Identifying Ripened and Overripe Fruits Using Hyperspectral Imagery

Sansayan Gajanithy

2104620

BEng Computers with Electronics

Supervisor: Dr Adrian Clark

Second Assessor: Dr Vishwanathan Mohan

Acknowledgements

My gratitude goes to everyone who supported and took part in this journey to the completion of this project. I am more than thankful for your aid and guidance. It has been a pleasure working with all of you.

A special thanks to my capstone supervisor Dr Adrian Clark, for the precise guidance, thoughtful feedback, and strong encouragement. His expertise in computer vision has significantly enriched my work.

I am thankful for the Hyperspectral dataset of fruits made available by the Cognitive Systems Research Group, University of Tuebingen. The recorded data for their project has been the backbone of my project, as these data have been utilized in the initial stages until I had access to the hyperspectral imager.

I am grateful for the efforts put in by the CSEE Head of school team, especially Dr Martin Reed at Essex University to accelerate the processes involved to grant me access to a hyperspectral imager in time to record data. The technicians deserve my gratitude for their efforts to finish the certain risk assessment for the imager.

My respect goes to Dr Vishwanathan Mohan, for his advice and wisdom. He had continuous positive expectations of me to complete the project successfully.

Abstract

The demand for resources has increased as the global population increased in the last century. The main demand is in the food supply chain. Governments worldwide are creating sustainable solutions to overcome the demand for food supply by developing agriculture industries and expanding the production of food. The key goal is to use technology to sustain food production and reduce wastage.

Hyperspectral imagers record information across the electromagnetic spectrum as spectral bands. These image bands generate more detailed information, enabling early detection of fruits' lifespan and reducing food wastage.

Fruits undergo subtle colour changes as they ripen, for some fruits like avocados, it may be very challenging for a human eye or regular cameras to detect the ripeness. The expectation is that spectral data analysis will lead to an observation of a change in the reflectance of the fruit as it ripens. There should be a shift in the wavelength across the spectrum as they ripen resulting in the life stage at which the fruit is present being identified. The observation and analysis will help the agriculture industry to identify the correct life stage of fruits and cultivate them at the correct time to increase the yield of production.

Table of Contents

Acknowledgements **2**

Abstract **3**

Chapter 1 – Prelude **8**

Chapter 2 – Literature Review **9**

2.1 The Growing world and the requirement for food supply *9*

2.2 What is Hyperspectral Imaging? *9*

2.3 Inspirations and related work *10*

2.4 The use of Hyperspectral data in food and agriculture industries *11*

2.5 Avocados *11*
2.5.1 Data Sample 12

2.6 Software and Hardware Requirements *12*

Chapter 3 – Legal, Ethical, Health and Safety related issues **13**

3.1 Health and safety considerations *13*

3.2 Legal Considerations *13*

3.3 Ethical Considerations *14*

3.4 social, economic, and cultural considerations *14*

3.5 Program rules and standards required by the industry. *14*

3.6 Sustainability and environmental considerations *15*

Chapter 4 – Main Text **16**

4.1 Hyperspectral Imager Access *16*

4.2 Hyperspectral Imager operation and connectivity *17*
4.2.1 Imager and Connections 17
4.2.2 Lumo Scanner 17
4.2.3 Aquisation of Data 18

4.3 Phase 1 – Visual Representation of Hyperspectral Data *20*
4.3.1 Exporting data into Python using spectral python. 20
4.3.2 Experimentation 21
4.3.3 K-means and graph generation 22
4.3.4 Matplotlib Animation Framework 23

4.4 Phase 2 – Data Processing *24*
4.4.1 Matching Regions 25
4.4.2 The Use of Masks 28
4.4.3 Mean value calculations 30
4.4.4 Avocado animations 30
4.4.5 Thresholding 32
4.4.6 Anomalies and difference in data 33

4.5 Thresholding vs Deep Learning *34*

Chapter 5 – Evaluation of data from the imager across environments. **35**

Chapter 6 – Conclusion **39**

6.1 conclusion of the Work *39*

6.2 Future work ... *39*

Chapter 7 – Project planning ... 40

7.1 Risk analysis ... *40*

7.2 Project progression ... *40*

7.3 The review of the project ... *42*

References ... 43

Table of Figures

Figure 1 The electromagnetic spectrum and the range of Hyperspectral Images [2] 10
Figure 2 Avocado life cycle. 12
Figure 3 Specim Fx10e 17
Figure 4 Avocado set 1, fruit 2 [back] (256 x 456).. 18
Figure 5 Detector view and waterfall display widgets were not functional.............................. 19
Figure 6 spectral resolution canvas. 20
Figure 7ground truth image of mango at day 1. 21
Figure 8 Mango spectral graph Day 1 of Fruit 1 23
Figure 9 spectral value plot of Prerecorded data. 24
Figure 10 spectral value plot independently recorded data by the researcher. 25
Figure 11 Image 1 with the region of interest. 26
Figure 12 Image 2 with the best matched region. 27
Figure 13 Mask of avocado with otsu's threshold. 29
Figure 14 Mask of Avocado with Harris corner detector’s threshold. 29
Figure 15 Avocado graphs of fruit 1 back side. 31
Figure 16 Avocado graphs of fruit 3 front side. 31
Figure 17Avocado graphs of fruit 4 back side. 31
Figure 18 Avocado graphs of fruit 5 front side. 32
Figure 19 Avocado measured with Halogen light and High room lighting. 35
Figure 20 Avocado measured with Halogen light and Low room lighting. 36
Figure 21 Avocado measured with only sunlight. 36
Figure 22 Bruised Avocados measured with Halogen lights and High room lighting. 37
Figure 23 Bruised Avocados measured with Halogen lights and Low room lighting. 38
Figure 24 Cumulative flow diagram showing progression. 41
Figure 25 Pie chart showing the types of issue created. 41

Table of Tables

Table 1 Thresholding Vs Machine Learning..34

Chapter 1 – Prelude

The main concept of the project is developing a vision algorithm to measure ripeness in fruit. Hyperspectral data captures multiple band information across the electromagnetic spectrum and this data can be used to understand and observe specific patterns in wavelength. Globally, there isn't much research involving hyperspectral data in agri-food industries. This is due to the expense of spectral imaging. However, at the University of Essex, hyperspectral imaging cameras are available and projects within agri-food technologies are strongly encouraged. This document introduces the concepts of Hyperspectral imaging and the vision system developed. The following chapters conclude the study made by the researcher.

This material starts with Chapter 2, a context review of the project explaining the global problem, the need for the project, the key concepts of hyperspectral imaging, data, and software requirements needed to begin the project. The review is a detailed analysis of the researcher's understanding of concepts on hyperspectral imaging. Chapter 3 lists all the concerns and considerations regarding the legal, ethical, health and safety of the project. Chapter 4 consists of the main text, which is the approaches to developing a processing algorithm. The chapter is divided into five sections. Section 4.1 reports on the imager access issue the researcher faced. Section 4.2 describes the data acquisition from the hyperspectral imager, which the researcher operated. The project is divided into two different phases. Phase 1 on visually representing spectral data is listed in section 4.3 and section 4.4 explains the experiments, and techniques used to build the processing system and the results obtained. Section 4.5 is a comparison of thresholding vs machine learning concepts. Chapter 5 is the evaluation performed of the environment in which the imager is used to acquire data. Chapter 6 concludes the work completed and explains the future work that could be done on the project. Finally, chapter 7 explains the project plan and management that were carried out.

Chapter 2 – Literature Review

2.1 The Growing world and the requirement for food supply

The population growth of the world is growing at a faster rate, and we are expected to reach over 9 billion people by the year 2050 according to UN estimations. This growth results in a chained reaction and increases the demand for food supply. There are several factors, which also contribute to the need for more food supply such as economic growth and dietary preferences. The demand for food is expected to increase by around 60 % by the year 2050 [1]. The challenge is to keep this supply available to the requirements, which may be difficult. One main reason behind this is the demand for laborers. The decrease in laborers farming and cultivating food is decreasing year by year due to the changing world. This creates a rift across agriculture industries and to tackle this issue, the use of machinery and technology to efficiently produce and cultivate food is implemented. The use of technology reduces human error during the cultivation of these fresh produce and reduces food wastage. Food wastage is one factor which prevents the produce from reaching consumers. Fresh produce is affected the most due to environmental constraints, that narrow the crop yield. Regular wastage of food occurs when the correct time of cultivation cannot be identified. With the help of technology, it is possible to identify and cultivate the produce at the correct time it is ready to be consumed. Governments are funding projects in agriculture and supply industries which use technology.

2.2 What is Hyperspectral Imaging?

Hyperspectral imaging is a technique, in which information is processed across the electromagnetic spectrum between the visible ranges of ultraviolet to infrared. The electromagnetic spectrum consists of all wavelengths and frequencies, which are both visible and non-visible to the human eye emitted by electromagnetic radiation. A typical RGB image captures blue, green, and red channels from a visible spectrum between the range of 380 – 740 nm. This channel range is increased in the hyperspectral images from 250 – 1500 nm [2]. This results in a detailed capture of the objects as the ranges are wider. Hyperspectral images are recorded as multiple bands across the spectrum, while the RGB only records three bands, thus allowing us to examine unique spectral signatures across bands at pixel levels.

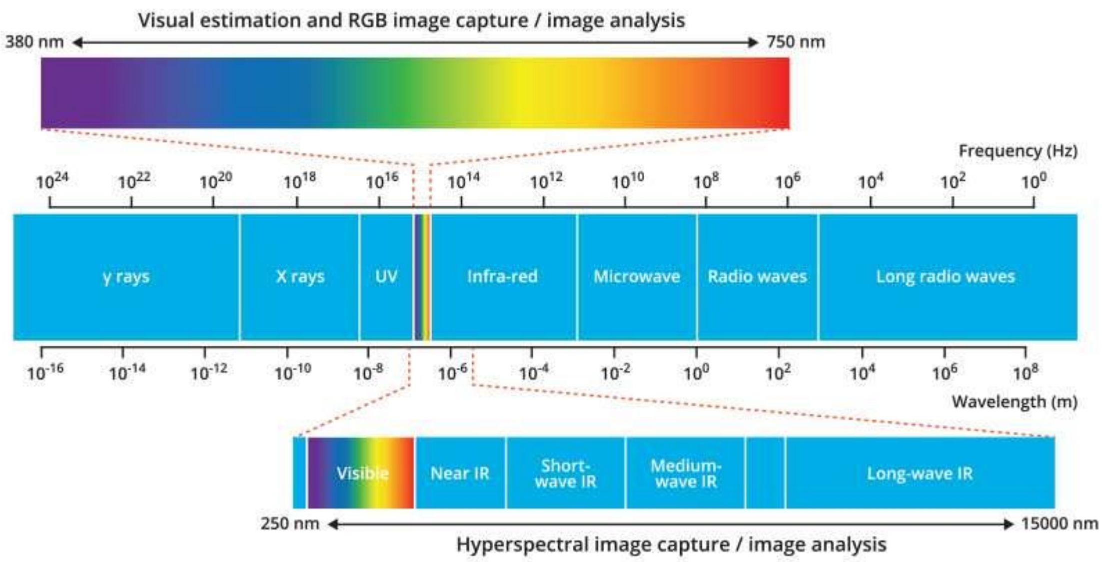

Figure 1 The electromagnetic spectrum and the range of Hyperspectral Images [2]

The interaction of light with objects, also known as spectroscopy, is used in hyperspectral imaging to extract spectral signatures of objects. These spectral signatures across bands determine the level of intensity of the object being captured. With this representation, the reflectance level of the object is determined and analyzed.

The spectral data is represented as raw data when it is captured through a hyperspectral imager and usually in the format of ENVI file format which includes a header file and a binary file. The header file holds information regarding the metadata of the image such as file type, number of samples and lines, number of bands, default bands for data visualization and more. The Binary file holds the raw information of the image captured [3]. Spectral data is often referred to as hypercube, data recorded are arranged in a 3D array M x N x C, where M x N is the spatial dimensions and C is the spectral dimensions [4]. When spectral data is recorded, spectral binding may typically occur, where several bands might merge into single bands.

2.3 Inspirations and related work

The project began with the inspiration and advancement of projects that were conducted across the world. Although this is a pre-made project listed by the Supervisor Dr Adrian Clark, the main reason this thesis has been conducted by the researcher is based on two projects with hyperspectral data. Vegetation analysis of multispectral images using Python, which sets the template to understand spectral information and their usefulness. The exploration of vegetation in Krishna and Godavari Districts made crucial contributions [5]. Another project that has inspired the researcher to work on Measuring the ripeness of fruit with Hyperspectral Imaging and Deep Learning. This project completed by the Cognitive Research Group at the University of Tuebingen is like this project which recorded fruits using a hyperspectral imager and measured the maturity with machine learning techniques [6]. The key difference is that their project is heavy on deep learning while the project to be conducted here wants to build an efficient processing algorithm and analysis of a suitable system for the study to be conducted. The initial stages used the data made available by this research and the latter stages converted onto the data recorded by the Specim Fx10e hyperspectral imager by the researcher. This inspirational work particularly used PyTorch to train data with deep learning algorithms and determine the ripeness of the fruit. It has been stated the observations made on avocados and kiwis were successful and a pattern visible in wavelength.

