUS-RF was the lowest performance in terms of AUC and PRC.

20.7.4 The Effect of Classifiers on Model Performance

These three ML algorithms employed in this research perform similarly in distinguishing non-infected and BSR-infected trees according to the AUC, PRC, and success rate. To maximise each classification system, we must comprehend their output.

Both of the following should be taken into account prior to implementing an algorithm: (1) Performance: while choosing a classification or regression algorithm, the overall output of the algorithm is an important factor to consider. (2) Robustness: while evaluating performance, it is critical to examine the robustness of the application instead of the consistency of the fitting. In this case, we rule out the possibility of over-generalisation. Restricting the number of models an algorithm can suggest can help to mitigate this. In this context, MLP and RF can both be deployed, and each offers a distinct set of benefits in addition to certain drawbacks. Furthermore, each of these methods is competent in dealing with ambiguity and over-fitting. In contrast to NB, this approach estimates the conditional probability of each class value while classifying a new instance. As the expected class, it chooses based on the highest probability. The approach estimates the necessary probability values using training data. In order to make calculation manageable, this method uses the naive

assumption that almost all input variables are conditionally independent of the class value.

In reality, each of these three classifiers can be used for a variety of purposes. There is no single algorithm that will work throughout each situation, according to ML. A lack of a single approach consistently surpasses all others, resulting in a wide variation in the results of an algorithm based on its applicability and dataset sizes. As a result, the best learning algorithm may be determined by comparing the results of various algorithms. In order to maximise the benefits of each model's strengths while minimising its weaknesses, it is beneficial to integrate models generated using a variety of methodologies.

20.7.5 The Effect of Data Imbalance on Classification

As a supervised learning process, classification relies heavily on the training dataset. Training level has a significant impact on the accuracy of the resulting classifier. In this circumstance, the imbalanced aspect of the datasets is a significant drawback. Due to the infrequent occurrence of minor classes, the classifier's training is insufficient, resulting in erroneous predictions. In the case of multiclass classifiers, this imbalance results in the underrepresentation of entries, leading to their complete disregard. The majority of classifiers implicitly assume their data to be balanced; therefore, standard classifiers have a majority bias.

Countering data imbalance is typically one of the most important research topics in real-time categorisation. The fundamental assumption of classifiers that operate on data is that the data are balanced. Therefore, the weight assigned to each sample is equal. However, in the presence of imbalanced data, this form of operation results in a classifier that favours the majority classes. Minority groups can even be disregarded during the rule-making procedure if the level of imbalance is sufficiently great. For this case, data balancing methods have been developed. Modifying existing algorithms to improve the weightage of minority classes, boosting their contribution levels, or sampling might be used to manage data.

This research explores the RUS, ROS, and SMOTE imbalanced techniques for addressing the imbalance issue. Even though this is not a comprehensive list, it serves as a good starting point for further research. This research indicates that the ROS technique outperforms the single (without class imbalance) approach. It is more effective at categorising non-infected and BSR-infected oil palm trees. Furthermore, no single technique or model is optimal across all issues, so it is highly suggested to test a variety of strategies and models to see which one is most effective.

20.8   Conclusions

In conclusion, by simply employing the Tmax feature, RF can predict BSR disease with a level of accuracy that is comparatively outstanding compared to MLP and NB, which have a level of acceptable accuracy. The potential use of thermal data and an unbalanced data method are two significant benefits extracted from this research. These approaches can be used in conjunction with ML algorithms to categorise oil palm trees that have been infected with *G. boninense*. In the future, investigations with samples of varying degrees of severity will be utilised to investigate temperature features and identify oil palm trees that have been infected with BSR disease.

References

Abdi, A. M. (2020). Land cover and land use classification performance of machine learning algorithms in a boreal landscape using sentinel-2 data. *GIScience and Remote Sensing, 57*(1), 1–20. https://doi.org/10.1080/15481603.2019.1650447

Abdullah, A. H., et al. (2012). Hand-held electronic nose sensor selection system for basal stamp rot (BSR) disease detection. In *Proceedings - 3rd international conference on intelligent systems modelling and simulation, ISMS 2012* (pp. 737–742).

Ahmadi, P., et al. (2017). Early detection of *Ganoderma* basal stem rot of oil palms using artificial neural network spectral analysis. *Plant Disease, 101*(6), 1009–1016. http://apsjournals.apsnet.org/doi/10.1094/PDIS-12-16-1699-RE (September 14, 2018).

Ali, A., Shamsuddin, S. M., & Ralescu, A. L. (2013). Classification with class imbalance problem: A review. *International Journal of Advances in Soft Computing and its Applications, 7*(3), 176–204.

Ariffin, D., & Idris, A. S. (1993). A selective medium for the isolation of ganoderma from disease tissues. In *PORIM international palm oil conference. progress, prospects challenges towards the 21st century.* PORIM.

Ariffin, D., Idris, A. S., & Abdul, H. H. (1989). Significance of the black line within oil palm tissue decayed by *Ganoderma boninense*. *Elaeis, 1*(1), 11–16.

Avdelidis, N., & Moropoulou, A. (2003). Emissivity considerations in building thermography. *Energy and Buildings, 35*, 663–667.

Balasundram, S. K., Golhani, K., Redmond, S. R., & Vadamalai, G. (2020). Precision agriculture technologies for management of plant diseases. In I. U. Haq & S. Ijaz (Eds.), *Plant disease*

management strategies for sustainable agriculture through traditional and modern approaches (Sustainability in plant and crop protection) (pp. 259–274). Springer. http://link.springer.com/10.1007/978-3-030-35955-3 (May 10, 2021).

Bazilian, M. D., Kamalanathan, H., & Prasad, D. K. (2002). Thermographic analysis of a building integrated photovoltaic system. *Renewable Energy, 26*, 449–461.

Bejo, S. K., et al. (2015). Identification of healthy and BSR-infected oil palm trees using color indices. *International Journal of Agricultural and Biosystems Engineering, 9*(8), 876–879.

Bejo, S. K., et al. (2018). Detecting basal stem rot (BSR) disease at oil palm tree using thermal imaging technique. In *14th international conference on precision agriculture June* (pp. 1–8).

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Bock, C. H., et al. (2020). From visual estimates to fully automated sensor-based measurements of plant disease severity: Status and challenges for improving accuracy. *Phytopathology Research, 2*(1), 1–30. https://doi.org/10.1186/s42483-020-00049-8. (May 10, 2021).

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32. https://link.springer.com/article/10.1023/A:1010933404324 (January 17, 2021).

Buchlin, J. M. (2010). Convective heat transfer and infrared thermography (IRTH). *Journal of Applied Fluid Mechanics, 3*(1), 55–62.

Cerdeira, F., Vázquez, M. E., Collazo, J., & Granada, E. (2011). Applicability of infrared thermography to the study of the behavior of stone panel as building envelopes. *Energy and Buildings, 43*, 1845–1851.

Chang, C.-W., Lee, H.-W., & Liu, C.-H. (2018). A review of artificial intelligence algorithms used for smart machine tools. *Inventions, 3*(3), 41. http://www.mdpi.com/2411-5134/3/3/41 (January 16, 2021).

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Philip Kegelmeyer, W. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Chong, K. P., Dayou, J., & Alexander, A. (2017). Current detection methods of *G. boninense* infection in oil palm. In *Detection and control of ganoderma boninense in oil palm crop* (pp. 13–20). Springer.

Chung, G. (2011). Management of ganoderma diseases in oil palm plantations. *Planter, 87*(1022), 325–339.

Estabrooks, A., Jo, T., & Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence, 20*(1), 18–36.

Fokaides, P. A., & Kalogirou, S. A. (2011). Application of infrared thermography for the determination of the overall heat transfer coefficient (U-Value) in building envelopes. *Applied Energy, 88*, 4358–4365.

Fotouhi, S., Asadi, S., & Kattan, M. W. (2019). A comprehensive data level analysis for cancer diagnosis on imbalanced data. *Journal of Biomedical Informatics, 90*, 103089.

Fowotade, S. A., et al. (2019). Enhanced electrochemical sensing of secondary metabolites in oil palms for early detection of ganoderma boninense based on novel nanoparticle-chitosan functionalized multi-walled carbon nanotube platform. *Sensing and Bio-Sensing Research, 23*, 100274. https://doi.org/10.1016/j.sbsr.2019.100274

Frank, E., Trigg, L., Holmes, G., & Witten, I. H. (2000). Technical note: Naive bayes for regression. *Machine Learning, 41*(1), 5–25. https://link.springer.com/article/10.1023/A:1007670802811 (June 22, 2021).

Gogoi, N. K., Deka, B., & Bora, L. C. (2018). Remote sensing and its use in detection and monitoring plant diseases: A review. *Agricultural Reviews, 39*(4), 307–313.

Gomarasca, M. A. (2010). Basics of geomatics. *Applied Geomatics, 2*(3), 137–146. https://link.springer.com/article/10.1007/s12518-010-0029-6 (April 15, 2021).

Guo, X., et al. (2008). On the class imbalance problem. In *Proceedings - 4th international conference on natural computation, ICNC 2008* (pp. 192–201).

Harun, M. H., & Noor, M. R. M. (2006). *Canopy temperature difference (CTD) for detecting stress*. Kuala Lumpur.

Hashim, I. C., et al. (2021a). Classification of non-infected and infected with basal stem rot disease using thermal images and imbalanced data approach. *Agronomy, 11*(12), 2373. https://www.mdpi.com/2073-4395/11/12/2373/htm (March 30, 2022).

Hashim, I. C., et al. (2021b). Machine-learning approach using SAR data for the classification of oil palm trees that are non-infected and infected with the basal stem rot disease. *Agronomy, 11*(3), 532. https://doi.org/10.3390/agronomy11030532. (May 10, 2021).

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*(9), 1263–1284.

Hoens, T. R., & Chawla, N. V. (2013). Imbalanced datasets: From sampling to classifiers. In H. He & Y. Ma (Eds.), *Imbalanced learning: Foundations, algorithms, and applications* (pp. 43–59). Wiley.

Huang, C., Li, Y., Loy, C. C., & Tang, X. (2020). Deep imbalanced learning for face recognition and attribute prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 42*(11), 2781–2794.

Huang, W., et al. (2007). Identification of yellow rust in wheat using in-situ spectral reflectance measurements and airborne hyperspectral imaging. *Precision Agriculture, 8*(4–5), 187–197. https://link.springer.com/article/10.1007/s11119-007-9038-9 (May 10, 2021).

Husin, N. A., et al. (2020). Classification of basal stem rot disease in oil palm plantations using terrestrial laser scanning data and machine learning. *Agronomy, 10*(11), 1624.

Idris, A. S., & Rafidah, A. R. (2008). Polyclonal antibody for detection of *Ganoderma*. In *MPOB information series*, June 2008 (MPOB TT No. 405). www.mpob.gov.my (April 15, 2021).

Idris, A. S., Yamaoka, M., Noorhasimah, I., & Ariffin, D. (2003). PCR technique for detection of *Ganoderma*. In *MPOB information series*, June 2003 (MPOB TT No. 188) (pp. 1–4). http://palmoilis.mpob.gov.my/publications/TOT/tt188.pdf (April 15, 2021).

Immitzer, M., Atzberger, C., & Koukal, T. (2012). Tree species classification with random forest using very high spatial resolution 8-band WorldView-2 satellite data. *Remote Sensing, 4*(9), 2661–2693. http://www.mdpi.com/2072-4292/4/9/2661 (January 17, 2021).

Jamali, A. (2020). Land use land cover mapping using advanced machine learning classifiers: A case study of Shiraz City, Iran. *Earth Science Informatics, 13*(4), 1015–1030. https://link.springer.com/article/10.1007/s12145-020-00475-4 (June 29, 2021).

Johari, S. N. Á. M., et al. (2021). Detecting BSR-infected oil palm seedlings using thermal imaging technique. *Basrah Journal of Agricultural Sciences, 34*(Special Issue 1), 73–80.

Jones, H. G. (2004). Irrigation scheduling: Advantages and pitfalls of plant-based methods. *Journal of Experimental Botany, 55*(407), 2427–2436. https://academic.oup.com/jxb/article/55/407/2427/496045 (August 3, 2021).

Jones, H. G. (1999). Use of infrared thermometry for estimation of stomatal conductance as a possible aid to irrigation scheduling. *Agricultural and Forest Meteorology, 95*(3), 139–149.

Kamarudin, N., Seman, I. A., Moslim, R., & Mazmira, M. (2016). Current challenges on pests and diseases of oil palm cultivation. In *MPOB paper presented at the agriculture seminar*.

Khaled, A. Y., et al. (2018). Spectral features selection and classification of oil palm leaves infected by basal stem rot (BSR) disease using dielectric spectroscopy. *Computers and Electronics in Agriculture, 144*, 297–309. https://doi.org/10.1016/j.compag.2017.11.012

Khosrokhani, M., Bejo, S. K., & Pradhan, B. (2018). Geospatial technologies for detection and monitoring of ganoderma basal stem rot infection in oil palm plantations: A review on sensors and techniques. *Geocarto International, 33*(3), 260–276. https://www.tandfonline.com/doi/full/10.1080/10106049.2016.1243410.

Lee, J., Im, J., Kim, K., & Quackenbush, L. (2018). Machine learning approaches for estimating forest stand height using plot-based observations and airborne LiDAR data. *Forests, 9*(5), 268. http://www.mdpi.com/1999-4907/9/5/268 (January 16, 2021).

Li, M., Im, J., & Beier, C. (2013). Machine learning approaches for forest classification and change analysis using multi-temporal landsat TM images over Huntington wildlife forest. *GIScience and Remote Sensing, 50*(4), 361–384.

Liaghat, S., et al. (2014). Mid-infrared spectroscopy for early detection of basal stem rot disease in oil palm. *Computers and Electronics in Agriculture, 101,* 48–54.

Liakos, K. G., et al. (2018). Machine learning in agriculture: A review. *Sensors, 18*(8), 1–29.

Liebmann, F. E. (2010). *Infrared target temperature correction system and method.*

Liu, J., & Wang, X. (2021). Plant diseases and pests detection based on deep learning: A review. *Plant Methods, 17,* 1–18. https://doi.org/10.1186/s13007-021-00722-9. (May 10, 2021).

Liu, Z., et al. (2018). Application of machine-learning methods in forest ecology: Recent progress and future challenges. *Environmental Reviews, 26*(4), 339–350.

Longadge, R., Dongre, S. S., & Malik, L. (2013). Class imbalance problem in data mining: Review. *International Journal of Computer Science and Network, 2*(1). https://www.researchgate.net/publication/236651567_Class_Imbalance_Problem_in_Data_Mining_Review (June 29, 2021).

López, V., et al. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences, 250*(2–3), 113–141. https://doi.org/10.1016/j.ins.2013.07.007. (June 29, 2021).

Marius, P., et al. (2009). Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems, 8*(7), 579–588. https://www.researchgate.net/publication/228340819 (January 17, 2021).

Mee, C. Y., Balasundram, S. K., & Husni, A. M. H. (2017). Detecting and monitoring plant nutrient stress using remote sensing approaches: A review. *Asian Journal of Plant Sciences, 16*(1), 1–8.

