OIL PALM YIELD DATA COLLECTION USING IMAGE PROCESSING

BY

RACHEL YEE JEE SAN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS)

COMPUTER ENGINEERING

Faculty of Information and Communication Technology

(Kampar Campus)

JANUARY 2021

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: OIL PALM YIELD DATA COLLECTION USING IMAGE PROCESSING

Academic Session: 202101

I __________________RACHEL YEE JEE SAN____________________

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature) (Supervisor's signature)

Address:
66, LALUAN PANORAMA 4,
TAMAN PANORAMA RAPAT INDAH
31350 IPOH, PERAK.

Goh Hock Guan
Supervisor's name

Date: 14/4/2021

Date: 15/4/2021

OIL PALM YIELD DATA COLLECTION USING IMAGE PROCESSING

BY

RACHEL YEE JEE SAN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS)

COMPUTER ENGINEERING

Faculty of Information and Communication Technology

(Kampar Campus)

JANUARY 2021

DECLARATION OF ORIGINALITY

I declare that this report entitled "**OIL PALM YIELD DATA COLLECTION USING IMAGE PROCESSING**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ____________________________

Name : RACHEL YEE JEE SAN

Date : 14/4/2021

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to my supervisor, Dr. Goh Hock Guan for guiding me and supporting me throughout the Final Year Project. His invaluable guidance, suggestions and comments have brought me to where I am today in being able to complete my Final Year Project successfully.

Most importantly, I would like to thank my parents for giving me unconditional support and encouragement in the pursuit of this project. They have always been very supportive and this project would not have been possible without them.

ABSTRACT

This project is an automated drone program integrated with image processing for academic purpose. It will provide students with the methodology, concept and design of an autonomous drone with image processing. This will be illustrated through the training of an ANN for image processing and also provide the basic controls to run an automated drone. The motivation for this project is to solve the traditional way of manually counting oil palm fruits. Spending hours of observation in rough weather conditions is a tedious job and it could be a problem for elderly farmers who are no longer flexible in moving around the big oil palm plantations. In the area of image processing, this job involves different techniques such as pre-processing, feature extraction and ANN. The tools used in training the ANN is the TensorFlow Object Detection API. There are many algorithms in the Object Detection API and three common methods, Faster R-CNN, SSD and YOLO are reviewed for their suitability in object detection. In the end, Faster R-CNN is chosen because its accuracy is the best compared to others, since accuracy is a priority in detecting the production yield for oil palm fruits. This API is important in object classification and counting which serves as the final product in the system. Autonomous drone also plays a big role in this system as it helps in capturing the images from the oil palm plantation. This area involves techniques such as path finding and stabilising in order to control the drone. The completion of this project will take up to two semesters and is divided into two main fields. These two fields include the autonomous drone and image processing area, where each area will be carried out in each semester. In conclusion, an autonomous drone system integrated with image processing can make a huge impact in the field of agriculture, which can change this industry to a more efficient and time-saving industry in terms of calculating the production yield.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
API	Application Programming Interface
FFB	Fresh Fruit Bunches
GPS	Global Positioning System
GPU	Graphics Processing Unit
GUI	Graphic User Interface
HLL	High Level Language
LIDAR	Light Detection and Ranging
PA	Precision Agriculture
R-CNN	Region-based Convolutional Neural Networks
RGB	Red Green Blue
SAR	Synthetic Aperture Radar
SSD	Single Shot Detector
UAV	Unmanned Aerial Vehicle
WAP	Wireless Access Point
YOLO	You Only Look Once

CHAPTER 1
INTRODUCTION

1.1 Problem Statement and Motivation

Quantification of the amount of FFB by the traditional way of manually counting or using ground surveying to gather information of the location is an intensive task. It is almost impossible to obtain information accurately with these methods in a large oil palm plantation. In addition, the traditional process of manually counting is susceptible to inaccurate estimation, time consuming and expensive.

Furthermore, spending hours of human observation in rough weather conditions such as heavy rain, monsoon wind or even in the unpleasant hot sun, this will be an additional problem for the small-scale farmers. This will reduce their efficiency in conducting their daily routines and hence, leading to inaccurate information of the number of FFB to be harvested.

Moreover, there could be some elderly farmers who are no longer flexible in moving around the large oil palm plantation and jotting down the stock of the FFB in the area. As a result, they might need to spend a sum of money in hiring someone else in order to fulfil that task. Therefore, when topography is uneven and coverage is large, these traditional methods that are created mostly based on visual observation are often inaccurate.

Our motivation in this project is to be able to produce an autonomous drone system integrated with computer vision in order to lighten the workload of farmers around the country. We want to be able to improve the existing system of counting oil palm fruits where the UAVs used are slow, expensive and capture low resolution images which leads to the lack of accuracy in counting oil palm production yield. This system is meant to help farmers to enhance efficiency in counting oil palm FFB as they do not have to manually count the fruits ever again. It helps the farmers to save time as they can proceed to harvest the ripe oil palm fruits instead of spending hours in the sun trying to manually count the oil palm FFB.

1.2 Project Objectives

The objective of this final year project is to replace the traditional way of manually counting FFB with the automated way of counting oil palm tree FFB. We want to overcome the existing problems where manually counting is time consuming and inefficient. Time consuming is always an important issue that determines the success of a framework. With the combination of image processing and UAV technology, quantification of oil palm tree FFB can be taken within minutes.

The objective in this project is to prove that, with image processing techniques, an automated remote sensing platform through UAV can be realized into different types of plantation to ease the workload of the farmers. This process is not only efficient, but it also brings convenience to all farmers throughout the country. It has always been a tedious task to manually count oil palm tree FFB but now, every record is digitised to reduce the work of storing data into the computer, which indirectly further improve the efficiency in carrying out these tasks on a daily basis.

1.3 Project Scope

In this paper, we are interested in developing an autonomous remote sensing platform by integrating UAV and a machine-vision system for quantification of oil palm tree fruit bunches. This will be an improvement plan of the traditional way of counting oil palm FFB. Farmers are often faced with many problems such as extreme weather conditions or movement inflexibility. Therefore, to overcome this problem for farmers throughout the country, the autonomous remote sensing platform by integrating UAV and a machine-vision system is introduced. In order to realize this project, a few methods are proposed which includes image pre-processing, feature extraction and training the ANN.

1.4 Expected Contributions from the Project

At the end of this project, farmers will be able to observe the quantification of oil palm FFB through a Python GUI on a PC by integrating the use of UAVs. They will no longer have to spend hours under the hot sun to manually count the oil palm FFB ever again. This is an efficient and effective approach in facilitating farmers in oil palm plantations to minimize these tedious tasks on a daily basis.

By using an autonomous drone system integrated with image processing, a virtual helping hand is lent to these farmers in order to save time and improve efficiency to produce better results. Therefore, it can potentially benefit farmers around the country in reducing their workload and enhancing their lifestyle in terms of computer vision and unmanned aerial vehicles.

This oil palm detection and counting system will be able to detect oil palm FFB in the images captured by the Parrot AR. Drone 2.0. It will then show the oil palm production yield to the user as requested.

1.5 Organisation of the Report

The details of this final year project are shown in the subsequent chapters. Firstly, Chapter 1 mentions the problem statement, motivation, objectives and scope of this project. In Chapter 2, the technologies and existing systems are reviewed. Then, the system development model, hardware and software system requirements, functional requirements, system testing and performance, expected challenges, the project milestone and estimated cost are listed out in Chapter 3. And then, Chapter 4 describes the system architecture, functional modules, system flow and the GUI design. In Chapter 5, the hardware setup, software setup, setting and configuration as well as the system operation are listed out. Next, Chapter 6 showed the system testing and performance metrics, testing setup and result, project challenges and objectives evaluation. Finally, Chapter 7 reports the conclusion and recommendation that can be used to further enhance this project in the future.

CHAPTER 2
LITERATURE REVIEW

2.1 Review of the Technologies

In this chapter, an overall literature review on hardware platforms, programming language and the current existing systems are introduced. A short summary on the review of the technology and current existing systems are also listed out in a table.

2.1.1 Hardware Platform

UAVs have shown capability of collecting data through agricultural remote sensing. UAVs are typically low airspeed, light weight and low-cost aerial vehicles that are suitable for obtaining information. An aircraft that is capable of performing automated flight operations without the handling of a human pilot and is equipped with communications systems, automatic control, sensors and necessary data processing unit are defined as an UAV or drone by Cai et al. (2011).

UAVs have multiple benefits. They can acquire high-resolution images, flexible in terms of timing of missions and altitude, safer than piloted aircrafts, less expensive, and can be deployed repeatedly and quickly. These images allow for monitoring of distinct patterns, gaps, patches and plants at some of the sites where formerly have been impossible (Franklin et al. 2006). A solar-powered UAV is used to obtain multi-spectral and high-spatial resolution images by numerous researchers who are developing the UAV based agricultural remote sensing systems (Herwitz et al. 2004). The implementation of wireless technology has made it easier to download images and remotely control the operation of a camera in -real time. An up-looking quantum sensor and five downwards-looking digital camera were equipped on the UAV and used by Hunt, Walthall and Daughtry (2005). To obtain the visible and near-infrared images, the lens' filters were altered. The possibility of observing the oil palm fruit at an early stage were displayed on the system. Besides, an aerial vehicle which could be remotely piloted for observing agricultural and natural resources was developed by Mark and Hardin (2005). The system was capable of obtaining high-resolution images with a GPS showing the exact location of the image.

Quantification of FFB and mature fruits from UAV stream images is the first stage towards implementing PA in oil palm plantations. In smaller scale oil palm plantations, it is possible to compute oil palm production yield with all the accessible high-tech imaging sensors as well as using remote sensing techniques and real-time image processing. These techniques can be pixel-based or object-based (Blaschke, Feizizadeh & Hölbling 2014), template matching (Ahuja & Tuli 2013), image analysis, learning algorithms methods for classification (Kalantar et al. 2017) and analysing an image for useful information. The suitability of the various sensors available for the use of UAVs in PA were studied by Maes et al. (2019) where important perspectives could be provided. Although sensors can be changed accordingly for each application, it can still be costly to most of the small-scale farmers.

Advantages of using automated UAV is the lowered cost with each operation flight and their reasonable price that makes it appropriate in plantation monitoring applications for academic research. The likelihood of collecting high-resolution images from aerial vehicles at different angles is being assessed through this plan for quantification of FFB and mature fruits as well as fly inside and over oil palm plantations.

Therefore, satellite and UAV-based remote sensing have been used by professionals (Srestasathiern & Rakwatin 2014) for applications such as vegetation cover assessment (Breckenridge et al. 2006), vegetation mapping (Kalantar et al. 2017), crop monitoring (Jensen 2016) and forest fire applications (Ambrosia et al. 2003). Thus, drone technology (Xiongkui et al. 2017) and agricultural robotics (Shamshiri et al. 2018) have made a huge difference in the accuracy and speed of sending out crucial information. Digital agriculture (Shamshiri 2017) offers many chances for automated farming tasks inside oil palm plantations by surveillance of air or ground and software that processes data to predict or estimate oil palm yields.

UAV can be constructed on an unmanned vehicle armed with several sensors, using GPS positioning technologies and communication technologies to acquire high-resolution images of the FFB. Models based on remote sensing retrieval are then used after processing the data (Sugiura et al. 2005). The different types of UAVs are multi-rotors, flying wing, helicopters, blimps, fixed-wings (Table 1) and are chosen depending on the objective as well as budget.

Specification	Description				
	Multi-rotor	Helicopter	Fixed-wing	Blimps	Flying wing
					
Model	DJIS1000+	AXH-E230	Bat-3	CB3000	Pathfinder-Plus
Manufacturer	DJI technology	AVIX	MLB Co.	Beijing CSCA Co.	AeroVironment
Materials	Carbon fiber, High strength performance engineered plastics	Carbon fiber, aluminum alloy	Carbon fiber, engineered plastics	Kevlar fibers, fiber optic, electrical cores	Carbon fiber, Nomex, Kevlar, plastic sheeting, plastic foam
Cost	Low	Medium	Medium	High	very high
Power/Motors	Eight electric, 0.5 kw max each	One BLDC motors	Two-stroke engine	One oil engine	Eight (8) solar-electric, 1.5 kW max each
Gross weight/kg[a]	6	15	56	300	318
Payload capacity/kg[b]	7	15	9	10	67.5
Speed/m s^{-1}	12	23	33	15	14
Endurance/h[c]	0.25	0.8	6	12	15
Altitude ceiling/m	500[d]	3,000	3,000	120	25,000

Table 2-1-1-1 Typical types of UAV used

Unmanned helicopters also have the ability to land vertically and take-off, hover and fly sideways. The payload of an unmanned helicopter is bigger than the payload of a multi-rotor UAV where big sensors can be supported, like LIDAR. Nevertheless, the procedure being too complex, high maintenance cost, lack of hovering and loud noise are some of the limitations of unmanned helicopters (Sugiura et al. 2005). The fixed-wing UAV is considered to have high flying speed and extended flight time but it has a limitation for this application. This device lacks the ability to hover and high altitudes and velocities can cause image distortion. (Herwitz et al. 2004).

Multi-rotor UAVs have the ability to hover and have low cost, low landing and take-off requirements. Still, the biggest disadvantages of multi-rotor UAVs are the comparatively lower payload, sensitivity to weather and the short flight time (Zhang & Kovacs 2012). Old-style UAV uses metal materials for the body like aluminium and steel (Molina & Colomina 2014). To reduce the weight of the UAV, prolong flight time and body strength can be further enhanced. Hence, an assortment of strong composite and lightweight resources have been broadly involved and have become an alternative for the main supplies for the body of UAVs.

UAV engines can be separated into two main classes which are electric and oil engines. Oil engines are known for their long working time and strong wind resistance. They also are known for being bulky, having poor reliability and producing big vibration which could cause serious image distortion (Tian & Xiang 2011). On the other hand, electric engines have the pros of low cost, low maintenance, having small vibrations and safe, which makes it an alternative to quantification of FFB for UAVs. However, the weak wind resistance and short flight endurance time limited its use in an oil palm plantation at large scale. Propulsion systems that are silent and low-altitude are important in meeting the expectations of different sizes of UAVs, especially medium and small-sized ones (Verhoeven 2009).

Fluorescence sensors, infrared thermal sensors, spectral sensors, visible light imaging sensors and LIDAR equipped in UAV platforms can acquire the texture and colour, which then is used to observe the different growth stages of the oil palm fruit (Zhang & Kovacs 2012). Since the UAV's payload capacity has its limitation, equipped sensors should meet the requirements of small size, low power consumption, light weight and high precision. UAV payload and advancement of commercial products such as SAR, three-dimensional camera, LIDAR, hyperspectral camera, infrared thermal imager, multispectral camera and digital camera (RGB) are some of the main sensors which the UAV equipped considering the cost (Chapman et al. 2014).

UAV is most commonly equipped with digital cameras, which can quickly obtain colour or grayscale images for quantification of FFB in yield estimation (Ballesteros et al. 2014). RGB camera is most frequently used by UAV as it has the benefits of low working environment requirements, requires simple data processing, convenient operation, light weight and is low cost. Even beneath both cloudy and sunny environment, data can still be collected but exposure should be set according to the climate environment to keep away from excessive or inadequate exposure on images. However, this technique is inadequate to precisely count oil palm fruits because of the restriction in the light bands’ visibility.

Spectral imaging sensors equipped by UAV can acquire spectral reflectance and absorption characteristics of fruits, which is used to monitor the growth of the crops and to foretell the production yield (Overgaard et al. 2010). UAVs are also commonly equipped with hyperspectral and multispectral imaging sensors. These sensors especially multispectral imaging sensors have the capability of recording and sensing radiations from visible and invisible parts of a spectrum, because of their ability of being high efficiency at work, fast frame imaging and low cost. Unfortunately, their discontinuous spectrum, low spectral resolution and low number of bands and are some of the limitations (Berni et al. 2009). Numerous continuous spectra and narrow bands can be obtained by hyperspectral imaging sensors. Hyperspectral imagers have high spectral resolution and contains higher band information as compared to multispectral imagers and spectral characteristics and differences of the oil palm fruit in the field can be precisely replicated (Zarco-Tejada et al. 2012).

Ground information can be effectively and swiftly obtained by using UAV remote sensing platforms, particularly for FFB monitoring. Flight speed and flight altitude should be constantly adjusted depending on weather conditions and could be caused by the farmland environment being too complicated. On the other hand, flight altitude and speed should be lowered to a certain height to acquire thorough data on the quantification of FFB (Mathews & Jensen, 2013).

Nowadays, most UAVs have driving systems that are automatic. GPS is used to carry out the adjustment of altitude, position and height as well as mount a pressure gauge to the UAV to avoid the impact of human factors on flight safety as well as decrease the strength of controlling manually (Pajares 2015). For many fixed-wing and multi-rotor UAVs, payload can reach up to 3 and 5 kg. In addition, it is important to have an ejection frame equipped to the UAV for take-off at the specific site as well as for opening the parachute decided by an artificial judgment, which could complicate the fixed-wing UAVs' operation. Besides, the body of the UAV has to be bigger when a maximum weight of 5 kg payload is surpassed.

Overall, multi-rotor UAVs are better in terms of convenience and stability for quantification of FFB and mature fruits for yield estimation (Yang et al. 2017). However, autonomy and speed are part of the limitation for rotary wings. As the battery technology develops with time, multi-rotor UAVs will have the ability to travel continuously with a time limit greater than one hour in the future.

The range for wavebands starting from near-infrared to visible light for optical sensors are currently equipped on the UAVs, such as digital cameras, hyperspectral sensors and multispectral sensors (Yang et al. 2017). However, sensors deployed by UAV might have the disadvantages of the difficulty to acquire quantitative information as they are usually employed for qualitative analysis. Furthermore, digital cameras commonly used by UAVs lack calibration which could lead to inaccuracy of parameter analysis.

Vibration in the UAV platform can cause noticeable distortion in the absence of a few systems. A shock absorber or a stabilization platform which is constantly active can be used to reduce the vibrations produced. However, it is difficult to calibrate accurately, where the application for this kind of sensor can be extremely affected. Thus, it is essential for sensors to be installed to the UAV for the purpose of quantification of FFB and mature fruits for yield estimation.