2.4 The use of Hyperspectral data in food and agriculture industries

Hyperspectral image-based analysis is used in several industries like mining, waste management, space, agriculture and more. These images are predominantly used for their detailed image output. In the agriculture industry, it is often used to identify the quality of produce, diseases and infections, maturity of the crops and more.

The argument is that hyperspectral data with a combination of machine learning algorithms would be able to efficiently increase the production of yield and match the food supply requirements. The main crops that could experience benefits are vegetable and fruit crops. Plant varieties contain higher amounts of water content and some types of fruit and veg do not go through substantial colour change. The observation of band channel information in fruits can determine their maturity by projecting the reflectiveness of the skin as the fruit ripens. A regular RGB has only three reference channels (red, green, and blue) making it difficult to figure out a pattern in their spectral signature as it ripens. However, a hyperspectral camera produces a hypercube with images of the fruit in multiple spectral channels meaning that we can investigate the unique spectral signature of the fruit across the bands by producing a wavelength in a graph-based format. An observation can be made where a pattern in the wavelength at specific points. This project aims to observe the maturity of fruit using the hyperspectral data and processing an algorithm which identifies which stage the fruit is at in its life cycle by measuring the ripeness. The chosen fruit for this project is avocados.

2.5 Avocados

The main reason for avocados to be chosen for analysis is their properties, which makes it visually harder to identify their level of ripeness by the human eye. Avocados are climacteric fruit, meaning they ripen even after they are harvested due to their strong ethylene content. Ethylene is a natural gas which released to ripen fruits [7]. Avocados are usually harvested when they are unripe and exported all over the globe. The challenge of avocados when harvesting, the fruit needs to be separated. According to the OECD standards [8], firm avocados are selected to be harvested and exported. This segmentation involves avoiding fruits with deformities and anomalies. Then they are also separated by shape, and quality (class).

The ripening process of avocados can be made faster by releasing ethylene externally. The factors that can affect the rate of ethylene gas productivity are temperature, lighting, and airflow. Several materials were available online to ripen avocados at home faster. Certain fruits like bananas produce ethylene gas so avocados placed next to these fruits can ripen faster. The other methods include placing the fruit in a warmer place and covering the avocados with brown paper [9]. Naturally, avocados ripen for around a week when left alone. Figure 2 states the ripening cycle of avocados over 10 days. Avocados go through a subtle colour change as they ripen ranging from green when it is unripe, purple hue during a perfect condition to be eaten, and black when overripe. This estimation depends on the previous factors written and based on popular varieties like Hass and Fuerte.

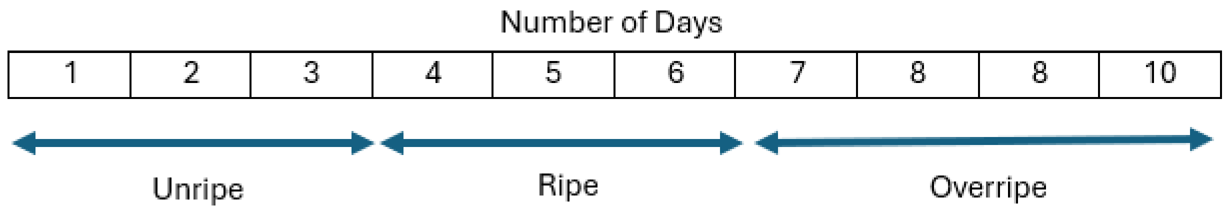

Figure 2 Avocado life cycle.

The recording of the avocados in this project is of the Hass variety and they are generally left alone to naturally ripen.

2.5.1 Data Sample

The sample size of the recorded data plays a crucial role in the outcome of the processing that can be used. The larger the sample data, a machine learning algorithm could be built. This is entirely for the larger set of data required as the training data. The data acquired must be a clear capture of the object. The resolution controls the size of the data. The higher resolution results in more unique spectral signatures of the fruit but it generally results in larger file size for storage. Therefore, a reasonable resolution is required for the images that should be captured. The data used in the initial stages were enough to generate a machine learning algorithm on its own but lacks integrity and if the independently recorded are different, a machine learning algorithm will not be used. See the section below.

2.6 Software and Hardware Requirements

This project is conducted in Python3 and requires spectral python (Spy) [10], which is a hyperspectral image processing and analytical library. The purpose of using Spy is to export spectral data and perform analysis. Spy provides a function to read, display and manipulate spectral data. When working with spectral data, graph animation is generated for a visual representation of the wavelengths, and it is utilized with matplotlib [11]. The program prefers to be executed in fragments so jupyter notebooks are used. The interaction of cells is effective when generating graphs and segmenting data by data. Several other libraries are used such as NumPy, SciPy and cv2.

Furthermore, the Hyperspectral data needs to be recorded and requires a Hyperspectral Imager. For this project, the data is recorded using Specim FX10e spectral imager by Specim Imaging Ltd. The camera needs to be controlled using either prebuilt software or a software development kit. Lumo scanner is used as the software application for this operation. See the section below for more information about the imager operation and connectivity [12] [13].

Chapter 3 – Legal, Ethical, Health and Safety related issues

3.1 Health and safety considerations

Any involved individual’s health and safety is prioritized in this project as per regulatory standards. The whole project is completed by the researcher with his device in computer laboratories at the CSEE department of Essex University and relevant accommodation. The key issues concerning the researcher’s health regarding eye constraints. The researcher has Keratoconus and recently has gone through cross-linking surgery in the past year. Therefore, He has been advised to take regular breaks in between screen time advised by the doctors. Typical schedules include 6-8 hours of sleep a day and 20-minute short breaks every 90 minutes of work.

There is work regarding the recording of the Hyperspectral data involved and requires access to a research lab. The potential issue regarding the issue is predominantly on the type of data (fruits) and hyperspectral imager. There are several risk assessments completed beforehand to access the images and as long the researcher follows the instructions and rules set by the supervisor and technicians, good health and safety are ensured. Here are some issues that may concern and affect the project regarding health and safety when in a research lab environment. Before granting access to the imager, the supervisor agreed terms and questioned all the allergy information of the researcher such as the allergy of fruit and allergy of being sensitive to latex. Contamination of samples used (avocados) in a research lab is a concern. The instruction is to use personal protective equipment (PPE) should be used such as gloves and wipes to sterilize and prevent the spreading of any contaminations. The spectral imager and lab environments are also a concern. The imager has components like Halogen lights attached to the main frame, which can result in burns and Optoelectronics and terahertz lab where the imager is located has worked on radioactive contents. The researcher has been aware of and followed the risk register to ensure safety.

3.2 Legal Considerations

The legality of the implementation of the project must be the researcher’s work except for smaller inspirational ideas. All the materials that the research has created should be independent. He is very aware of this, and it is independent. The researcher acknowledges the materials that he used as research and inspiration. All the sourced materials are correctly referenced. This ensures that this project follows all requirements under the copyright act [14] and avoids academic offences at the University of Essex. The software and hardware tools used in this project are either licensed or open source, making sure they cover all the intellectual property requirements. The imager and the licensing to use the software tools are thoroughly licensed by the CSEE department at the University of Essex and the department has granted the access required. The pre-recorded data that has been used in the initial stage is free to use. The recordings conducted with a hyperspectral imager, which was present at the research lab have limited access and require all laboratory policies to be followed. The research labs are supervised and restrict access outside the working hours of the university. The researcher followed all the guidelines and measures set in the policy to avoid offences and safety. These are basic contracts and agreements between the researcher and the university.

3.3 Ethical Considerations

CSEE projects at the University of Essex focus also on ethical reasoning when a study is conducted regarding logical issues which affect the users (humans). This project does not use any data from human subjects except from the researcher and potential clients who use the product to run the program. However, potential clients who could use this product could be affected by other ethical concerns. There no data set, which is concerning security issues and all research and studies completed have been transparent. This project was never intended to harm any life forms, but rather help to sustain the human population.

Commercial use of Hyperspectral data has very limited understanding in human subjects. Therefore, there is potential for it to be deemed unreliable and unsuitable. A clear demonstration will avoid this issue with successful results. Typically, spectral data are analyzed with deep learning algorithms. Deep learning and AI algorithms are also studied little by humans. Although this project wanted to utilize machine learning, it does not use machine learning analysis. Therefore, this issue is not a concern. When accessing the research lab for recordings, the facility is also a researcher’s responsibility and should always be taken care of.

3.4 social, economic, and cultural considerations

Social and economic considerations must be made of the future and present use of the study and its effects. There are no critical concerns regarding these issues as this is an individual project. The product developed is not commercially made available. Towards the future, if the project is made commercially available to the public, the concerns are related to compatibility and use. Socially, there are disagreements in technology with day-to-day usage. The product will have to go through general regulatory checks and should be deemed safe for use. Economically, this project’s past, present and future has been expensive. This project uses hyperspectral cameras which are expensive pieces of equipment. The University of Essex owns the spectral imager, which the researcher loans to record data on avocados. The data sample was also brought by the researcher due to time constraints when accessing recorded data. Thus, increasing expenditure.

The study in this project was entirely conducted in English. Therefore, the adaptability and understanding of this project may be harder to reach globally. The common understanding is that translation tools are available for the project to be converted to the desired language and understood. This is also the case for the hyperspectral data, the header file defining metadata is in English as well.

3.5 Program rules and standards required by the industry.

The resources used throughout this project as inspiration and research are referenced according to IEEE referencing standards, making it authentic and genuine. All the programs which the project uses are written in Python3 with additional libraries to aid the development process. All programming structurally follows the official Python programming style documentation PEP8 [15] and is written in modular design, increasing the readability of the program. All programs and libraries are contained within a virtual environment for Python to ensure the security of the project and required packages don’t clash with the main computer’s system. Further to ensure a higher level of safety, the Windows subsystem for Linux has been used to contain the directories and environment needed for the project. This is also a researcher’s preference.

The recording of the hyperspectral data also followed the standard procedures. The rules and guidelines for accessing the research lab, where the imager is present, have been followed. The researcher was aware of the environment and wrote down all the control settings before spectral data was recorded. Once the recording was completed, all the evacuation procedures of sanitizing, cleaning, and disposing of wastage were followed.

3.6 Sustainability and environmental considerations

The CSEE department at the University of Essex is a prestigious research institution ensuring the quality and output of the project are at full potential. The quality of education is excellent due to accreditation given by both BCS and IEEE for the researcher's degree to be completed. This project is the final year thesis module, part of the degree to be awarded.

In cost-wise consideration, this project typically involves the hyperspectral imager, which is expensive equipment. Without the data from the imager, the processing wouldn't commence. Pre-recoded data sets are available as free resources, but these data can differ. The imager will consume a larger amount of power. It is a factor of cleaner energy consideration. However, once the data is extracted, the analysis will only require a lower level of energy consumption with little GPU performance. The main power consumption is on the CPU of the computer used for analysis when running Python programs.

This project's primary objective is reducing the wastage of fruits such as avocados by identifying the level of ripeness. These successful outcomes will result in reducing the environmental damage to the world.

Chapter 4 – Main Text

4.1 Hyperspectral Imager Access

The project involved a hyperspectral imager to acquire data on avocados for processing data to identify the level of ripeness through the measurement of reflectance. The imager is owned by the CSEE department at the University of Essex. There was no study conducted with the use of the hyperspectral imager before this project began. The objective is to record enough data at the start of the project using the hyperspectral imager. The formal agreement meeting between the researcher Dr Adrian Clark and the researcher introduced the imager and it was present at the supervisor’s dedicated laboratory before the project. Once the project has been initialized, the first meeting reviewed the clear aims and discussions of the project. During this meeting, the imager was not present in the supervisor’s lab. The researcher has been introduced to the technician and requested access to an imager. The expectation is that the researcher would receive access to the imager the following week and be able to record spectral data upon completing a formal risk assessment. However, this wasn’t the case in this situation.

The following week, the researcher received all the manuals regarding the imager and its software. It was told that it may take a few weeks to access the imager due to the risk assessment and documentation being made. The researcher focused on understanding the manuals that were given and made a study on operating the image. See the section below to understand how the hyperspectral imager is operated.

A month after the wait, the supervisor told the researcher that the technicians wanted to implement further measures on the imager, and it would take time so he was advised to focus on the processing of these imager. At the start, the researcher desired to use a software development kit, which uses c programming to record data. However, due to the time constraints of the project, the pre-installed software is to be used (Lumo Scanner). During term 2 of the semester, the researcher contacted the CSEE Head of school to Fastrack the documentation to gain access with the advice of Dr Vishwanathan Mohan. The request has been a success as imager access has been granted. The supervisor and researcher completed all the formalities to gain entry. However, the timeframe that the imager has been given to access wasn’t sufficient for the researcher to record enough data for deep learning and accurate acquisition of avocados. Also, restrictions of the imager such as the researcher only could access the labs during working hours so data couldn’t be recorded during weekends.