Merlot, S., et al. (2002). Use of infrared thermal imaging to isolate arabidopsis mutants defective in stomatal regulation. *The Plant Journal, 30*(5), 601–609. https://onlinelibrary.wiley.com/doi/full/10.1046/j.1365-313X.2002.01322.x (August 3, 2021).

Mirani, A., Azeem, et al. (2021). Machine learning in agriculture: A review. *LUME, 10,* 5. www.ijstr.org (June 29, 2021).

More, S., & Singla, J. (2019). Machine learning techniques with IoT in agriculture. *International Journal of Advanced Trends in Computer Science and Engineering, 8*(3), 742–747.

Murphy, D. J., Kirstie, G., Russell, R., & Paterson, M. (2021). Oil palm in the 2020s and beyond: Challenges and solutions. *CABI Agriculture and Bioscience, 2*(1), 1–22. https://cabiagbio.biomedcentral.com/articles/10.1186/s43170-021-00058-3 (May 22, 2022).

Naidu, R. A., Perry, E. M., Pierce, F. J., & 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. *Computers and Electronics in Agriculture, 66*(1), 38–45.

New Straits Times. (2021). MPOB develops solution to control oil palm top stem rot. *New Straits Times.* https://www.nst.com.my/business/2021/10/732768/mpob-develops-solution-control-oil-palm-top-stem-rot (March 30, 2022).

Omran, E. S. E. (2017). Early sensing of peanut leaf spot using spectroscopy and thermal imaging. *Archives of Agronomy and Soil Science, 63*(7), 883–896.

Ouyang, X., Chen, Y., & Wei, B. (2017). Experimental study on class imbalance problem using an oil spill training data set. *British Journal of Mathematics & Computer Science, 21*(5), 1–9.

Padurariu, C., & Breaban, M. E. (2019). Dealing with data imbalance in text classification. In *Procedia computer science* (pp. 736–745). Elsevier.

Pan, L., Gu, L., Ren, R., & Yang, S. (2020). Land cover classification based on machine learning using UAV multi-spectral images. In *Earth observing systems XXV*, International Society for Optics and Photonics, 115011F. https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11501/115011F/Land-cover-classification-based-on-machine-learning-using-UAV-multi/10.1117/12.2566128.full (June 29, 2021).

Rakhra, M., et al. (2022). Implementing machine learning for smart farming to forecast farmers' interest in hiring equipment. *Journal of Food Quality, 2022,* 4721547.

Rebitanim, N. A., et al. (2020). GanoCare® improves oil palm growth and resistance against ganoderma basal stem rot disease in nursery and field trials. *BioMed Research International, 2020,* 1–16. https://doi.org/10.1155/2020/3063710

Rodriguez-Galiano, V. F., et al. (2012). An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS Journal of Photogrammetry and Remote Sensing, 67*(1), 93–104.

Saeidi, T., Ismail, I., Wen, W. P., & Alhawari, A. R. H. (2019). Dielectric properties of complete oil palm trunk sample (healthy and unhealthy). *AIP Advances, 9*, 75314. http://creativecommons.org/licenses/by/4.0/

Santoso, H., Tani, H., & Wang, X. (2017). Random forest classification model of basal stem rot disease caused by *Ganoderma Boninense* in oil palm plantations. *International Journal of Remote Sensing, 38*(16), 4683–4699.

Santoso, H., et al. (2019). Classifying the severity of basal stem rot disease in oil palm plantations using WorldView-3 imagery and machine learning algorithms. *International Journal of Remote Sensing, 40*(19), 7624–7646. https://www-tandfonline-com.ezaccess.library.uitm.edu.my/doi/abs/10.1080/01431161.2018.1541368 (February 27, 2021).

Sharma, A., Jain, A., Gupta, P., & Chowdary, V. (2021). Machine learning applications for precision agriculture: A comprehensive review. *IEEE Access, 9*, 4843–4873.

Siddiqui, Y., et al. (2021). Current strategies and perspectives in detection and control of basal stem rot of oil palm. *Saudi Journal of Biological Sciences, 28*(5), 2840–2849.

Sighicelli, M., Colao, F., Laia, A., & Patsaeva, S. (2009). Monitoring post-harvest orange fruit disease by fluorescence and reflectance hyperspectral imaging. *Acta Horticulturae, 817*, 277–284.

Singh, G. (1991). Ganoderma - The scourge of oil palm in the coastal area. In *Proceedings of ganoderma workshop, Bangi, Selangor, Malaysia, 11 September 1990* (pp. 7–35). Palm Oil Research Institute of Malaysia.

Spelmen, V. S., & Porkodi, R. (2018). A review on handling imbalanced data. In *Proceedings of the 2018 international conference on current trends towards converging technologies, ICCTCT 2018* (pp. 1–11). Institute of Electrical and Electronics Engineers Inc.

Stańczyk, U. (2013). Rough set and artificial neural network approach to computational stylistics. *Smart Innovation, Systems and Technologies, 13*, 441–470. https://link.springer.com/chapter/10.1007/978-3-642-28699-5_17 (January 17, 2021).

Sterling, T. M. (2005). Transpiration: Water movement through plants. *Journal of Natural Resources and Life Sciences Education, 34*(1), 123–123. https://acsess.onlinelibrary.wiley.com/doi/full/10.2134/jnrlse.2005.0123 (August 28, 2021).

Tahir, M. A., Kittler, J., Mikolajczyk, K., & Yan, F. (2009). A multiple expert approach to the class imbalance problem using inverse random under sampling. In *MCS 2009: Multiple classifier systems* (pp. 82–91). Springer. https://link.springer.com/chapter/10.1007/978-3-642-02326-2_9 (June 30, 2021).

Usamentiaga, R., et al. (2014). Infrared thermography for temperature measurement and non-destructive testing. *Sensors, 14*(7), 12305–12348. https://www.mdpi.com/1424-8220/14/7/12305/htm (August 26, 2021).

Wang, Z., & O'Boyle, M. (2018). Machine learning in compiler optimization. In *Proceedings of the IEEE* (pp. 1879–1901). Institute of Electrical and Electronics Engineers Inc.

Warghade, S., Desai, S., & Patil, V. (2020). Credit card fraud detection from imbalanced dataset using machine learning algorithm. *International Journal of Computer Trends and Technology, 68*(3), 22–28.

Yang, C. (2020). Remote sensing and precision agriculture technologies for crop disease detection and management with a practical application example. *Engineering, 6*(5), 528–532.

Zainol Hilmi, N. H., Idris, A. S., & Azmil, M. N. M. (2019). Headspace solid-phase microextraction gas chromatography-mass spectrometry for the detection of volatile organic compounds released from *Ganoderma boninense* and oil palm wood. *Forest Pathology, 49*(4), e12531. http://doi.wiley.com/10.1111/efp.12531 (April 15, 2021).

Zhang, J., et al. (2019). Monitoring plant diseases and pests through remote sensing technology: A review. *Computers and Electronics in Agriculture, 165*, 104943. https://doi.org/10.1016/j.compag.2019.104943

Chapter 21
Early Detection of Plant Disease Infection Using Hyperspectral Data and Machine Learning

Aiman Nabilah Noor Azmi, Siti Khairunniza-Bejo, Mahirah Jahari, and Ian Yule

Abstract Basal stem rot (BSR) caused by *Ganoderma boninense* (*G. boninense*) fungus is one of the most destructive diseases within oil palm plantations in Southeast Asia that result in losses of up to USD500 million annually. Breeding programmes to develop planting materials resistant to *G. boninense* involve a manual census to monitor the progress of the disease development associated with various treatments. It is prone to error due to a lack of experience and subjective human judgements. This current research is aimed to detect early *G. boninense* infections using visible-near-infrared (VIS-NIR) hyperspectral images where there are no BSR symptoms present. A total of 28 samples of oil palm seedlings at 5 months old were used whereby 15 of them were inoculated with the *G. boninense* pathogen. Five months later, spectral reflectance oil palm leaflets taken from fronds 1 (F1) and 2 (F2) were obtained from the top view of VIS-NIR hyperspectral images. The significant bands were identified based on the high separation between uninoculated (H) and inoculated (U) seedlings. The results indicated that the differences were evidently seen in the NIR spectrum. It was also demonstrated that there

A. N. N. Azmi

Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia

S. Khairunniza-Bejo (✉)

Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia

Smart Farming Technology Research Centre, Universiti Putra Malaysia, Serdang, Malaysia

Institute of Plantation Studies, Universiti Putra Malaysia, Serdang, Malaysia
e-mail: skbejo@upm.edu.my

M. Jahari
Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia

Smart Farming Technology Research Centre, Universiti Putra Malaysia, Serdang, Malaysia

I. Yule
Institute of Agriculture and Environment, Massey University, Palmerston North, New Zealand

was no requirement to separate F1 and F2 during the data pre-processing stage since the use of both fronds together produced acceptable classification accuracy. Various significant bands were later used as datasets for the development of detection models using various types of machine learning classifiers. Results have shown a large number of bands achieved a high level of classification accuracy of up to 100%, while a small number of bands achieved a slightly lower level of classification accuracy. In view of the economical aspect of hardware development, a linear support vector machine (SVM) which was developed using a single-band reflectance at 934 nm was identified as the best model for detection since it was not only economical, but also demonstrated a high score of accuracy (94.8%), sensitivity (97.6%), specificity (92.5%), and an area under the receiver operating characteristic curve (AUC) (0.95). The findings of these studies have the potential to significantly benefit the oil palm industry by successfully detecting asymptomatic *G. boninense* infection at an early stage, allowing for prevention of disease spread.

Keywords NIR · Asymptomatic infection · *Ganoderma boninense* · Non-destructive detection

21.1 Basal Stem Rot (BSR) Disease due to *G. boninense* Infection

Oil palm (*Elaeis guineensis*) is a palm species that has been extensively planted in Southeast Asia, primarily in Indonesia and Malaysia to attempt to fulfil the global demand for vegetable oil due to increasing population, income, and the growing biofuel market. Nevertheless, the production of oil palm in Southeast Asia has been affected by the never-ending basal stem rot (BSR) disease caused by the *G. boninense* pathogen. The symptoms of BSR are difficult to recognise at the early stages since there are no significant characteristics that can be visually detected by the naked eye (Govender et al., 2017; Rakib et al., 2019; Wulandari et al., 2018). The earliest visual symptom of *G. boninense* infection in oil palm seedlings can be seen by the presence of fruiting bodies at the bole, followed by partial yellowing of the leaves or mottling of the basal fronds and necrosis when more than 50% of the stem base has been internally destroyed (Naher et al., 2015). Proper management, including the use of planting materials (seedlings) that are resistant to *G. boninense*, can reduce the effects of the disease on the plant (Turnbull et al., 2014). Improving yields through better management and more disease-resistant cultivation not only prevents the spread of BSR disease in plantations and subsequent economic losses (Turnbull et al., 2014; Idris, 2009), but also reduces and sustains the pressure to expand plantation areas, which thus can build towards sustainable palm oil production. Since it is of utmost importance to distinguish between different levels of susceptibility and resistance to plant diseases, it is important to perform pathogen artificial inoculation at the nursery level to create an early screening test (Breton et al., 2009). Current nursery practice uses manual inspection which relies heavily on

the visible symptoms of the disease to monitor the progression of various treatment-related illnesses (Parker & Gilbert, 2007; Chung, 2012) and is performed by humans. This method is prone to error due to inexperience, subjective assessment, and asymptomatic cases. A lab-based method (Ariffin & Seman, 1993; Idris & Rafidah, 2008; Kandan et al., 2010; Madihah et al., 2014) is believed to be reliable for early detection of *G. boninense*. However, the procedure involves the collection of stems, which can lead to plant damage and eventual destruction. Other proposed methods include an electronic nose (e-nose) (Markom et al., 2009; Kresnawaty et al., 2020), electrical impedance (Khaled et al., 2018a, b), tomography (Mazliham et al., 2008; Arango et al., 2016), thermal images (Bejo et al., 2018; Mohd Johari et al., 2021), multispectral images (Bejo et al., 2018; Santoso et al., 2011), spectroscopy or hyperspectral data (Ahmadi et al., 2017; Azmi et al., 2020; Izzuddin et al., 2013; Lelong et al., 2010; Liaghat et al., 2014), and ground laser scanners (Bejo et al., 2018; Husin et al., 2020a, b, c; Nur et al., 2021). A comprehensive review of the sensors used to detect BSR by (Maryam et al., 2018) found that the scores differed in terms of accuracy and limitations of each method. Based on a detailed review of a near-infrared (NIR) spectroscopy sensor (Mohd Hilmi Tan et al., 2021), it can be concluded that *G. boninense* can be detected accurately using NIR spectra. In addition, spectroscopic techniques have proven to be more reliable than chemical and imaging techniques in reactions involving organic tissues. However, the spectroscopy approach has a limitation whereby the instrument can only make one measurement at a time for a small sample spot, requiring a longer data acquisition period. In contrast, hyperspectral cameras can cover a large area in a single imaging session, reducing the data acquisition time.

21.2 Hyperspectral Imaging

Hyperspectral imaging is a technology that integrates conventional spectroscopy and imaging techniques to obtain both spatial and spectral information about a scene. Hyperspectral imaging sensors measure the light reflected from each pixel of a two-dimensional area in narrow and contiguous wavelengths. The hyperspectral image data is in the form of a three-dimensional structure with one spectral and two spatial dimensions or a look-like "cube." Therefore, every pixel in the hyperspectral image contains a complete spectral signature equal to the number of cube bands, as shown in Fig. 21.1. Such signatures provide sufficient information to classify and analyse existing materials within an image.

In earlier times, hyperspectral imaging was often used for military space-borne mapping and detection (Dale et al., 2013). Later, the application of hyperspectral imaging has been extended to the agricultural field. The spectral reflectance of vegetation in the electromagnetic spectrum of 400–2500 nm may contain unique characteristics and information (Goetz et al., 1985). For example, the 400–675 nm range (the visible spectrum) is only suitable for examination of surface features due to the lack of penetration capability. The 675–1000 nm (the NIR spectrum) has a
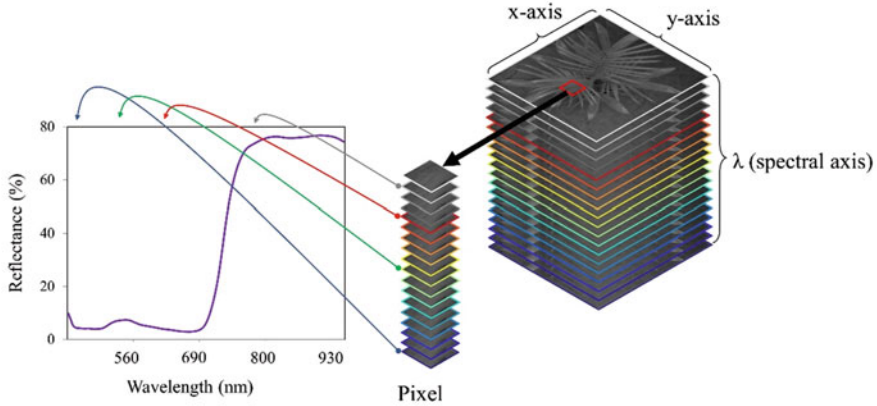
Fig. 21.1 Example of a hyperspectral image data cube. Each pixel consists of a complete reflection spectrum at its band

better penetration capability hence the range is suitable for internal quality assessment while 1000–2500 nm [short-wave infrared (SWIR)] is sensitive to the moisture content of the sample (Huang et al., 2007).