In a nutshell, UAVs deploy sensors with the intention to conduct the operation in a more convenient and flexible way, gaining access to high spatial resolution and data as a crucial way to acquire information on the quantification of FFB. However, because the limitation of a single sensor is in the area of remote sensing information, multiple sensors can be combined to acquire and integrate data by using UAV. In addition, as the quality of the image can be influenced by many kinds of factors, exploring strategies for acquiring high quality images are necessary for data processing and object detection.

2.1.2 Programming Language

The Python programming language is excellent at integrated tasks. It is commonly used as a free, open source and high-level language which is exceptionally interpreted, dynamic, multiparadigm and scripting. Besides, Python can support object-oriented programming features and is often used as a general-purpose programming language. This programming language is much simpler to learn and has an easier syntax as compared to Java, C and C++ programming languages.

In addition, Python is equally well-known for applications that are desktop-based. It also has different and wide-ranging support for Python libraries such as pillow, pip, matplotlib, PyLab, Networkx, NumPy, and many more libraries as well. Some of the fields where Python really stands out are machine learning, data science, symbolic, and numeric computations. Moreover, it is used in other famous fields like image processing, website development, games and big data analytics. Python is also implemented by big companies like Google, YouTube, Walt Disney, NASA as well as other companies.

Image processing integrated with Python is a very effective and efficient process for carrying out tasks such as examining the digitization of images for extraction of required information. Many tasks such as enhancing, blurring, zooming, inverting the image, improving the quality, applying text on the images, converting to greyscale, recovering and performing image restoration is possible with this programming language. In a study, different operations have been performed on a set of images in Python with the usage of libraries and other functions for the user to have an easier understanding towards the concepts of image processing and Python. This is useful for resolving the real-world problems in a very effective and efficient manner. (Gujar et al. 2016)

2.1.3 Summary of Technologies Review

Current Existing Technologies	Advantages	Disadvantages	Critical Comments
Fixed-wing UAVs	1) Fast flying speed 2) Long flight time	1) Lack of hovering ability 2) Image distortion caused by higher velocities and altitudes 3) Requires a minimum flight speed before they stall 4) It has a much more complex operation, which makes it riskier 5) Data acquisition is limited in quantification of oil palm fruits	This approach is used best for long distance tasks, such as surveillance and mapping. However, it has a complex operation, which could affect the process of these tasks.

Multi-rotor UAVs	1) Ability to hover 2) Low cost 3) Low take-off and landing requirements	1) Lower payload 2) Short flight time 3) Sensitive to weather	This approach is used best in aerial photography work for a short period of time on a small-scale plantation. However, their endurance and speed are limited, which makes them not suitable for aerial mapping on a large scale.
UAV with oil engines	1) Long working time 2) Strong wind resistance	1) Bulky 2) Poor reliability 3) Producing big vibrations which constantly lead to image distortion	This approach is used best in aerial photography work for a long period of time. However, images captured might be distorted due to the vibrations produced by the engine.

UAV with electric engines	1) Low maintenance 2) Produce small vibrations 3) Low cost 4) Safe	1) Weak wind resistance 2) Short flight endurance time limited its use in an oil palm plantation at large scale	This approach is used best in aerial photography work for a short period of time on a small-scale plantation. However, their limited endurance time makes them not suitable for aerial mapping on a large scale.
UAVs equipped with fluorescence sensors, infrared thermal sensors, spectral sensors, visible light imaging sensors and LIDAR	1) Early observation of different growth stages of the oil palm fruit	1) High cost 2) Difficult to acquire quantitative information as they are employed for qualitative analysis 3) Difficult to calibrate accurately	This approach is used best in obtaining accurate information with visible details. However, they are expensive and is difficult to calibrate.

UAVs equipped with digital cameras	1) Low cost 2) Convenient operation 3) Light weight 4) Simple data processing 5) Low working environment requirements	1) Inaccurately count oil palm fruits because the visible light bands have a limitation 2) Lack adjustment which affects the accuracy of parameter analysis	This approach is used best in simple aerial photography. However, if the images captured are needed for image processing, inaccurate information might be obtained.
UAVs with spectral imaging sensors	1) Acquire spectral reflectance and absorption characteristics of fruits to monitor crop growth 2) Low cost 3) High work efficiency	1) Low spectral resolution 2) Low number of bands 3) Discontinuous spectrum	This approach is used best if a high efficiency UAV is needed for aerial photography work. However, some information may be lost due to its discontinuous spectrum.

Table 2-1-3-1 Summary of Technologies Review

2.2 Review of the Existing Systems/Applications

Understanding the oil palm detection and counting algorithm clearly before beginning a progress is essential for developing an accurate oil palm system. Therefore, a review on current existing systems is carried out in order to identify the areas which are under-performing and make further improvements on these areas.

2.2.1 Usage of Logistic Regression Model for Verification of Oil Palm Detection System (Rueda et al. 2016)

In this existing system, unmanned aerial vehicles are used to capture images and then a logistic regression model will be used to categorize them. Firstly, the acquired images will go through the process of photogrammetry software to create orthomosaics. After that, these images will undergo a developed computer vision algorithm in order to be analysed. In order to generate candidates in image pyramids, a logistic regression model will be used to classify the windows and a sliding window technique, which is a form of visual descriptor is used to accurately replicate the image texture.

In the final stage, a non-maximum suppression algorithm is applied in order to better enhance the final decision. Different images were used to validate the system as compared to other images which has been through the training process. The reason to carry out this process is to allow the determination of how each and every parameter could affect the behaviour of the system. In a nutshell, it is possible to sum up that the size of the sliding windows and median filter are some of the most appropriate constraints to enhance the performance of the system. This study focuses on a logistic regression model which is used to verify an oil palm counting and detection system for general UAVs. However, this model does need to depend on the proper presentation of data in order to run.

2.2.2 Oil Palm Age Estimation and Counting Using LiDAR Data and WorldView-3 Imagery with Regression Analysis and Integrated OBIA Height Model (Rizeei et al. 2018)

In this study, object-based image analysis was integrated in a support vector machine algorithm or otherwise known as SVM was applied for the counting of oil palm. Four various SVM kernel types with its own segmentation parameters were used to test the sensitivity in order to gain an ideal coverage of crown delineation. Crown extraction of a tree is integrated with multi-regression methods and height model to precisely make a rough estimation on the age of the trees. The multi-regression model was used to attain the most ideal model for oil palm age estimation through the different sizes of multi-kernel. At the same time, five various oil palm plantations were used in order to train these models.

The relationship between the height of a tree and the age of a tree was significant in supporting the model with a kernel size for young oil palm trees under the polynomial regression function, while on the other hand, older trees such as 22 years old and above are more suited for the exponential regression function. Generally, machine learning and remote sensing practices are beneficial in detecting and monitoring oil palm plantation in order to gain the maximum production yield. To sum it all up, Worldview-3 satellite and LIDAR airborne imagery as well as an all-up-to-date method for oil palm counting and age estimation is useful in this study. However, it is not practical in most applications as it simplifies the modern world problems by assuming the existence of a linear relationship among the data.

2.2.3 Automated Detection and Identification of Oil Palm from Normalized Cross Correlation and Multi-Scale Clustering (Wong-in et al. 2015)

This proposed method is able to solve the problem of oil palm identification from UAV images when the distance between the oil palms is too near together which could lead them to be identified as a single fruit. This process of normalized cross correlation and multi-scale clustering consists of distinguishing oil palms from other objects, eliminating non-tree components from the image, counting the amount of oil palm FFB and lastly, detecting each distinct oil palm.

Normalized cross correlation and an ideal low-pass filter can be used to identify and separate oil palm fruits from other objects in this study. Then, the proposed method of using erosion and multi-scale clustering is used to distinguish each discrete oil palm fruit from a tree. This technique was assessed by having a digital camera attached to the remote aircraft going through the oil palm plantations in different areas of Thailand and obtaining 21 sets of images from the aircraft. To conclude it all, this new proposed method is used to extract information from the aerial images in order to identify and detect oil palms regardless of their sizes using more distinct characteristics like texture, size and shape. However, it has a high computational cost in delivering this high-speed information especially when radio frequency signals are present.

2.2.4 Summary of Existing Systems

Existing System	Advantages	Disadvantages	Critical Comments
Usage of Logistic Regression Model for Verification of Oil Palm Detection System (Rueda et al. 2016)	1.Scaling of input features is not required 2. Too many computational resources are not required 3.Easy to implement and efficient to train	1. High reliance on proper presentation of data 2. Overfitting can occur where a random error is described instead of the relationship between variables	This approach is good when the system only has 2 classes. However, errors can occur when there are more classes because it can only separate the input into 2 regions by a linear boundary. The data must also be ensured that it is linearly separable.
Oil Palm Age Estimation and Counting Using LiDAR Data and WorldView-3 Imagery with Regression Analysis and Integrated OBIA Height Model (Rizeei et al. 2018)	1.Large coverage of oil palm plantation area 2.Modeling speed is fast	1. Requires linear data in order to run 2. Assumes a linear relationship between variables, which could lead to inaccuracy in oil palm counting	This approach is used best when the data obtained has a linear relationship. However, it is not practical in most applications as it simplifies the modern world problems by assuming the existence of a linear relationship among the data.

Automatic Oil Palm Detection and Identification from Multi-scale Clustering and Normalized Cross Correlation (Wong-in et al. 2015)	1.High speed algorithm 2.Better performance 3.Scale to large datasets	1.High computational cost	This approach is used best in obtaining high quality, high spatial resolution images. It also has a high speed in delivering information. However, it has a high computational cost especially when radio frequency signals are used in delivering high-speed information.

Table 2-2-4-1 Summary of Existing Systems

2.3 Concluding Remark

A few of the hardware platform and programming language platform are discussed and studied in this chapter. A summary of the technologies review is then briefly described in Section 2.1.3. In addition, three existing oil palms counting and detection systems have been discussed in this chapter as well. Effort on the study of oil palm counting systems definitely play a significant role in obtaining the most accurate results. However, it is also worthy to consider other object detection algorithms and oil palm detection systems that could produce a better result with their own roles and configurations.

CHAPTER 3
SYSTEM METHODOLOGY

3.1 System Development Models

There are four main system development models which are the waterfall model, prototyping model, iterative enhancement model and the spiral model.

3.1.1 Waterfall Model

The Waterfall model was the first system development model to be introduced and it is very easy to use and understand. In a Waterfall model, overlapping in the phases is not allowed and each phase must be finalized before the next phase can commence. This model is the earliest system development approach that was used for the development in software.

The whole process of software development in the Waterfall model is divided into distinct phases. The output of one phase acts as the input for the next phase consecutively. This means that any phase in the development process begins only if the previous phase is complete. The waterfall model is a chronological process in which progress is seen as a waterfall steadily flowing downwards through the phases of requirements, design, implementation, verification and maintenance.

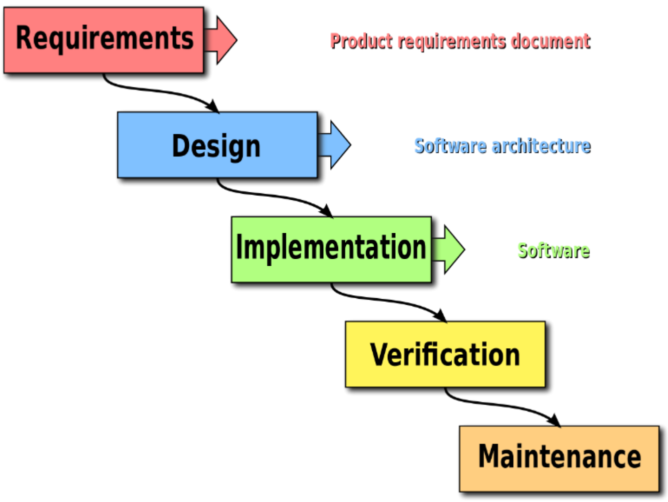


Figure 3-1-1-1 Waterfall Model

3.1.2 Prototyping Model

The prototyping model is a system development method where a prototype is assembled, tested and then revised as necessary until a satisfactory outcome is achieved from which the complete product or system can be developed. This model works the best in situations where project requirements are not known beforehand. It is an iterative as well as a trial-and-error process that takes place between users and developers.

In this system development model, the system is implemented partially before the analysis phase therefore giving the users an opportunity to see the early development of the product. The process begins with enquiring requirements from the customers and developing the partially finished paper model. This piece of document is used only to build the initial prototype of the system, supporting the most basic functionalities as desired and chosen by the customer. The prototype is then further refined to eliminate the problems once the customer clearly figures them out. This process continues until the customer has completely approved the prototype and finds the model to be of satisfactory level.

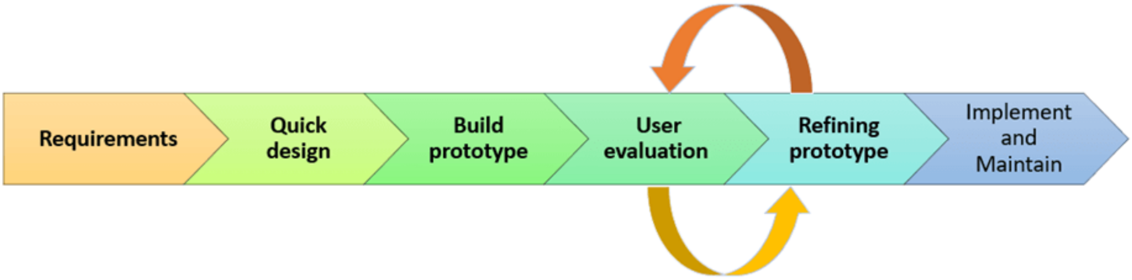


Figure 3-1-1-2 Prototyping Model

3.1.3 Iterative Enhancement Model

An iterative enhancement model or else known as incremental model consists the features of a waterfall model but in an iterative manner. In the implementation phase, the project is split into small subsystems referred to as increments that are individually implemented. This model includes several phases in which each phase generates an increment. At the very beginning of the development process, these increments are identified and for each increment, the whole process from the beginning of requirements gathering to the final phase of the delivery of the product will be implemented.

The basic concept of this model is to begin the process with requirements and consecutively improve the requirements until the final software has been executed. This method comes in handy as the software development procedure is simplified and the implementation of increments by increments makes it much easier than implementing the whole system itself.

Every stage of this model will be adding some new functionality to the product and handing it onto the next phase. The first increment is commonly referred to as a core product and is usually used by the user for a thorough evaluation. Such a process results in the formation of a plan for the subsequent increment. This plan determines the adjustments to the product made accordingly to the user's needs. The incremental process, which also contains the distribution of the increments to the user, will continue to execute until the software is fully developed.

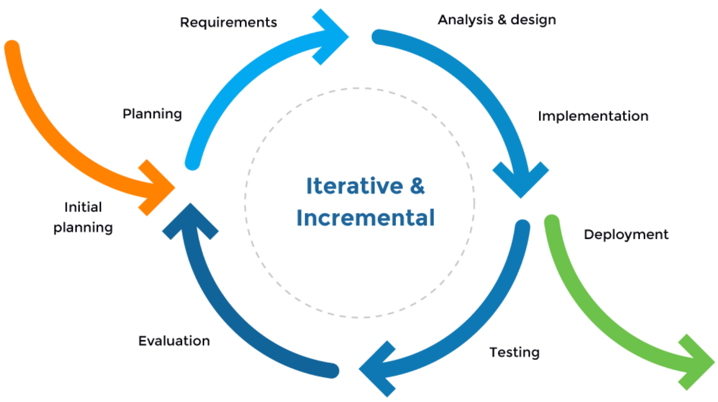


Figure 3-1-1-3 Iterative Enhancement Model

3.1.4 Spiral Model

The spiral model is one of the most significant system development models, which offers support for risk handling. In its representation, it does seem like a spiral with many twists. The precise number of loops of the spiral remains unidentified and it may differ depending on the project. Each single loop of the spiral is referred to as a phase of the system development process. The precise number of phases required to develop a project may vary accordingly depending on the project risks. The radius of the spiral at any single point symbolizes the cost of the project and the angular dimension symbolizes the development made until now in the current phase.

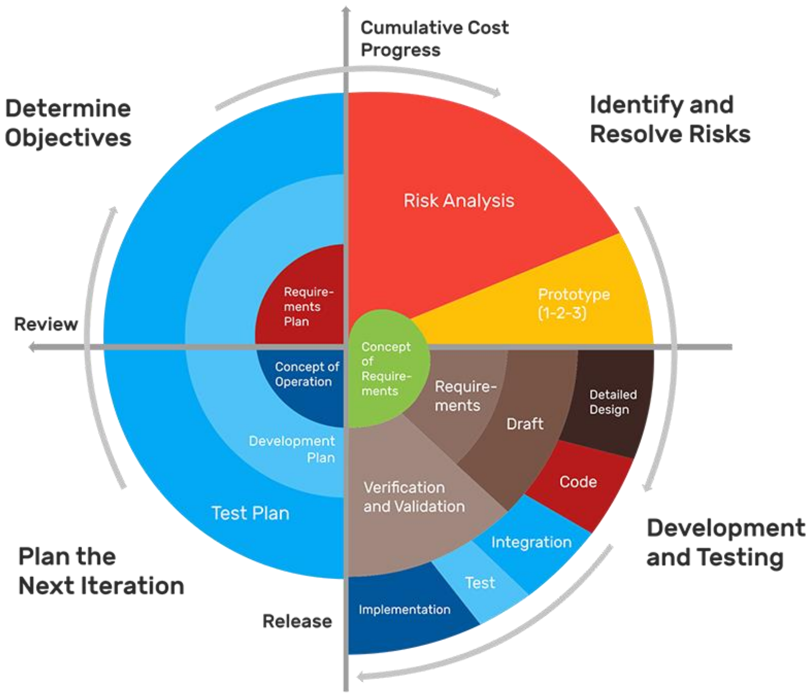


Figure 3-1-1-4 Spiral Model

3.1.5 Selected Model

The model that we have chosen for my project is the iterative enhancement model. This model fits my project scope in the image processing area especially during the training of the ANN. The requirement of this project is to be able to train the network to detect oil palm FFB with the highest accuracy. At the beginning of this project, we used a very small dataset to train the network. When we run a few sample images through the network, it was found out that the system has a poor detection towards the oil palm FFB in the images. Because of this, we went back to the start of the training process and train the network with a bigger dataset this time. We noticed that there was a better detection on the oil palm FFB this time. Consecutively, we repeatedly trained the network with a bigger dataset every time for the system to reach the highest accuracy. This training process matches the idea of the iterative enhancement model where the requirement is iteratively enhanced until the final software is implemented and all the requirements are fulfilled.