The researcher selected week 29 of the academic year to conduct a study of the environmental factors which affect the data. The researcher had planned to use the imager. However, when the user accessed the research lab facility, the imager that was present at that time was removed as the technician needed to arrange space for the CE301 open days the following week. Therefore, the researcher, requested to have access to the imager before the open day. Arrangements were made by the technician to gain access to do the recordings on Monday of week 30. Therefore, the study is only conducted within a day affecting the accuracy of the results observed.

4.2 Hyperspectral Imager operation and connectivity

The hyperspectral imager used for this project is the Specim FX10e multispectral camera by Specim Spectral Imaging Ltd. Lumo scanner is used as the software to control and acquire the spectral data. This section covers all information needed for the imager to be operated and data to be acquired.

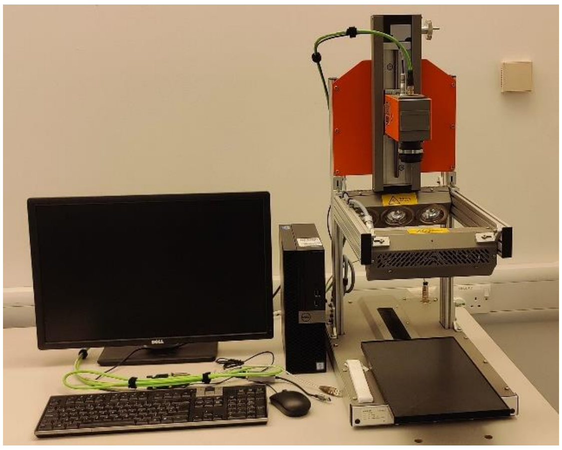

Figure 3 Specim Fx10e

4.2.1 Imager and Connections

The spectral imager consists of several components. There is a base with motors installed and a sample tray above the base. The samples are placed on the tray, then the motor will move the tray to acquire lines. Above this base, the image sensors and set of warmer halogen lights are mounted on a frame. Hyperspectral imagers require the need for more light than regular cameras as artificial illumination is used to drive the separate wavelengths into separate bands.

Each component is synchronized through power connectors. There are two separate cables given to connect the imager to a computer. The operation range of the camera is 400 nm to 1000nm with a data interface using GigE vision to transfer information. GigE uses an ethernet cable to connect to a computer. The other cable is the control cable (super VGAwhich is to control the motors and components. The blinking of LEDs will show the status of the camera.

- LED not lit - power is off.
- LED blinking green is in startup - starting.
- LED stable green - power is on, not recording.
- LED blinking green - power on and recording.

The resolution of the data is determined through the region of interest on which the sensor focuses and the speed of the tray movement. The image is decided by the spectral binning to set the number of bands that need to be acquired. The height of the image is determined by the speed of the tray, which moves below the sensor. For further information on the imager, refer to imager documentation on GitLab and the Specim Fx10e manual [12].

4.2.2 Lumo Scanner

The Lumo scanner is an older version of the pre-built software deployed by Specim. The minimum computer requirements to download the software are an up-to-date CPU, 4 GB RAM with 256 GB storage for data, a 64-bit operating system and a screen resolution of 1024 by 800. The software interface has operational views to control the imager and acquire data.

1. Firstly, the setup view initializes all the setting parameters required before operating the imager. This includes fields such as defining communication links, definition of directories, reference band setup and calibrations. In this view, calibration packages must be inputted.
2. The second interface has an adjustment view for adjusting the data according to the user's preference. The view has 4 widgets to visualize the sensor's view. Detector widgets show the current frame in greyscale, which the sensor is focusing on. The waterfall view shows a live colour image of the region. The live input of the camera could help the user to set the region of interest. The rest of the widgets in the adjusted view are spatial and spectral widgets to show the pixel values and wavelength information with RGB reference points.
3. The final view is the capture view, and it is used to capture data with controls to mitigate the data acquired and its contents. These include setting the naming fields of the data.

Please refer to the Lumo scanner manual for further information [13].

4.2.3 Aquisation of Data

The hyperspectral imager has been used and spectral data of avocados are extracted each day of its ripening cycle. The dataset acquired for this project is day 1-9. The exception is that the data is not recorded during days 6 and 7 as the research facility, where the imager is present only accessible during work hours of 9-5 between Monday to Friday. The data set contains 5 fruits and each fruit's front, and back sides are recorded.

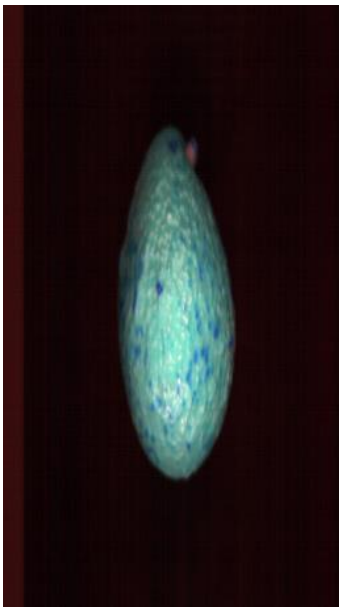

Figure 4 Avocado set 1, fruit 2 [back] (256 x 456)

The quality of the data recorded wasn't accurate as the avocado captured does not fill all dimensions of the image extracted as output, as shown in Figure 4 above. It is a result of the waterfall display widget in the adjustment view of the scanner not functioning. Figure 5 shows the non-functional display widgets of the detector and waterfall as it only shows a white panel instead of the live capture feed. Therefore, an exact calculation and adjustment couldn't be finished, so the data recorded is an estimation.

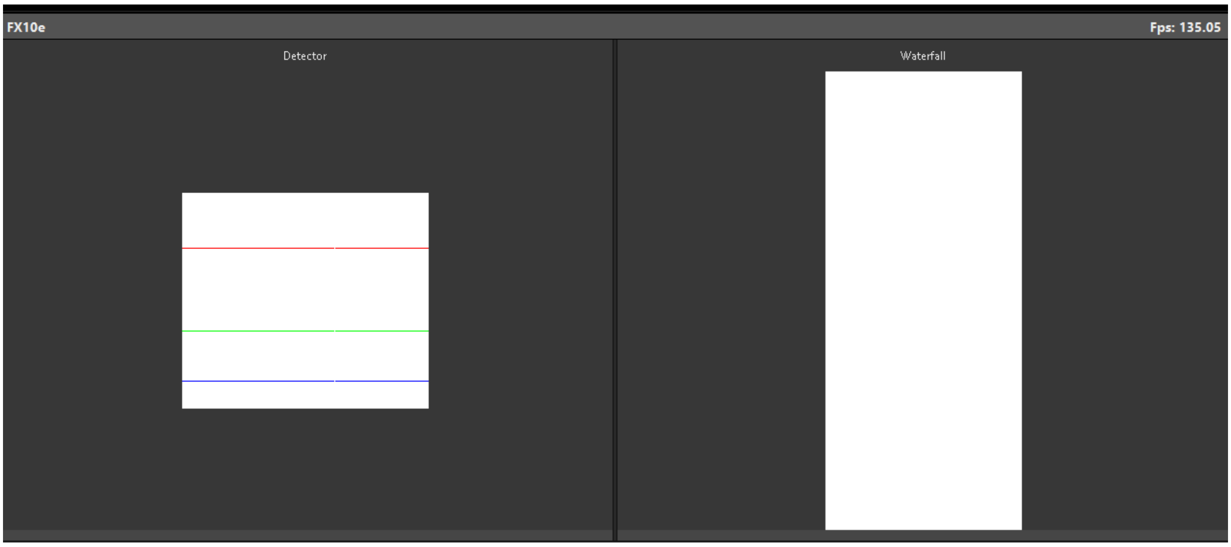

Figure 5 Detector view and waterfall display widgets were not functional.

Here are the Lumo scanner settings used for the avocado data acquired:

Frame rate: 135 fps.

Exposure time: 3.20 milliseconds

Spectral binning:2

Spatial binning: 4

Clock used: Internal.

Field of view: 90 mm

With these value settings, an estimate of the image's metadata can be observed. The maximum spectral band available is 448. The spectral binning combines spectral bands and reduces the observable data size. The number of bands the image has is 448/2 = 224. The spatial binning determines the sample per line (horizontal axis). In spectral data, the x-axis (horizontal) refers to the number of samples and lines refer to the y-axis (vertical) pixels. the maximum sample limit of the imager is 1024 and according to this calculation, the number of sample pixels is 1024/4 = 256. The time in which the imager operates is around 3.4 seconds as observed. The speed determines the lines. In this instance, the speed is controlled through the field of view. The calculation for the number of lines is the frame rate multiplied by the time of the recording. Here the number of lines is 135 X 3.40 = 459 pixels. These calculations provide the important metadata of the image of resolution and the number of bands.

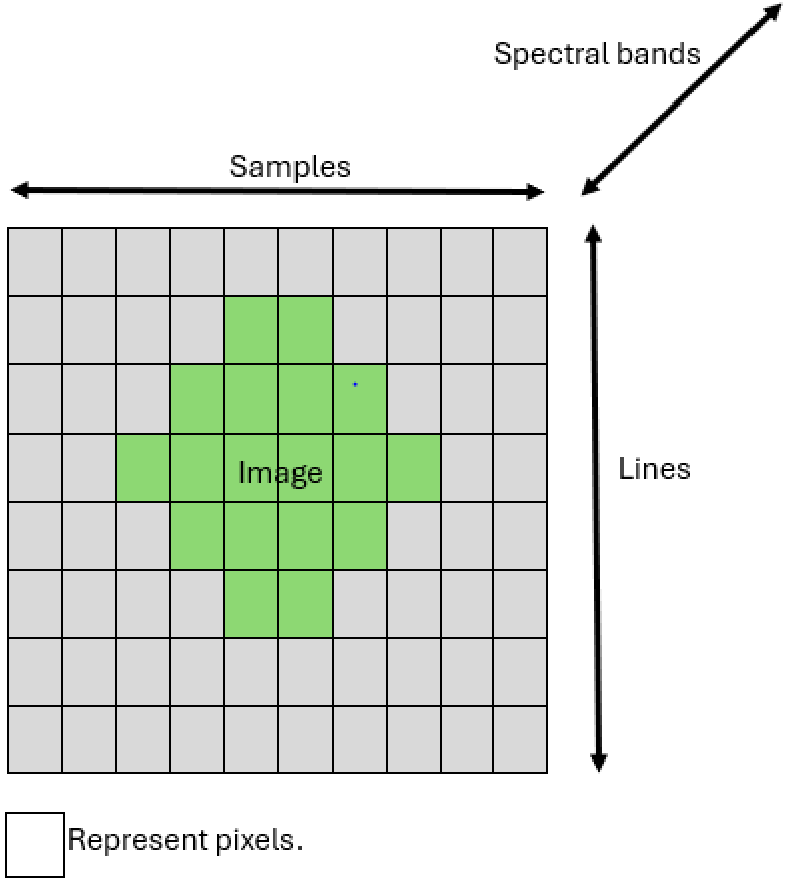

Figure 6 spectral resolution canvas.

The RGB reference channels are set between the capture range of the image for it to be converted to RGB. The reference band values for set 1 are {56, 144, 196}, with these values, the shift in wavelength can be observed across spectral bands. The temperature of the spectral sensor is vital to have a clear output of the data. Therefore, keeping it cooler will not produce any certain blurring within the data.

4.3 Phase 1 – Visual Representation of Hyperspectral Data

The initial stage of the project primarily focused on understanding hyperspectral imaging concepts and visually representing hyperspectral data using spectral Python (Spy) for further analysis in phase 2 of the project. These include experimentation of displaying data and observing a certain pattern in wavelength for later production of the processing algorithm. The spectral data used during the initial stages are pre-recorded and the set images used are the mangos due to the lower file size around 15 megabytes, making the program run faster. The resolution which the images of the mangos is 64 X 64. The latter stages used the avocado data for analysis and processing, which was in higher resolution.

4.3.1 Exporting data into Python using spectral python.

Before displaying spectral data, the raw file and the header file must be read into the Python program. The header file (.HDR) contains the metadata is first read as it holds the spatial and spectral information of the image. Then the binary data is read based on the information gathered from the header file and placed into a 3D matrix. When importing the spectral data into Python, a clear path declaration must be made for these data files and the following command reads the data.

```
Image = envi .open ("header file path" , "raw file path").
```

The processing and efficiency of the data read could be improved by the load() method at the end of the file reading, which loads information into a NumPy array. The certain read_band loads a single band data into an array and the read_bands read multiple channel information to an array. Also, regions of the image can be read. For more information, refer to the spectral Python notes [10]

4.3.2 Experimentation

The experimentation with the techniques in spectral python developed a certain level of understanding of how data can be mitigated and made ready for processing. Experiment.ipynb file has been made available in Git, which shows a step-by-step guide based on mango hyperspectral data. Towards the start of the file, it shows hyperspectral data being read and displaying information being processed. There are several examples of displaying types of images. These include the image, image selected to several bands, image at a specific band and ground truth images. The ground truth image is a very interesting set of data to observe as the images show the true colour reference classification of the image, projecting the inside contents of the fruit. There could be a pattern observed towards the center of the mango seed on moisture content as the green class labels represent moisture levels and it reduces as the fruit ripens resulting in the decrease in the green label as shown in Figure 7.