Hyperspectral imagery has been clearly proven acceptable for evaluating crop conditions by collecting enormous quantities of raw data that require further processing to facilitate a wide range of applications. For example, the imagery may be used to investigate water status (Ezenne et al., 2019), vigour assessment (Zhang et al., 2020), biomass estimation and yield prediction (Li et al., 2020), post-harvest (Torres et al., 2019), and disease monitoring (Abdulridha et al., 2020). This technique has not yet been used extensively in the detection of BSR in Malaysia.

21.3 Machine Learning

Machine learning is a combination of the scientific research of algorithms and statistical models to perform a detailed analysis that depends on patterns and inferences (Mitchell, 1997). There are two types of machine learning algorithms; supervised learning and unsupervised learning. Supervised learning develops mathematical models using a training dataset and generates an inferred function to allow the estimation of output values. After sufficient training, the system is able to provide predictions for any new inputs. In contrast, unsupervised learning only includes the inputs from a dataset and discovers any structure in the dataset using grouping or clustering data points. The cluster usually consists of samples with identical properties over the entire dataset (Behmann et al., 2014). Thus, unsupervised learning could be used as the first step before the supervised learning process.

Furthermore, there are two types of machine learning algorithms, namely parametric and non-parametric. Parametric algorithms require prior knowledge to
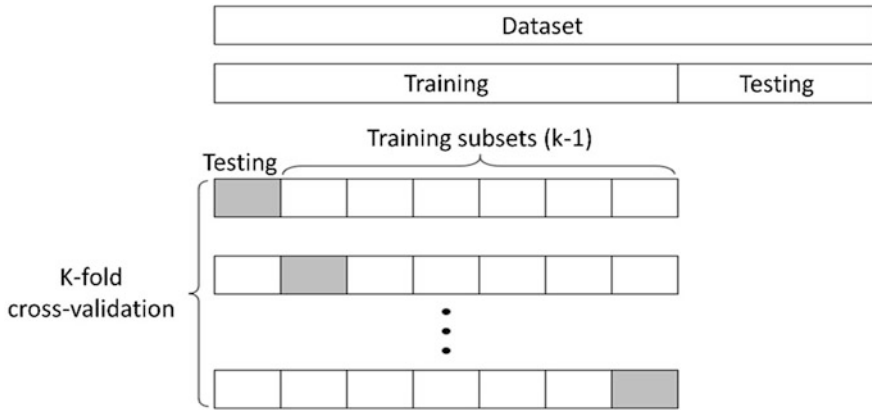
Fig. 21.2 Visual representation of training/testing split and cross-validation

regulate the relationship between predictors and targets. These algorithms have a fixed structure, are faster to train and require fewer data points, but may not be as powerful. In contrast, non-parametric algorithms do not require prior knowledge. However, these algorithms are slower to train and have a higher model complexity depending on the number of training data points but can result in more powerful models (Kotlar et al., 2019). Examples of parametric algorithms are logistic regression, discriminant analysis, and Naïve Bayes. In contrast, nearest neighbour, decision trees, radial basis function (RBF) kernel SVMs, and ensemble are considered as non-parametric algorithms due to the application of kernel matrix by computing the pair-wise distances between the training points.

In practice, there are two types of datasets used for machine learning, namely training and testing datasets. The training dataset is used by learning classifiers to construct a model, while the testing dataset (also known as a validation dataset) is used to evaluate the performance of the predictive model. The K-fold cross-validation technique can be introduced to reduce bias and variance in the training dataset. This cross-validation method splits the training dataset into equal size K subsets (Fig. 21.2). In each iteration, one subset is reserved as a testing dataset while the remaining k-1 subsets left over are used as training data. Hence, each subset becomes a testing set exactly once and becomes a training set k-1 times. Additionally, this cross-validation method may also improve the effectiveness of the model.

Machine learning (ML) has been applied in various fields in agriculture including crop yield estimation (Su et al., 2017), disease detection (Ferentinos, 2018), weed detection (Pantazi et al., 2017), crop quality (Hu et al., 2017), species recognition (Grinblat et al., 2016), animal welfare (Matthews et al., 2017), livestock production (Morales et al., 2016), water management (Feng et al., 2017), and soil management (Morellos et al., 2016). It can be used to facilitate strategic plantation management by providing early plant health information, and for improving the accuracy of disease severity classification from 80% (Husin et al., 2020a, b) to 85% (Husin et al., 2020c). Various types of machine learning (ML) techniques have been used to

detect G. boninense infection both in the nursery and in plantation fields using different types of datasets such as odour, electrical impedance, laser scanning data, spectral reflectance, and synthetic aperture radar (SAR) data. Differences in accuracy assessments are due to different types of datasets and situations, as well as different classifiers with different properties, and different types of classifiers are sensitive to different parameter optimisations (Husin et al., 2020a, b, c).

21.4   Research Design

This paper presents progress work on the early detection of *G. boninense* infection in oil palm seedlings using hyperspectral data and machine learning (ML) techniques (Fig. 21.3). It started with the identification of suitable wavelengths and was then followed by the detection process using ML techniques. Support vector machine (SVM), a well-known ML technique for disease detection, was first used by applying various numbers of bands as datasets. After that, the capability of seven ML classifiers was tested. In order to overcome the economic issues, the capability of a small number of bands was later used as datasets for SVM models.

Therefore, the research commenced with sample preparation for the uninoculated seedlings, defined as healthy seedlings (H) in this research; and seedlings inoculated (U) with *G. boninense*. This was followed by the experimental setup of the image acquisition which included all considerations to minimise the effects of uneven environmental illumination, the spectral extraction of the leaflets of frond 1 (F1) and frond 2 (F2) and removal of outliers. It continued with spectral analyses of the extracted raw spectra, which also identified the significant bands using t-test statistical analysis. Finally, it also included the methods used to develop machine learning models for the classification of the H and U seedlings using SVM, seven ML classifiers and finally SVM with band reduction.
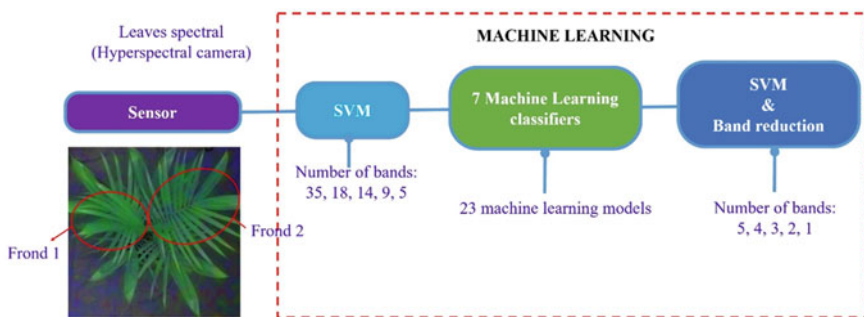

Fig. 21.3 Progress work on the early detection of *G. boninense* infection in oil palm seedlings using hyperspectral data and Machine Learning techniques

21.4.1   Research Area

The research was conducted at the UPM Transgenic Greenhouse (2°59′33.10″N 101°43′19.16″E), Serdang, Malaysia, from 24 January 2019 to 24 June 2019. The greenhouse had dimensions of 12 m length × 6 m width × 4 m height and was made up of polycarbonate to provide protection against UV light, and was equipped with air conditioners, humidifiers, thermal screens, humidity, and temperature sensors. The temperature and humidity inside the greenhouse were set at 27 °C and 90%, respectively, following the procedure proposed by Oettli et al. (Oettli et al., 2018).

21.4.2   Preparation of Samples

A total of 28 oil palm seedlings (commercial standard crosses of Dura x Pisifera, DxP) at the age of 4 months old were obtained from Sime Darby Plantation Research Sdn. Bhd., Banting, Malaysia. During 2 h of travelling to UPM, every seedling was wrapped in individual plastic and transported in a box to maintain the quality of the seedlings. As soon as the seedlings arrived, the seedlings were allowed to acclimatise to the greenhouse condition for 1 month before inoculation. Only 28 oil palm seedlings were used in this research due to the limitation of the greenhouse size. Every seedling was arranged according to standard nursery practice to ease the maintenance of the greenhouse. Two of the 28 oil palm seedlings were used for the laboratory-based *G. boninense* infection confirmation test, resulting in 26 oil palm seedlings available for complete analysis.

21.4.2.1   Artificial Inoculation

Fifteen out of the 28 oil palm seedlings were transplanted with rubberwood blocks (RWBs) colonised with *G. boninense* pathogen, referred to as inoculated seedlings. Two of the 15 inoculated seedlings were used for the polymerase chain reaction (PCR) confirmation test. Meanwhile, the remaining 13 seedlings were transplanted with uncolonised RWBs referred to as uninoculated seedlings. The seedlings were transplanted at 5 months old. The transplanting process was achieved following the method of Naidu et al. (Naidu et al., 2018). After transplanting, the seedlings were arranged according to the standard triangular arrangement of the oil palm nursery with a spacing of 1 m. All seedlings were watered 1 L daily at 10.00 am using a drip irrigation system and fertilised every month using 50 g nitrogen (N), phosphorus (P), and potassium (K) fertiliser.

21.4.2.2 Polymerase Chain Reaction (PCR)

A polymerase chain reaction (PCR) test was used to confirm the *G. boninense* infection after 2 months of inoculation, whereby two inoculated seedlings were sent to the Bacteriology Laboratory, Faculty of Agriculture, UPM. The Polymerase chain reaction (PCR) is a molecular technique to detect and identify pathogens by amplifying the specific or targeted parts of a Deoxyribonucleic Acid (DNA) sequence. The part needed for the test was the root because the *G. boninense* infection starts at the roots of the oil palm. Figure 21.4 shows an example of an infected seedling and the associated PCR result. As shown in Fig. 21.4a, the infected seedling did not show any visible symptoms related to *G. boninense* infection such as fungal mass or foliar symptoms such as yellowing of leaves despite testing positive with the *G. boninense* pathogen. The positive result indicated that the *G. boninense* pathogen had penetrated and infected the tested roots. The specific band size was approximately 160 to 170 bp (Fig. 21.4b) that were obtained from the roots, which authenticated the presence of the *G. boninense* infection. Further confirmation was conducted using gene sequencing analysis between the specific bands and *G. boninense* (taken from the GenBank dataset). The result showed 99.5 to 100% similarity index.

21.4.3 Hyperspectral Imaging

21.4.3.1 Image Acquisition

The image acquisition process was conducted inside the greenhouse. The hyperspectral camera used in this research was a FireflEYE S185 snapshot camera manufactured by Cubert GmbH, Ulm, Germany. The camera provided wavelengths ranging from 450 nm to 950 nm that covered the visible (blue, green, and red) to NIR regions. The spectral was 125, with a spectral sampling of 4 nm. The snapshot
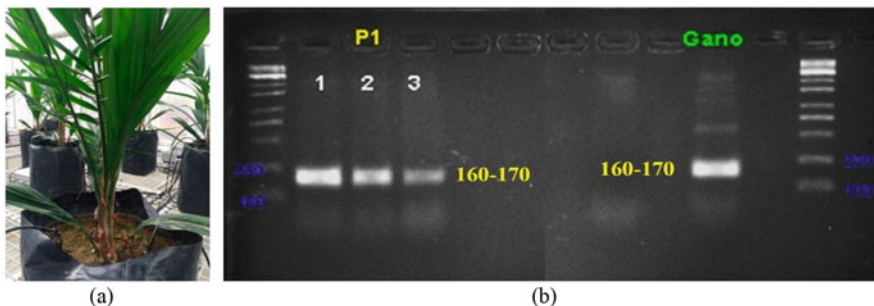

Fig. 21.4 Sample of an infected seedling. (**a**) Condition of an infected seedling without symptoms i.e., no fruiting bodies or of older leaves. However, the disease was confirmed by the (**b**) PCR amplification using a specific primer of *G. boninense*
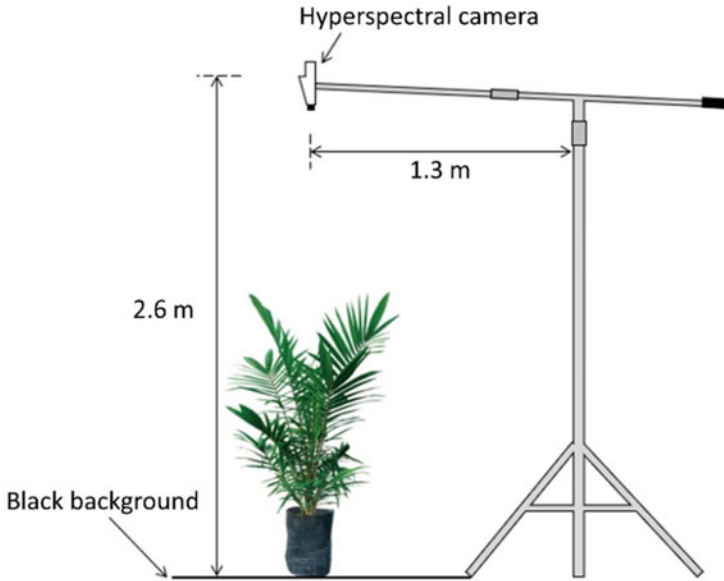
Fig. 21.5 Illustration of hyperspectral image acquisition setup inside the greenhouse

acquisition mode was operated by two charge-coupled devices (CCD) with a pinhole and prisms for spatial and spectral separation.

The camera was mounted horizontally on a custom tripod and was positioned 2.6 m from the ground level, as shown in Fig. 21.5. The system was controlled by Cube-Pilot software (Cubert GmbH, Ulm, Germany) supplemented by the manufacturer. The top view images of the oil palm seedlings at an age of 5 months after the inoculation process were taken on a sunny clear day from 11:00 am to 2:00 pm local time to receive natural illumination.

The camera was calibrated with white and dark references before each image acquisition to reduce the effects of illumination and detector sensitivity. The integration times were almost the same. A dark calibration was performed by closing the lens of the camera, while the white calibration was performed by placing a provided white rectangular board (99% light reflection) flat and close to the lens. Each collected spectrum was calibrated as:

$$\text{Reflectance} = \frac{\text{Image} - \text{Dark}}{\text{White} - \text{Dark}} \tag{21.1}$$

The white and dark calibration was tested before the actual image acquisition to ensure good output image quality. It was achieved by observing the spectral reflectance of the white reference board. If the reflectance of the white reference board was stagnant at 100% in all bands, the calibration was accepted, and the image acquisition process could be continued. One seedling was imaged at a time, and the sample

Fig. 21.6 Sample image taken from the top view of a seedling using a hyperspectral camera
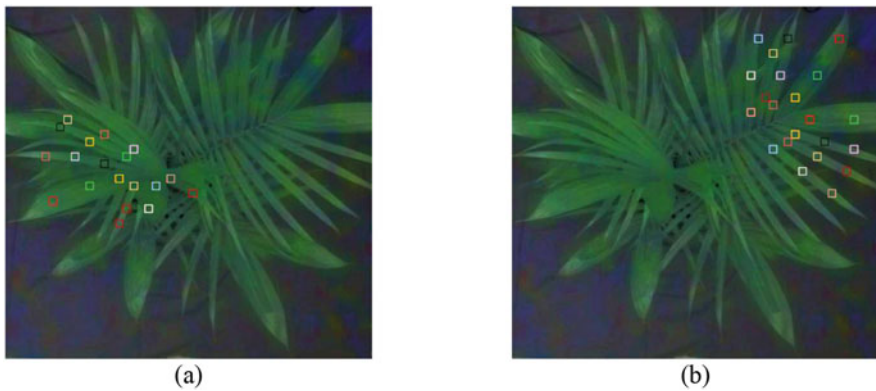

Fig. 21.7 Spectral extraction using a Cube-Pilot of (**a**) frond 1 and (**b**) frond 2

was put on a 2 m × 2 m black background board. Since black reflects no light, the light captured by the camera sensor while taking the image was purely reflected by the fronds of the seedling. Figure 21.6 shows a sample of an image taken from the top view of the seedlings to imitate the actual application in a plantation, which uses an unmanned aerial vehicle (UAV) to capture the image.