3.2 System Requirement

System requirements are a description of the system's services, functions and operational constraints. The hardware and software system requirements are listed out as below.

3.2.1 Hardware

Parrot AR. Drone 2.0

Figure 3-2-1-1 Parrot AR. Drone 2.0

The Parrot AR. Drone 2.0 is a quadcopter that is developed by Parrot SA Company. The AR. Drone is operated with an internal built-in 468 MHz ARM9-processor. This drone also consists of 128MB RAM. AR and can run on a Linux operating system. In addition, a mini-USN connector is available for the attachment of software and for other add-ons such as GPS sensor. Furthermore, the AR. Drone has a mounted integrated wireless card inside to provide network connection.

Frontal and vertical mounted cameras are some of the on-board electronic devices used for the purpose of obstacle detection. This UAV relies deeply on various types of sensors to fly such as a 3-axis gyroscope 2000° per second precision capabilities, 3-axis accelerometer with +/-50 mg precision, a 3-axis magnetometer with precision up to 6°, a 60 FPS vertical armed QGVA sensor type camera for measure ground speed, a pressure sensor with +/- 10 Pa precision, and lastly, an ultrasound sensor for measurement of ground altitude. All of these sensors are installed and used during flights for stabilization and cameras onboard are used to provide visual feedback to the user from the drone.

Laptop

Figure 3-2-1-2 Laptop

Laptop is used in developing and implementing Python code as well as object detection algorithms for the quantification of oil palm FFB. It is also used to establish laptop Wi-Fi connection to the Parrot AR. Drone 2.0.

3.2.2 Software

Python version 3.7.7

Figure 3-2-2-1 Python Software

Python is an object-oriented and high-level programming language. Python is simple and has an easy to learn syntax which highlights readability and thus the cost of software maintenance is reduced. Python also supports packages and modules, which encourages the modularity of the program and re-usage of code.

The minimum hardware system requirements for Python version 3.7.7 is a x86 64-bit CPU (Intel/AMD architecture), 4GB RAM and 5GB free disk space. The minimum requirements for operating systems are Windows 7 or 10, 64-bit Mac OS X 10.11 or higher and 64-bit Linux: RHEL 6/7.

<u>TensorFlow version 1.15.2</u>

Figure 3-2-2-2 TensorFlow Software

TensorFlow is an open-source platform designed for machine learning. It has an inclusive, flexible environment of libraries, tools and resources that lets students and researchers push the advancement in machine learning as well as easily build and deploy machine learning applications.

The TensorFlow 1.15.2 version is supported on Python 2.7, 3.4, 3.5, 3.6 and 3.7. The minimum hardware requirements for CPU-enabled version of TensorFlow is same as the hardware requirements needed for Python. However, if GPU-enabled version of TensorFlow is used, the hardware system requirements are different. The requirements are 64-bit Linux operating system, Python 2.7, CUDA 7.5 GPU (CUDA 8.0 required for Pascal GPUs) and cuDNN v5.1 or cuDNN v6 if the TensorFlow version is 1.3.

One limitation is that TensorFlow is not supported on Python 32-bit systems. It is only supported on Python 64-bit systems. Another requirement to consider is the amount of memory needed for TensorFlow. If the version of TensorFlow used is GPU-enabled, the amount of GPU memory is taken into consideration and if the version of TensorFlow used is CPU-enabled, the amount of RAM is taken into consideration. As long as the graph and all of its constants, variables and data can fit into the memory space, there should be no problems.

LabelImg

Figure 3-2-2-3 LabelImg Software

LabelImg is an image annotation tool which provides bounding boxes for images after labelling. It is written in Python and uses Qt for its interface. The output data are then saved as XML files in PASCAL VOC, which is the format used by ImageNet or it could also be saved as XML files in YOLO format.

LabelImg is also supported on Windows, macOS and Linux operating systems. Besides, this software is also supported on Anaconda, which is an open-source and free distribution of the programming languages, R and Python. It is also supported on Docker, which is a tool to make it much easier to create and run different applications by using containers.

3.3 Functional Requirement

Hardware Interface:

In this project, on-board frontal and vertical cameras are used for gaining visual feedback from the Parrot AR. Drone 2.0. In addition, pressure sensors will be depended heavily on in order to keep the drone steady. On-board pressure sensors are used to perform any flight adjustment accordingly and sustain a constant position up in the air regardless of wind and altitude.

Software Interface:

In this project, the Python libraries to perform the oil palm counting and detection are studied and implemented. It is implemented by performing image pre-processing, feature extraction and training the ANN through the images captured by the Parrot AR. Drone 2.0.

3.4 Expected System Testing and Performance

The expected system performance for this project is targeted at the accuracy of counting oil palm fruits. Many image processing systems tend to have low accuracy on object detection. Our project aims to produce results with high accuracy in detecting oil palm fruits. This will allow the farmers to carefully plan out their production and earnings by the quantity of the oil palm fruits. A system with high accuracy can help improve productivity and enhance efficiency in agricultural fields as well as reduce the workload of the farmers.

The expected system testing is to primarily test the autonomous drone by running a piece of Python script which connects to the drone through a WAP. To verify this, we will run the program in the campus compound and observe whether if the drone is able to fly around and capture images on its own without a remote control. If the drone is able to perform all of this using only a piece of code, this verification test is considered successful. On the other hand, to test the accuracy of detecting oil palm FFB, we will be using sample images from the Internet to test it out. If these sample images can achieve high accuracy upon using the image processing program, this test will also be considered successful. To determine high accuracy, we can compare the output count value from the system with the actual value from what we observe on the image itself.

3.5 Expected Challenges

The expected challenge in this project is to train the network to be of highest accuracy. In order to train a network to the highest accuracy, we will need a huge dataset and also take other criteria into consideration. Some of the criteria may include training the network to be able to detect other objects. In our case where the location is in an oil palm plantation, we will need to train the network to recognise trees, branches, leaves and other objects that may be present in an oil palm plantation. This is to prevent the system from mistakenly detecting other objects as oil palm FFB. If this were to happen, the system could indirectly lead to the inaccuracy of counting oil palm FFB, which defeats the whole purpose of our project.

3.6 Project Milestone

Task	Project Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Review Project Proposal														
Literature Review on Technology														
Study on Existing Systems														
Outline System Design														
System Prototyping														
System Testing 1														
System Debugging														
System Testing 2														
Design Graphic User Interface														
Presentation														
Project Documentation														

Table 3-6-1 Project Milestone for FYP1

Task	Project Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Review FYP1 Report														
Discussion with supervisor														
Collect enough dataset to train the network														
Improve accuracy of system														
System Testing 1														
Improve efficiency of system														
System Testing 2														
Improve Graphic User Interface														
Presentation														
Project Documentation														

Table 3-6-2 Project Milestone for FYP2

3.7 Estimated Cost

Items	For FYP Development	For Commercialisation
Parrot AR. Drone 2.0	Free (provided in lab)	RM 1,558.00
1.500 mAh LiPo battery	Free (provided in lab)	RM 99.00
Python license	Free (open-source license)	Free (open-source license)
TensorFlow (under Apache License 2.0)	Free	Free (allows for commercial use)
LabelImg (under MIT license)	Free	Free (allows for commercial use)

Table 3-7-1 Estimated Cost for this Project

3.8 Concluding Remark

In a nutshell, analysis on system methodology is carried out. Different system development models are studied in Section 3.1 and a brief explanation on each system development model is outlined. A thorough explanation on the selected system development model for this project is also included as it informs the user on the importance of choosing the right model. In Section 3.2, the software and hardware system requirements are also stated out. Functional requirements in Section 3.3 allows the user to have an understanding of the system functionality and behaviour.

In section 3.4, the expected system testing and performance is stated out. The expected challenges of this project are also listed out in section 3.5. Project milestones in section 3.6 are also illustrated in a diagram to help readers to understand the time allocation for this project. Last but not least, the estimated cost for this project is outlined as well in section 3.7. This allows the readers to have a glimpse on the estimated cost for the purpose of final year project as well as for the purpose of commercialization.

CHAPTER 4
SYSTEM DESIGN

4.1 System Architecture

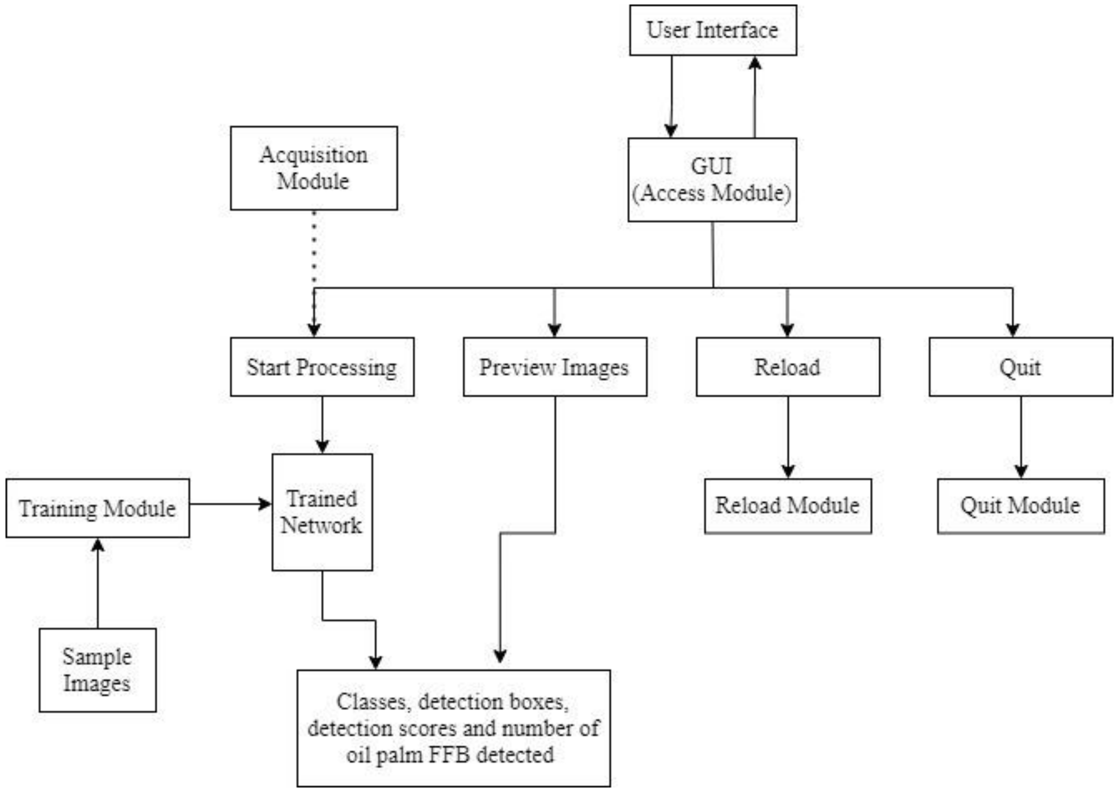


Figure 4-1-1 System Architecture

In this system, the input to the GUI will be the user interface while the output to the GUI depends on the user. The input to the 'Start Processing' option is the acquisition module as it provides the images to go through the trained network. If 'Start Processing' is chosen, the output will be the number of oil palm FFB detected on each image as well as the cumulative amount of oil palm FFB detected on all the images. If 'Preview Images' is chosen, the output to this module is the input images before detection and the output images after detection. If 'Reload' is chosen, the output will be the reload module or if 'Quit' is chosen, the output of this module will be the quit module.

At the same time, the input to the training module is the sample images while the output to this module is the classification and the counting of the oil palm FFB, which is also the final product as well.

4.2 Functional Modules in the System

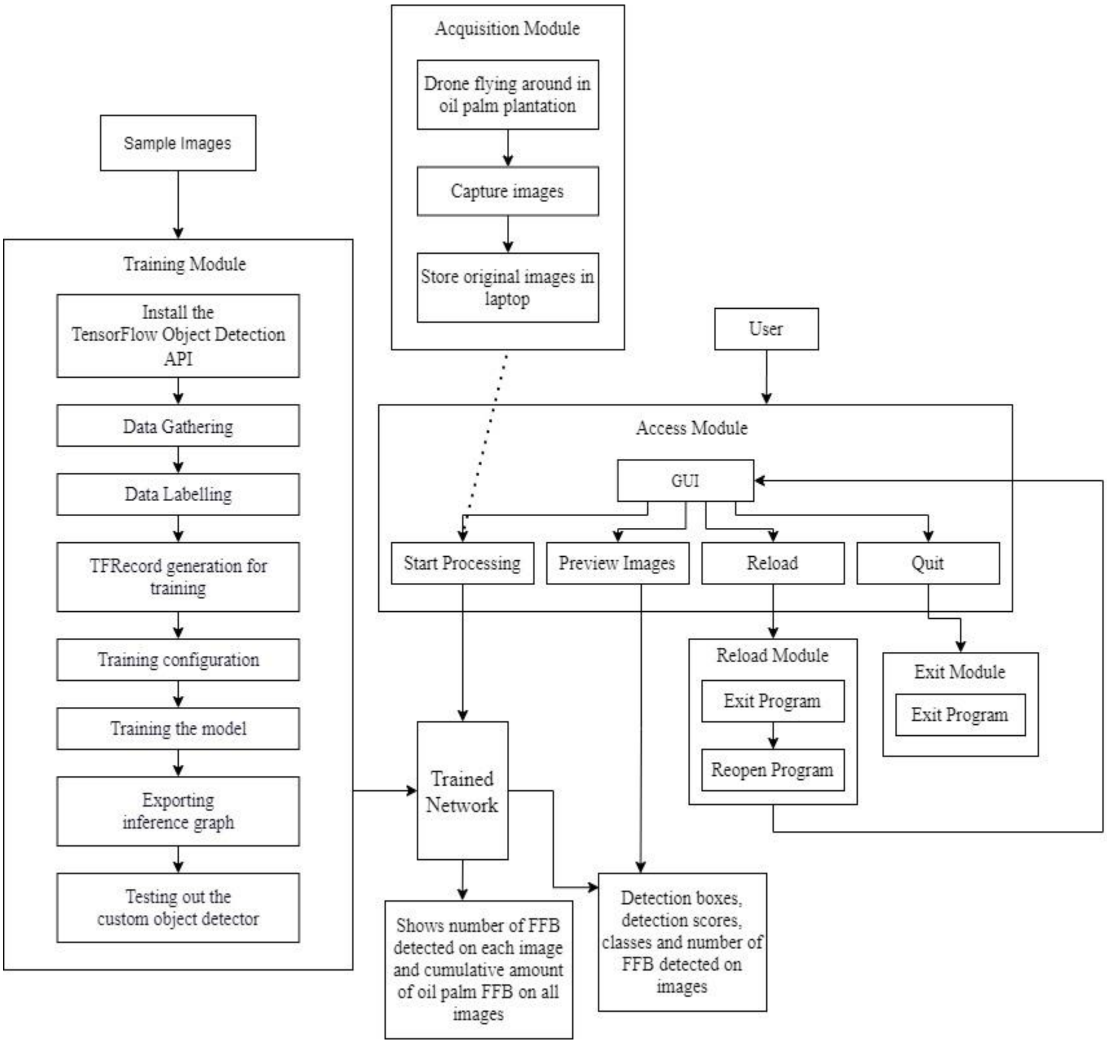


Figure 4-2-1 Functional Modules

In this project, there are 5 functional modules in the system. The first module is the acquisition module, where this module is in charge of acquiring images from the UAV and storing these images in our laptop for the use of quantification of oil palm FFB later.

The second module is the training module where a batch of 200 sample images is used to train the network in order to produce a custom oil palm FFB detector. These images will go through the process of data labelling, TFRecord generation, training configuration, model training and exporting the inference graph. After all of these steps, a trained network will be produced.

The third module is the access module where the user will be able to access the GUI through the Python software. The user is then able to choose from 4 options in the GUI which is the ‘Start Processing’, ‘Preview Images’, ‘Reload’ and ‘Quit’ buttons. The ‘Start Processing’ button will allow the user to observe on the console the number of FFB detected on each image and the cumulative amount of oil palm FFB detected on all images after undergoing the training module. The ‘Preview Images’ button will allow the user to observe the images before going through the ANN (before detection) and the images after going through the ANN (after detection).

The ‘Reload’ button will bring us to the fourth module in the system which is the reload module. This module will exit the program and then reopen back the program itself to allow the after-detection images to be updated into the program and to be shown to the user.

Lastly, the ‘Quit’ button will bring us to the final module in the system which is the quit module. This module allows the user to quit the program with ease.

4.3 System Flow

There are many steps in training the ANN as shown below in the design block diagram such as installing the object detection API, gathering and labelling data, generating TFRecord for training, configuring training, training the model, exporting inference graph and finally, testing out the custom object detector.