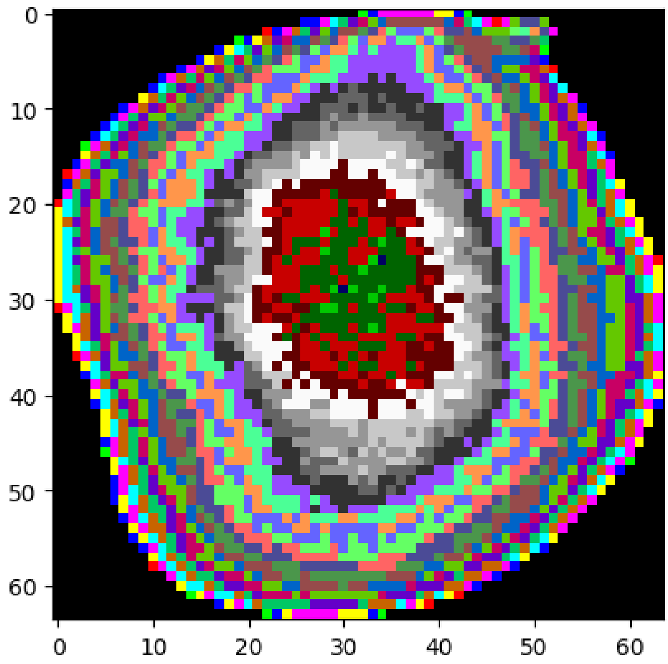

Figure 7ground truth image of mango at day 1.

At the end of the experimentation, it shows a graph-based representation of the image data. Typically, the graph created would be an observation of a pattern in which there is a change in the reflectance level of pixels across the spectrum.

4.3.3 K-means and graph generation

When plotting wavelength information, the values extracted in this study are the mean values of the spectral bands and represent the reflectance of the fruit. During Phase 1, this is obtained through K-means. However, K-means are not used during processing as it was not deemed efficient.

K-means is an algorithm which groups values together into clusters. Typically, it is used for faster computability of programs by reducing the larger set of data. Grouping of data into clusters based on the distance between the data point to the cluster [16]. In spectral Python, the same concepts were used to obtain the mean value of the image. K-means is included in SPy library as a function kmeans( ). Within this function, the spectral image, number clusters and iteration of the samples are given as parameters. The algorithm randomly assigns the number of clusters within the image and then the closest pixel values are assigned to the cluster centroid in iteration. Then the mean of the cluster is figured out using NumPy. The return of the function is two NumPy arrays containing band information and cluster points. Here is an example of how the K-mean calculation can be called:

```
image = spectral data

(bandinfo, cluster) = kmeans(image, 10, 20)
```

The pseudo-code above shows that K-means is performed for images with 10 clusters. The number of iterations is 20 per cluster.

After the extraction of the mean values, it is plotted as a graph showing the wavelength using matplotlib. The pseudo-code here shows how it can be performed.

```
Import matplotlib as plt

plt.figure()

plt.xlabel(“Bands”)

plt.ylabel(“refeltance”)

for i in range (bandlegth):

      plt.plot(cluster[i])

plt.plot()
```

After observation of the data, the limits of the graph can be set with plt.ylim(). Figure 8 shows the output of a graph generated using K-means of mango.

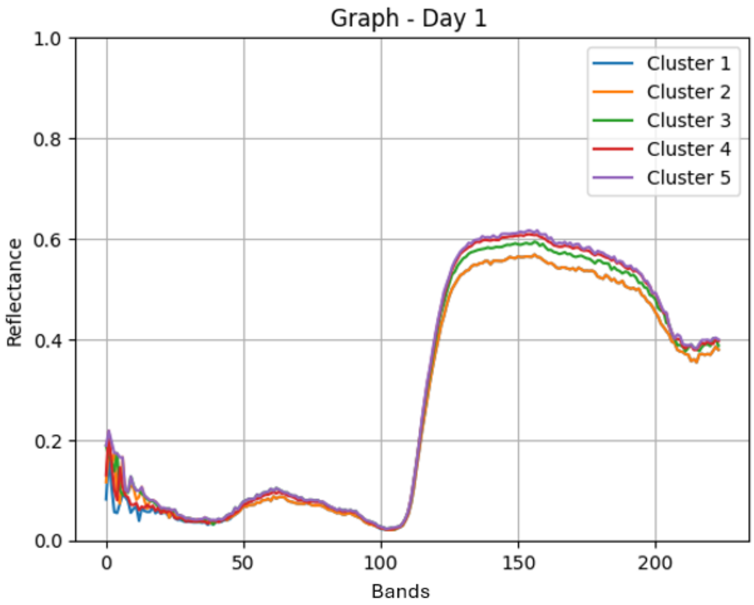

Figure 8 Mango spectral graph Day 1 of Fruit 1

The graph above shows the first data observed from the mango set during phase 1. It has 5 clusters and between the ranges of 0 to 1. In this graph, the hike in reflectance is seen between the band 100 to 224. The expectation is that there will be a downfall in the reflectance as the mango ripens. A visual must be made to observe the change from day 1 to 10. This is where the animations were implemented. The minimum and maximum range of the images in the mango set is between 0 – 0.6 truncated to 2 decimal places on average.

4.3.4 Matplotlib Animation Framework

To visually identify the change in reflectance, an animation in the form of a GIF is generated using the matplotlib animation framework. It is the first form of processing when measuring the ripeness. To make a GIF in matplotlib, utilize the FuncAnimation framework which draws subplots and iteratively updates the main plot with the subplots generated. Then pillow writes the output in GIF format [17]. Here is the pseudo-code for creating the animation.

```
Def draw_ani (frame):
      Perform extraction values.
      Plot values.

Graph, sp = plt.subplots()
Frames = specify number
Animation = FuncAnimation (Graph, draw_ani, Frames,
inerval time)
Animation.save ("path", writer= "pillow")
```

When the animation was created, the pattern observed showed that the reflectance level saw a fall as the days progressed. This is visible between bands 100 – 224. At the start of the mango life cycle, the maximum reflectance level was above 0.6 and by day 10, the value saw a fall to 0.4 at band 150. This representation made clear the researcher can perform processing to identify ripeness. Although K-means with animations showed a successful outcome in results, its approach wasn't preferred as the mean calculations can be conducted for the entire image across bands rather than K-means calculating the mean for only a certain value of the image. For the mango animations, please refer to Gitlab.

4.4 Phase 2 – Data Processing

Phase 2 consists of creating a processing algorithm to identify the ripeness of avocados. The previous phase used mango data. However, the key objective of the project is to use avocados. During the start of this phase, the data sets used were pre-recorded and it required good storage management as the file size of avocado data sets is larger due to the resolution of the images being higher. The resolution of the images is in the ranges between 270-290 samples and 320-340 samples. The minimum and maximum pixel values for these sets are between the ranges of 0-0.5, the same as the mango data used earlier. However, this differs from the data recorded with Specim FX10e during this project. The latter stages conducted processing on the independently recorded data. The average maximum and minimum pixel values for those datasets are between 200 – 800. See the data acquisition section for information on the data recorded. This is one of the factors that affected and prevented deep learning study not to being conducted as the data is varied. However, the same rules are applied when processing these data, therefore the difference was not an issue with thresholding.

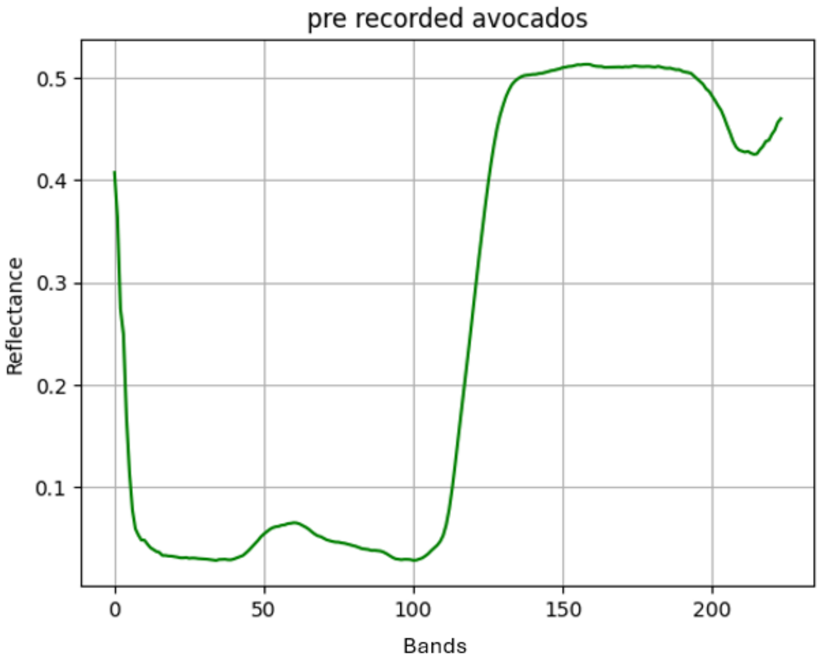

Figure 9 spectral value plot of Prerecorded data.

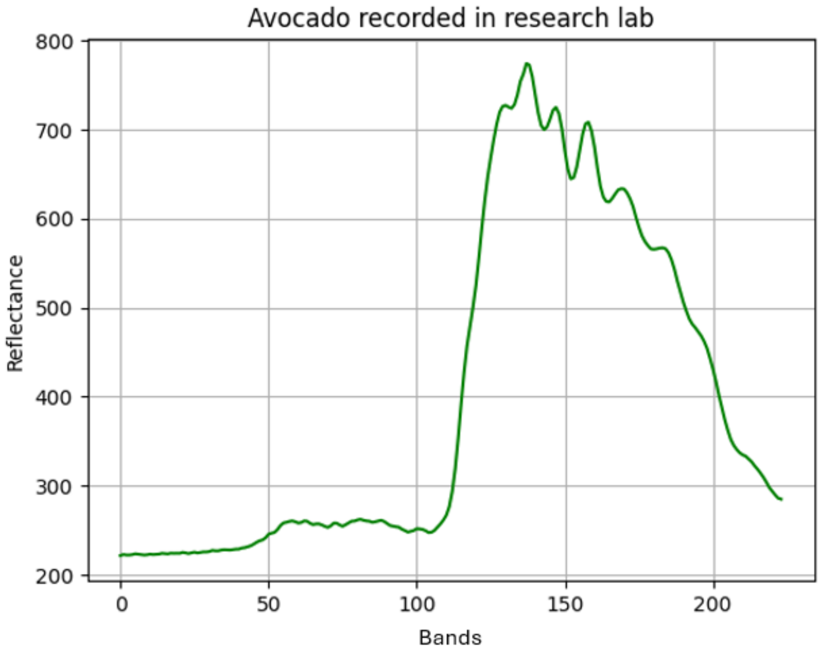

Figure 10 spectral value plot independently recorded data by the researcher.

4.4.1 Matching Regions

The main problem between each of the day recordings is the deviation of avocados. This is due to the handling of the fruit each time when is recorded. The researcher might move them, affecting the accuracy of the results. Also, measuring the ripeness across the entire image wasn't wise as the background could force false values to be generated. Therefore, an efficient way of measuring the data should be made only focusing on certain regions of the imager, primarily on the object (avocados). There was consideration of a feature used the matching regions between images such as matching regions with SIFT or cross-correlation.

The first experiment is made, trying to match a region between two regions of avocados using cross-correlation. Cross correlation measures the similarities within two separate data, where one data is sifted slightly. In this case, avocado is sifted during recording between days. The application used in this experiment uses correlate2d of SciPy. Here is an example pseudo code to perform cross-correlation [18].

```
import scipy.signal import correlate2d

img1 = open and load the data 1 of a band
img2 = open and load the data 2 of a band
#define the region to target
pixelX = number within the image range
pixelY = number within the image range
size = size to create the region

target = img1 [pixelX: PixelX+size , pixelY: pixelY+size]

bestmatch =0
bestlocation = None

# Iterate over the rows and coloums of the imager 2
for x in (img2.shape[0] - size):
      for y in (img2.shape[1] - size):
```

```
                    # define image2 target region
                    target2 = img2[x:x+size, y:y+size]

                    #calaculate the corrattion
                    Corraltion = correlate2d(target, target2,
        mode='valid')[0, 0]

                    # Update best match
                    if Corraltion > bestMatch:
                bestMatch = Corraltion
                bestLoction = (x, y)
    # plot the best match using matplotlib
```

This is an example of how cross-correlation is performed, for the full demonstration please refer to GitLab. The results gathered between day 1 and day 2 of a set at the 200th band show cross-correlation isn't efficient when matching regions. The best match is not a clear match, and this could be because the surface of the avocado is similar in colour and cross-correlation underperforms in finding the shift when the matches are similar. The figures below show the matched regions in avocados. There are several factors which could affect this such as the tilt in avocados, the resolution, and the potential shift in wavelength as the day progresses making it difficult to match regions. Therefore, a better approach should be implemented to segment the avocados.