21.4.3.2 Spectral Extraction

The acquired hyperspectral images were processed to obtain the spectra of the uninoculated (H) and inoculated (U) leaves from the leaflets of frond 1 (F1) and frond 2 (F2). Both fronds were clearly visible in the aerial view images because they were at the top of the list of fronds. The inclination of the fronds could be used to distinguish Frond 1 and Frond 2 because F1 is the youngest expanded frond and thus less inclined than F2. The spectra were extracted randomly from the first four leaflets on the left and right sides of F1 and F2 (Fig. 21.7). In order to minimise spectral

reflectance variations caused by frond inclinations, only four leaflets were included in this research (Rapaport et al., 2014). Each frond yielded an average of 20 spectral readings. As a result, the spectra collected from F1 were 558 (263 for H seedlings and 295 for the U seedlings), while F2 was 564 (277 for H and 287 for the U seedlings).

21.4.3.3   Significant Bands for BSR Detection

The average spectra of the H and U seedlings from F1 and F2 were calculated and plotted against reflectance to reveal visible and near-infrared spectrum patterns, as well as the differences between the H and U. One spectral reflectance consisted of 125 bands. However, not all of them contain information relevant to the research, whereby some may contain content redundancy and irrelevant information. Therefore, the bands were filtered based on the first 35 bands (30% of the total) that provided high separation values between the H and U. These bands were then subjected to a result of a t-test statistical analysis, achieved using SPSS statistical software (IBM SPSS Statistics 25, IBM, New York, USA), where the means of H and U treatments were compared using hypothesis testing to determine whether the treatments were different from each other at these bands. The bands that obtained $P \leq 0.05$ were considered significant as they contained the most information concerning differences between the H and U seedlings and were thus used as input to develop the machine learning classification models.

Figure 21.8 shows the result of reflectance analysis for the H and U seedlings sampled from leaflets of F1 and F2 at week-20 after inoculation. As shown in this
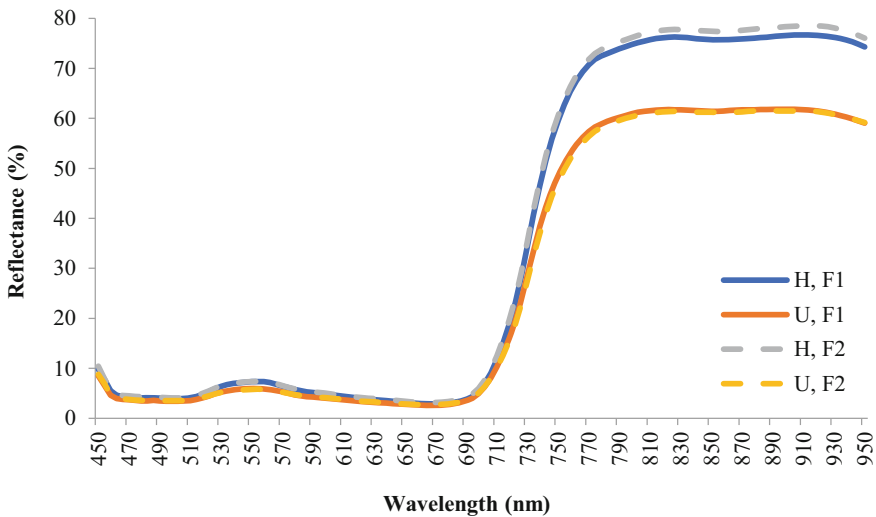

Fig. 21.8   Average reflectance of uninoculated (H) and inoculated (U) oil palm seedlings for frond 1(F1) and frond 2 (F2)

Table 21.1 Significant bands of frond 1 (F1) and frond 2 (F2) as identified by t-test statistical analysis

Frond number	Total significant bands	Significant band (nm)
F1	35	810, 814, 818, 822, 826, 830, 834, 838, 842, 846, 850, 854, 858, 866, 870, 874, 878, 882, 886, 890, 894, 898, 902, 906, 910, 914, 918, 922, 926, 930, 934, 938, 942, 946, 950
F2	35	814, 818, 822, 826, 830, 834, 838, 842, 846, 850, 854, 858, 862, 866, 870, 874, 878, 882, 886, 890, 894, 898, 902, 906, 910, 914, 918, 922, 926, 930, 934, 938, 942, 946, 950
F12	35	814, 818, 822, 826, 830, 834, 838, 842, 846, 850, 854, 858, 862, 866, 870, 874, 878, 882, 886, 890, 894, 898, 902, 906, 910, 914, 918, 922, 926, 930, 934, 938, 942, 946, 950

figure, F1 and F2 yielded almost similar reflectance patterns for both the H and U seedlings. The U seedlings presented lower reflectance compared to H at the near-infrared (NIR) range (750–950 nm), with maximum differences of 15.4% and 17.3% for F1 and F2, respectively. The reflectance pattern generated by the U seedlings was typical for diseased plants, with low reflectance in the NIR wavelengths due to the destruction of xylem, which thus reduced chlorophyll pigments and caused water deficiency.

The reflection patterns of F1 and F2 were about the same for the H and U, but H for F2 produced slightly higher reflectance than H for F1 in the NIR spectrum. This may indicate that the older leaves produced higher reflectance than the younger leaves. This idea was supported by Rapaport et al. (2014), who found an increases in NIR reflectance of Cabernet Sauvignon in the second week when the fourth leaf (young) of the control treatment moved to the eighth nodal position (old leaf position), which concluded that age variability mainly influenced the differences in reflectance spectra. Furthermore, the higher reflectance of H in F2 presented higher differences between H and U of F2, indicating higher degradation of the mesophyll cell wall, which was similar to symptoms of *G. boninense* mentioned by Naidu et al., 2018 and Nisfariza et al. (Nisfariza et al., 2010), where old leaves showed symptoms earlier than young leaves.

A new dataset, named F12 was then introduced to identify the band based on the separation values of the H and U seedlings, which were later confirmed using statistical test analysis. The F12 dataset took both the reflectance values of F1 and F2. Table 21.1 summarises the 35 significant bands obtained from F1, F2, and F12 at 450–950 nm. Based on the table, some bands were significant only in F1 and vice versa. Considering only 35 bands instead of 125 bands could make future hardware design easier and more economical by avoiding analytical problems with unwanted bands.

21.5 BSR Detection

21.5.1 BSR Detection Using SVM

Different support vector machine (SVM) kernel functions namely linear, Gaussian RBF, and polynomial together with the significant bands of F1, F2, and F12 were used to develop classification models to differentiate between H and U seedlings (Azmi et al., 2020). After the classification models were constructed, an optimisation process was carried out to determine the optimal number of bands that could classify H and U seedlings with a high accuracy as shown in Fig. 21.9. If the accuracy was greater than 85%, the current number of significant bands was reduced by 50%. Otherwise, the current number of significant bands was increased by 50%. Separate classification models were developed using F1, F2, and F12 to determine the suitable fronds to differentiate between the seedlings. As a result, the number of bands was optimised from 35 bands to 18, 9, 14, or 5 bands. Figures 21.10 and 21.11 show the classification accuracy of SVM models developed using the significant bands of F1, F2, and F12.

The results of F1 as shown in Fig. 21.10 indicate that SVM models developed using 35 and 18 bands had achieved 100% accuracy when differentiating H and U seedlings. Overall, linear SVM, fine Gaussian SVM, and medium Gaussian SVM were the good F1 classifiers that scored 100% accuracy for 35 and 18; and 98.6% for 9 and 5 bands. This demonstrated that these models were not sensitive to band reduction because they still provided high classification accuracy even when using 9 and 5 bands with a finely detailed distinction between H and U. On the other hand, the Quadratic SVM and Coarse Gaussian SVM were very sensitive to the band
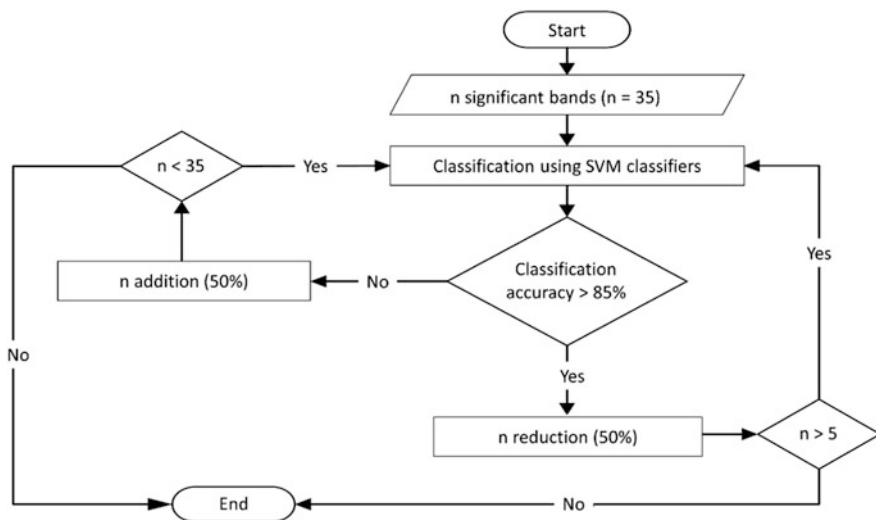

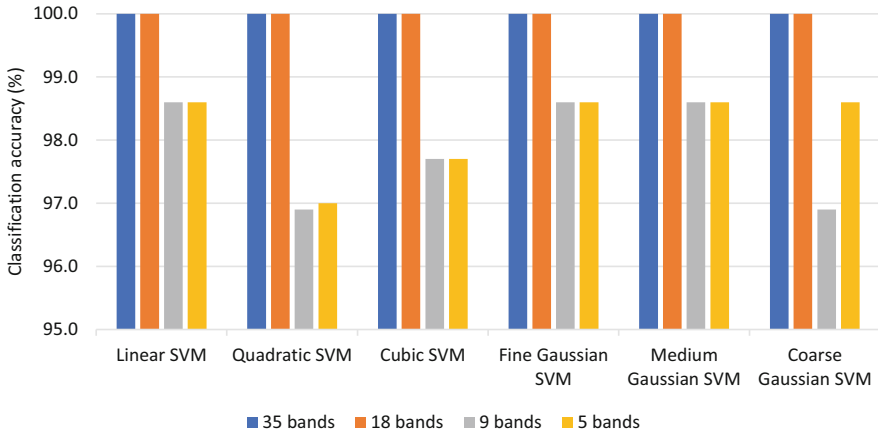Fig. 21.9 Flowchart of band number optimisation process for reflectance dataset

Fig. 21.10 Classification accuracies obtained using different numbers of significant bands of F1
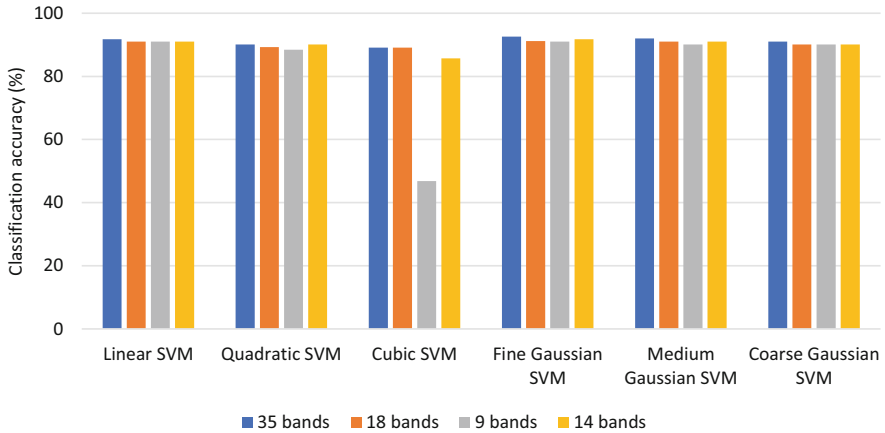

Fig. 21.11 Classification accuracies obtained using a different number of significant bands of F2

reduction, whereby the use of 9 bands reduced the accuracy by at least 3%. However, the usage of 5 bands slightly increased the classification accuracy, which indicated 5 bands were better at classifying the H and U than 9 bands.

For F2, as shown in Fig. 21.11, fine Gaussian SVM obtained the highest accuracy of all bands with 92.6% accuracy when using 35 bands. All models achieved classification accuracies greater than 90% for 35 bands, except the Cubic SVM model, which had a classification accuracy of 89.1%. Classification accuracy when using 18 bands was slightly lower than 35 bands. For example, although Fine Gaussian SVM gained the highest classification accuracy for 18 bands, its performance was lowered by 2% compared to when using 35 bands, making it the classifier with the highest percentage of loss. In contrast, the Cubic SVM was the only classifier that maintained the same accuracy as 35 bands, with an accuracy of
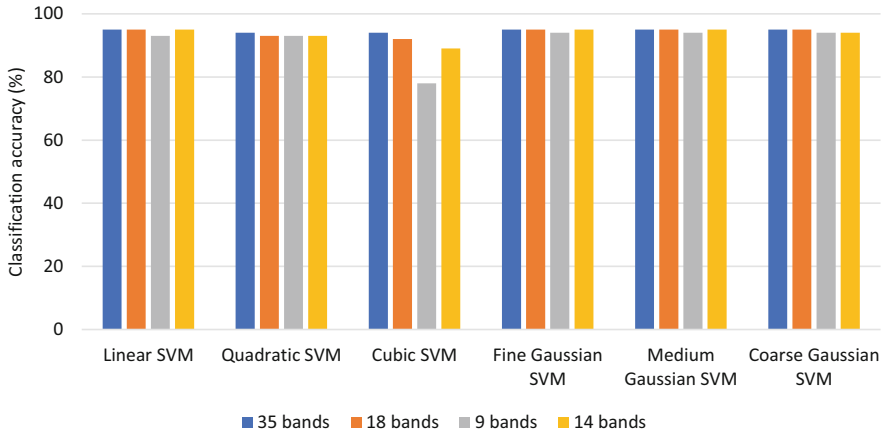
Fig. 21.12 Classification accuracies obtained using a different number of significant bands of F12

89.1%. However, this kernel performed less accurately than others. The optimisation of bands from 18 to 9 significantly impacted the accuracy of the Cubic SVM, which dropped from 89.1% to 46.8%. In contrast, the other models only experienced a slight decrease in classification accuracy except for the linear SVM and coarse Gaussian SVM, which maintained the same accuracy as the 18 bands.