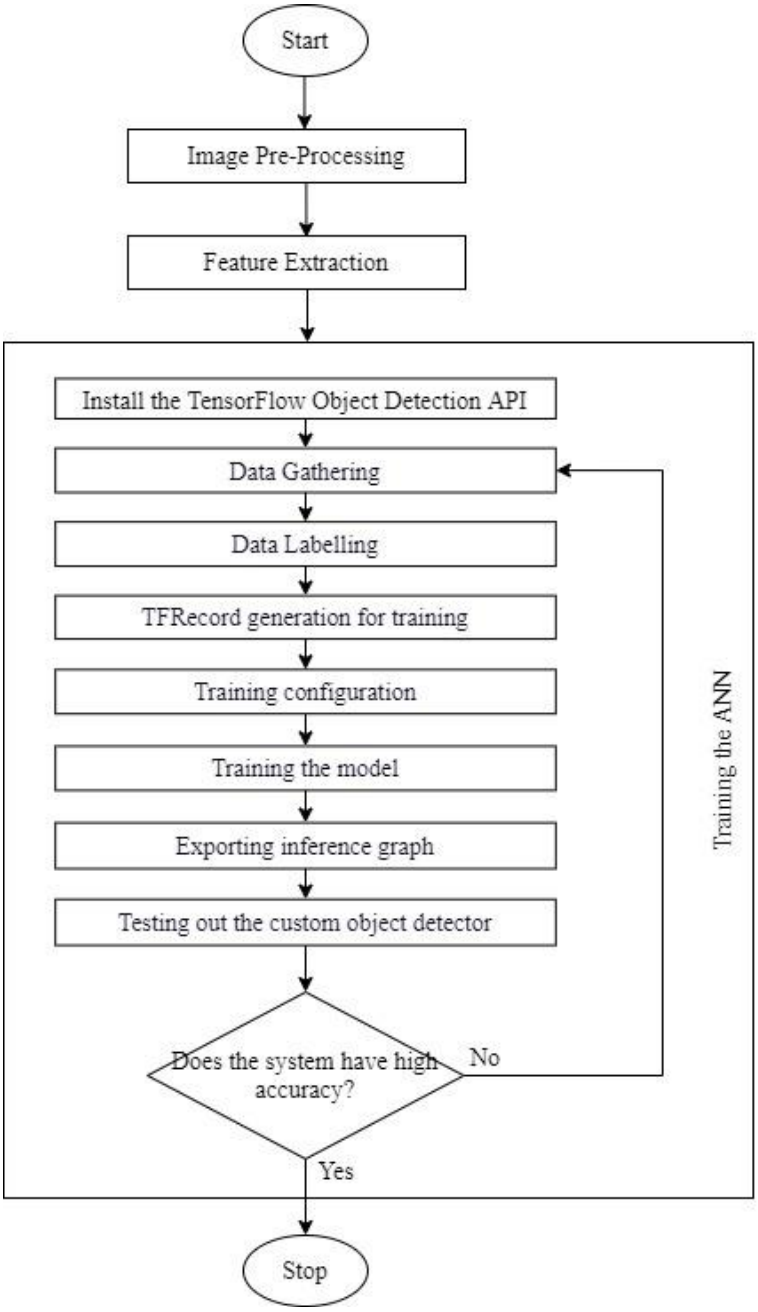


Figure 4-3-1 System flow of this project

First of all, we will have to install the TensorFlow Object Detection API. This object detection API can be defined as the framework for producing an Artificial Neural Network that resolves object detection problems, which in our case is to detect oil palm FFB. We installed this API by using Python Package Installer (pip) into a specified directory.

Before we begin to train our custom object detector, we will need to gather some data. We will need a lot of sample images which differ from each other in order to train a strong classifier. The more the sample images, the higher the accuracy of detecting oil palm fruits will be. These sample images should consist of different angles of oil palm FFB, diverse lighting conditions and varying backgrounds. We decided to download 35 images from the Internet to act as data for the training process later. However, these images are of different sizes and some may have high resolutions. Therefore, we want to change them to a consistent size and scale them so the training process will perform faster.

Before we start to label our images, we will have to move 80 percent of our images into the training directory and the 20 percent into our test directory for validation of the system later. Since now we have moved our sample images, we can start to label the images in the training directory. We will need a labelling software in order to label these data. LabelImg is the software that we will be using to label these images. Basically, using this software, we will create a bounding box for the object we want to detect, which in our case, will be the oil palm FFB. This process of labelling will be repeated for all our sample images.

Now that we have our images labelled, we will need to generate TFRecords which will serve as input data for the training process. TFRecords is a file for storing data as a sequence of binary strings. We will be using two Python scripts in order to successfully generate TFRecords. The first Python script will transform xml to csv files appropriately. After that, we can generate TFRecords by using command lines to run the second Python script to produce TFRecord files.

Then, we will proceed on to the training configuration step. We will need to create a label map which maps an id to a name. Our label map will only consist of 1 object, since we are only detecting oil palm FFB. Next, we will create a training configuration file. There are many models to choose from but we will be using the faster_rcnn_inception model. This model will then be copied into the training folder and a few lines will need to be altered in the configuration files. For example, we will need to change the number of objects that we want to detect, the input paths to the TFRecords files and also the path to the label map file.

Later, we will be training the model. The model will be trained until it reaches an acceptable loss value, where the lower the loss value, the stronger the classifier will be. We can then terminate the training process by pressing the key CTRL + C. The last step will be to export the inference graph. We will need to find out the highest step number from the training directory and look for the file with the maximum index, which will then be used to run the model.

Finally, we will be able to test out our custom object detector by using the images in our test directory. If these images can reach a high detection score, it means that our custom object detector has successfully been implemented and the validation of the system is successful as well.

4.4 GUI Design

Figure 4-4-1 GUI Design of oil palm production yield system

In this project, we have created a GUI for an easier approach to observe the quantification of oil palm FFB. This GUI shows the before detection images and the after-detection images for the user to easily observe the detection carried out on these images. The user will be able to observe the classes, detection boxes, detection scores and the number of oil palm fruits detected on the output images.

There are four buttons that we have created on the GUI which is the 'Start Processing', 'Preview Images', 'Reload' and 'Quit' buttons. When the 'Start Processing' button is pressed, the number of oil palm FFB detected on each image will be shown on the console as well as the cumulative amount of oil palm FFB detected on all the images. The program will also write the output images to a folder for the user to have easier access to these images and also for future reference. This is to ensure that they would not have to run the program again in order to observe the detection carried out on these images every time they need it.

In order to preview the images, we will need to click on the ‘Reload’ button. This allows the output images to be updated into the program. The ‘Reload’ button allows images to be easily updated into the program if any future images are captured by the drone or being processed in the program itself. The ‘Preview Images’ button will show the input images before going through the ANN and the output images after going through the ANN. This allows the user to easily compare between the input and output images as well as observe if the system has accurately detected the oil palm FFB. Lastly, the ‘Quit’ button allows the user to quit the program easily.

4.5 Concluding Remark

In this project, system architecture is illustrated and explained in Section 4.1 with a diagram to allow the user to clearly understand how the system works. Furthermore, function modules and their respective functions are also listed out in Section 4.2. A complete system flow chart is also attached in Section 4.3. This flow chart gives the user a clearer picture on the flow of the system. Finally, the GUI design figures are attached in Section 4.4.

CHAPTER 5
SYSTEM IMPLEMENTATION

5.1 Hardware Setup

<u>Laptop</u>

In this project, a laptop is used to train the neural network for the image classifier and object detection modules. This laptop is also used to retrieve the images captured from the drone, pass these images through the neural network, implement an oil palm detection and counting system as well as design the GUI in Python. The oil palm counting system is capable of displaying results to the user through the GUI by displaying the classes, detection boxes, detection scores and the number of oil palm fruits detected on the drone images.

<u>Parrot AR. Drone 2.0</u>

Parrot AR. Drone 2.0 is used to capture images in an oil palm plantation. Before we begin, check the items included in the package. There are a total of 8 items which includes:

Figure 5-1-1 Items in Parrot AR. Drone 2.0 package

A) 1 unit of Parrot AR. Drone 2.0 main body

B) 1 unit of indoor hull protector (for indoor flight)

C) 1 unit of outdoor hull protector (for outdoor flight)

D) 2 units of Lithium-Polymer batteries (1,000 mAh and 1,500 mAh)

E) 1 unit of battery AC charger with international adaptors

F) 1 unit of multilingual instruction manual

G) Decal stickers for drone

After checking if all of the items are complete, we can start to setup the drone. Below are the following instructions for the Parrot AR. Drone 2.0 setup:

1) Charge the battery as soon as you unbox the package as it takes 1.5 hours to charge completely
2) Take the drone out from the box
3) Put on the indoor hull protector in Figure 5-1-2 for the drone if you are flying indoors or the outdoor hull protector in Figure 5-1-3 if you are flying outdoors

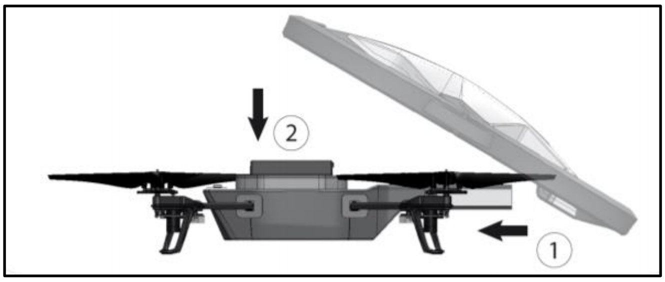


Figure 5-1-2 Indoor Hull Protector

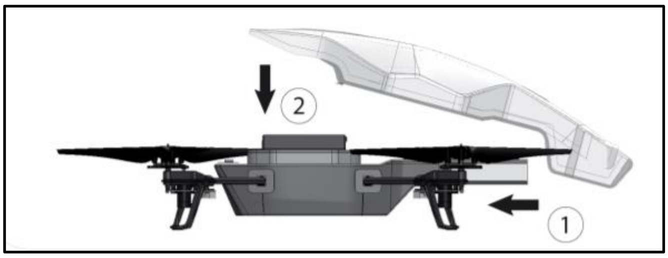


Figure 5-1-3 Outdoor Hull Protector

4) After the battery has finished charging, the LED indicator will turn green as shown in Figure 5-1-4

Figure 5-1-4 LED indicator turn green

5) Unplug the battery and insert the battery into the drone's battery compartment
6) Make sure the battery is correctly secured by using the mechanism as shown in Figure 5-1-5

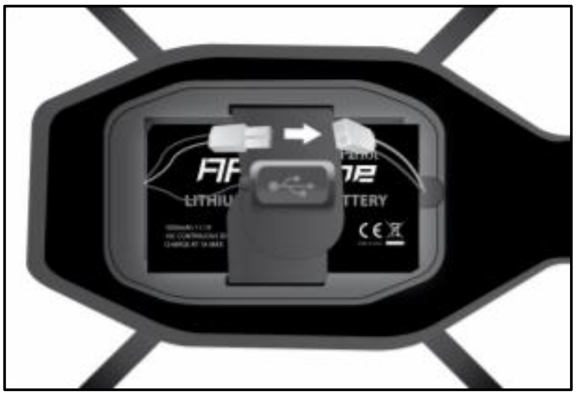


Figure 5-1-5 Attachment mechanism for the battery

7) After the battery has been secured correctly, the drone will start to operate by rotating its 4 motor blades consecutively in a few seconds
8) The LED indicators near the rotating blades will then turn green, indicating the drone is ready as shown in Figure 5-1-6

Figure 5-1-6 LED indicators turn green near rotating blades

9) Once the drone is ready, it will broadcast a Wi-Fi hotspot. User can perform a search on the list of Wi-Fi network and search for a network that begins with "ardrone2" as shown in Figure 5-1-7

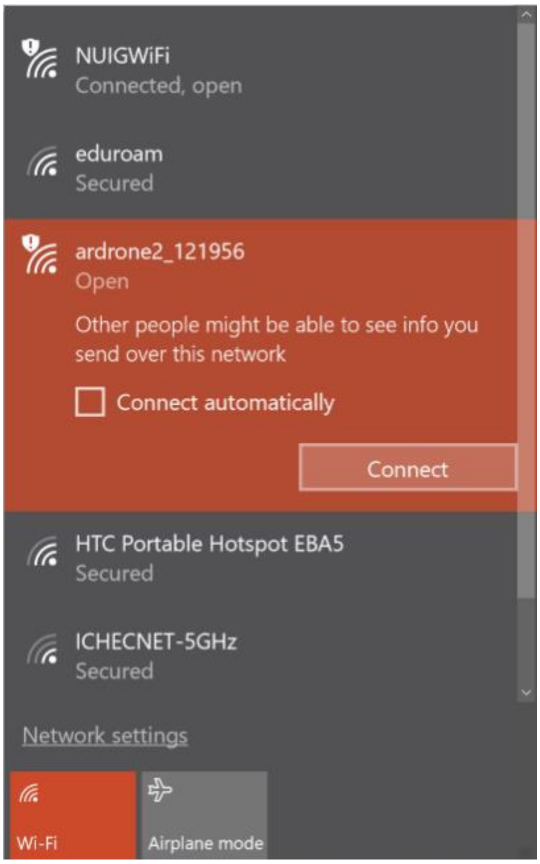


Figure 5-1-7 Parrot AR. Drone 2.0 Wi-Fi hotspot

10) When the drone is connected to the laptop, user can start to control the drone using an application to capture oil palm FFB images in an oil palm plantation

Note: If you would like to know more about the Parrot AR. Drone 2.0, the complete user guide can be found at: (https://www.bhphotovideo.com/lit_files/121124.pdf).

5.2 Software Setup

Python Installation

Before we can start the project, we need to download Python. The Python version that we used in this project is Python 3.7.3. The installation for Python is very direct as well as easy to install and the installation guide can be found here at: (https://www.python.org/downloads/). In order to run TensorFlow, we have to take note to only download Python 64-bit installer as 32-bit installers are currently not supported on TensorFlow. Below are the following instructions to run Python on your laptop:

1) Download the Python installer version of your choice (Requires version 3.6 – 3.8)
2) Open and unzip the installer
3) Run the installer as administrator on your laptop
4) Follow the steps as directed by the installer
5) You now have the Python program on your laptop

TensorFlow Installation

Now that we have downloaded Python, we can start to download TensorFlow. The TensorFlow version that we used in this project is TensorFlow 1.15.2. The installation guide can be found at: (https://www.tensorflow.org/install/pip#system-install).

There are 4 total mechanisms that we can install TensorFlow which is through "native" pip, Docker, virtualenv and Anaconda. In this project, we are using "native" pip to install TensorFlow. Below are the following instructions to install TensorFlow on your laptop:

1) Check for your python and pip version using the following commands:

```
> python3 --version
> pip3 --version
```

2) If your pip is not up-to-date, you can upgrade pip using the following command:

```
> python -m pip install --upgrade pip
```

3) After you have checked your pip and python version, you can start to install TensorFlow using the following command:

```
> pip3 install --user --upgrade tensorflow
```

*Please note that TensorFlow takes some time to install as it contains many files.

4) To check if Tensorflow is installed correctly, we can run a simple program in Python Shell to verify it.

```
import tensorflow as tf
print(tf.reduce_sum(tf.random.normal([1000, 1000])))
```

* If TensorFlow is installed successfully, a tensor will be returned in the above program.

Note: During the implementation of this project, TensorFlow 2 is now available and supported for Python 3.6-3.8. If you wish to install TensorFlow 2, you can find the installation guide at: (https://www.tensorflow.org/install).

TensorFlow Object Detection API

After downloading TensorFlow, we can now download the Object Detection API for TensorFlow. The installation guide to download the API can be found at: (https://gilberttanner.com/blog/live-object-detection). There are 2 ways to download the API, which are “native” pip and Docker. In this project, we are using “native” pip to download the API.

Note: The download for TensorFlow 2 Object Detection API is now available. The installation guide for this API can be found at: (https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html).

5.3 Setting and Configuration

In this project, there are two essential folders named tensorflow and tensorflow_object_counting_api. These folders contain necessary source codes to run the oil palm detection and counting system. In the tensorflow folder, it consists of the TensorFlow Object Detection API as well as all the models needed to train the neural network for the image classifier. On the other hand, in the tensorflow_object_counting_api folder, it consists of the source code used for counting the oil palm fruits detected in the output images. In order to set up all the necessary source codes to run the system, the instructions are stated as below:

1) Firstly, copy the ‘tensorflow’ folder into your preferred path (hard disk).
2) In the tensorflow/models/research/object_detection/ directory, make sure all the paths for xml_to_csv.py, generate_tfrecord.py, export_inference_graph.py and model_main.py corresponds to your preferred path.
3) After that, make sure the faster_rcnn_inception_v2_pets.config in object_detection/training/ directory corresponds to your path as well.
4) Now, proceed to copy the ‘tensorflow_object_counting_api’ folder into the same path as your ‘tensorflow’ folder.
5) Make sure all the image path, frozen_inference_graph.pb path and labelmap.pbtxt path is set correctly in the main.py file.
6) Now that all the paths have been set, the system can be executed by running the executor.py file.

5.4 System Operation

Once all of the settings and configurations from section 5.3 have been set up properly, the system should be executing smoothly. A Python GUI will then pop up, showing users the input images before and the output images after going through the neural network. The Python GUI is shown below in Figure 5-4-1.

Figure 5-4-1 Python GUI

The GUI consists of four buttons, which includes the 'Start Processing', 'Preview Images', 'Reload' and 'Quit' buttons.

When the 'Start Processing' button is selected as shown in Figure 5-4-2, the console will show the number of oil palm fruits detected on each image as well as the cumulative count of all the oil palm fruits detected in a folder. The console is shown below in Figure 5-4-3.

Figure 5-4-2 'Start Processing' button is selected

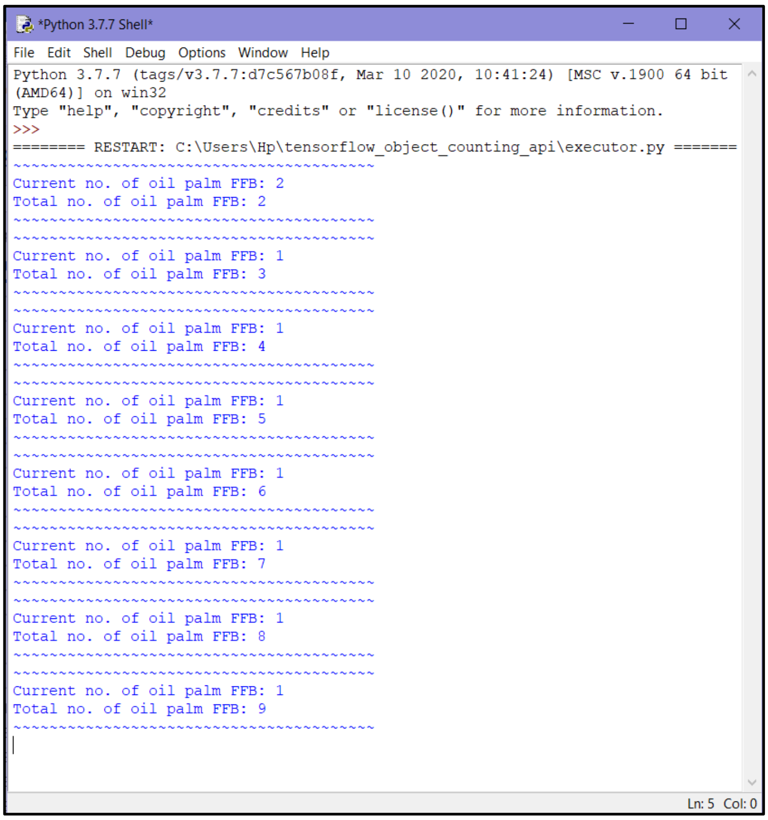


Figure 5-4-3 Console showing the number of oil palm FFB detected on each image as well as the cumulative count of all the oil palm FFB

Meanwhile, while the input images are being processed and shown to the console, the output images are being written to a folder. This can be seen in Figure 5-4-4, which shows the images after being passed through the neural network.