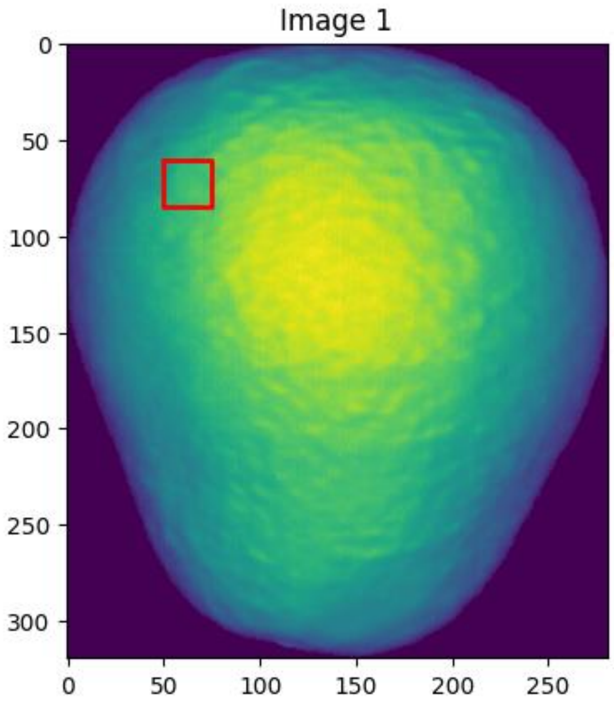

Figure 11 Image 1 with the region of interest.

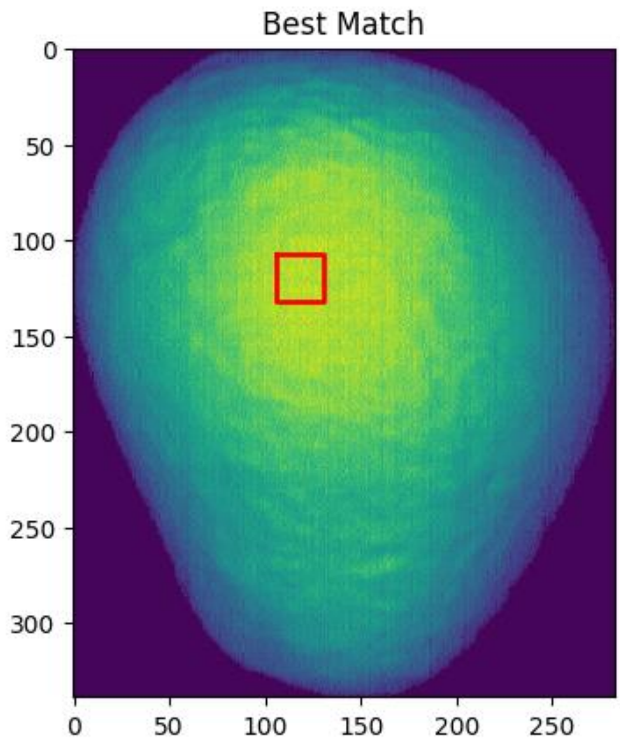

Figure 12 Image 2 with the best matched region.

4.4.2 The Use of Masks

Masking the foreground, which is the avocado from the background is a better approach to segmenting and processing the image. This technique also makes improvements to the program as all values of the avocados within the image are averaged to produce reflectance. There were different methods for producing masks of the image. The masks are created using OpenCV routine cv2.threshold which takes in greyscale image and threshold value to return a mask. The different method of producing masks is based on the calculation of the threshold value. The threshold value is used to segment the foreground from the background.

The first calculations of the threshold are calculated using Otsu's threshold [19]. It involves iterating through all pixel values within the image array and determining whether the value falls under the background or foreground. The problem with implementing otsu thresholding is the value range error created when loading the data into a NumPy array where the spectral data is not an 8-bit unsigned integer. Therefore, before performing ostu threshold, convert the band data into an unsigned integer. The following pseudo-code will convert and perform ostu threshold.

```
im = np.asarray(single_band_data* 255 , dtype='uint8')

ret, mask = cv2.threshold(im, 0, 255,
cv2.THRESH_BINARY+cv2.THRESH_OTSU)
```

Another, approach to creating the mask is calculating the threshold with a Harris Conner detector, which detects corners of a region. In this case, the detector will find the corners of the avocado and process bounding box values to create the mask. Once the bounding box is created, the mean intensity will determine the threshold value for cv2 to create the mask. The following code computes the threshold using the Harris corner detector.

```
im = single band data
# default values as Parameters for corner detection
blockSize = 2
ksize = 3
k = 0.04
# finding corners from corner harris
corners = cv2.cornerHarris(im, blockSize, ksize, k)
# Find bounding box around the corners
non_zero = np.where(corners > 0.01 * corners.max())
x, y, w, h = np.min(non_zero[1]), np.min(non_zero[0]),
np.max(non_zero[1]) - np.min(non_zero[1]), np.max(non_zero[0]) -
np.min(non_zero[0])
avo_reg= im[y:y+h, x:x+w]
# Find the mean of the intensity
mean_intensity = np.mean(avo_reg)
threshold_value = int(mean_intensity)
ret, mask = cv2.threshold(im, threshold_value, 255, cv2.THRESH_BINARY)
```

The code above is the Python implementation of the opencv Harris corner detection [20].

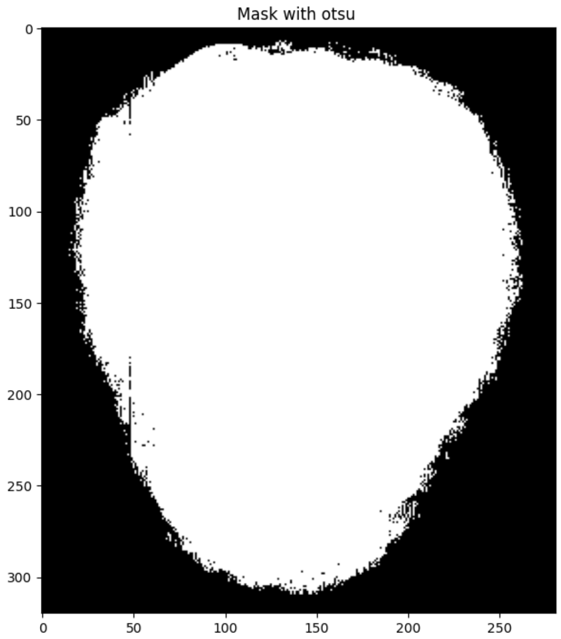

Figure 13 Mask of avocado with otsu's threshold.

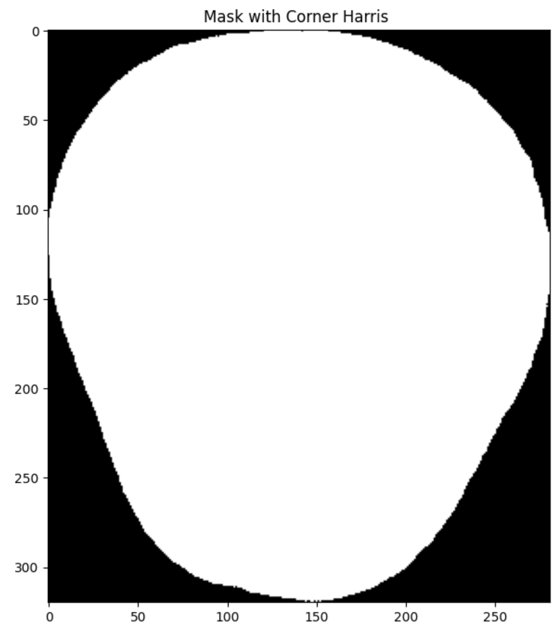

Figure 14 Mask of Avocado with Harris corner detector's threshold.

Figures 13 and 14 show the masks created with different methods which calculated the thresholds. The mask produced for the same avocado image shows that the mask generated from otsu's threshold values creates a greater range of noise compared to the mask created with the Harris corner detector's mean intensity as the threshold. The noise created in Otsu's calculation is likely to be from certain regions of the foreground's (avocado) surface and is similar in colour to the background. This results in some classification during thresholding to be masked out. However, the Harris corner detector's mean intensity only detects corners of the avocado and forms points around the avocado, masking the entire avocado. This observation made clear that the mask generated from the Harris corner detector will be used during processing.

4.4.3 Mean value calculations

Once masks were produced for the avocados, the next stage of the processing involved calculating the reflectance level. One suggestion was to determine a single band of the data, in which an observable downfall in the mean value can be seen. However, the argument is that, by looking at a single band of data, accuracy in the avocado's reflection might be a concern due to anomalies of mean pixel values for that certain band of data. Even though this approach will reduce the run time of the program as other bands were not used for processing, the processing algorithm might not be efficiently working. Therefore, the researcher calculated the mean value of the image in each band and averaged out the entire set of mean values across the spectral bands. Masks were used to only calculate the mean of the avocados (foreground) across the 224 bands. The computer program followed the following protocol to calculate the mean of the hyperspectral image.

1. Create a mask of a sample band for the mean calculation of avocados.
2. Apply the mask and calculate the mean for each band within the hyperspectral data.
3. Calculate the average of the mean values extracted.

In the case of the project, the extracted average for each day is the mean reflectance and it will show a declining pattern.

4.4.4 Avocado animations

To figure out which range in bands with the data a pattern can be observed, animations are used. The animation framework created is the same implementation of the animation with mangos during phase 1 of the project. Firstly, the animations showed that there was a difference in the pixel values between the pre-recorded data and the researcher's data as mentioned above. Towards the end of processing, the need for pre-recorded data was not needed and the researcher used the data recorded from the Specim Fx10e at the University of Essex. The wavelength generated with the set of avocados showed that there was a downfall in the reflectance between the spectral band of 0 -100. However, the pre-recorded data showed a different pattern as there was a decline in reflectance between the band of 100 – 224. This meant that the two data sets were in contrast. It may be a result of different settings, in which the data sets are captured. The animations are available within the researcher's GitLab for the whole set of avocados. The following graphs are the first and last days of avocado fruits captured. The animation does not include visual representations for days 6 and 7 as the days are not recorded.

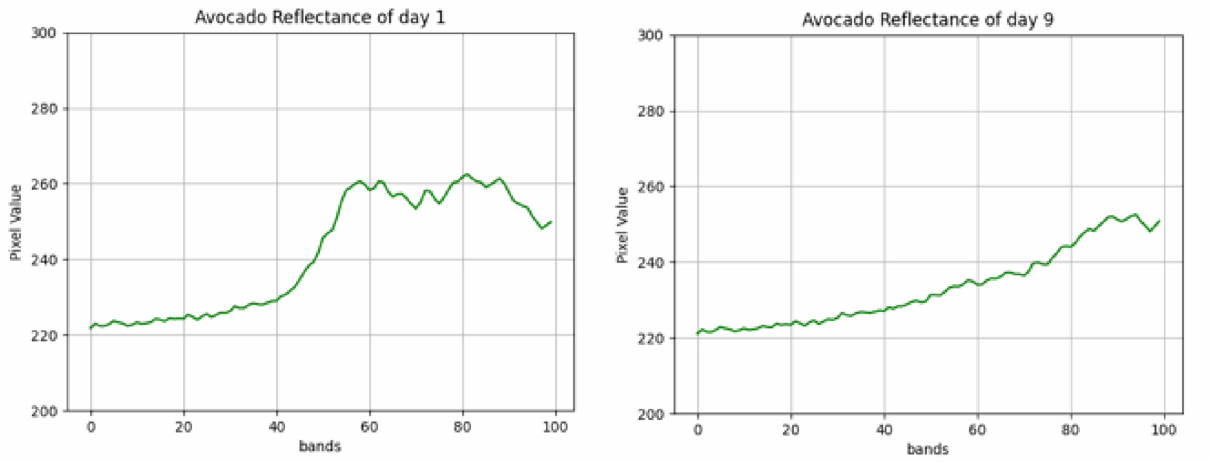

Figure 15 Avocado graphs of fruit 1 back side.

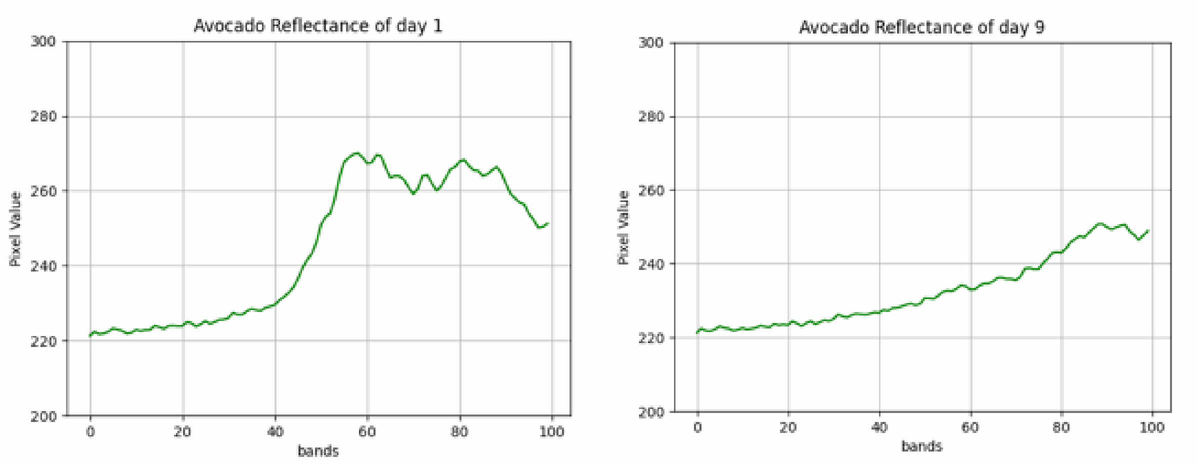

Figure 16 Avocado graphs of fruit 3 front side.