For F12, as shown in Fig. 21.12, the highest classification accuracy of 95.3% was achieved by linear SVM and fine Gaussian SVM when using 18 bands. Meanwhile, the majority of the models achieved a classification accuracy of more than 90%, with half of the models achieving 94.9% accuracy when using 35, 18, and 14 bands. Only the 9 bands of the coarse Gaussian SVM model outperformed the 14 bands with 0.4% increases in classification accuracy. As for the Cubic SVM, the classification accuracy achieved gradually decreased as the number of bands decreased, initially 94.3% for 35 bands, 91.7% for 18 bands, then reduced to 78.2% for 9 bands and increased to 88.9% for 14 bands.

In a nutshell, the results demonstrated the NIR wavelengths had a high potential to distinguish healthy and asymptomatic *G. boninense* seedlings with up to 100% accuracy. Furthermore, the number of bands affected the classification accuracy of the SVM models as the models developed using a high number of bands tended to yield high accuracy, which ensured better prediction. In terms of frond number, the SVM models developed with F1 produced more than 97% accuracy, whereas models developed with F2 only produced a classification accuracy of more than 90%. Meanwhile, the classification accuracy obtained by F12 outperformed the accuracy of F2 with an average difference of 4.8%, which showed that the F12 data could improve the accuracies obtained by F2, and thus confirmed that the combination of both fronds could be used to detect the *G. boninense* infection in oil palm. This result was consistent with Shafri et al. (Shafri et al., 2011), who used a maximum likelihood classification technique with a combination of F1 and F2 to determine the health status of oil palm seedlings and obtained a net accuracy of 82%.

However, our F12 method achieved more than 90% accuracy even when fewer bands were used, e.g., 9 bands. Combining F1 and F2 reflectance would simplify the process of BSR disease detection using an aerial view approach since both fronds can be seen clearly from the top view image, and no image pre-processing process would be needed to separate the fronds.

21.5.2 BSR Detection Using Various Types of ML

This section presents the capability of 23 ML classifiers to classify the H and U seedlings using 35 significant bands (Azmi et al., 2021). These bands were extracted from F12, as listed in Table 21.1. The optimisation process was carried out to determine the optimal number of bands to improve the classification accuracy obtained by the ML models. Exploration runs were applied where significant bands were reduced and increased during the optimisation process as shown in Fig. 21.13. If there was improvement of the average accuracy of 23 models, the number of bands in the next classification process was reduced by 50%. However, if the accuracy was decreased, the number of bands was increased by 50%. The optimisation process ended if the number of bands became greater than 35 or there was no improvement in the average accuracy for at least two neighbouring band reduction processes. The performance time for each model for each significant band was also recorded in order to assess the capability of the model.

Table 21.2 shows the results of the ML models with F-scores greater than 95% in the 35, 18, 14, 11, and 9 bands with their performance times. The 35, 18, and
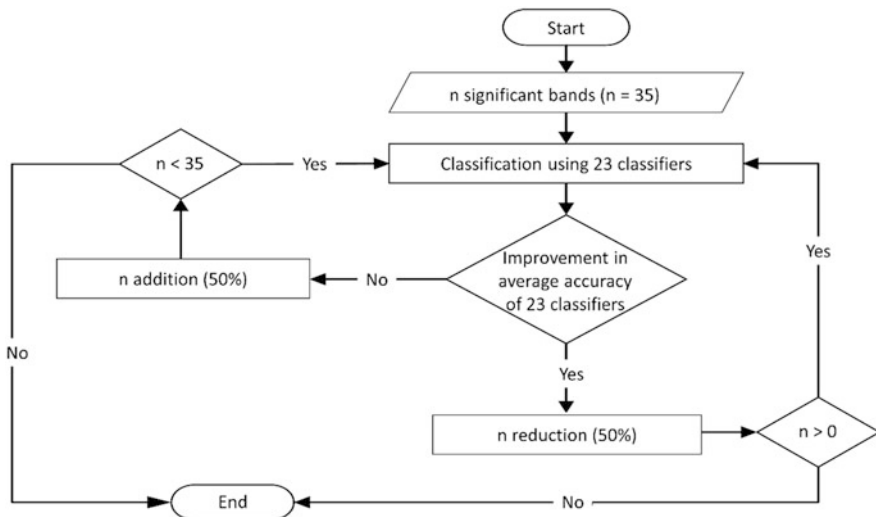

Fig. 21.13 Flowchart of optimisation of number of bands for reflectance dataset

Table 21.2 Classification models that achieved more than 95% *F*-score with their performance times

| | Performance time (s) | | | | |
| | Number of significant bands | | | | |
Classification model	35	18	14	11	9
Linear SVM	–	4.0248	–	–	–
Quadratic SVM	–	–	–	11.6320	–
Fine Gaussian SVM	6.1060	4.6642	3.8789	–	–
Medium Gaussian SVM	4.8147	–	3.7307	–	–
Coarse Gaussian SVM	–	–	–	–	1.7124
Medium kNN	5.7937	4.8772	3.9551	–	–
Cubic kNN	8.9532	6.1620	4.8110	–	–

Table 21.3 The five most significant bands of F12

Total significant bands	Significant band (nm)
5	926, 930, 934, 938, 942
4	930, 934, 938, 942
3	930, 934, 938
2	934, 938
1	934

14 bands each have four models that obtained an F-score above 95%, whereas the 11 and 9 bands each have only one model. A high F-score indicated that the model was less sensitive to band optimisation, implying that high classification accuracy may be achieved even when the model was developed with a small number of datasets. Since several models achieved high F-score values, the best model was determined based on the performance time. It was observed that the type of algorithm and the number of bands used had an effect on model performance. For example, the cubic SVM took the longest time to develop models for 35, 18, and 14 bands as compared to the other algorithms. In addition, the quadratic SVM provided the longest performance time with 11.6320 s, even when using only 11 bands. Overall, the best model for classifying the H and U seedlings was a coarse Gaussian SVM with 9 bands because it had the shortest performance time of 1.7124 s.

21.5.3 BSR Detection Using SVM and a Small Number of Bands

This section presents the capability of a small number of bands for BSR detection Khairunniza-Bejo et al. (2021). The five most significant bands extracted from F12 as shown in Table 21.3 were used as datasets. The number of bands was later reduced to 4, 3, 2, and 1. The performance of the classification models was assessed using a confusion matrix where accuracy, sensitivity, and specificity were
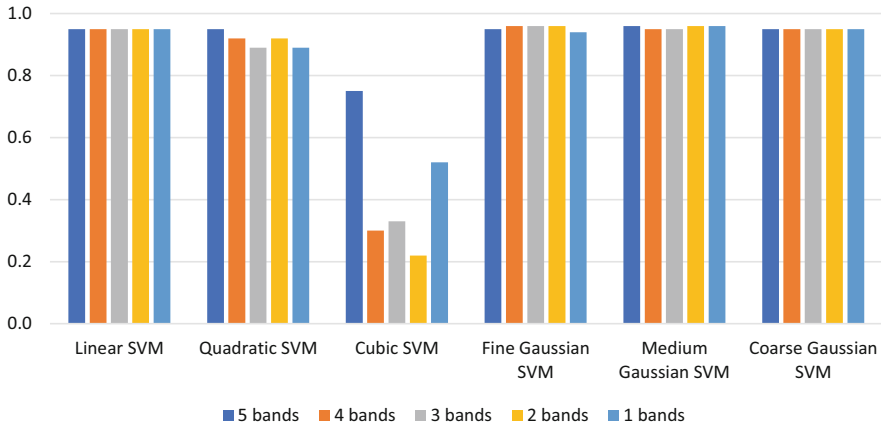
Fig. 21.14 Performance of each SVM model for AUC in each reduction in the number of wavelengths

calculated. Furthermore, the performance of the classification models was further evaluated using the receiver operating characteristic (ROC) and the area under the ROC curve (AUC). A ROC curve illustrates the performance of a classification overall categorisation thresholds. Meanwhile, the AUC quantifies the area beneath the ROC curve A model with 100% correct predictions would have an AUC of 1.0, while one with 100% incorrect predictions would have an AUC of 0.0. An AUC less than 0.5 cannot discriminate between positive and negative class values.

Figure 21.14 shows the performance of each SVM model for AUC in each reduction number of wavelengths. Based on this figure, the SVM models performed exceptionally well in classification, with an average AUC of 0.91 to 0.96, except for the cubic SVM, which had the lowest average AUC value of 0.42. This means that there is a high chance that most of the models were able to distinguish between the H and U seedlings. Since the models performed well in terms of classification, the accuracy, sensitivity, and specificity of each model were assessed using a coefficient of variance (CV) to determine the data dispersion. In general, almost all models yielded a good classification accuracy of over 90% in average accuracy, sensitivity, and specificity.

Overall, as summarised in Table 21.4, the linear SVM model had the lowest average CV for accuracy and specificity, as well as the second-lowest value for sensitivity. The lowest CV score of the linear SVM indicated that the reduced number of wavelengths did not affect the consistency of the model to detect a *G. boninense* infection. The linear SVM models maintained excellent accuracy (94.80%), sensitivity (97.60%), specificity (92.50%), and AUC (0.95), even when the number of wavelengths was optimised to 1. As a result, the best detection model was found to be the linear SVM with datasets from 934 nm. The model could distinguish between healthy and infected seedlings excellently and improved the classification accuracy obtained by Azmi et al. (Azmi et al., 2020) by 1.80%.

Table 21.4 Mean (x) and coefficients of variations (CV) for accuracy, sensitivity, and specificity of each SVM model

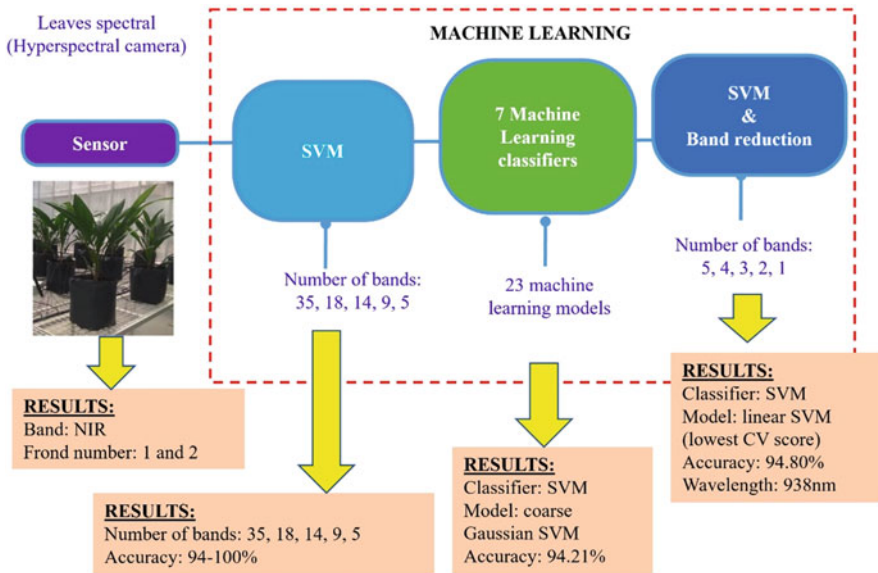| SVM Model | Accuracy | | Sensitivity | | Specificity | |
	Mean, x (%)	CV (%)	Mean, x (%)	CV (%)	Mean, x (%)	CV (%)
Linear	94.58	0.16	91.98	0.38	97.00	0.00
Quadratic	85.58	7.34	88.18	8.20	83.50	15.48
Cubic	47.28	30.32	44.55	49.06	49.50	21.16
Fine Gaussian	94.58	0.26	91.83	0.60	97.00	0.00
Medium Gaussian	94.43	0.16	90.65	0.45	97.50	0.59
Coarse gaussian	94.55	0.35	90.68	0.57	97.75	0.51

Fig. 21.15 Summary of the findings for the research of the early detection of *G. boninense* infection in the oil palm seedlings using hyperspectral data and machine learning techniques

21.6 Conclusion

This chapter presents the capability of hyperspectral imaging to detect an early stage of *G. boninense* infection in oil palm seedlings using machine learning techniques. In this research, the seedlings showed no visible symptoms after 20 weeks of inoculation, which indicated an early stage of *G. boninense* infection. However, the PCR test conducted after 4 weeks of inoculation confirmed the infection in the seedlings. Figure 21.15 summarises the findings of this research. The hyperspectral reflectance showed significant differences where the U seedlings yielded low reflection in the NIR spectrum as compared to the H seedlings. The NIR spectrum is

superior in detecting disease for asymptomatic infection due to its capability to obtain information of the interior tissues, compared with VIS which can only obtain exterior information that is invisible at the early stage of infection. The result also demonstrated that the F12 could be used for early detection of *G. boninense* infection in oil palm seedlings since F12 produced slightly lower classification accuracy than F1 but higher than F2. Therefore, it is acceptable for not separating between F1 and F2 during the pre-processing stage. This information is useful because this approach can be expanded to the more practical application using an unmanned aerial vehicle (UAV) where the identification of fronds could be challenging in aerial images. Indeed, SVM was identified as the best ML classifier in detecting BSR using various types of datasets extracted from NIR. Furthermore, a large number of bands (35 and 18) could achieve 100% accuracy, while a small number of bands achieved a slightly lower level of classification accuracy. Considering the economic aspect for hardware implementation in the future, a linear SVM which was developed using a single-band reflectance at 934 nm was identified as the best model for detection with a high score of accuracy (94.8%), sensitivity (97.6%), specificity (92.5%), and the area under the receiver operating characteristic curve (AUC) (0.95). The findings of this research have the potential to significantly benefit the oil palm industry by successfully detecting asymptomatic *G. boninense* infection, allowing for adequate treatment time and preventing disease spread. As a result, improved plantation management could result in higher yields, increased sustainability of palm oil production, and protection of the environment and communities in palm oil-producing regions. For future work, the method developed in this research could be implemented in an open environment or a real nursery to confirm its reliability for field application.

References

Abdulridha, J., Ampatzidis, Y., Roberts, P., & Kakarla, S. C. (2020). Detecting powdery mildew disease in squash at different stages using UAV–based hyperspectral imaging and artificial intelligence. *Biosystems Engineering, 197*, 135–148. https://doi.org/10.1016/j.biosystemseng.2020.07.001

Ahmadi, P., Muharam, F. M., Ahmad, K., Mansor, S., & Abu Seman, I. (2017). Early detection of *Ganoderma* basal stem rotof oil palms using artificial neural network spectral analysis. *Plant Disease, 101*, 1009–1016. https://doi.org/10.1094/PDIS-12-16-1699-RE

Arango, M., Martínez, G., & Torres, G. (2016). Advances in the interpretation of tomographic images as an early detection method of oil palm affected by basal stem rotin Colombia. *Plant Disease, 100*, 1559–1563. https://doi.org/10.1094/PDIS-12-15-1473-RE

Ariffin, D., & Seman, I. A. (1993) A selective medium for the isolation of *Ganoderma* from diseased tissues. In: *PORIM international palm oil conference progress prospects challenges towards the 21st century*, Kuala Lumpur, Malaysia, 9–14 September 1993.