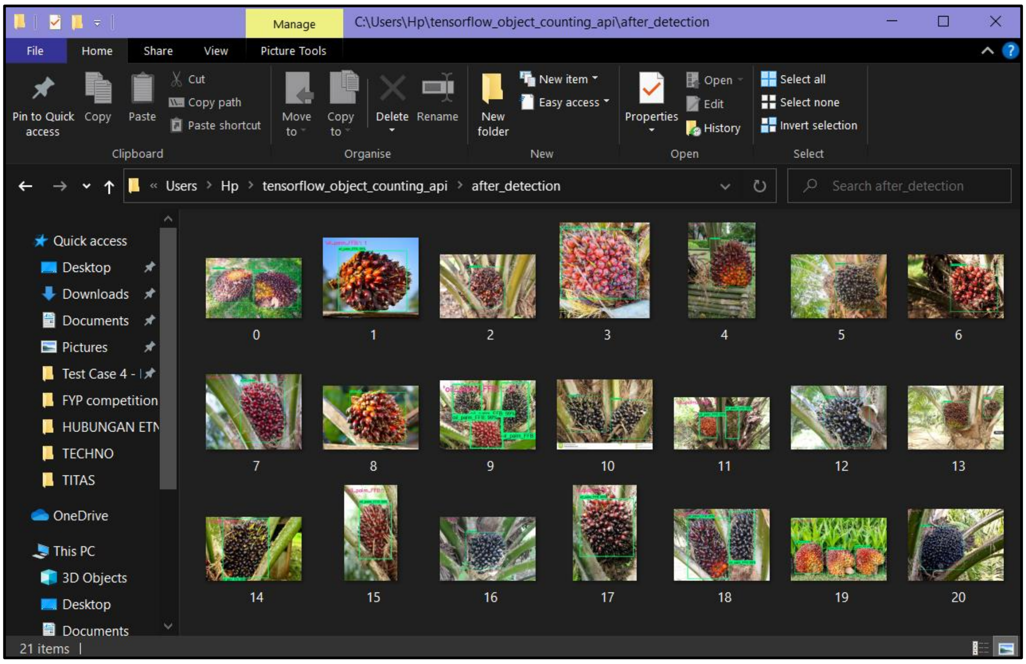


Figure 5-4-4 Output images written to a folder

On the other hand, when the 'Preview Images' button is selected, you can view the input images and the output images on the GUI. In short, it acts as a 'Next' button. This is seen in Figure 5-4-5, when the image changes from one to another on the GUI.

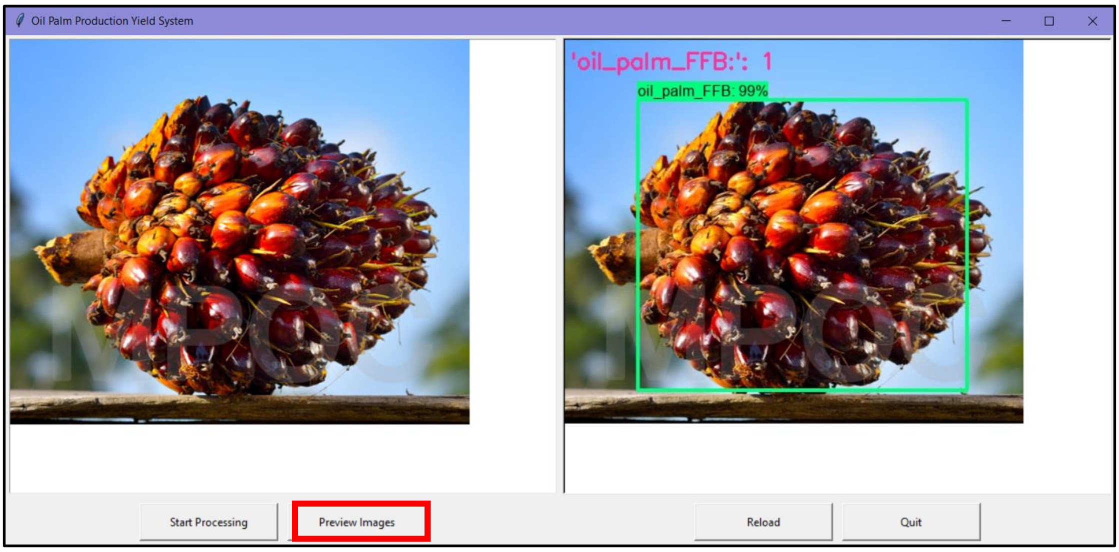


Figure 5-4-5 'Preview Images' button is selected

When the 'Reload' button is selected as shown in Figure 5-4-6, the new images that is captured and received from drone will be uploaded to the folder and then can be processed by clicking the 'Start Processing' button again. The GUI will disappear and reappear after a second when the 'Reload' button is clicked.

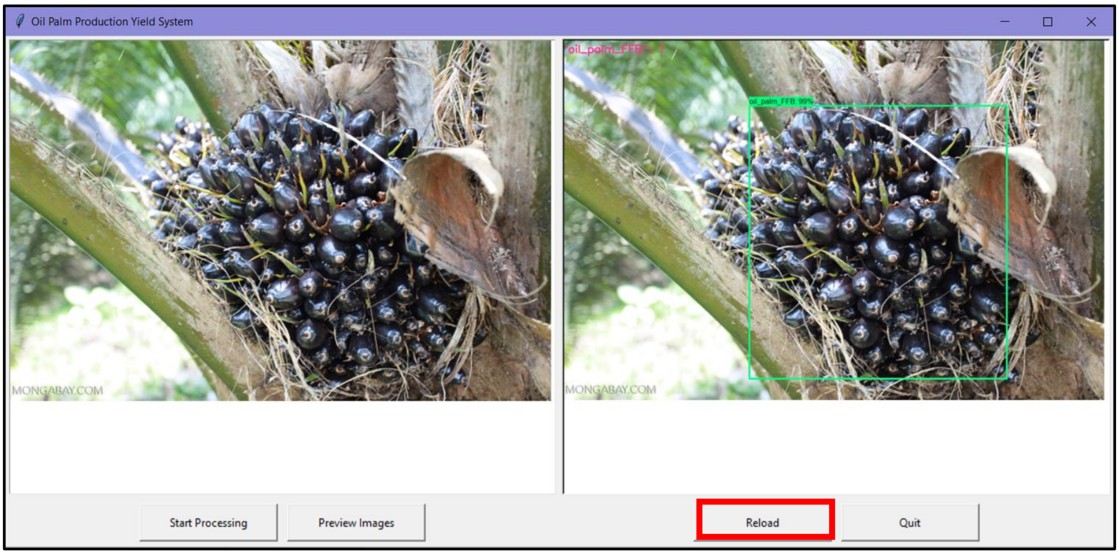


Figure 5-4-6 'Reload' button is selected

When the ‘Quit’ button is selected as shown in Figure 5-4-7, a message box will prompt for user to confirm if they really want to quit the program. This can be seen in Figure 5-4-8.

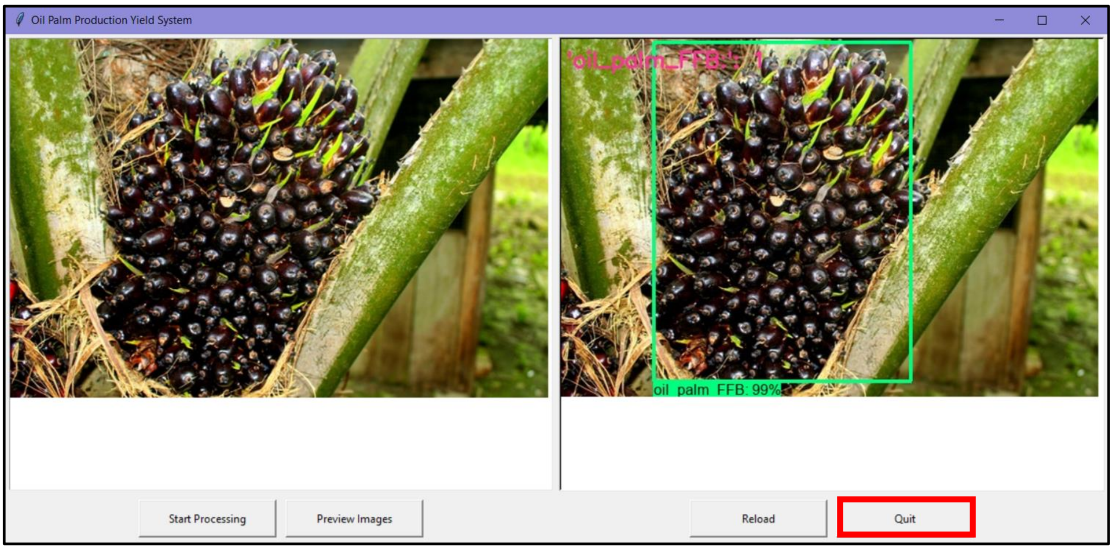


Figure 5-4-7 ‘Quit’ button is selected

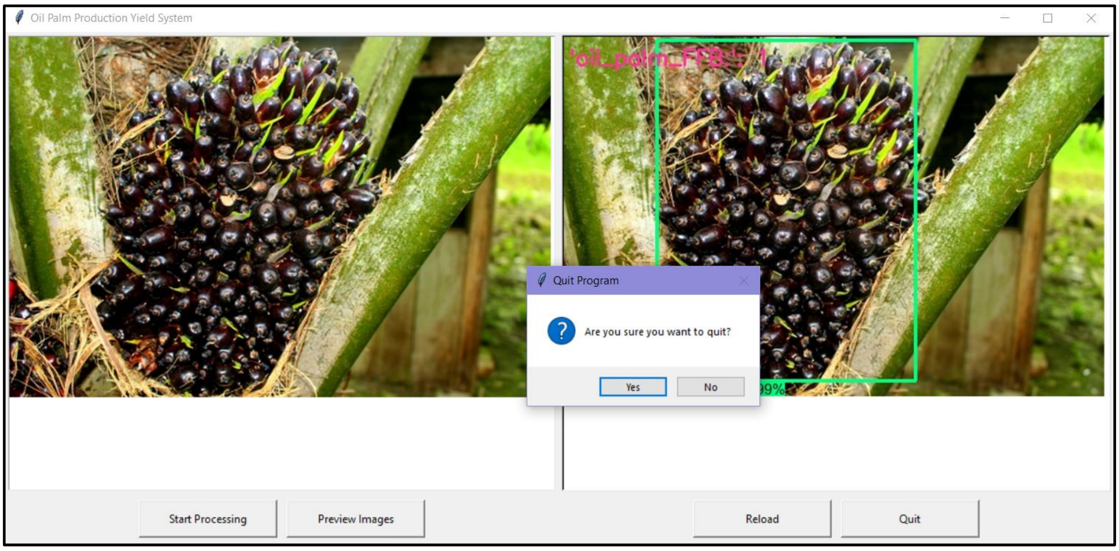


Figure 5-4-8 Message box prompt for user’s selection

5.5 Concluding Remark

In this project, the hardware setup is illustrated in Section 5.1 with diagrams to allow users to have a clearer understanding on the setup procedure. The software setup is listed out in Section 5.2 with commands for the software installation. In section 5.3, the setting and configuration to run the system is listed out. Finally, the system operation is shown in section 5.4 to allow users to understand how the system runs.

CHAPTER 6
SYSTEM EVALUATION AND DISCUSSION

6.1 System Testing and Performance Metrics

A sequence of testing is carried out to test the accuracy of the oil palm detection and counting system. In this project, a total of 8 test cases (conditions) will be carried out to assess the performance of the system. There are many conditions which could occur when processing images from a drone such as bad lighting, overlapping oil palm FFB, oil palm captured at different angles or even under normal conditions as well. To test this out, each of the condition will be carried out in 20 trails, summing them up to a total of 160 trails for all of the 8 conditions. This is to ensure that the system will be able to detect oil palm FFB with a high accuracy under any circumstances, be it normal or poor environmental conditions. The acceptance requirement for this system testing is that the number of correct results must achieve a minimum of 80% accuracy.

Condition 1	Without any restraint (same angle, no overlap, good lighting)
Condition 2	Bad Lighting on Oil Palm Fruits
Condition 3	Overlapping Oil Palm Fruits
Condition 4	Different Angles of Oil Palm Fruits
Condition 5	Bad Lighting & Overlapping of Oil Palm Fruits
Condition 6	Overlapping & Different Angles of Oil Palm Fruits
Condition 7	Bad Lighting & Different Angles of Oil Palm Fruits
Condition 8	Bad Lighting, Overlapping & Different Angles of Oil Palm Fruits

Table 6-1-1 Conditions for System Testing

6.2 Testing Setup and Result

The tables below show the number of correct classification results for each of the system testing conditions as stated under section 6.1 for the oil palm detection and counting system.

Test Case 1: Without any restraint (same angle, no overlap, good lighting)

Number of trials	Actual Count	System Count	Correct/Wrong
1	2	2	Correct
2	1	1	Correct
3	1	1	Correct
4	1	1	Correct
5	1	1	Correct
6	1	1	Correct
7	1	1	Correct
8	1	1	Correct
9	1	1	Correct
10	3	3	Correct
11	2	2	Correct
12	2	2	Correct
13	1	1	Correct
14	2	2	Correct
15	1	1	Correct
16	1	1	Correct
17	1	1	Correct
18	1	1	Correct
19	2	2	Correct
20	3	3	Correct

Table 6-2-1 Results for Test Case 1

Number of Trials: 20 Number of Correct Results: 20 Number of Wrong Results: 0 Success Rate: 20/20 * 100% = 100% Failure Rate: 0/20 * 100% = 0%

Figure 6-2-1 Sample Image in Test Case 1

Figure 6-2-1 shows a sample image that will be used in Test Case 1. The images in Test Case 1 are of the same angle, has no overlapping oil palm fruits and has good lighting obtained from the drone.

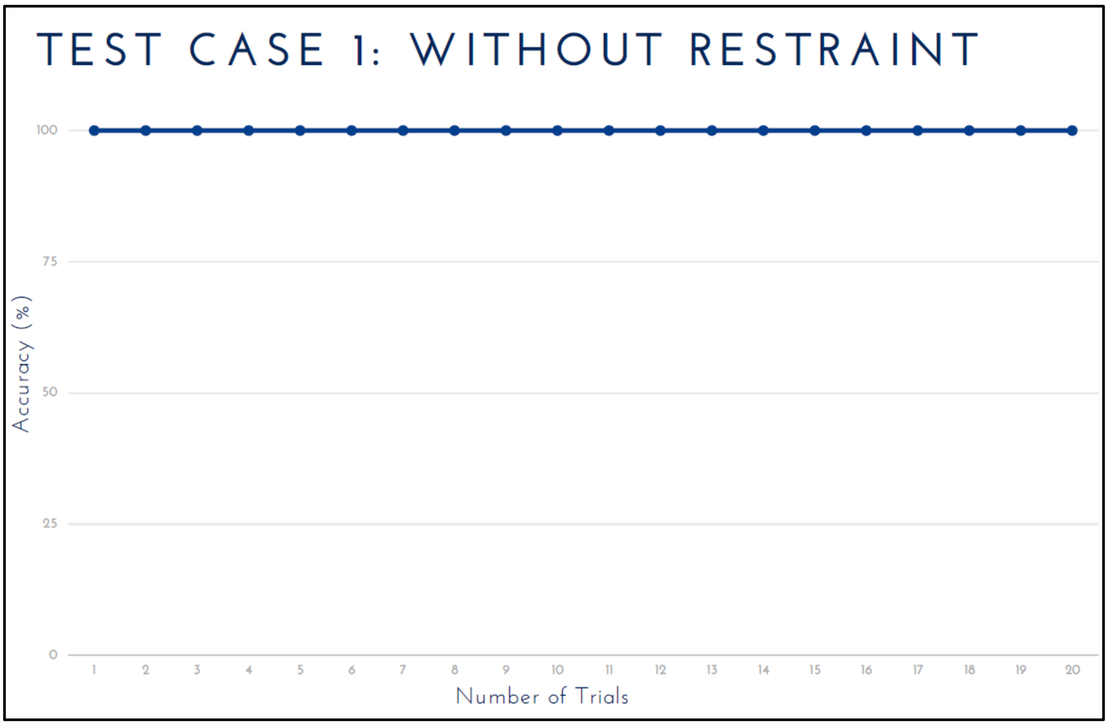


Figure 6-2-2 Accuracy rate for all trials for Test Case 1

Based on the results above, the outcome for Test Case 1 is considered as accurate as it produces 20 correct results out of 20 trials. The success rate for this test case is 100% while the failure rate is 0%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-2 shows the accuracy rate for all of the trials for Test Case 1.