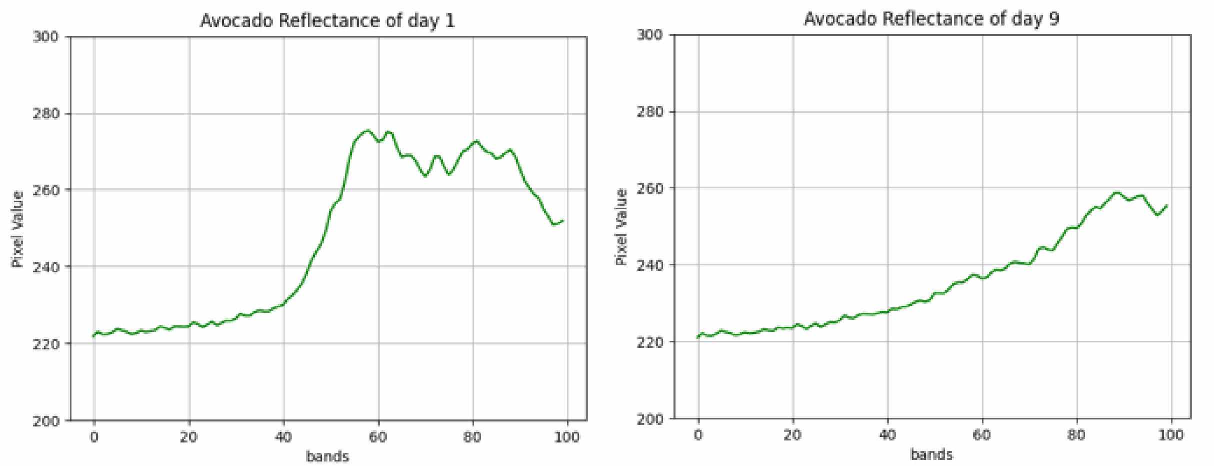

Figure 17Avocado graphs of fruit 4 back side.

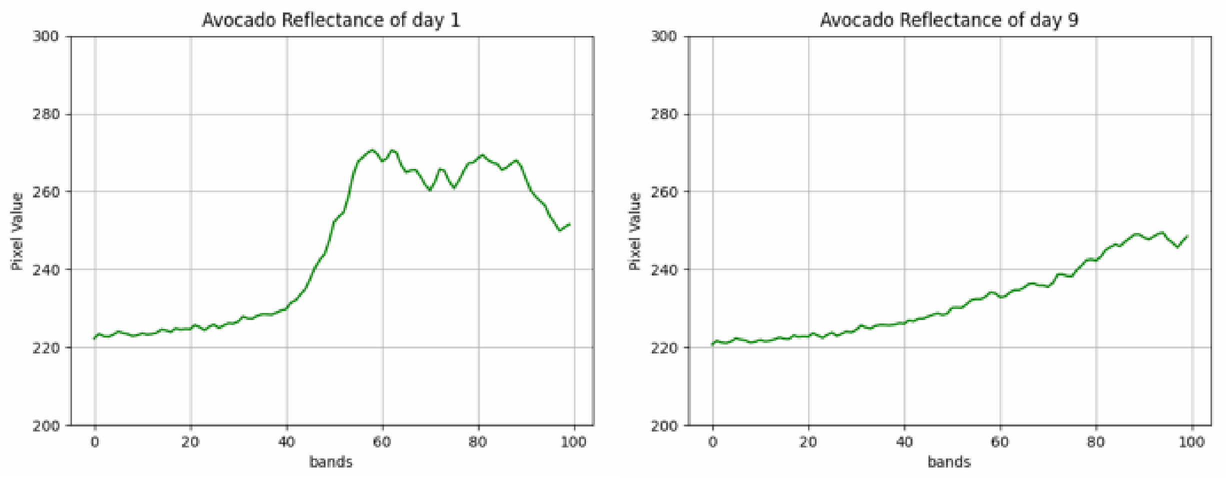

Figure 18 Avocado graphs of fruit 5 front side.

The observations made clear that avocados between bands 0 -100 see a downfall in reflectance and the pattern of all fruits have similar pixel values. Narrowing the spectral bands even more will make precise measurements of the mean between data sets. From band 100 to 224, the values, the observed pattern does not show a unique pattern, but rather a confusing pattern. Some data sets showed a hike in value and sometimes the reflectance remained the same or dropped slightly. Typically, the surface pattern of avocados shifts to a warmer tone. This means that bands which are close to the infrared spectrum see an increase in pixel values and the reflectance levels of channels closer to ultraviolet dropped significantly. In this instance, band 0 is closer to ultraviolet and band 224 is closer to infrared. As avocados ripen, it absorbs UV radiation at higher levels to produce vitamins. Therefore, avocados reflecting UV light are minimal during the last stages of their maturity. This could conclude why the bands closer ultraviolet spectrum had a decline in pixel value.

4.4.5 Thresholding

After the observation of data through animation and the extraction of an accurate average of the mean across bands between data sets, thresholding is implemented to segment the data sets into the three stages of maturity, which the avocado goes through. These are unripe, perfectly ripe and overripe. The thresholding in this program is called within the animation framework and the terminal prints the output of data with mean value and maturity status. Although, in a real working environment, this would be different. The average mean values generated is 200 – 275 between bands 0 and 100. Thresholding is a simple command which uses if statements to determine the status according to the average mean value. These are some of the terminal outputs gathered from the program. For the full set of results, please refer to GitLab.

```
day 1 MEAN reflectance: 246.3888655090332 Status: Unripe.
day 2 MEAN reflectance: 245.64269577026369 Status: Unripe.
day 3 MEAN reflectance: 241.57880126953125 Status: Unripe.
day 4 MEAN reflectance: 240.87230773925782 Status: Perfectly ripe.
day 5 MEAN reflectance: 240.23869232177734 Status: Perfectly ripe.
day 8 MEAN reflectance: 235.2064306640625 Status: Overripe.
day 9 MEAN reflectance: 235.17894348144532 Status: Overripe.
# Avocado fruit 1 Back side
```

```
day 1 MEAN reflectance: 241.6311671447754 Status: Unripe.
day 2 MEAN reflectance: 240.91874710083007 Status: Perfectly ripe.
day 3 MEAN reflectance: 238.26942810058594 Status: Perfectly ripe.
day 4 MEAN reflectance: 238.34311279296875 Status: Perfectly ripe.
day 5 MEAN reflectance: 236.5416554260254 Status: Perfectly ripe.
day 8 MEAN reflectance: 235.05789489746093 Status: Overripe.
day 9 MEAN reflectance: 233.24847076416015 Status: Overripe.
# Avocado fruit 1 Front side


day 1 MEAN reflectance: 244.61593963623048 Status: Unripe.
day 2 MEAN reflectance: 243.63191192626954 Status: Unripe.
day 3 MEAN reflectance: 239.65690322875977 Status: Perfectly ripe.
day 4 MEAN reflectance: 237.02306442260743 Status: Perfectly ripe.
day 5 MEAN reflectance: 236.5262351989746 Status: Perfectly ripe.
day 8 MEAN reflectance: 233.86727935791015 Status: Overripe.
day 9 MEAN reflectance: 232.60194747924805 Status: Overripe.
# Avocado fruit 3 Front side


day 1 MEAN reflectance: 243.35680557250976 Status: Unripe.
day 2 MEAN reflectance: 242.8214602661133 Status: Unripe.
day 3 MEAN reflectance: 242.4616456604004 Status: Unripe.
day 4 MEAN reflectance: 239.5559098815918 Status: Perfectly ripe.
day 5 MEAN reflectance: 238.57110580444336 Status: Perfectly ripe.
day 8 MEAN reflectance: 233.3743391418457 Status: Overripe.
day 9 MEAN reflectance: 235.39611282348633 Status: Overripe.
# Avocado fruit 4 Front side
```

This thresholding shows the expected results of the project. The average mean calculation is accurate enough to determine the maturity. However, the data recorded are store-bought avocados, individually picked for the unripe fruits. There could be a mixing of avocados during the packaging of the fruits cultivated on different days. Typically, avocados ripen for a week and every 3 days the maturity state changes. This could be one reason why some days when the fruit is ripened early.

4.4.6 Anomalies and difference in data

Anomalies on a thresholding algorithm cannot be avoided for most of the programs when thresholding is used. There are some datasets of avocados that show faulty results. Certain avocados have higher reflectance levels, and they fall gradually back to normal levels as the days progress. This is an issue due to the inaccuracies in data. The main reason is inaccuracies within the avocados themselves and the recording decisions. These are some of the anomalies within avocado processing.

```
day 1 MEAN reflectance: 247.52310134887696 Status: Unripe.
day 2 MEAN reflectance: 249.36359100341798 Status: Unripe.
day 3 MEAN reflectance: 244.67507598876952 Status: Unripe.
day 4 MEAN reflectance: 239.94313674926758 Status: Perfectly ripe.
day 5 MEAN reflectance: 239.488212890625 Status: Perfectly ripe.
day 8 MEAN reflectance: 235.050018157959 Status: Overripe.
day 9 MEAN reflectance: 236.44203125 Status: Perfectly ripe.
# Avocado fruit 2 Backside segments day 9 as perfectly ripe even
though it is overripe.
```

```
day 1 MEAN reflectance: 246.93408767700194 Status: Unripe.
day 2 MEAN reflectance: 247.67748123168946 Status: Unripe.
day 3 MEAN reflectance: 246.41232391357423 Status: Unripe.
day 4 MEAN reflectance: 243.26483154296875 Status: Unripe.
day 5 MEAN reflectance: 242.51825271606447 Status: Unripe.
day 8 MEAN reflectance: 234.15928100585938 Status: Overripe.
day 9 MEAN reflectance: 235.44706146240233 Status: Overripe.
# Avocado fruit 4 Backside remains unripe for most days.
```

The accuracy of results can be better handled during recording and processing. There are several ways to make the program more accurate. The first way is by reducing the sample size of bands, in which the mean values are calculated. Currently, the mean values of the data are averaged between the bands 0-100. By reducing this range to 0-75, the program is likely to produce better results. Secondly, stricter measures could be placed to avoid any mix-up of data. This also includes eliminating any unusual avocados. The environments in which the data were recorded could also affect the information gathered from the sensor. Further to avoid inaccuracies, machine learning could be used to cast out unusual data.

4.5 Thresholding vs Deep Learning

This project used a simple thresholding to process hyperspectral data to measure the maturity of avocados. This project could have benefited from machine learning. However, time constraints regarding accessing the imager, restricting to have smaller recordings of avocados and machine learning's requirements to have a larger set of data made it not to be used. Both thresholding and machine learning have several advantages and disadvantages. The table below lists a comparison between thresholding and machine learning [21] [22].

Table 1 Thresholding Vs Machine Learning

Thresholding	Machine learning
Simple to use and implement. These also reduce the cost and time.	Complex to implement but very easy to use. It is very expensive and requires a lot of time.
Can be used with fewer data sample, with human intervention. However, handling a larger set of data is a struggle.	Requires a larger set of data as the training set. It avoids the interventions of human intervention, making the output more reliable.
Accuracy can be an issue when there is a human intervention and when handling a larger set of data.	Accuracy is always improved with larger training data.
Require low storage management as the data handling is live.	Requires large storage to store space for program and training sets.
Little security and privacy issues as data is not stored.	Security and privacy are a concern due to the larger set of data.

Chapter 5 – Evaluation of data from the imager across environments.

The final objective of this project is to evaluate the data recording from the hyperspectral imager under different environments. During the study of the evaluation, the researcher faced an imager access issue. This issue is also addressed in the hyperspectral imager access section. There were certain factors that affect hyperspectral data during recordings like lighting, temperature and the sample being used. As the evaluation is short due to issues, this study focused predominantly on the lighting of the environment and the sample used for recording.

Firstly, the hyperspectral imager used in this experiment (Spacim Fx10e) records information under two spectrums of halogen and sunlight. Halogen spectrum outputs of the images are recorded from the Halogen light source, which is mounded to the body frame of the camera. This light source emits heat affecting the bands, which are significantly closer to the infrared spectrum. Therefore, a typical pattern in which the data is observed will be a representation of bands closer to infrared being higher in pixel values than the bands closer to the ultraviolet range. Sunlight spectrum on the other hand relies on the sun's lighting source making a balanced pattern across the wavelength.