Azmi, A. N. N., Bejo, S. K., Jahari, M., Muharam, F. M., Yule, I., & Husin, N. A. (2020). Early detection of *Ganoderma boninense* in oil palm seedlings using support vector machines. *Remote Sensing, 12*, 3920. https://doi.org/10.3390/rs12233920

Azmi, A. N. N., Khairunniza-Bejo, S., Jahari, M., Muharram, F. M., & Yule, I. (2021). Identification of a suitable machine learning model for detection of asymptomatic *Ganoderma boninense* infection in oil palm seedlings using hyperspectral data. *Applied Sciences, 11*, 11798. https://doi.org/10.3390/app112411798

Behmann, J., Steinrücken, J., & Plümer, L. (2014). Detection of early plant stress responses in hyperspectral images. *ISPRS Journal of Photogrammetry and Remote Sensing, 93*, 98–111. https://doi.org/10.1016/j.isprsjprs.2014.03.016

Bejo, S., Abdol-Lajis, G., Abd-Aziz, S., Abu-Seman, I., & Ahamed, T. (2018). Detecting basal stem rot (BSR) disease at oil palm tree using thermal imaging technique. In: *14th International conference on precision agriculture*, Montreal QC, Canada, 24–27 June 2018.

Breton, F., Miranti, R., Lubis, Z., Hayun, Z., Setiawati, U., Flori, A., & De Franqueville, H. (2009). Implementation of an early artificial inoculation test to screen oil palm progenies for their level of resistance and hypothesis on natural infection: *Ganoderma* disease of the oil palm. In 16th *International oil palm conference and expopalma challenges in sustainable oil palm development*, Cartagena de Indias, Columbia, 22–25 September 2009.

Chung, G. F. (2012). Effect of pests and diseases on oil palm yield. In O. M. Lai, C. P. Tan, & C. C. Akoh (Eds.), *Palm oil* (pp. 163–210). AOCS Press. https://doi.org/10.1016/B978-0-9818936-9-3.50009-5

Dale, L. M., Thewis, A., Boudry, C., Rotar, I., Dardenne, P., Baeten, V., & Pierna, J. A. F. (2013). Hyperspectral imaging applications in agriculture and agro–food product quality and safety control: A review. *Applied Spectroscopy Reviews, 48*, 142–159. https://doi.org/10.1080/05704928.2012.705800

Ezenne, G. I., Jupp, L., Mantel, S. K., & Tanner, J. L. (2019). Current and potential capabilities of UAS for crop water productivity in precision agriculture. *Agricultural Water Management, 218*, 158–164. https://doi.org/10.1016/j.agwat.2019.03.034

Feng, Y., Peng, Y., Cui, N., Gong, D., & Zhang, K. (2017). Modeling reference evapotranspiration using extreme learning machine and generalized regression neural network only with temperature data. *Computers and Electronics in Agriculture, 136*, 71–78. https://doi.org/10.1016/j.compag.2017.01.027

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture, 145*, 311–318. https://doi.org/10.1016/j.compag.2018.01.009

Goetz, A. F., Vane, G., Solomon, J. E., & Rock, B. N. (1985). Imaging spectrometry for earth remote sensing. *Science, 228*, 1147–1153. https://doi.org/10.1126/science.228.4704.1147

Govender, N. T., Mahmood, M., Seman, I. A., & Wong, M. Y. (2017). The phenylpropanoid pathway and lignin in defense against *Ganoderma boninense* colonized root tissues in oil palm (*Elaeis guineensis Jacq*). *Frontiers in Plant Science, 8*, 1395. https://doi.org/10.3389/fpls.2017.01395

Grinblat, G. L., Uzal, L. C., Larese, M. G., & Granitto, P. M. (2016). Deep learning for plant identification using vein morphological patterns. *Computers and Electronics in Agriculture, 127*, 418–424. https://doi.org/10.1016/j.compag.2016.07.003

Hu, H., Pan, L., Sun, K., Tu, S., Sun, Y., Wei, Y., & Tu, K. (2017). Differentiation of deciduous–calyx and persistent–calyx pears using hyperspectral reflectance imaging and multivariate analysis. *Computers and Electronics in Agriculture, 137*, 150–156. https://doi.org/10.1016/j.compag.2017.04.002

Huang, W., Lamb, D. W., Niu, Z., Zhang, Y., Liu, L., & Wang, J. (2007). Identification of yellow rust in wheat using in–situ spectral reflectance measurements and airborne hyperspectral imaging. *Precision Agriculture, 8*, 187–197. https://doi.org/10.1007/s11119-007-9038-9

Husin, N. A., Khairunniza–Bejo, S., Abdullah, A. F., Kassim, M. S. M., & Ahmad, D. (2020a). Study of the oil palm crown characteristics associated with basal stem rot (BSR) disease using stratification method of point cloud data. *Computers and Electronics in Agriculture, 178*, 105810. https://doi.org/10.1016/j.compag.2020.105810

Husin, N. A., Khairunniza-Bejo, S., Abdullah, A. F., Kassim, M. S., Ahmad, D., & Azmi, A. N. (2020b). Application of ground-based LiDAR for analysing oil palm canopy properties on the occurrence of basal stem rot (BSR) disease. *Scientific Reports, 10*, 1–16. https://doi.org/10.1038/s41598-020-62275-6

Husin, N. A., Khairunniza-Bejo, S., Abdullah, A. F., Kassim, M. S., Ahmad, D., & Aziz, M. H. (2020c). Classification of basal stem rot disease in oil palm plantations using terrestrial laser scanning data and machine learning. *Agronomy, 10*, 1624. https://doi.org/10.3390/agronomy10111624

Idris, A. S. (2009). Basal stem rotin Malaysia-Biology economic importance epidemiology detection and control. In *International workshop on awareness detection and control of oil palm devastating diseases*, Kuala Lumpur, Malaysia, 1 January 2009.

Idris, A. S., & Rafidah, R. (2008). Enzyme linked immunosorbent assay-polyclonal antibody (ELISA-PAb). *MPOB Information Series, 430*, 1–4.

Izzuddin, M. A., Idris, A. S., Wahid, O., Nishfariza, M. N., & Shafri, H. Z. M. (2013). Field spectroscopy for detection of *Ganoderma* disease in oil palm. *MPOB Information Series, 532*, 630.

Kandan, A., Bhaskaran, R., & Samiyappan, R. (2010). *Ganoderma*: A basal stem rotdisease of coconut palm in South Asia and Asia pacific regions. *Phytopathology and Plant Protection, 43*, 1445–1449. https://doi.org/10.1080/03235400802536527

Khairunniza-Bejo, S., Shahibullah, M. S., Azmi, A. N. N., & Jahari, M. (2021). Non–destructive detection of asymptomatic *Ganoderma boninense* infection of oil palm seedlings using NIR–hyperspectral data and support vector machine. *Applied Sciences, 11*, 10878. https://doi.org/10.3390/app112210878

Khaled, A. Y., Aziz, S. A., Bejo, S. K., Nawi, N. M., & Seman, I. A. (2018a). Spectral features selection and classification of oil palm leaves infected by basal stem rot (BSR) disease using dielectric spectroscopy. *Computers and Electronics in Agriculture, 144*, 297–309. https://doi.org/10.1016/j.compag.2017.11.012

Khaled, A. Y., Aziz, S. A., Bejo, S. K., Nawi, N. M., Seman, I. A., & Izzuddin, M. A. (2018b). Development of classification models for basal stem rot (BSR) disease in oil palm using dielectric spectroscopy. *Industrial Crops and Products, 124*, 99–107. https://doi.org/10.1016/j.indcrop.2018.07.050

Kotlar, A. M., Iversen, B. V., & de Jong van Lier, Q. (2019). Evaluation of parametric and nonparametric machine learning techniques for prediction of saturated and near saturated hydraulic conductivity. *Vadose Zone Journal, 18*, 1–13. https://doi.org/10.2136/vzj2018.07.0141

Kresnawaty, I., Mulyatni, A. S., Eris, D. D., Prakoso, H. T., Triyana, K., & Widiastuti, H. (2020). Electronic nose for early detection of basal stem rotcaused by *Ganoderma* in oil palm. In *IOP conference series: Earth and environmental science* (p. 012029). IOP Publishing.

Lelong, C. C., Roger, J. M., Brégand, S., Dubertret, F., Lanore, M., Sitorus, N. A., & Caliman, J. P. (2010). Evaluation of oil-palm fungal disease infestation with canopy hyperspectral reflectance data. *Sensors, 10*, 734–747. https://doi.org/10.3390/s100100734

Li, B., Xu, X., Zhang, L., Han, J., Bian, C., Li, G., & Jin, L. (2020). Above–ground biomass estimation and yield prediction in potato by using UAV–based RGB and hyperspectral imaging. *ISPRS Journal of Photogrammetry and Remote Sensing, 162*, 161–172. https://doi.org/10.1016/j.isprsjprs.2020.02.013

Liaghat, S., Mansor, S., Ehsani, R., Shafri, H. Z. M., Meon, S., & Sankaran, S. (2014). Mid-infrared spectroscopy for early detection of basal stem rotdisease in oil palm. *Computers and Electronics in Agriculture, 101*, 48–54. https://doi.org/10.1016/j.compag.2013.12.012

Madihah, A. Z., Idris, A. S., & Rafidah, A. R. (2014). Polyclonal antibodies of *Ganoderma boninense* isolated from Malaysian oil palm for detection of basal stem rotdisease. *African Journal of Biotechnology, 13*, 3455–3463. https://doi.org/10.5897/AJB2013.13604

Markom, M. A., Shakaff, A. M., Adom, A. H., Ahmad, M. N., Hidayat, W., Abdullah, A. H., & Fikri, N. A. (2009). Intelligent electronic nose system for basal stem rotdisease detection. *Computers and Electronics in Agriculture, 66*, 140–146. https://doi.org/10.1016/j.compag.2009.01.006

Maryam, K., Khairunniza-Bejo, S., & Biswajeet, P. (2018). Geospatial technologies for detection and monitoring of *Ganoderma* basal stem rot infection in oil palm plantations: A review on sensors and techniques. *Geocarto International, 33*, 260–276. https://doi.org/10.1080/10106049.2016.1243410

Matthews, S. G., Miller, A. L., PlÖtz, T., & Kyriazakis, I. (2017). Automated tracking to measure behavioural changes in pigs for health and welfare monitoring. *Scientific Reports, 7*, 1–12. https://doi.org/10.1038/s41598-017-17451-6

Mazliham, M. S., Pierre, L., & Idris, A. S. (2008). Towards automatic recognition and grading of *Ganoderma* infection pattern using fuzzy systems. *Engineering, Computing & Technology, 19*, 1–6. https://doi.org/10.5281/zenodo.1061936

Mitchell, T. M. (1997). Does machine learning really work? *AI Magazine, 18*, 11–20. https://doi.org/10.1609/aimag.v18i3.1303

Mohd Hilmi Tan, M. I. S., Jamlos, M. F., Omar, A. F., Dzaharudin, F., Chalermwisutkul, S., & Akkaraekthalin, P. (2021). *Ganoderma boninense* disease detection by near-infrared spectroscopy classification: A review. *Sensors, 21*, 3052. https://doi.org/10.3390/s21093052

Mohd Johari, S. N. A., Khairunniza-Bejo, S., Abdol Lajis, G., Jeffery Daim, L. D., Neoh, B. K., Yap, Y. C., & Ithnin, N. (2021). Detecting BSR infected oil palm seedling using thermal imaging technique. *Basrah Journal of Agricultural Sciences, 34*, 73–80. https://doi.org/10.37077/25200860.2021.34.sp1.8

Morales, I. R., Cebrián, D. R., Blanco, E. F., & Sierra, A. P. (2016). Early warning in egg production curves from commercial hens: An SVM approach. *Computers and Electronics in Agriculture, 121*, 169–179. https://doi.org/10.1016/j.compag.2015.12.009

Morellos, A., Pantazi, X. E., Moshou, D., Alexandridis, T., Whetton, R., Tziotzios, G., & Mouazen, A. M. (2016). Machine learning based prediction of soil total nitrogen organic carbon and moisture content by using VIS–NIR spectroscopy. *Biosystems Engineering, 152*, 104–116. https://doi.org/10.1016/j.biosystemseng.2016.04.018

Naher, L., Intan, S., Mokhtar, B., & Sidek, N. (2015). Trichoderma harzianum T32 growth and antagonistic performance against *Ganoderma boninense* on different culture media. In *3rd international conference on biological chemical & environmental science (BCES-2015)*, Kuala Lumpur, Malaysia, 21–22 September 2015.

Naidu, Y., Siddiqui, Y., Rafii, M. Y., Saud, H. M., & Idris, A. S. (2018). Inoculation of oil palm seedlings in Malaysia with white–rot hymenomycetes: Assessment of pathogenicity and vegetative growth. *Crop Protection, 110*, 146–154. https://doi.org/10.1016/j.cropro.2018.02.018

Nisfariza, M. N., Idris, A. S., Shafri, Z. H., Steven, M., & Boyd, D. (2010). Hyperspectral derivative band ratios of oil palm stress associated with *Ganoderma* basal stem rot disease. In: *MRSS 6th international remote sensing & GIS conference and exhibition 2010*, Kuala Lumpur, Malaysia, 28–29 April 2010.

Nur, A. H., Khairunniza-Bejo, S., Abdullah, A. F., Kassim, M. S., & Ahmad, D. (2021). Multi-temporal analysis of terrestrial laser scanning data to detect basal stem rotin oil palm trees. *Precision Agriculture, 23*, 1–26. https://doi.org/10.1007/s11119-021-09829-4

Oettli, P., Behera, S. K., & Yamagata, T. (2018). Climate based predictability of oil palm tree yield in Malaysia. *Scientific Reports, 8*, 1–13. https://doi.org/10.1038/s41598-018-20298-0

Pantazi, X. E., Tamouridou, A. A., Alexandridis, T. K., Lagopodi, A. L., Kontouris, G., & Moshou, D. (2017). Detection of Silybum marianum infection with Microbotryum silybum using VNIR field spectroscopy. *Computers and Electronics in Agriculture, 137*, 130–137. https://doi.org/10.1016/j.compag.2017.03.017

Parker, I. M., & Gilbert, G. S. (2007). When there is no escape: The effects of natural enemies on native invasive and non-invasive plants. *Ecology, 88*, 1210–1224. https://doi.org/10.1890/06-1377

Rakib, M. R. M., Borhan, A. H., & Jawahir, A. N. (2019). The relationship between SPAD chlorophyll and disease severity index in *Ganoderma*-infected oil palm seedlings. *Journal of the Bangladesh Agricultural University, 17*, 355–358.