Test Case 2: Bad Lighting on Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	1	1	Correct
2	2	2	Correct
3	3	3	Correct
4	1	1	Correct
5	1	1	Correct
6	1	1	Correct
7	1	1	Correct
8	1	1	Correct
9	2	2	Correct
10	2	2	Correct
11	3	3	Correct
12	2	2	Correct
13	3	3	Correct
14	1	1	Correct
15	1	1	Correct
16	1	1	Correct
17	2	2	Correct
18	1	1	Correct
19	2	2	Correct
20	2	2	Correct

Table 6-2-2 Results for Test Case 2

Number of Trials: 20

Number of Correct Results: 20

Number of Wrong Results: 0

Success Rate: 20/20 * 100% = 100%

Failure Rate: 0/20 * 100% = 0%

Figure 6-2-3 Sample Images in Test Case 2

Figure 6-2-3 shows sample images that will be used in Test Case 2. The images in Test Case 2 have bad lighting, either overexposed to light (80% of light exposure) or underexposed to light (20% of light exposure).

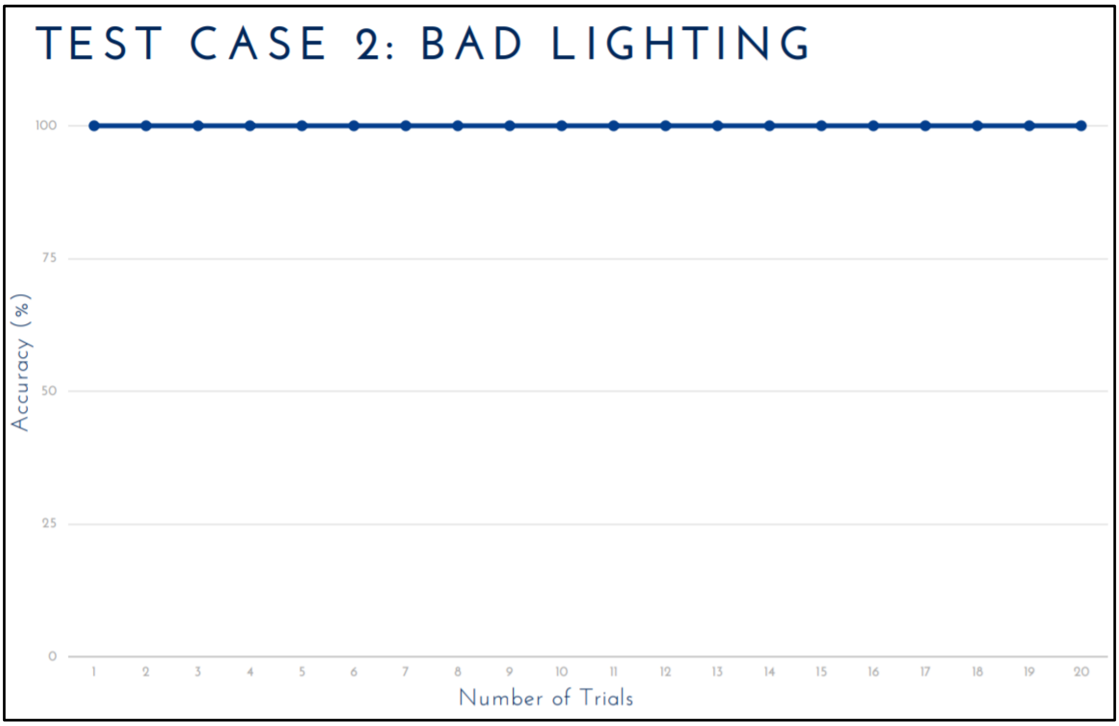


Figure 6-2-4 Accuracy rate for all trials for Test Case 2

Based on the results above, the outcome for Test Case 2 is considered as highly accurate as it did not have any wrong results. The success rate for this test case is 100% while the failure rate is 0%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are easily available. Figure 6-2-4 shows the accuracy rate for all of the trials for Test Case 2.

Test Case 3: Overlapping Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	5	5	Correct
2	6	6	Correct
3	3	3	Correct
4	4	4	Correct
5	3	3	Correct
6	4	4	Correct
7	5	5	Correct
8	4	5	Wrong
9	3	3	Correct
10	5	5	Correct
11	5	5	Correct
12	2	2	Correct
13	7	7	Correct
14	2	2	Correct
15	3	3	Correct
16	5	5	Correct
17	4	3	Wrong
18	5	5	Correct
19	5	4	Wrong
20	6	6	Correct

Table 6-2-3 Results for Test Case 3

Number of Trials: 20

Number of Correct Results: 17

Number of Wrong Results: 3

Success Rate: 17/20 * 100% = 85%

Failure Rate: 3/20 * 100% = 15%

Figure 6-2-5 Sample Image in Test Case 3

Figure 6-2-5 shows sample images that will be used in Test Case 3. The images in Test Case 3 consists of overlapping oil palm fruits (other oil palm fruits or leaves or branches obstructing the oil palm fruit itself).

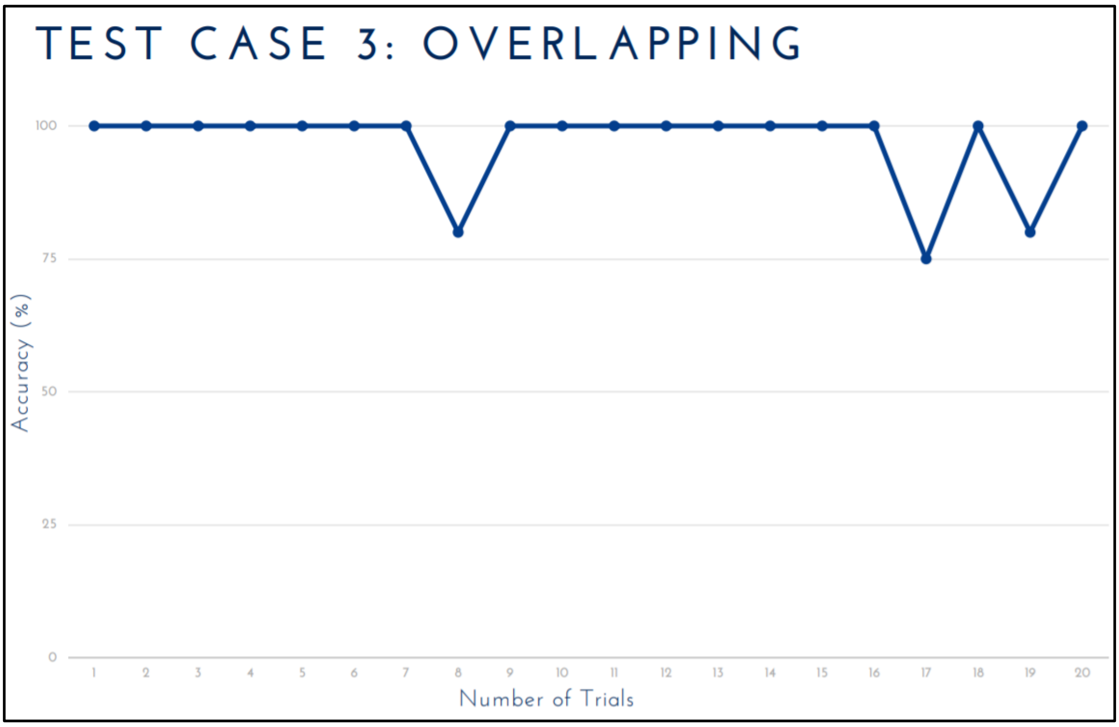


Figure 6-2-6 Accuracy rate for all trials for Test Case 3

Based on the results above, the outcome for Test Case 3 is considered as accurate as it produces 17 correct results out of 20 trials. The success rate for this test case is 85% while the failure rate is 15%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-6 shows the accuracy rate for all of the trials for Test Case 3.

Test Case 4: Different Angles of Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	3	3	Correct
2	1	1	Correct
3	1	1	Correct
4	1	1	Correct
5	3	2	Wrong
6	2	2	Correct
7	1	1	Correct
8	1	1	Correct
9	3	3	Correct
10	1	1	Correct
11	1	1	Correct
12	2	2	Correct
13	3	2	Wrong
14	2	2	Correct
15	1	1	Correct
16	1	1	Correct
17	2	2	Correct
18	1	1	Correct
19	1	2	Wrong
20	2	2	Correct

Table 6-2-4 Results for Test Case 4

Number of Trials: 20

Number of Correct Results: 17

Number of Wrong Results: 3

Success Rate: 17/20 * 100% = 85%

Failure Rate: 3/20 * 100% = 15%

Figure 6-2-7 Sample Image in Test Case 4

Figure 6-2-7 shows sample images that will be used in Test Case 4. The images in Test Case 4 consists of oil palm fruits captured at different angles by the drone (side view, bottom view or top view).

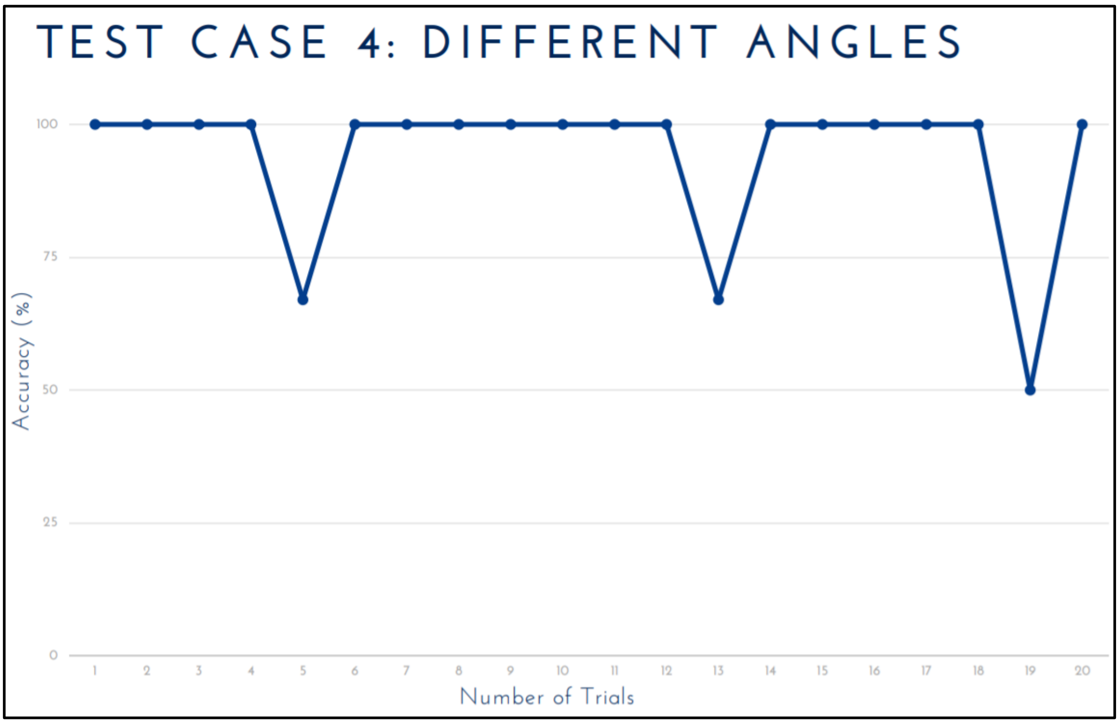


Figure 6-2-8 Accuracy rate for all trials for Test Case 4

Based on the results above, the outcome for Test Case 4 is considered as accurate as it produces 17 correct results out of 20 trials. The success rate for this test case is 85% while the failure rate is 15%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-8 shows the accuracy rate for all of the trials for Test Case 4.

Test Case 5: Bad Lighting and Overlapping of Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	5	5	Correct
2	2	2	Correct
3	5	5	Correct
4	8	7	Wrong
5	5	5	Correct
6	4	4	Correct
7	5	5	Correct
8	6	6	Correct
9	2	2	Correct
10	2	1	Wrong
11	7	7	Correct
12	5	5	Correct
13	3	3	Correct
14	4	4	Correct
15	2	2	Correct
16	6	6	Correct
17	5	5	Correct
18	2	2	Correct
19	5	5	Correct
20	3	3	Correct

Table 6-2-5 Results for Test Case 5

Number of Trials: 20

Number of Correct Results: 18

Number of Wrong Results: 2

Success Rate: 18/20 * 100% = 90%

Failure Rate: 2/20 * 100% = 10%

Figure 6-2-9 Sample Image in Test Case 5

Figure 6-2-9 shows sample images that will be used in Test Case 5. The images in Test Case 5 consists of overlapping oil palm fruits captured from the drone with bad lighting.

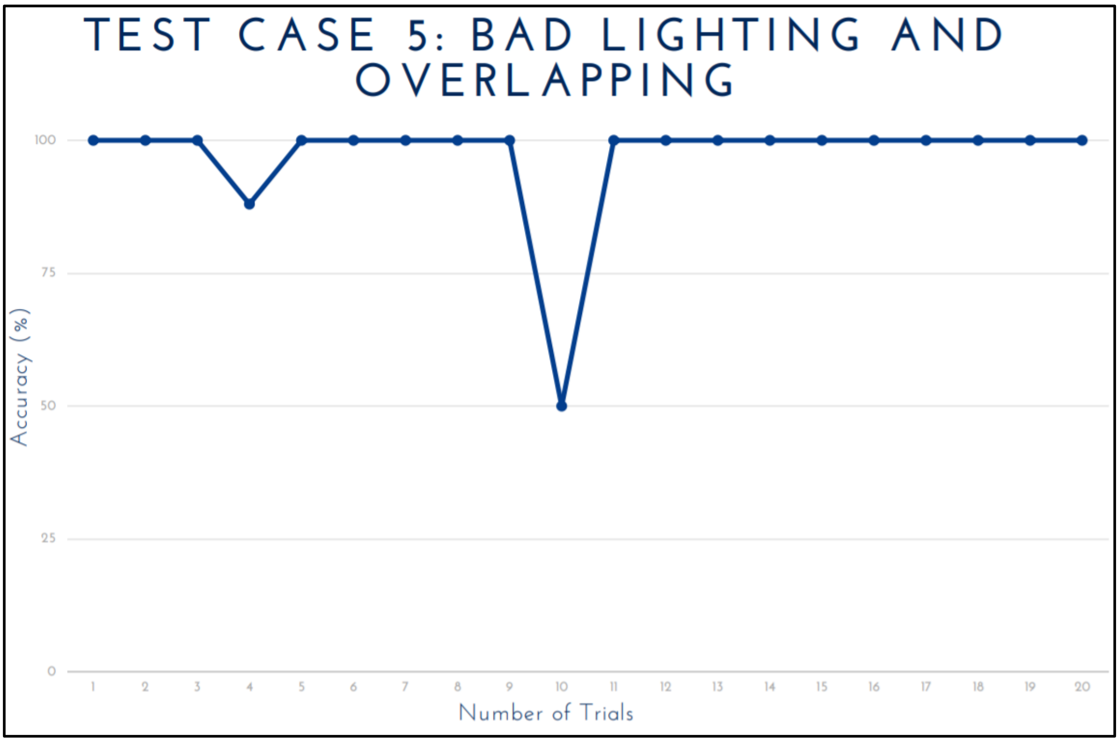


Figure 6-2-10 Accuracy rate for all trials for Test Case 5

Based on the results above, the outcome for Test Case 5 is considered as accurate as it produces 18 correct results out of 20 trials. The success rate for this test case is 90% while the failure rate is 10%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-10 shows the accuracy rate for all of the trials for Test Case 5.

Test Case 6: Overlapping and Different Angles of Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	4	4	Correct
2	4	4	Correct
3	3	3	Correct
4	3	3	Correct
5	2	2	Correct
6	3	3	Correct
7	2	2	Correct
8	5	5	Correct
9	2	2	Correct
10	4	4	Correct
11	2	2	Correct
12	2	2	Correct
13	4	4	Correct
14	5	5	Correct
15	5	5	Correct
16	2	2	Correct
17	3	3	Correct
18	3	3	Correct
19	8	7	Wrong
20	5	5	Correct

Table 6-2-6 Results for Test Case 6

Number of Trials: 20 Number of Correct Results: 19 Number of Wrong Results: 1 Success Rate: 19/20 * 100% = 95% Failure Rate: 1/20 * 100% = 5%

Figure 6-2-11 Sample Image in Test Case 6

Figure 6-2-11 shows sample images that will be used in Test Case 6. The images in Test Case 6 consists of overlapping oil palm fruits captured from the drone at different angles

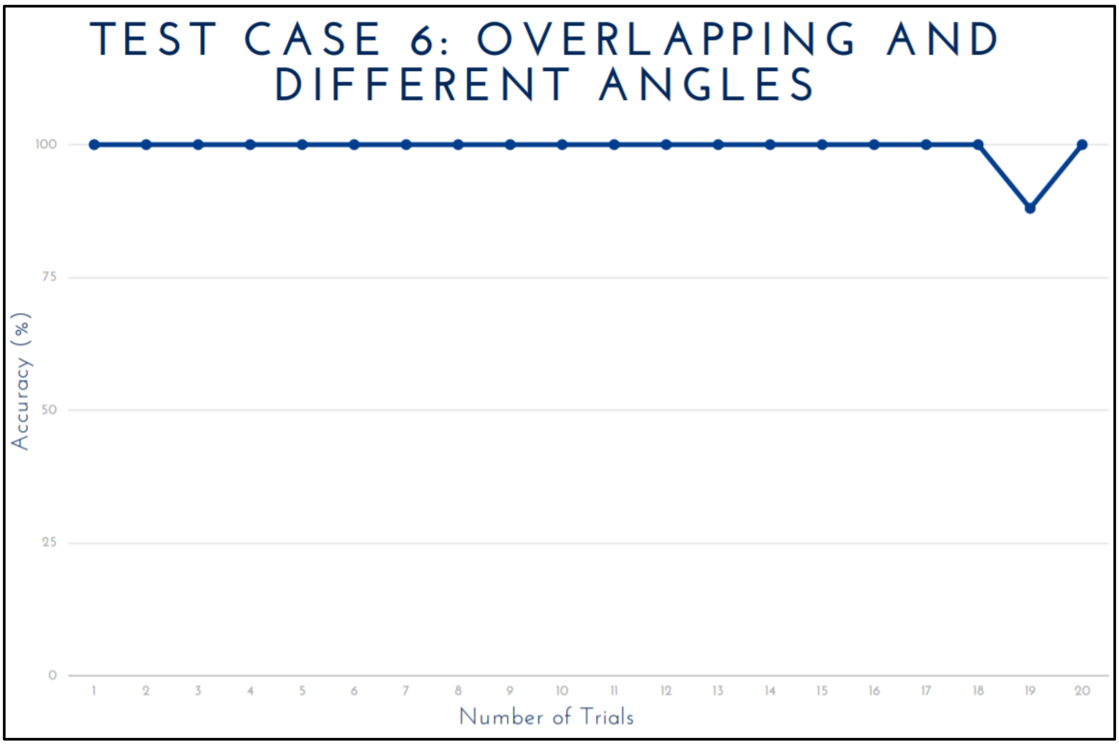


Figure 6-2-12 Accuracy rate for all trials for Test Case 6

Based on the results above, the outcome for Test Case 6 is considered as accurate as it produces 19 correct results out of 20 trials. The success rate for this test case is 95% while the failure rate is 5%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-12 shows the accuracy rate for all of the trials for Test Case 6.