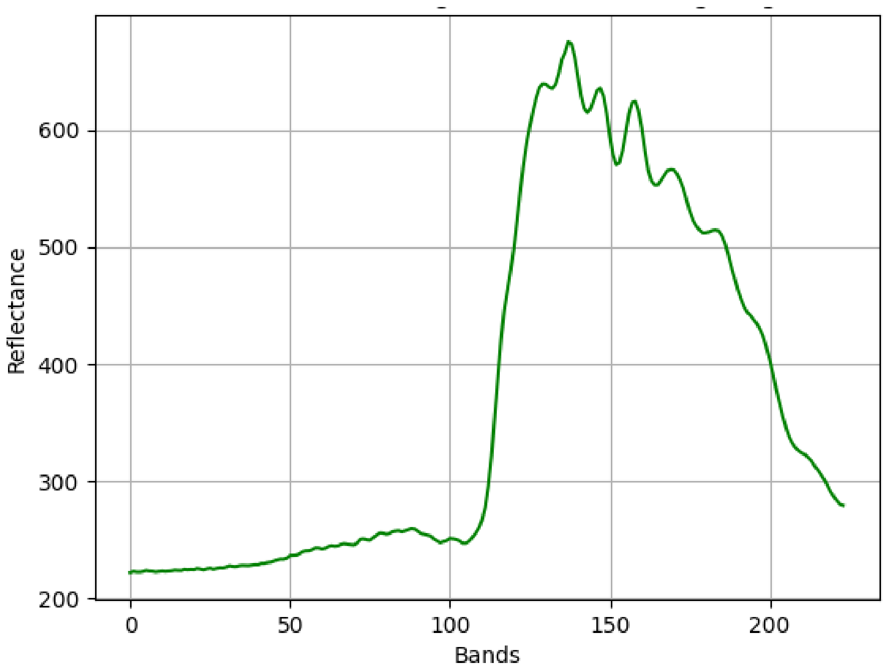

Figure 19 Avocado measured with Halogen light and High room lighting.

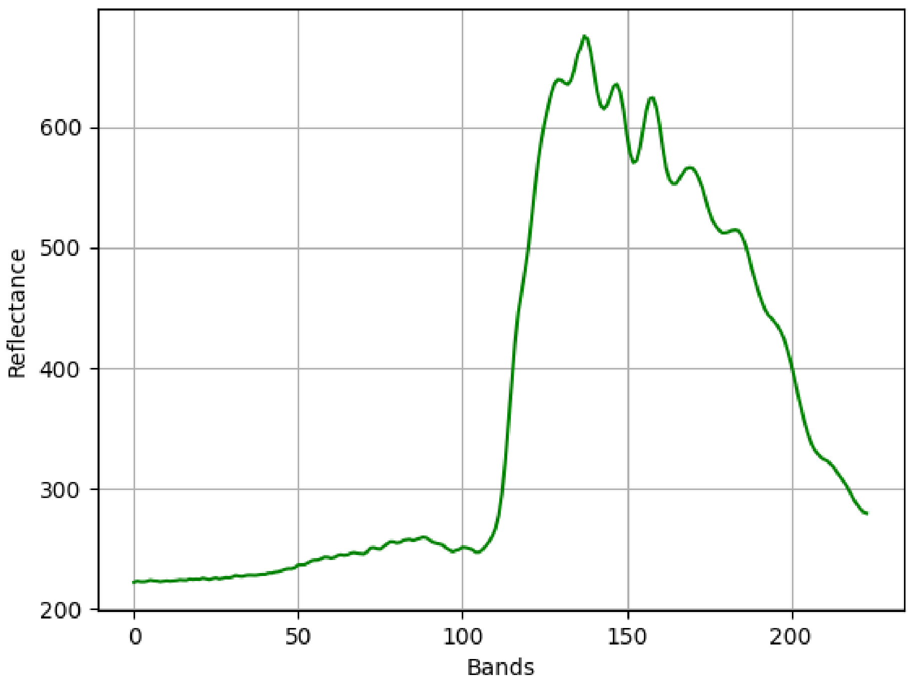

Figure 20 Avocado measured with Halogen light and Low room lighting.

Figures 19 and 20 show the representation of avocados with halogen light source. The only difference between these figures is the additional external source of light. Figure 19 used normal room lighting while figure 20 didn't use any external light source. Mean average is performed for both sets of avocados in this condition. The values generated in figure 19 is 369.81 (2 DP) and for Figure 20 is 369.74 (2 DP). The results suggest that the reflectance values are similar and states that external light source will not affect the data. Halogen lights are stronger and will prevent any alteration made by the external light source.

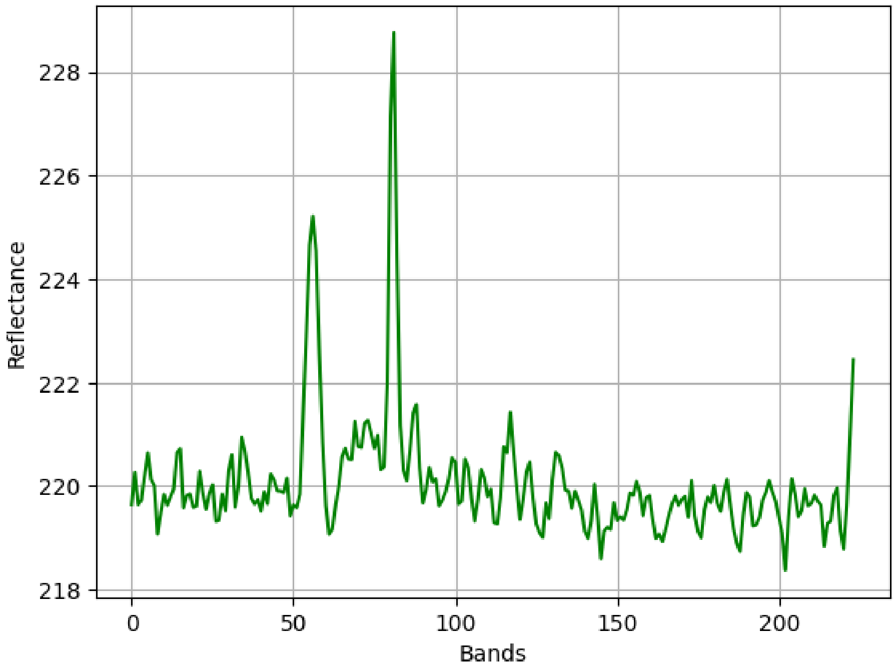

Figure 21 Avocado measured with only sunlight.

Figure 21 shows an output of avocado which only used the sunlight during recording. The halogen light source is turned off from the imager by the power supply from the base being removed. The average mean value for the pixels across the bands is 220.06 (2 DP). This value is significantly lower than figures 19 and 20. This is due to the intensity of the light source. Sun light is a natural source, and the intensity of room light varies across days. During this evaluation, the wet weather blocked majority of the sunlight meaning the recording is conducted is with slightly darker environment. The researcher only had access to imager for only one day to study environments, therefore observations are not made on another bright day. This may have affected accuracy in results. However, the representation in figure 21 shows even spread of reflectance across bands except for few larger hikes. These hikes may represent optimal RGB reference bands. This conclude that sunlight source is an unreliable light source during recording as the data can change depending on the intensity.

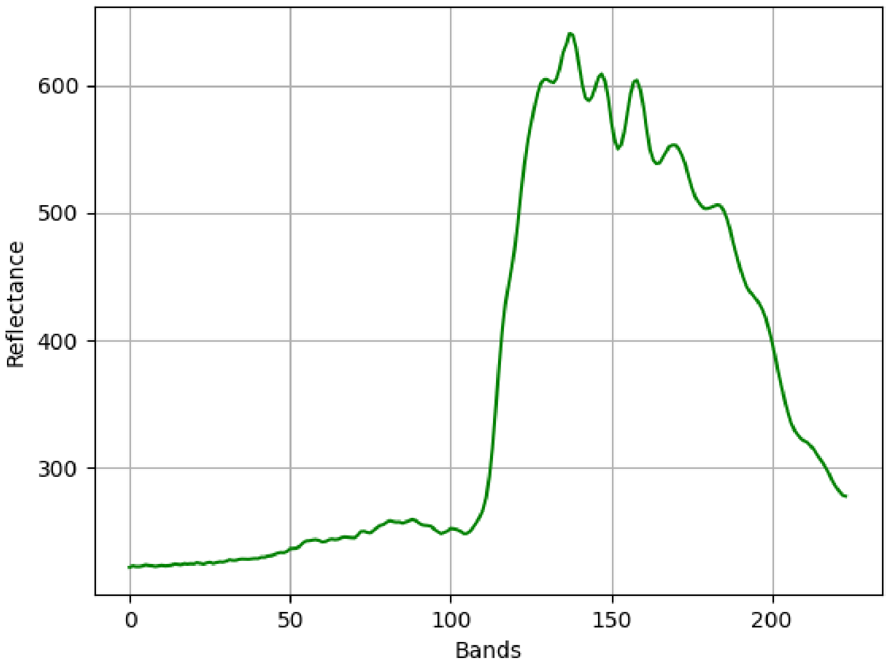

Figure 22 Bruised Avocados measured with Halogen lights and High room lighting.

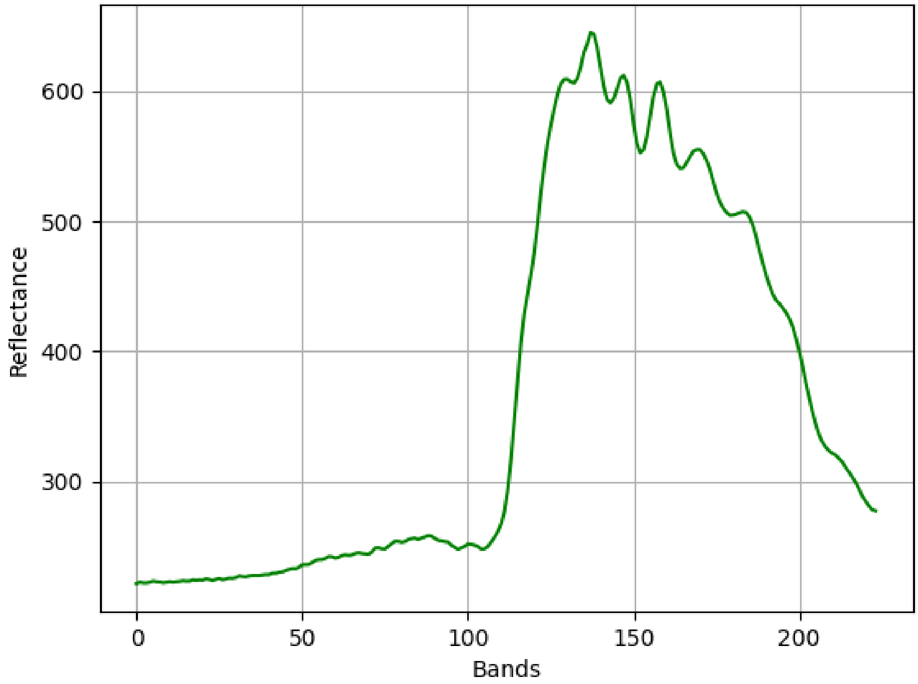

Figure 23 Bruised Avocados measured with Halogen lights and Low room lighting.

The researcher also conducted study on the samples recorded. The study is based on weather damages to avocados affect the representation and able to be identified. Certainly, avocado conducted in the previous evaluations is specifically damaged by researcher to have bruising. The expectation is that there should be a slight change in reflectance. The avocados were measured with halogen light source and figure 22 also had an external source of room light. Figure 23 is measured with no external light source. the average mean values program generated is 362.17 (2 DP) for figure 22 and 363.00 (2 DP) for figure 23. This result showed a decline in reflectance level of avocado when it is damaged and recorded under halogen lights. Although, there is a decline observable, it is not a larger decline as the average pixel level is only drops by 6. The graph representation is having similar structure to figure 19 and 20.

Several other factor could have also changed the data such as temperature and resolution. Resolution affects the data as the number of pixels being averaged. The higher the resolution of the image, the clearer the data recorded and information in pixels. However, the larger resolution will increase the size of the data, requiring a larger storage. Temperature affects the data by producing more infrared radiation. The more heat surrounds the recording, will affect bands closer to infrared range. In this Evaluation, regular room temperature is applied. The camera setting could also affect the data as each setting in RGB references, exposure setting, and the speed of the tray movement will affect the data.

With this evaluation, it can be concluded that the best environment in which the imager to be used. The imager should be using halogen light source and measure data with a regular room temperature. The size of data should be kept to a reasonable resolution around 250 to 300 samples and 300-400 lines. Avocados used for measuring should be in good conditions and have no damages during lab scanning.

Chapter 6 – Conclusion

6.1 conclusion of the Work

As mentioned in the prelude, the project implemented a vision system to identify the maturity of fruits, in this case, avocados. There we two phases to this project. Phase 1 enriched the knowledge and understanding of spectral imaging to the researcher. This stage created a visual representation of hyperspectral data in the form of an animation framework which is used throughout the project to observe and perform processing. There are certain features such as K-means to measure the reflectance is proved undesirable when building a vision system in this case. Therefore, a better approach is developed to extract the average to plot and process data.

The approach in phase 2 is focused on building a certain vision system. The first objective is matching regions of avocados between the data sets. The first experiment on matching regions with cross-correlation proved that it is difficult to match regions between the fruit as the surfaces of avocados are similar in texture, so the correlation fails to find the best possible region. This meant that the use of masks was required to separate avocados from the background. The Harris corner detection is effective in creating masks with lower levels of noise compared to otsu's method. The mean values are extracted for each band, and they are averaged to produce an average mean value across bands. These values are the reflectance which is processed using thresholding. There was a successful outcome when measuring the maturity of avocados.