Rapaport, T., Hochberg, U., Rachmilevitch, S., & Karnieli, A. (2014). The effect of differential growth rates across plants on spectral predictions of physiological parameters. *PLoS One, 9*, e88930. https://doi.org/10.1371/journal.pone.0088930

Santoso, H., Gunawan, T., Jatmiko, R. H., Darmosarkoro, W., & Minasny, B. (2011). Mapping and identifying basal stem rotdisease in oil palms in North Sumatra with QuickBird imagery. *Precision Agriculture, 12*, 233–248. https://doi.org/10.1007/s11119-010-9172-7

Shafri, H. Z., Anuar, M. I., Seman, I. A., & Noor, N. M. (2011). Spectral discrimination of healthy and *Ganoderma*–infected oil palms from hyperspectral data. *International Journal of Remote Sensing, 32*, 7111–7129. https://doi.org/10.1080/01431161.2010.519003

Su, Y. X., Xu, H., & Yan, L. J. (2017). Support vector machine–based open crop model (SBOCM): Case of rice production in China. *Saudi Journal of Biological Sciences, 24*, 537–547. https://doi.org/10.1016/j.sjbs.2017.01.024

Torres, I., Sánchez, M. T., Cho, B. K., Garrido-Varo, A., & Pérez-Marín, D. (2019). Setting up a methodology to distinguish between green oranges and leaves using hyperspectral imaging. *Computers and Electronics in Agriculture, 167*, 105070. https://doi.org/10.1016/j.compag.2019.105070

Turnbull, N., de Franqueville, H., Breton, F., Jeyen, S., Syahoutra, I., Cochard, B., & Durand-Gassellin, T (2014). Breeding methodology to select oil palm planting material partially resistant to *Ganoderma boninense*. In *5th quadrennial international oil palm conference Bali*, Nusa Dua Convention Center, Indonesia, 17–19 June 2014.

Wulandari, Y. R. E., Felicia, F., Arifin, A. R., & Suwanto, A. (2018). EgMLP1 gene expression in oil palm Ramet infected with *Ganoderma boninense*. *International Journal of Oil Palm, 1*, 71–78.

Zhang, T., Fan, S., Xiang, Y., Zhang, S., Wang, J., & Sun, Q. (2020). Non–destructive analysis of germination percentage germination energy and simple vigour index on wheat seeds during storage by VIS/NIR and SWIR hyperspectral imaging. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy, 239*, 118488. https://doi.org/10.1016/j.saa.2020.118488

Chapter 22
Strategic Short Note: Development of an Automated Speed Sprayer for Apple Orchards in Japan

Ryozo Noguchi

Abstract Pesticide application is essential for stable fruit production in Japan. Pesticide application in orchards is often performed by Speed Sprayers (SS), which is equipped with a large column tank and spraying system from underneath the canopy. However, there is risk in long time operation for the operator and as well as the environment due to a large amount of spraying of pesticides. Therefore, automated driving of SS and automated pesticide application technology in smart agriculture are important in addressing the decrease in the number of farm workers and freeing them from pesticide exposure due to pesticide spraying using advanced sensors. Thus, it was aimed to improve the stability of straight driving control and turning control by using not only GNSS but also optical markers, image processing using camera images, and LiDAR. In this chapter, the development of an automated speed sprayer for apple orchards in Japan was discussed with recent advances in sensors and its perceptions. Therefore, the digitalization and feedback of the spraying condition provides an opportunity to enable a shift to environment- and ecosystem-friendly agriculture for orchard-based systems.

Keywords ArUco marker · Autonomous control · GNSS · LiDAR · RGB camera · Speed sprayer

22.1 Introduction

22.1.1 Apple Production and Pesticide Spraying

The current production of apples in Japan is approximately 750,000 tons. There is a high demand for Japanese apples overseas, and approximately 50,000 tons are exported to Taiwan and other countries. Aomori Prefecture produces the most apples of any prefecture in Japan, with an area of approximately 20,000 ha and an annual

R. Noguchi (✉)
Graduate School of Agriculture, Kyoto University, Kyoto, Japan
e-mail: noguchi.ryozo.8j@kyoto-u.ac.jp

Fig. 22.1 Apple cultivation methods (from left to right: normal tree cultivation, cropped tree cultivation, and high-density cropping, with target yields of approximately 40 t/ha, 50 t/ha, and 60 t/ha at maturity, respectively) (Apple Research Institute)

production volume of approximately 450,000 tons. In Aomori Prefecture, a shift is underway from conventional regular tree cultivation to dwarf cultivation and high-density dwarf cultivation to improve the efficiency and productivity of apple farm operations (Fig. 22.1).

Apple cultivation requires a large amount of agricultural work throughout the year, including fertilization, pollination, flower, and fruit picking, pest control, bagging, weed management, weeding, limbing, pole setting, bagging, reflective sheeting, leaf plucking, fruit turning, harvesting, fruit selection, pruning, and rough skin scraping. Pest control by spraying agricultural chemicals, which takes place from April to August, needs to be done about 12 times a year. In addition, in recent years, Aomori Prefecture has seen many outbreaks of black star disease, and pesticide application time has increased due to the need for careful pest control.

In many cases, pesticide application by using speed sprayers (SS) without a cabin requires the wearing of pest control clothing, which increases both temperature and humidity, making SS operators significantly uncomfortable. In addition, there are reports of a shortage of SS operators due to the aging of the workforce, pesticide exposure of operators, hearing impairment due to noise, fatigue due to vibration, and fatal accidents due to collisions with tree branches and fruit tree trellis wires. Furthermore, apple cultivation requires good drainage and sufficient solar radiation to improve sweetness, and many sloping fields located in mountainous areas cause the accident in turning over of SS (Fig. 22.2). In addition, SS is generally operated by men. As a result, there are many cases of family farmers abandoning their apple orchard operations due to SS accidents; then there are high expectations from the field for automated pesticide spraying technology.

22.2 Past Efforts to Automate SS and Development Goals

In the past, an automated SS had achieved the same level of chemical application efficiency as like manned SS by applying electric current to an induction cable placed underground and using a sensor to detect the magnetic field generated. However, this technology has not been widely used due to the occurrence of cable

Fig. 22.2 Pesticide application in a sloping field (Soma Village, Aomori Prefecture)

breakage caused by rodents and the high cost of installation. In addition, markers laid on orchard surfaces are easily destroyed or lost due to weed control or plowing.

On the other hand, the spread of high-precision satellite positioning enabled inexpensive and highly accurate automation in outdoor vehicles, contributing greatly to the automation of agricultural machinery. However, it has been suggested that in orchards, stable use of high-precision satellite positioning is difficult due to the influence of fruit tree branches and leaves. Therefore, we aimed to improve the stability of straight driving control and turning control by using not only GNSS but also optical markers, image processing using camera images, and LiDAR.

Since orchards, where conventional SS are used, have a variety of fruit tree species and cultivation methods, the orchard environment has a significant impact on the development policy of automated SS. In this project, with the cooperation of the Aomori Apple Research Institute, we focused on the automation of pesticide application in the cultivation of dwarf trees. With an eye toward practical application, the SSA-V1002CDX (Maruyama Co., Ltd.) was used as the first prototype, and the SSA-V602C-DX (Maruyama Co., Ltd.) as the second prototype, to perform automated driving and spraying using SS vehicles currently in use.

22.3 GNSS Application

A multi-band high-precision satellite positioning device capable of transmitting reference station signals via specified low-power communications was fabricated. In addition, we have confirmed that the signal-receiving performance is good during the period when the fruit tree branches and leaves are at their peak growth stage in the apple orchard at the Apple Research Institute. Furthermore, we confirmed that the receiving performance was satisfactory on all driving paths even in the apple orchard that had been flattened by cutting down the slope. Therefore, it was confirmed that self-position estimation by GNSS alone was sufficient for driving accuracy. Furthermore, it was found that the system can precisely reproduce the

Fig. 22.3 Speed sprayer (SSA-V602C-DX (Maruyama Co., Ltd.)) tested in the experiment, and scenes of automatic driving, automatic spraying, and pillow turning by self-position estimation using GNSS in a dwarf apple orchard (Aomori Apple Research Institute)

Fig. 22.4 Speed sprayer (SSA-V1002CDX (Maruyama Co., Ltd.)) used in the experiment and automatic water application in a jointed pear orchard (T-PIRC farm, University of Tsukuba)

driving route, driving speed, timing of pesticide spraying, etc., by learning teacher signals in advance at the Apple Research Institute by workers who specialize in SS, and it is clear that driving with sufficient accuracy for practical use is possible (Fig. 22.3).

22.4 ArUco Markers and GNSS Application

An RTK-GNSS base station was installed in the T-PIRC farm at the University of Tsukuba, and ArUco markers were installed in a joint pear orchard, which was close to the row of fruit trees in the apple orchard for autonomous driving experiments (Fig. 22.4).

First, the potential use of ArUco markers for self-location estimation in orchards was examined. The results showed that the recognition rate of ArUco markers by the RGB camera was 79% at 15 m, 92% at 10 m, and 99% at 5 m from the mounting position. In addition, ArUco markers were recognized with an accuracy of more than
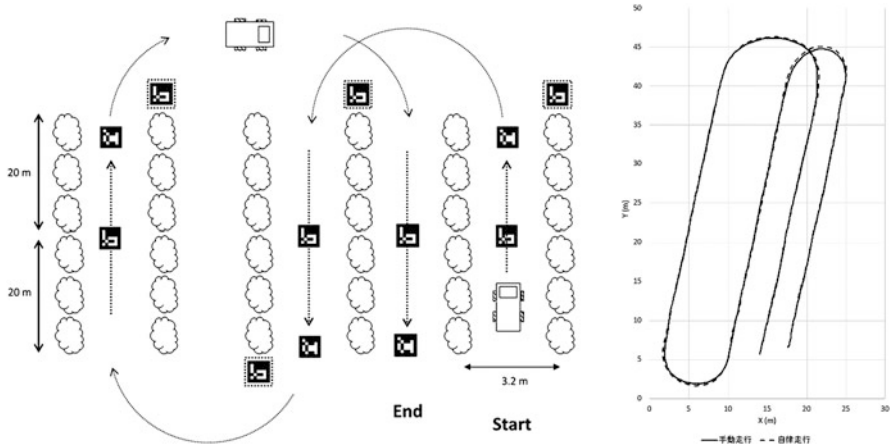
Fig. 22.5 Traveling paths and trajectories in a joint-tailored pear orchard

90% regardless of forward or backlight. However, the recognition rate dropped to less than 25% in backlit conditions where sunlight directly hit the lens and caused white-out.

Next, automatic SS driving was examined using GNSS signals. When single-band high-precision satellite positioning was available, the GNSS, ArUco marker, and azimuth angle by azimuth sensor were used, and when the ArUco marker indicating the start of turning was recognized, the SS shifted to turning. As a result, the average difference in distance from the ideal path was 0.14 m in the driving experiment in a joint-tailored pear orchard, confirming sufficient driving accuracy (Fig. 22.5).

22.5   LiDAR Application

LiDAR is a sensor that measures the position of obstacles by the reflection time of an infrared laser. Among them, 2D LiDAR has the advantages of low disturbance in outdoor environments, relatively low cost, and low computational complexity. In this study, a two-dimensional LiDAR (Hokuyo Electric, UST-10LX) was installed in front of an SS to investigate automatic driving by generating a path using the potential method (Khatib, 1990). The use of the potential method in this study was to determine the path based on the position information of obstacles (fruit trees), so the potential method could not be used during turning. Therefore, in addition to the algorithm of the potential method for straight driving, a turning algorithm was added to move to the next orchard driving path and return to straight driving. As a result, the mean square deviation was 0.21 m and the maximum error was 0.36 m on the orchard test course. Since the vehicle width of the SS used in this study was 1.45 m and the orchard targeted was approximately 3.20 m wide, the driving path had a
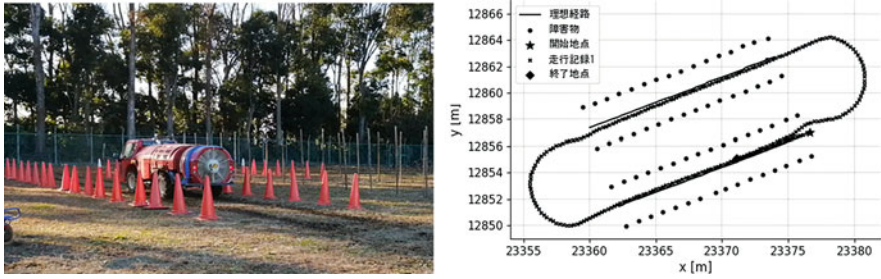
Fig. 22.6 Driving paths and trajectories on the orchard test course

margin of approximately 0.85 m on either side of the vehicle, which was sufficient for adequate accuracy (Fig. 22.6).

22.6 Future Tasks

The development of automated SS in this task to date has shown a sufficient potential to achieve automated driving and spraying in orchard fields. However, in the use of SS in orchard fields, if the SS in operation needs to be replenished with chemicals, it will be necessary to travel from the location where the pesticide application is interrupted to the location where the replenishment of chemicals was interrupted. In addition, to reduce the amount of extra pesticide application as much as possible, it is necessary to add functions such as collection and storage of pesticide application operation data and real-time remote data transmission. In addition, there are issues to be addressed for practical application, such as how to use information from LiDAR to determine the level of branches and leaves that are safe for collision with the SS when they are extending into the driving path in the orchard. Furthermore, it is still being confirmed whether LiDAR can safely stop an automated SS while the SS is confronted by humans or other sudden obstacles.

Reference

Khatib, O. (1990). *Real-time obstacle avoidance for manipulators and mobile robots, autonomous robot vehicles* (pp. 396–404). Springer.

Chapter 23
The Spectrum of Autonomous Machinery Development to Increase Agricultural Productivity for Achieving Society 5.0 in Japan

Tofael Ahamed

Abstract Automation levels have different definitions across the Japanese, European, and US standards. Japanese automation starts with Level 0, which is the guidance referring to the full control of humans over the machine. Level 1 is called function-specific automation and refers to guidance; it is the first involvement of a vehicle controlling any function such as acceleration, steering, and braking through the system. Level 2 refers to the optimization and coordination with multiple functions controlled by the vehicle at the same time through an adaptive cruise control system and lane-keeping assistance (combine function automation). Level 3 refers to the most functions controlled by the vehicle and human assistance if needed. This level is referred to as assisted autonomy. One of the common examples is operator-assisted autonomy with one manned tractor backing up an unmanned tractor. Advancing from supervised autonomy to full autonomy is referred to as Level 4, which is a significant progress for Japanese companies. The safety regulations from the Japanese government need to meet at the same time to adopt full autonomy in Japanese agriculture.

Keywords Automation level · Function-specific automation · Combine function automation · Operator- assisted autonomy · Unmanned tractor · Full autonomy

23.1 Introduction

The agricultural field in Japan is facing a critical problem when it needs to consider aging farmers whose average age is 67 without replacement by successors (Jöhr, 2012; The Japan Times, 2018). Despite standing in the top ten largest agricultural businesses, with billions of dollars coming from exporting numerous products such as traditional cuisines, Japan's agricultural labor sector shrank 22% in the last

T. Ahamed (✉)
Faculty of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
e-mail: tofael.ahamed.gp@u.tsukuba.ac.jp

10 years (Goedde et al., 2016). More specifically, the number of full-time farmers was 1.7 million in 2014, declining from 2.2 million a decade earlier. This explains the rise in abandoned farmland to 420,000 hectares, which is twice that in the previous 20 years (Japan Today, 2016). The rapid labor decline phenomenon has also led to the doubling in workload for each farmer regardless of the low safety, possibly caused by elderly health (Qiu et al., 2014). Considering this circumstance of labor shortages in the near future among farmers' communities in Japan, the Japanese government is urgently moving toward technology solutions, which refer to autonomous machinery such as self-driving tractors (Torii, 2000; Li et al., 2009). The machine is designed with a high-technology combination of sensors, cameras, and internet applications to take over some primary and hard work, such as steering, harvesting, and sorting (Mousazadeh, 2013). For instance, the government focuses on deploying the introduction of self-driving tractors by hoping that such machines will help farmers deal with the fieldwork and reduce the tasks at hand.