Test Case 7: Bad Lighting and Different Angles of Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	3	3	Correct
2	2	2	Correct
3	2	2	Correct
4	1	1	Correct
5	1	1	Correct
6	1	1	Correct
7	1	1	Correct
8	2	2	Correct
9	1	1	Correct
10	2	2	Correct
11	2	2	Correct
12	1	1	Correct
13	2	2	Correct
14	1	1	Correct
15	1	1	Correct
16	1	1	Correct
17	1	1	Correct
18	3	3	Correct
19	1	1	Correct
20	3	3	Correct

Table 6-2-7 Results for Test Case 7

Number of Trials: 20 Number of Correct Results: 20 Number of Wrong Results: 0 Success Rate: 20/20 * 100% = 100% Failure Rate: 0/20 * 100% = 0%

Figure 6-2-13 Sample Image in Test Case 7

Figure 6-2-13 shows sample images that will be used in Test Case 7. The images in Test Case 7 consists of oil palm fruits captured from the drone with bad lighting and at different angles

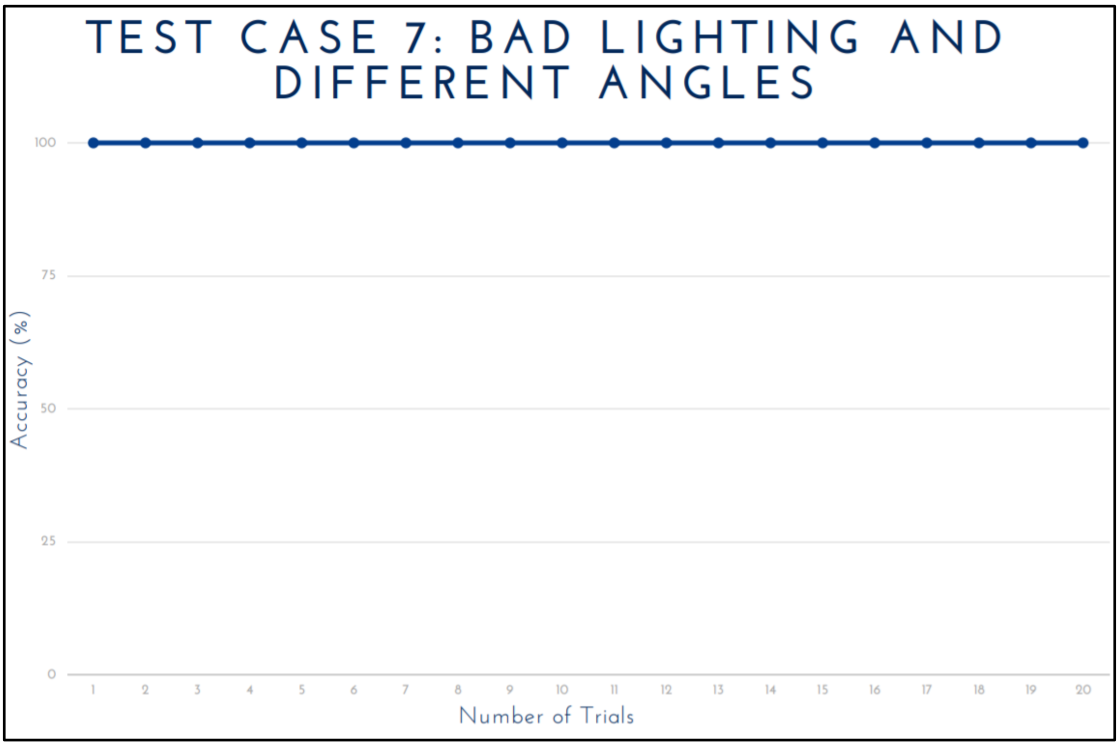


Figure 6-2-14 Accuracy rate for all trials for Test Case 7

Based on the results above, the outcome for Test Case 7 is considered as accurate as it produces 20 correct results out of 20 trials. The success rate for this test case is 100% while the failure rate is 0%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-14 shows the accuracy rate for all of the trials for Test Case 7.

Test Case 8: Bad Lighting, Overlapping and Different Angles of Oil Palm Fruits

Number of trials	Actual Count	System Count	Correct/Wrong
1	5	5	Correct
2	2	2	Correct
3	4	4	Correct
4	4	4	Correct
5	3	3	Correct
6	3	3	Correct
7	3	3	Correct
8	3	3	Correct
9	3	3	Correct
10	2	2	Correct
11	3	3	Correct
12	2	2	Correct
13	2	2	Correct
14	2	2	Correct
15	5	5	Correct
16	4	4	Correct
17	2	2	Correct
18	2	2	Correct
19	5	5	Correct
20	4	4	Correct

Table 6-2-8 Results for Test Case 8

Number of Trials: 20

Number of Correct Results: 20

Number of Wrong Results: 0

Success Rate: 20/20 * 100% = 100%

Failure Rate: 0/20 * 100% = 0%

Figure 6-2-15 Sample Image in Test Case 8

Figure 6-2-15 shows sample images that will be used in Test Case 8. The images in Test Case 8 consists of overlapping oil palm fruits captured from the drone with bad lighting and at different angles

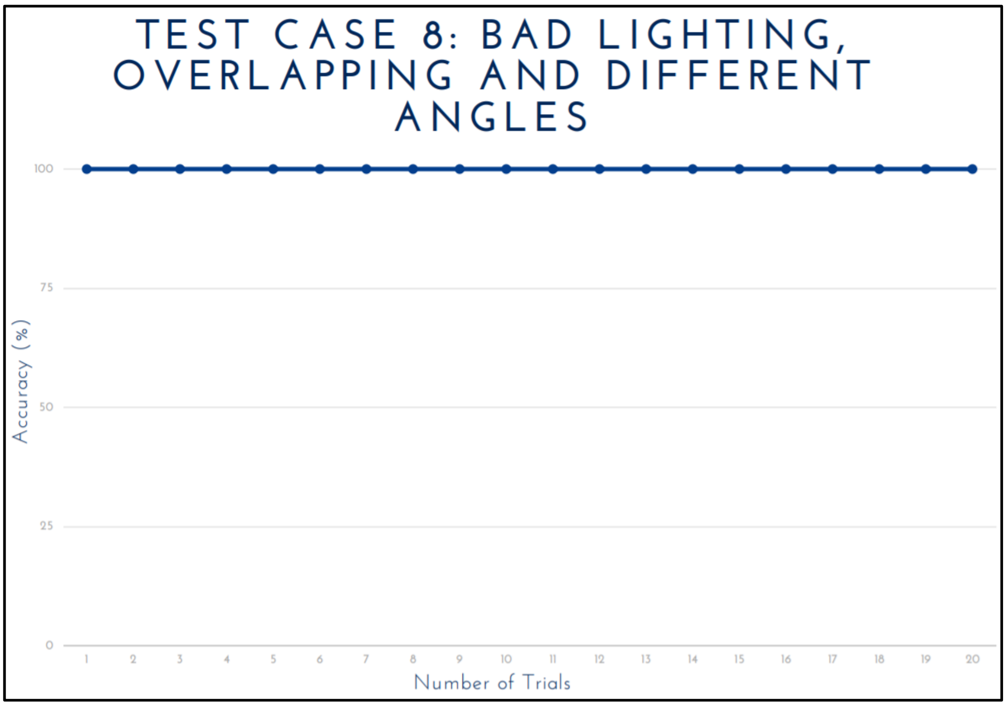


Figure 6-2-16 Accuracy rate for all trials for Test Case 8

Based on the results above, the outcome for Test Case 8 is considered as accurate as it produces 20 correct results out of 20 trials. The success rate for this test case is 100% while the failure rate is 0%. The reason for this is because the images are taken under different environmental conditions and the gathering of images for training are adequately available. Figure 6-2-16 shows the accuracy rate for all of the trials for Test Case 8.

6.3 Project Challenges

This project definitely has its own challenges than expected initially. The implementation of the system would be very difficult if some of these challenges are not solved immediately. Below are some challenges faced while completing this project:

- Adequate dataset – In order to train an oil palm detection and counting system with high accuracy, an adequate amount of dataset is needed to achieve it. It has been difficult to obtain adequate dataset for this project, especially during this Covid-19 pandemic, because there are cross-state and cross-district restrictions at the moment. Therefore, I was not able to visit a real oil palm plantation and capture enough dataset to train the system to the highest accuracy. This made me turn to online sources, but the images obtained are also not enough and very limited as well.

- System efficiency – To train a system with the highest efficiency, high computing resources are needed such as large memory space and power consumption. Due to limited memory space, the oil palm detection and counting system has not yet reach the highest level of efficiency.

- Limitation in machine learning – In machine learning, it takes time to completely understand and learn the concepts of machine learning. Due to limited knowledge about machine learning, the oil palm detection and counting system has not yet reach the highest level of performance. This system will take a longer research and development time to be improved in the near future.

6.4 Objectives Evaluation

The first objective of this project is to replace the traditional way of manually counting FFB with the automated way of counting oil palm tree FFB. This have been achieved by designing a GUI for the oil palm detection and counting system. Through this way, it allows farmers to automatically count oil palm FFB by just pressing a few buttons. The GUI consists of four buttons, which allows farmers to process the images captured by the drone, reload the new images that has been captured by the drone, preview the before detection and after detection images as well as quit the program.

The second objective of this project is to prove that, with image processing techniques, an automated remote sensing platform through UAV can be realized into different types of plantation to ease the workload of the farmers. This have been achieved by training and deploying the neural network to perform object detection and counting on the images that were captured by the drone in an oil palm plantation. This has been accomplished as seen in section 5.4, the image processing techniques were applied in the system, which acts as an artificial neural network and passes the images captured from the drone through it as well as produces results on the images, such as displaying the classes, detection boxes, detection scores and the number of oil palm fruits detected on the drone images.

With all of these objectives accomplished in this project, counting oil palm fruits is now easier than ever. It has become more time-saving and efficient in helping farmers complete this daily tedious job. A routine which once takes hours now only require minutes to complete with the help of image processing techniques and automated remote sensing platform through UAV.

6.5 Concluding Remark

In this project, the system testing and performance metrics were listed out in Section 6.1 to allow users to understand our test cases. In Section 6.2, the testing setup and results are recorded for the oil palm FFB detection and counting system. The project challenges are stated out in Section 6.3. Lastly, the evaluation of the project objectives is listed out in Section 6.4.

CHAPTER 7
CONCLUSION AND RECOMMENDATION

7.1 Conclusion

As a conclusion, the traditional way of manually counting oil palm fruits is an intensive task and has always been a distress among farmers. It is almost impossible to obtain accurate information with these methods in a large oil palm plantation. Besides, the traditional process of counting is prone to inaccurate estimation, time-consuming and expensive. Spending hours of human observation in rough weather conditions adds another problem for the small-scale farmers. This could reduce their efficiency in doing their tasks and could also cause health problems such as heat stroke for them. There are also elderly farmers who are no longer flexible in moving around the large oil palm plantation and jotting down the stock of the FFB in the area. As a result, a huge sum of money is needed in hiring someone else to fulfil that task.

Our motivation in this project is mainly to be able to produce an autonomous drone system integrated with computer vision in order to enhance the efficiency of the farmers' tasks on a daily basis as well as to reduce their workload. The existing system of manually counting oil palm fruits is slow and prone to inaccuracy, which could directly affect the outcome of the oil palm production yield. Therefore, this system is meant to help farmers to change to a new way of counting oil palm FFB by using UAVs and image processing, which is much faster, time-saving and at the same time, produce accurate results in counting oil palm FFB.

The solutions used in this project are divided into three main steps of image processing, which includes image pre-processing, feature extraction and training the ANN. In pre-processing, resizing on the images are performed. Next, common features are extracted from different images at different angles for the feature extraction process. Lastly, the ANN is trained to produce end results which shows the classes, detection boxes, detection scores and the number of oil palm FFB detected on the images. By using these few solutions, an autonomous drone system integrated with image processing is achieved in order to help farmers in different plantations to better enhance their efficiency as well as reduce their workload on a daily basis.

7.2 Recommendation

In this project, there are still many enhancements and improvements that can be done to the oil palm detection and counting system. Firstly, the dataset for the oil palm FFB images can be further improved by adding different weather conditions images such as noise, different light exposure and many others condition. This is because these images can enhance the overall accuracy of the system and affect the outcome on the drone images.

Next, the efficiency of the system can also be enhanced by using different algorithm models and computing resources. Different algorithm models can be tested out to see which model has the highest speed and accuracy in machine learning. Besides, the usage of GPU in laptop is recommended because Graphics Card have large number of cores, which leads to improved computation of multiple parallel processes. Large memory space in laptop is also recommended as well, as neural networks need to store input data, parameters as well as activations when an input passes through the network.

Lastly, the type of drone used in this project can be further enhanced. Parrot AR. Drone 2.0 has a very short flight time, which last for only about 10-12 minutes. This could be very inconvenient to gather dataset in an oil palm plantation as they are huge and requires a longer flight time to capture images. A drone with better battery life is recommended for this project such as the DJI Mavic 2 Pro which has a flight time of 34 minutes and capture high quality images and send a livestream straight from the drone's camera itself.

BIBLIOGRAPHY

Ahuja, K & Tuli, P 2013, *Object recognition by template matching using correlations and phase angle method.* International Journal of Advanced Research in Computer Science and Electronics Engineering, vol.2, pp. 1368-1373.

Ambrosia, VG, Wegener, SS, Sullivan, DV, Buechel, SW, Dunagan, SE, Brass, J, Stoneburner, J & Schoenung, SM 2003, *Demonstrating UAV-Acquired Real Time Thermal Data over Fires*. Photogrammetric Engineering and Remote Sensing, vol. 69, no. 4, pp. 391–402.

Ballesteros, R, Ortega, JF, Hernandez, D & Moreno, MA 2014, *Applications of georeferenced high-resolution images obtained with unmanned aerial vehicles. Part I: description of image acquisition and processing.* Precision Agriculture, vol.15, no.6, pp. 579-592.

Berni, JAJ, Zarco-Tejada, PJ, Suarez, L & Fereres, E 2009, *Thermal and narrowband multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle*. IEEE Transactions on Geoscience Remote Sensing, vol.47, no.3, pp.722–738.

Blaschke, T, Feizizadeh, B & Hölbling, D 2014, *Object-based image analysis and digital terrain analysis for locating landslides in the Urmia Lake Basin, Iran*. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol.7, no.12, pp.4806-4817.

Breckenridge, RP, Harbour, JL & Lee, RD 2006, 'Using Unmanned Aerial Vehicles to Assess Vegetative Cover and Identify Biotic Resources in Sagebrush Steppe Ecosystems: Preliminary Evaluation': *proceedings of the Eleventh Forest Service Remote Sensing Applications Conference*, Salt Lake City, Utah.

Cai, GW, Chen, BM & Lee, TH 2011, *Unmanned Rotorcraft Systems. Springer Science & Business Media*, Verlag, London.

Chao, HY, Baumann, M, Jensen, A, Chen, YQ, Cao, YC, Ren, W, McKee, M 2008, 'Band reconfigurable Multi-UAV-based Cooperative remote sensing for real time water management and distributed irrigation control': *proceedings of the 17th IFAC World Congress*, Seoul, Korea, vol.41, no.2, pp.11744-11749.

Colomina, I & Molina, P 2014, *Unmanned aerial systems for photogrammetry and remote sensing: A review*. ISPRS Journal of Photogrammetry and Remote Sensing, vol.92, pp.79–97.

FAO world data, 2016. Available from: <http://faostat.fao.org/>.

Franklin, P, Pearlstine, L, Dewitt, B, Smith, S, Watts, A, Ifju, P, Lee, K & Bowman, S 2006, *Autonomous unmanned aerial vehicle (UAV) for ecological research.*

Gujar, H, Mhatre, P, Ghanate, S, Chile, et al. 2016, *Python Based Image Processing.*

Herwitz, SR, Johnson, LF, Dunagan, SE, Higgins, RG, Sullivan, DV, Zheng, J, Lobitz, BM, et al. 2004, *Imaging from an Unmanned Aerial Vehicle: Agricultural Surveillance and Decision Support.* Computers and Electronics in Agriculture, vol.44, no.1, pp.49–61.

Kalantar, B, Idrees, MO, Mansor, SB, Halin, AA 2017, *Smart counting–oil palm tree inventory with UAV.* Coordinates e-Navigation Assistance System for Cooperative Collision Avoidance at Sea, vol.13, pp.17-22.

Maes, WH & Steppe, K 2019, *Perspectives for remote sensing with unmanned aerial vehicles in precision agriculture.* Trends Plant Science, vol.24, no.2, pp.152 -164.

Mark, J & Hardin, PJ 2005, 'Applications of Inexpensive Remotely Piloted Vehicles (Rpvs) for Collection of High-Resolution Digital Imagery.': *proceedings of the 20th Biennial Workshop on Aerial Photography, Videography, and High Resolution Digital Imagery for Resource Assessment*, Weslaco.

Mathews, AJ & Jensen, JLR 2013, *Visualizing and quantifying vineyard canopy LAI using an unmanned aerial vehicle (UAV) collected high density structure from motion point cloud.* Remote Sensing, vol.5, pp.2164–2183.

Overgaard, SI, Isaksson, T, Kvaal, K & Korsaeth, A 2009, *Comparisons of two hand held, multispectral field radiometers and a hyperspectral airborne imager in terms of predicting spring wheat grain yield and quality by means of powered partial least squares regression.* Journal of Near Infrared Spectroscopy, vol.18, pp.247–261.