This is a time-concerning project as it had issues regarding the access of the imager. This is due to the imager not being used for the most part and requiring a lot of risk assessments and documentation. After a strong appeal to the technicians by the researcher to provide access to the imager, the imager access is granted. Data has been recorded using the imager. These recorded data differed from the pre-record data set provided to use during the waiting period for access. This is due to the different settings and acquisition processes in which the data is recorded. An evaluation of the data has been made of the imager's best optimal way to record data. It is suggested that the imager should record data with a halogen light source.

6.2 Future work

There were several processes and objectives changed during this project due to issues. If this project is conducted without these issues, the outcome of this project is even better. The project shows promise in hyperspectral imaging as a technology of the future. In an ideal real-world environment, manufacturing factories of fruit may use spectral imagers in a conveyor belt to identify the maturity status of the fruit and package fruits accordingly.

This project will provide a strong understanding of the spectral imaging concepts and vision systems that could be implemented. This project needs to be improved in the future in several factors. Firstly, the operation of the imager and acquiring an accurate dataset. This understanding of the imager should be better for good processing. Accuracy in the vision system is a must as it is the first line of operation when developing a commercial product. The project could benefit from deep learning algorithms. This approach may be the backbone of future technologies and this project lacked the time and datasets required to study using deep learning models. If there is a commercial objective of this project, there should be a plan regarding the design of a commercial robotic model to package fruit in a factory environment with the inclusion of hyperspectral imagers.

Chapter 7 – Project planning

The main concern with this project is less availability of accurate resources. There wasn't enough research based on hyperspectral imaging and its commercial use in agri-food technology. A project should have competition for a better future. The challenge of this project is on where to start. With the guidance of the project supervisor, the researcher implemented these plans to implement a process algorithm.

The researcher made analysis and planning before this project began and worked according to this plan even with the risk of not accessing hyperspectral imager for most of the project. An agile design process has been used to direct the project towards the end. Kanban board is drawn in Jira and issues are created as epics, tasks, and subtasks to fragment the project. This made the project simpler. Weekly meetings were arranged and held between the researcher and the supervisor. All programs are backed up and stored in GitLab dedicated to the researcher. These contained the documentation of research material, programming content and results.

7.1 Risk analysis

The researcher made a risk register covering all the risks and solutions of the project during challenge week as a requirement for all projects under CE301 and submitted it to the project portal and the supervisor. The were no further risk requirements for the progression of the project. However, to gain access to the hyperspectral imager, there was the requirement of risk assessments made for the imager and the lab facility by the technicians and the researcher signed and understood all the risk assessments made and agreed to the contract. The risk register made by the researcher is available via GitLab and the imager risk register is available on request.

7.2 Project progression

As mentioned above, weekly meetings are held with the supervisor, and he reviews the progress of the project. During weekly meetings, the supervisor gave feedback on the certain implementation of the task and directed the researcher on the next steps that should be implemented. The supervisor during meetings gave technical advice to the vision system developed. He is also the personal tutor to the researcher and reviewed any issues the researcher faced outside of the project.

In Jira, the researcher plans the issues and reviews their status frequently. The kanban board design includes the prior creation of epics, which are the objectives of the project. The epics have tasks and subtasks which are to be completed. To efficiently perform tasks, priority has been set to tasks and the highest priority tasks are completed first. Some epics were pushed back to a later date for completion as the project faced delays. The risk is reported and observed. This plan is very clearly observed through the number of issues and diagrams. The following figures are obtained to show that there has been continuous management.

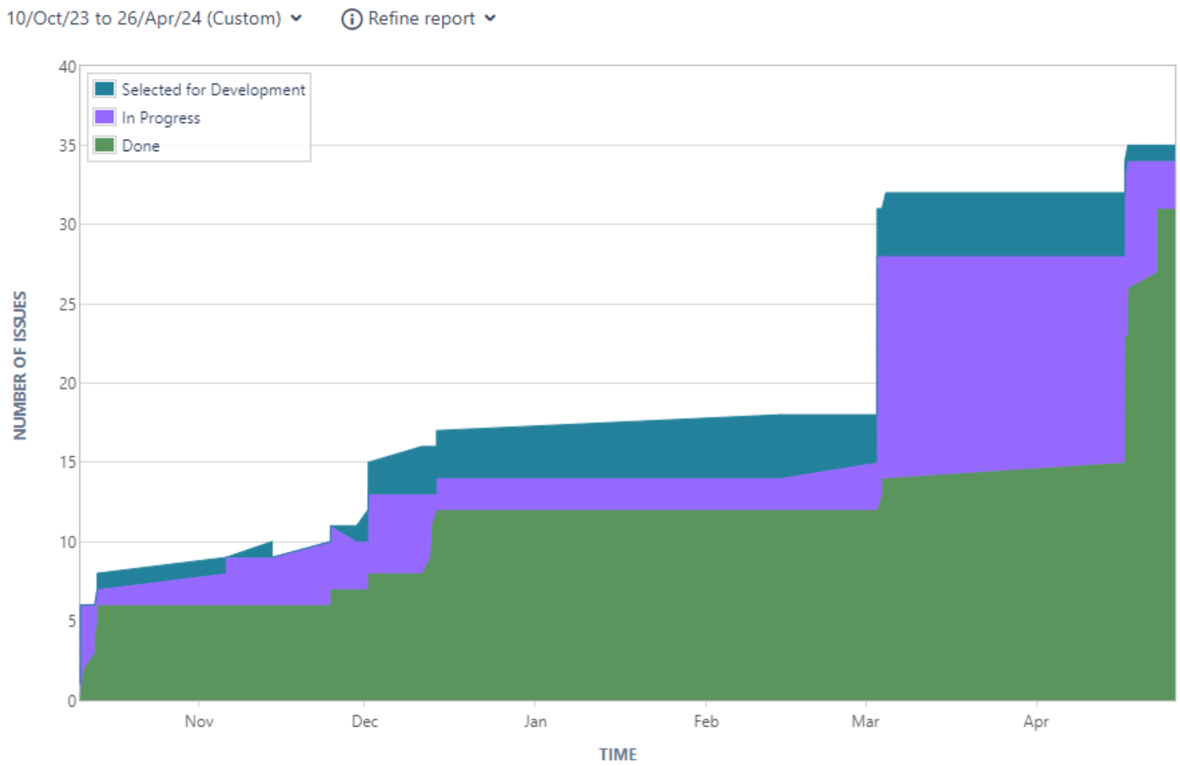

Figure 24 Cumulative flow diagram showing progression.

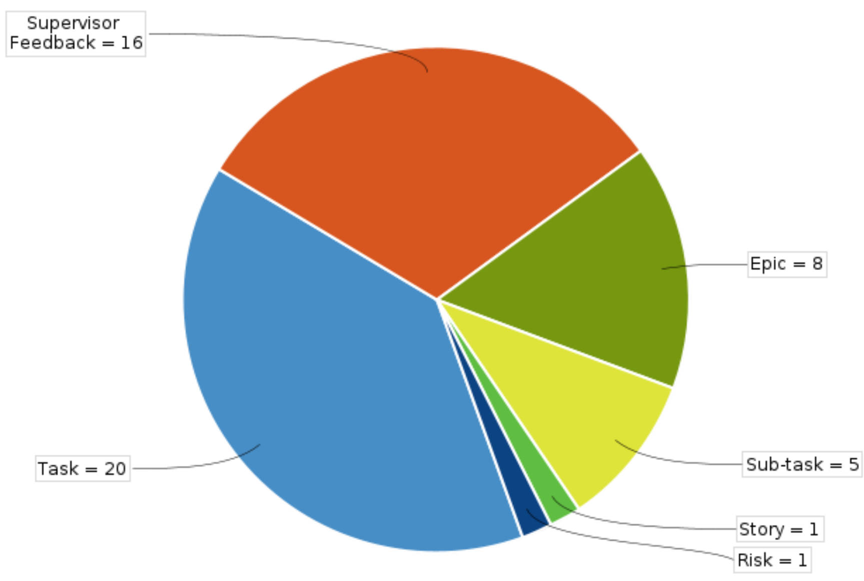

Figure 25 Pie chart showing the types of issue created.

7.3 The review of the project

The planning and progression of this project made the researcher understand vision systems and techniques applied very well. This is done through the modularity of the created tasks which are implemented during the project. The research layout is designed well so that it adapts to changes within the project. This is a positive sign of the agile processes working.

The project focused on reading for the first few weeks and the next epic of using an imager to record data was pushed back. The suggestion is that to focus on processing has been approached by the researcher and programming began with research and experimentation of features for the vision system. There is the analysis of features, and the best and most accurate techniques were selected for processing. Programming codes were developed to adapt to any changes in the data set. Building a processing algorithm isn’t a straightforward approach as it has research and review for better outcomes of results.

References

[1] Maarten Elferink and Florian Schierhorn (April 7th, 2016) "Global Demand for Food Is Rising. Can We Meet It?" available https://hbr.org/2016/04/global-demand-for-food-is-rising-can-we-meet-it

[2] Specim Imaging Ltd "WHAT IS HYPERSPECTRAL IMAGING: A COMPREHENSIVE GUIDE" available https://www.specim.com/technology/what-is-hyperspectral-imaging/

[3] Prediktera AB, "Description of hyperspectral file format" available Prediktera Evince Breeze hyperspectral file format.pdf

[4] MathWorks , Inc. "Getting started with hyperspectral image processing" available https://ww2.mathworks.cn/help/images/getting-started-with-hyperspectral-image-analysis.html

[5] Sushma Barma, Sumanjali Damarla, Sudheer Kumar Tiwari- November 2020 "Semi-Automated Technique for Vegetation Analysis in Sentinel-2 multi-spectral remote sensing images using Python"

[6] Leon Amadeus Varga, Jan Makowski, Andreas Zell - Apr 2021 Germany "Measuring the ripeness of Fruit with Hyperspectral Imaging and Deep learning"

[7] Climacteric fruits versus non-climacteric fruits by Agrotonomy (May 7, 2023) available https://agrotonomy.com/climacteric-fruits-versus-non-climacteric-fruits/

[8] OECD (2020), Avocados, International Standards for Fruit and Vegetables, OECD Publishing, Paris, available https://doi.org/10.1787/44a6e19b-en-fr-es.

[9] Avocado Ripening and Handling (19th February 2023) available https://www.catalyticgenerators.com/blog/avocado-ripening-and-handling

[10] Welcome to Spectral Python (SPy) — Spectral Python 0.21 documentation. [online] Available at: https://www.spectralpython.net/

[11] Matplotlib (2012). Matplotlib: Python plotting — Matplotlib 3.1.1 documentation. [online] Matplotlib.org. Available at: https://matplotlib.org/.

[12] Specim fx10 – user manual 2.2 , Specim, Spectral Imaging Ltd.

[13] Lumo Scanner – user guide 1.2, Specim Spectral Imaging Ltd.

[14] UK Copyright Service. (1998). UK Copyright Law. Copyright, Designs and Patents Act the UK Copyright Service. [online] Available at: https://copyrightservice.co.uk/copyright/uk_law_summary#:~:text=The%20Copyright%2C%20Designs%20and%20Patents%20Act%201988&text=It%20gives%20the%20creators%20of.

[15] van Rossum, G., Warsaw, B. and Coghlan, N. (2001). PEP 8 – Style Guide for Python Code [online] peps.python.org. Available at: https://peps.python.org/pep-0008/.

[16] A. Sachan Title: "Understanding K-means Clustering in Machine Learning" URL: https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1

[17] GeeksforGeeks. (2020). Matplotlib.animation.FuncAnimation class in Python. [online] Available at: https://www.geeksforgeeks.org/matplotlib-animation-funcanimation-class-in-python

[18] docs.scipy.org. (n.d.). scipy.signal.correlate2d — SciPy v1.13.0 Manual. [online] Available at: https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.correlate2d.html

[19] Adrian F. Clark. Computer Vision: Lecture notes for CE316 and CE866 (chapter 5.2 Low level vision). University of Essex, UK, 2021.

[20] docs.opencv.org. (n.d.). OpenCV: Harris Corner Detection. [online] Available at: https://docs.opencv.org/4.x/dc/d0d/tutorial_py_features_harris.html.

[21] OF THE BENEFITS AND THE DRAWBACKS OF THE IMAGE SEGMENTATION THRESHOLDING TECHNIQUES Available https://www.researchgate.net/figure/OF-THE-BENEFITS-AND-THE-DRAWBACKS-OF-THE-IMAGE-SEGMENTATION-THRESHOLDING-TECHNIQUES-18_tbl1_353940574

[22] www.javatpoint.com. (n.d.). Advantages and Disadvantages of Machine Learning - Javatpoint. [online] Available at: https://www.javatpoint.com/advantages-and-disadvantages-of-machine-learning.

This reference introduces the concepts of spectral imaging and enriches throughout the project.

Randall B. Smit January 2012 “Introduction to Hyperspectral imaging” available https://www.microimages.com/documentation/Tutorials/hyprspec.pdf