Generally, both government and the agricultural machinery industry's anticipation of the new technology is reducing the workload and production costs while boosting productivity and uniformity. These benefits eventually strengthen the competitiveness of Japanese agriculture regardless of the aging population. Moreover, most of the machines produced by Japanese manufacturers aim at rice cultivation and hopefully can be exported to countries with the same culture, for instance, Southeast Asia and China (Yaoming et al., 2005). The market for autonomous machinery in agricultural applications is expanding rapidly, creating opportunities for domestic and international competition. Therefore, this chapter highlights the spectrum of autonomy to address the current state of development, challenges, industrial anticipation, and outlook to bring the technologies to the market at a certain level of autonomy to address the acute labor shortages in agricultural field operations.

23.2 Autonomous Machinery: Japanese Spectrum and Current State of Development

The autonomous spectrum in Japan is essential for innovation guidance, as the low range is conclusively a significant barrier to the development in the country and its product's strength in the global market. Slightly different from the USA, the Japanese spectrum refers to five ascending levels of autonomy from zero to four corresponding to the percentage of automated control capacity (Fig. 23.1). According to the guideline for autonomous vehicles mentioned in the METI report of Japan Ministry of Economy, Trade, and Industry in 2014: Driverless tractor development is already advanced in the US, where John Deer, CASE, AGCO, and others have working technology that can send tractors on preprogrammed routes. In Japan, the development of autonomous machines is emerging rapidly (Japan Ministry of Economy, Trading and Industry, 2015). The effort to realize the aim is supported in many ways. If referred to the ascending levels of autonomy (US), the
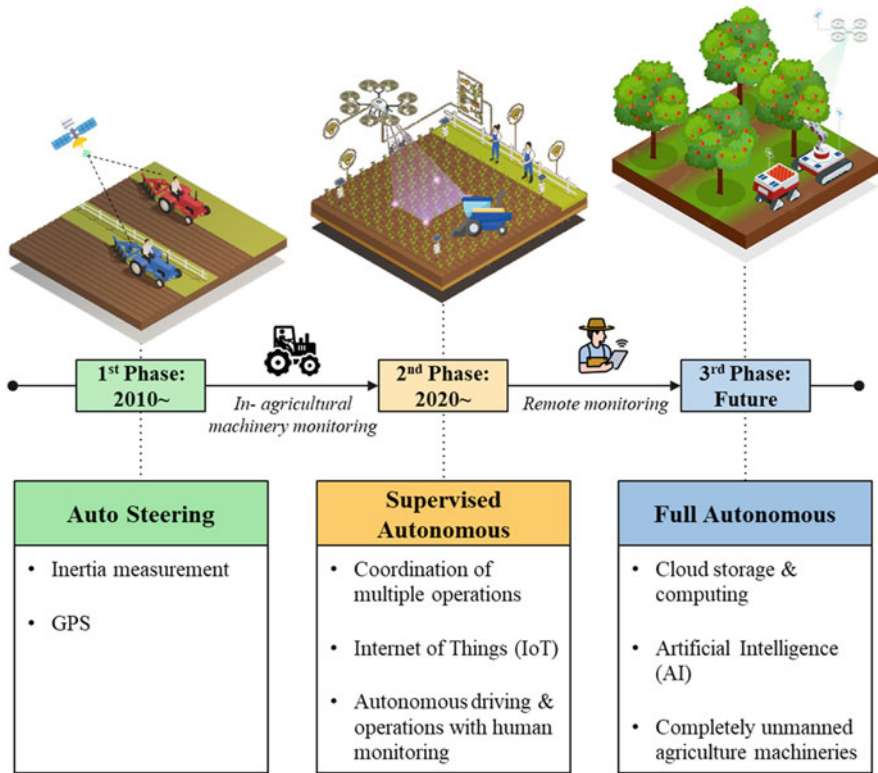
Fig. 23.1 The development direction of the autonomous spectrum of agricultural machinery in Japan

development of autonomous machines in Japan is between levels (3) operator-assisted autonomy and (4) supervised autonomy.

Japanese tech companies ranging from small scales to large firms, including Mitsubishi, Fujitsu, and Panasonic, are racing into research development investment to gain the best opportunities and profits in both the domestic and global markets. India and the Asia-Pacific countries are regarded as the most excitable and potential places for sales (The Japan Times, 2018). The autonomous machine market is already being exploited by the vegetable-growing industry. The "agri-tech" business has featured both Japanese and international media, with reports of indoor "vertical farms" and automated greenhouses gaining the most coverage (Goto, 2012; Kozai et al., 2015). The ongoing process emphasizes robotic automation using continuously updated IT systems and sensors to manipulate environmental factors, evidently enhancing work efficiencies and crop yield and product quality. For example, in all-robot farms, all operations are performed by robots from replanting young seedlings to the end of harvesting crops.

23.3   Solution Developments

Although Kubota has started selling the country's first autonomous tractors on a trial basis, the machines still need to be monitored as Level 3 of the autonomy. Kubota assumes that farmers will operate two tractors at a time, one with a driver and the other unmanned. In addition, the monitoring is aligned with the developed safety guidelines available that only allow the usage of autonomous machinery with on-site human monitoring (Kubota, 2017). The Ministry of Agriculture, Forestry and Fisheries (MAFF) has set objectives for achieving the commercialization of autonomous driving systems on farming lands and unmanned remote monitoring systems. However, to put these methods into practical use, many hurdles remain. Therefore, the next step is to use artificial intelligence (AI) to transform the agricultural robot into a smart robot through a deep learning process guided by farmers with support from technical experts (Yanmar, 2017). In addition, the challenges remaining for the most difficult technique are making the tractor turn without colliding and reducing the time for the exact time. Additionally, safety guidelines also need to be prepared for remote monitoring. It is difficult for the industry to achieve the target set by the government, and at the same time, it is making significant progress in putting the machine into practical use and remote monitoring. The work or any instruction for autonomous machines first needs to be preprogrammed. The machinery, which depends on GPS and other wireless technologies, is available to locate its position. It has been equipped with front/rear/left obstruction detection sensors, and the installation of a safety system, such as sounding an alert, slowing down, and stopping if an obstruction is detected within a specified distance (Zhang et al., 2018). The evolution of machines into smart robots is robust due to AI technology that utilizes machine intelligence and human-thinking ability to process various data to make predictions, recommendations, and decisions.

23.4   Industrial Anticipation and Outlook

Japanese manufacturers are aiming at fully automated machines that successfully support labor savings with high productivity. Moreover, intensive human supervision is not required but is a priority authorized with remote control. Their business model is continually releasing new technology for new businesses and keeping up with the trend. Technology companies are expected to automate many agricultural tasks according to 1 year of cultivation work in an integration system. This sophisticated idea is developed with the combination of cameras, GNSS, and AI learning processes. The increasingly complex and powerful software and hardware allow machines to achieve better interactions and full automation. A wide range of equipment has been introduced to support their customers doing the work with the best-trusted quality. In machinery control, the ISOBUS integrated system is for full control over and helps the machine achieve its task. The operation program is

designed to relieve work stress and provide a better overview of the work process. In the case of optimization, the precision of work such as seeds, fertilizers, and pesticide applications is improved to obtain the optimal result most comfortably. Autonomous operation replicating farmers will be the best target for machines.

23.5 Market Commercialization with Geographic Application Targets

Market commercialization by the major Japanese manufacturing companies is steadily adapting GNSS guidance and autosteering systems. However, smart agricultural machinery market size and forecasting show that the major contributing parts, such as software support, including complex environmental support, production support, and cultivation support software, must be properly addressed. Furthermore, marketing support solutions and management support solutions need to be ensured once the autonomous units are ready for use by the farmer. Additionally, technology support should cover GNSS guidance, autosteering, vehicle robot system and precision agriculture technology. The development phases of market commercialization have expanded from 2010 when it was only autosteering and GPS guidance. Autonomous applications need to be more reliable under human supervision. Therefore, market commercialization targets more unmanned systems under supervised support with the help of artificial intelligence (Fig. 23.2).

Kubota, the leading machinery manufacturer, aims to provide an autonomous tractor model for small tractors. The existing autonomous tractor developed between 60 and 100 hp., and these tractors are aimed at lowland cultivation, such as rice plantations, while upland and dry-field farming is supported by the M7 series. The M7 series tractor is a flagship model with a powerful engine of between 130 and 170 horsepower (Kubota, 2017). This is due to the expansion of large-scale farmers and farming operations. Kubota aims to expand the large-scale dry-field agricultural equipment business in Europe and North America. The new M7 models also won the Good Design Award 2016 and are excellently equipped with an autosteering function lining up with the Farm Pilot series supported by GPS.

23.6 Conclusions

A significant shortage of the agricultural labor force is threatening Japanese sustainable development in agriculture and food production. This circumstance highly motivates mechanization transformation, which creates a large market opportunity for autonomous machinery companies. To meet market demands, Japanese firms aim to develop integrated autonomous systems for crop plantation and prediction that will bring the best value to users as well as financial and technical support for
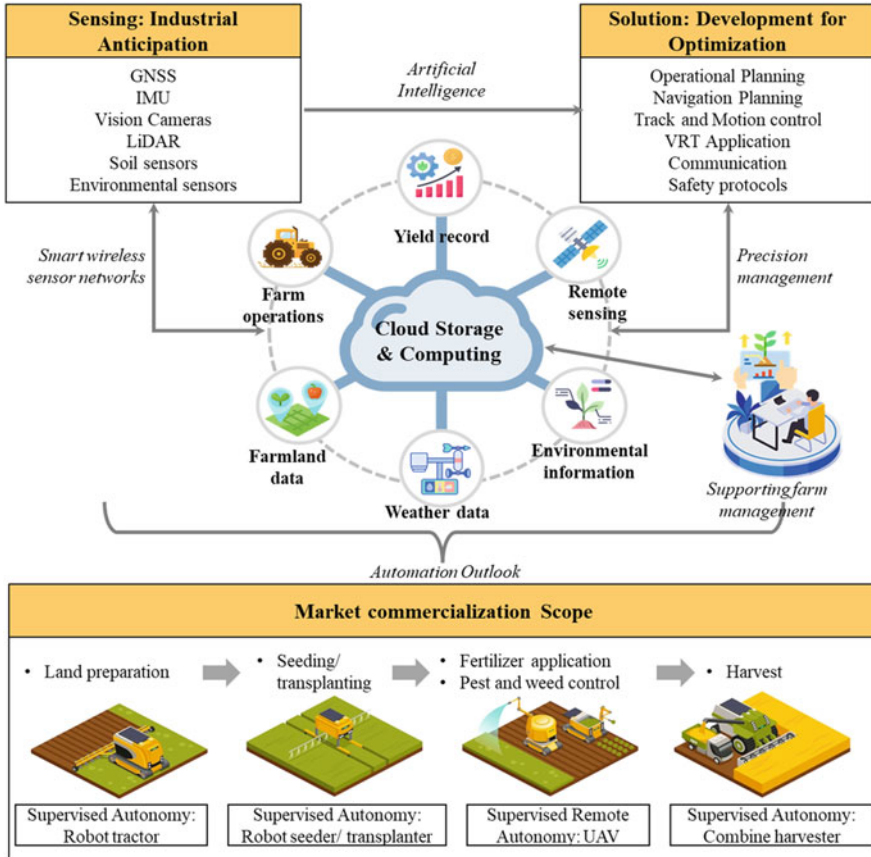
Fig. 23.2 Industrial anticipation and outlook for commercialization of autonomous solution

ensuring long-term productivity. In addition, the Japanese government's full support for autonomous machinery development in agriculture applications is also expected to meet new safety standards for supervised autonomy. Leading firms by Kubota Corp are close to half of the current market value running for establishing completely autonomous tractors. Assisting the development, the spectrum of autonomous machines consisting of five ascending levels acts as a guideline for both government and companies to further develop new products and establish a flexible legal system. In addition, industrial sectors have solutions for the market, which can be maximized for increasing machine productivity, user benefits, and conformation.

References

Goedde, L., Fischer, J., Denis, N., Tanaka, M., & Yamada, Y. (2016). *Empowering Japanese agriculture for global impact*. McKinsey Japan. Retrieved from https://www.mckinsey.com/~/media/mckinsey/global%20themes/asia%20pacific/strengthening%20japanese%20agriculture%20to%20maximize%20global%20reach/empowering-japanese-agriculture-report.ashx

Goto, E. (2012). Plant production in a closed plant factory with artificial lighting. In *VII international symposium on light in horticultural systems 956* (pp. 37–49).

Japan Ministry of Economy, Trading and Industry. (2015). *Summary of Japan's robot strategy*. Retrieved from http://www.meti.go.jp/english/press/2015/pdf/0123_01c.pdf

Japan Today. (2016). Japan's 'agri-tech' farming revolution. Retrieved from https://japantoday.com/category/tech/japans-agri-tech-farming-revolution

Jöhr, H. (2012). Where are the future farmers to grow our food? *International Food and Agribusiness Management Review, 15*(Special Issue A), 9–11.

Kozai, T., Niu, G., Takagaki, M., & (Eds.). (2015). *Plant factory: An indoor vertical farming system for efficient quality food production*. Academic Press.

Kubota. (2017). *Kubota*. Available at: http://www.kubota-global.net/news/2017/20170125.html

Li, M., Imou, K., Wakabayashi, K., & Yokoyama, S. (2009). Review of research on agricultural vehicle autonomous guidance. *International Journal of Agricultural and Biological Engineering, 2*(3), 1–16.

Mousazadeh, H. (2013). A technical review on navigation systems of agricultural autonomous off-road vehicles. *Journal of Terramechanics, 50*(3), 211–232.

Qiu, Z., Chen, B., & Takemoto, K. (2014). Conservation of terraced paddy fields engaged with multiple stakeholders: The case of the Noto GIAHS site in Japan. *Paddy and Water Environment, 12*(2), 275–283.

The Japan Times. (2018). *Japan may be in hot pursuit of robotics, but on the farm it's still all about human hands*. Retrieved from https://www.japantimes.co.jp/news/2018/12/25/business/japan-may-hot-pursuit-robotics-farm-still-human-hands/

Torii, T. (2000). Research in autonomous agriculture vehicles in Japan. *Computers and Electronics in Agriculture, 25*(1–2), 133–153.

Yanmar. (2017). *Will the robot be our Savior? The future of agriculture changed by technology*. Available at: https://www.yanmar.com/en_th/about/ymedia/product/agri_robot.html

Yaoming, L., Lizhang, X., Zhongping, X., & Lingli, D. (2005). Research advances of rice planting mechanization in Japan. *Transactions of The Chinese Society of Agricultural Engineering, 11*, 041.

Zhang, Y., Gao, P., & Ahamed, T. (2018). Development of a rescue system for agricultural machinery operators using machine vision. *Biosystems Engineering, 169*, 149–164.