Pajares, G 2015, *Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs).* Photogrammetric Engineering and Remote Sensing, vol.81, pp.281–329.

Rizeei, HM, Shafri, HZM, Mohamoud, MA, Pradhan, B, et al. 2018, *Oil Palm Counting and Age Estimation from WorldView-3 Imagery and LiDAR Data Using an Integrated OBIA Height Model and Regression Analysis*. Journal of Sensors, vol. 2018.

Rueda, C, Miserque, J & Laverde, R 2016, *Validation of an oil-palm detection system based on a logistic regression model.* 2016 IEEE ANDESCON, Arequipa, pp.1-4.

Shamshiri, R, Hameed, IA, Balasundram, SK, Weltzien, C, Yule, IJ, Grift, TE et al. 2018, *Adapting simulation platforms and virtual environments for acceleration of agricultural robotics: A perspective of digital farming.* International Journal of Agricultural and Biological Engineering, vol.11, no.4, pp.1-25.

Shamshiri, RR 2017, ‘Integration of smart sensors and robotics in increasing agricultural productivity with higher yields at lower costs.’: *proceedings of Asian Space Technology Summit.*

Srestasathiern, P & Rakwatin, P 2014, *Oil palm tree detection with high resolution multi-spectral satellite imagery*. Remote Sensing, vol. 6, no. 10, pp. 9749–9774.

Sugiura, R, Noguchi, N & Ishii, K 2005, *Remote-sensing technology for vegetation monitoring using an unmanned helicopter*. Biosystems Engineering vol. 90, pp. 369–379.

Verhoeven, GJJ 2009, *Providing an archaeological bird's-eye view - an overall picture of ground-based means to execute low-altitude aerial photography (LAAP) in archaeology.* Archaeological Prospection, vol. 16, pp. 233–249.

Wong-in, T, Kaewkongka, T, Cooharojananone, N, Lipikorn, R 2015, *Automatic Oil Palm Detection and Identification from Multi-scale Clustering and Normalized Cross Correlation*. Industrial Engineering, Management Science and Applications 2015. Lecture Notes in Electrical Engineering, vol. 349.

Xiang, HT & Tian, L 2011, *Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV).* Biosystems Engineering, vol.108, pp.174–190.

Xiongkui, H, Bonds, J, Herbst, A & Langenakens, J 2017, *Recent development of unmanned aerial vehicle for plant protection in East Asia.* International Journal of Agricultural and Biological Engineering, vol.10, no.3, pp.18.

Yang, G, Liu, J, Zhao, C, Li, Z, Huang, Y, Yu, H, Xu, B, Yang, X, Zhu, D, Zhang, X, Zhang, R, Feng, H, Zhao, X, Li, Z, Li, H & Yang, H 2017, *Unmanned Aerial Vehicle Remote Sensing for Field-Based Crop Phenotyping: Current Status and Perspectives.* Frontiers in Plant Science, vol.8, pp.1111.

Yang, Z 2010, *Fast template matching based on normalized cross correlation with centroid bounding.* International Conference on Measuring Technology and Mechatronics Automation, vol.2, no. 1, pp. 224–227.

Zarco-Tejada, P, Gonzalez-Dugo, V & Berni, JAJ 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, vol.117, pp.322–337.

Zhang, CH & Kovacs, JM 2012, *The application of small unmanned aerial systems for precision agriculture: a review.* Precision Agriculture, vol.13, pp.693–712.

APPENDIX A – FINAL YEAR PROJECT BI-WEEKLY REPORT

FINAL YEAR PROJECT WEEKLY REPORT

(Project I)

Trimester, Year: Y3T3	**Study week no.: 1 & 2**
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system

2. WORK TO BE DONE

- Finding enough dataset to train the network
- Improving accuracy of system
- Improving efficiency of system
- System testing
- Improving graphic user interface
- Project documentation

3. PROBLEMS ENCOUNTERED

- No problems for the time being

4. SELF EVALUATION OF THE PROGRESS

- Knowledge on machine learning improved.

Supervisor's signature

Student's signature

Trimester, Year: Y3T3	**Study week no.: 3 & 4**
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system
- Collected enough dataset to train the network

2. WORK TO BE DONE

- Improving accuracy of system
- Improving efficiency of system
- System testing
- Improving graphic user interface
- Project documentation

3. PROBLEMS ENCOUNTERED

- No problems for the time being

4. SELF EVALUATION OF THE PROGRESS

- Learnt how to tackle most of the problems occurred while re-training a network (some of it may be due to the images)

Supervisor's signature

Student's signature

Trimester, Year: Y3T3	**Study week no.: 5 & 6**
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system
- Collected enough dataset to train the network
- Improving accuracy of system

2. WORK TO BE DONE

- Improving efficiency of system
- System testing
- Improving graphic user interface
- Project documentation

3. PROBLEMS ENCOUNTERED

- High accuracy of system is hard to achieved as big dataset is needed to train the network

4. SELF EVALUATION OF THE PROGRESS

- Learnt more on how to improve accuracy by adding images with different weather conditions to the dataset

Supervisor's signature

Student's signature

Trimester, Year: Y3T3	**Study week no.: 7 & 8**
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system
- Collected enough dataset to train the network
- Improved accuracy of system
- Improving efficiency of system

2. WORK TO BE DONE

- System testing
- Improving graphic user interface
- Project documentation

3. PROBLEMS ENCOUNTERED

- Having difficulty with improving efficiency of the system

4. SELF EVALUATION OF THE PROGRESS

- Have better understanding towards the overall system

Supervisor's signature

Student's signature

Trimester, Year: Y3T3	Study week no.: 9 & 10
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE
[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system
- Collected enough dataset to train the network
- Improved accuracy of system
- Completed system testing
- Improving efficiency of system

2. WORK TO BE DONE

- Improving graphic user interface
- Project documentation

3. PROBLEMS ENCOUNTERED

- Having difficulty on improving the system to the highest efficiency

4. SELF EVALUATION OF THE PROGRESS

- Learnt how to conduct a complete system testing in this project

Supervisor's signature

Student's signature

Trimester, Year: Y3T3	Study week no.: 11 & 12
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system
- Collected enough dataset to train the network
- Improved accuracy of system
- Completed system testing
- Improved efficiency of system
- Improved graphic user interface

2. WORK TO BE DONE

- Project documentation

3. PROBLEMS ENCOUNTERED

- Compiling the final report

4. SELF EVALUATION OF THE PROGRESS

- Better understanding towards improving accuracy and efficiency of system

Supervisor's signature

Student's signature

Trimester, Year: Y3T3	Study week no.: 13
Student Name & ID: Rachel Yee Jee San 1701436	
Supervisor: Dr. Goh Hock Guan	
Project Title: Oil Palm Yield Data Collection Using Image Processing	

1. WORK DONE
[Please write the details of the work done in the last fortnight.]

- Reviewed FYP1 report
- Discussed about improving the accuracy and efficiency of the system
- Collected enough dataset to train the network
- Improved accuracy of system
- Completed system testing
- Improved efficiency of system
- Improved graphic user interface
- Completed project documentation

2. WORK TO BE DONE

- Prepare presentation slides
- Make minor configuration on the system for demonstration

3. PROBLEMS ENCOUNTERED

- Preparing for demonstration

4. SELF EVALUATION OF THE PROGRESS

- Have confidence in demonstrating the full functionalities of the system

Supervisor's signature

Student's signature

APPENDIX B - POSTER

OIL PALM YIELD DATA COLLECTION USING IMAGE PROCESSING

by Rachel Yee Jee San

INTRODUCTION

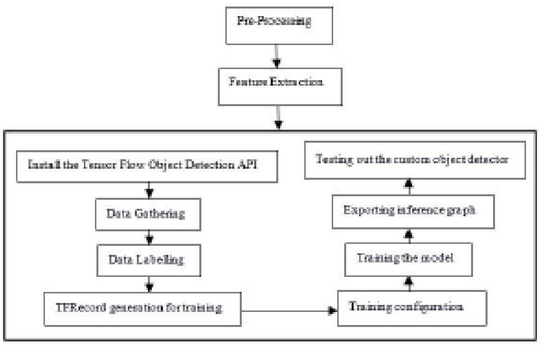
This project is an automated drone program integrated with image processing for quantification of oil palm tree fruits. It is designed to overcome the existing problems where manually counting is time-consuming and inefficient. Farmers are often faced with many problems such as extreme weather conditions and movement inflexibility. With the combination of image processing and UAV technology, quantification of oil palm tree FFB can be done indoors and the results can be produced within minutes.

DISCUSSION

In this project, we will be using the Tensorflow Object Detection API to train the ANN. There are many algorithms in the Object Detection API and 3 common methods, faster-rcnn, SSD and YOLO are reviewed for their suitability in object detection. In the end, faster-rcnn is chosen because its accuracy is the best compared to others, since accuracy is a priority in detecting the production yield for oil palm fruits.

METHODS

There are 3 main steps used in this project which is image pre-processing, feature extraction and training the ANN.



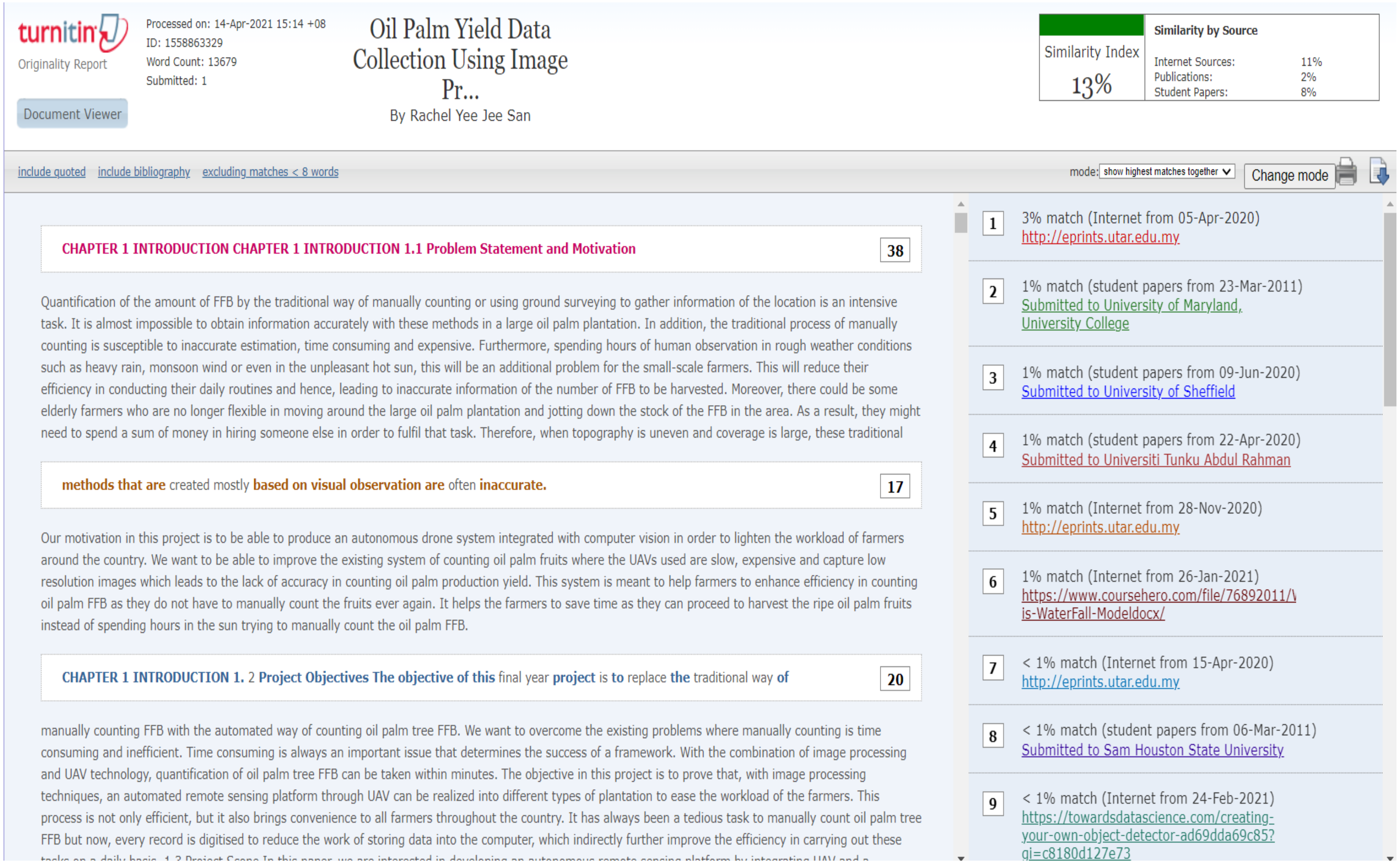
CONCLUSION

Our motivation in this project is to be able to produce an autonomous drone system integrated with computer vision in order to lighten the workload of farmers around the country. By using these few solutions, we have achieved a high accuracy oil palm detection and counting system.

FINAL SYSTEM

The final system will display results to the user through the GUI by displaying the classes, detection boxes, detection scores and the number of oil palm fruits detected on the images captured by the drone.

APPENDIX C - PLAGIARISM CHECK SUMMARY



turnitin
Originality Report
Document Viewer

Processed on: 14-Apr-2021 15:14 +08
ID: 1558863329
Word Count: 13679
Submitted: 1

Oil Palm Yield Data Collection Using Image Pr...
By Rachel Yee Jee San

Similarity Index
13%

Similarity by Source
Internet Sources: 11%
Publications: 2%
Student Papers: 8%

include quoted include bibliography excluding matches < 8 words

mode: show highest matches together Change mode

CHAPTER 1 INTRODUCTION CHAPTER 1 INTRODUCTION 1.1 Problem Statement and Motivation 38

Quantification of the amount of FFB by the traditional way of manually counting or using ground surveying to gather information of the location is an intensive task. It is almost impossible to obtain information accurately with these methods in a large oil palm plantation. In addition, the traditional process of manually counting is susceptible to inaccurate estimation, time consuming and expensive. Furthermore, spending hours of human observation in rough weather conditions such as heavy rain, monsoon wind or even in the unpleasant hot sun, this will be an additional problem for the small-scale farmers. This will reduce their efficiency in conducting their daily routines and hence, leading to inaccurate information of the number of FFB to be harvested. Moreover, there could be some elderly farmers who are no longer flexible in moving around the large oil palm plantation and jotting down the stock of the FFB in the area. As a result, they might need to spend a sum of money in hiring someone else in order to fulfil that task. Therefore, when topography is uneven and coverage is large, these traditional

methods that are created mostly based on visual observation are often inaccurate. 17

Our motivation in this project is to be able to produce an autonomous drone system integrated with computer vision in order to lighten the workload of farmers around the country. We want to be able to improve the existing system of counting oil palm fruits where the UAVs used are slow, expensive and capture low resolution images which leads to the lack of accuracy in counting oil palm production yield. This system is meant to help farmers to enhance efficiency in counting oil palm FFB as they do not have to manually count the fruits ever again. It helps the farmers to save time as they can proceed to harvest the ripe oil palm fruits instead of spending hours in the sun trying to manually count the oil palm FFB.

CHAPTER 1 INTRODUCTION 1. 2 Project Objectives The objective of this final year project is to replace the traditional way of 20

manually counting FFB with the automated way of counting oil palm tree FFB. We want to overcome the existing problems where manually counting is time consuming and inefficient. Time consuming is always an important issue that determines the success of a framework. With the combination of image processing and UAV technology, quantification of oil palm tree FFB can be taken within minutes. The objective in this project is to prove that, with image processing techniques, an automated remote sensing platform through UAV can be realized into different types of plantation to ease the workload of the farmers. This process is not only efficient, but it also brings convenience to all farmers throughout the country. It has always been a tedious task to manually count oil palm tree FFB but now, every record is digitised to reduce the work of storing data into the computer, which indirectly further improve the efficiency in carrying out these

1 3% match (Internet from 05-Apr-2020)
http://eprints.utar.edu.my

2 1% match (student papers from 23-Mar-2011)
Submitted to University of Maryland, University College

3 1% match (student papers from 09-Jun-2020)
Submitted to University of Sheffield

4 1% match (student papers from 22-Apr-2020)
Submitted to Universiti Tunku Abdul Rahman

5 1% match (Internet from 28-Nov-2020)
http://eprints.utar.edu.my

6 1% match (Internet from 26-Jan-2021)
https://www.coursehero.com/file/76892011/\ is-WaterFall-Modeldocx/

7 < 1% match (Internet from 15-Apr-2020)
http://eprints.utar.edu.my

8 < 1% match (student papers from 06-Mar-2011)
Submitted to Sam Houston State University

9 < 1% match (Internet from 24-Feb-2021)
https://towardsdatascience.com/creating-your-own-object-detector-ad69dda69c85?gi=c8180d127e73

APPENDIX D – FM-IAD-005

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	RACHEL YEE JEE SAN
ID Number(s)	17ACB01436
Programme / Course	BACHELOR OF INFORMATION TECHNOLOGY (HONS) COMPUTER ENGINEERING
Title of Final Year Project	OIL PALM YIELD DATA COLLECTION USING IMAGE PROCESSING

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 13 % **Similarity by source** Internet Sources: 11 % Publications: 2 % Student Papers: 8 %	
Number of individual sources listed of more than 3% similarity: 1	The mapping of 3% is due to the template.
Parameters of originality required and limits approved by UTAR are as follows: (i) Overall similarity index is 20% and below, and **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor	Signature of Co-Supervisor
Name: Goh Hock Guan	Name:
Date: 15/4/2021	Date:

APPENDIX E – CHECKLIST FOR FYP2

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	17ACB01436
Student Name	RACHEL YEE JEE SAN
Supervisor Name	DR. GOH HOCK GUAN

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Front Cover
√	Signed Report Status Declaration Form
√	Title Page
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result – Form Number: FM-IAD-005)

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report. ______________________ (Signature of Student) Date: 14/4/2021	Supervisor verification. Report with incorrect format can get 5-mark (1 grade) reduction. ______________________ (Signature of Supervisor) Date: Goh Hock Guan