Heriot-Watt University

F20PA Final Deliverable

Investigating The Performance of Multi-Scale/Kernel CNNs with Attention at Analyzing Hyperspectral Images

Author:
Ahmed Sherif

Supervisor:
Dr. Hadj Batatia

A thesis submitted in fulfilment of the requirements for the degree of BSc (Hons)

in the

School of Mathematical and Computer Sciences

August 2024

Student Declaration of Authorship

Course code and name:	F20PA Research Methods and Requirements
Type of assessment:	**Deliverable**
Coursework Title:	Deliverable 1
Student Name:	Ahmed Sherif Ahmed Moussa Abdelfattah
Student ID Number:	H00340434

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature *(type your name):* *Ahmed Sherif*

Date: *24/11/2023*

Abstract

Hyperspectral Imaging (HSI) has become more popular in the field of precision agriculture and food quality inspection due to its non-destructive procedure. This study investigates the performance of multi-scale and multi-kernel CNN with and without attention mechanisms at classifying hyperspectral images from the deepHS-Fruit datset. While pre-built architectures exhibit superior robustness and performance, their computational cost exceeds the processing capabilities of commonly available emedded systems. This limitation effectively hinders the deployment of these models in real-time tasks. This study aims to develop lightweight models whilst maintaining the same performance. In this study we propose two new architectures that implement multi-scale (MSDCNN) and multi-kernel (MKDCNN) concepts respectively and we take it a step further by integrating an attention mechanism (MSDCNNA and MKDCNNA). The models are then evaluated on aforesaid dataset achieving the following results. As a baseline a random forest(RF) algorithm achieved an accuracy of 74.74%, 70.69%, 56.45%, and 61.84% on each of the following subsets respectively: Avocado VIS, Avocado NIR, Kiwi VIS, Kiwi NIR. In contrast HSCNN achieves 48.42%, 44.82%, 38.71%, and 42.11% in said order of subsets. Our proposed models achieved the following results: MSDCNN scores 55.8%, 44.82%, 37.63%, and 42.11%, whilst MKDCNN scores 66.32%, 44.82%, 35.48%, and 44.74%. Adding the attention layer we are presented with the following results: MSDCNN scores 37.37%, 24.14%, 29.03%, and 28.95%, whislt MKDCNN scores 64.21%, 44.82%, 37.63%, and 43.42%. The interprertation of these reults are discussed and some recommendations are proposed.

Acknowledgements

Firstly, I would like to thank my supervisor Hadj Batatia for suggesting the topic, for sharing his knowledge and expertise, and for his guidance and support throughout the project.

Also, I would like to express my genuine gratitude to my parents who have provided me with endless sincere support and encouragement throughout my university study.

Contents

List of Figures

List of Tables

Abbreviations

HSI	**H**yper **S**pectral **I**maging
VIS	**VIS**ible range
NIR	**N**ear **I**nfra **R**ed
SWIR	**S**hort **W**ave **I**nfra **R**ed
PLSR	**P**artial **L**east **S**quares **R**egression
PCA	**P**rincipal **C**omponent **A**nalysis
ANN	**A**rtificial **N**eural **N**etwork
CNN	**C**onvolutional **N**eural **N**etwork
AE	**A**uto**E**ncoder
GAN	**G**enerative **A**dversarial **N**etwork
RMSE	**R**oot **M**ean **S**quared **E**rror
R^2	**C**oefficient of **D**etermination
MoSCoW	**M**ust **S**hould **C**ould **W**ould
SPA	**S**uccessive **P**rojection **A**lgorithm
SFS	**S**equential **F**orward **S**election
SBS	**S**equential **B**ackward **S**election
SVM	**S**upport **V**ector **M**achine
DT	**D**ecision **T**ree
RF	**R**andom **F**orest
KNN	**K**-**N**earest **N**eighbour
MLP	**M**ulti **L**ayer **P**erceptron
ReLU	**Re**ctified **L**inear **U**nit
ELU	**E**xponential **L**inear **U**nit
LOOCV	**L**eave **O**ne **O**ut **C**ross **V**alidation

Symbols

f	frequency	Hz
λ	wavelength	nm

Chapter 1

Introduction

Nowadays there is a high demand for enhancing the tools for fruits inspection, due to heavy daily consumption. Experiments (Li et al., 2017, Lu et al., 2017) have showed that inspection based on chemical usage is considered a destructive method. Moreover, chemical inspection and assessment is limited due to the adopted operations and procedure (Lu et al., 2017). Hence introducing a non-destructive fruits inspection procedure will be considered more secure for the customers. Recently it has been proven that Hyperspectral Imaging (HSI) is a great non-destructive technology for fruits inspection and agriculture analysis (Li et al., 2017). Figure 1.1 captures the trend of HSI in food science over the last 20 years (Vignati et al., 2023).

The first usage of HSI was during the 1980s, in different sectors of remote sensing. This includes agriculture, forests as well as atmospheric and climate changes (Lu et al., 2020). In the 1990s, HSI technology showed promising results in food science (Gowen et al., 2007) (Li et al., 2017), and in chemical imaging for post-harvest studies (Nicolaï et al., 2007).

1.1 Motivation

Human inspection (for fruits and vegetables) is basically constrained in terms of operator speed and subjectivity, as well as the small quantity of product acquired (Li et al., 2017, Lu et al., 2017). While chemical analysis approaches are destructive, time-consuming, labour-intensive, often unrepresentative, and unsustainable (Li et al., 2017, Lu et al.,

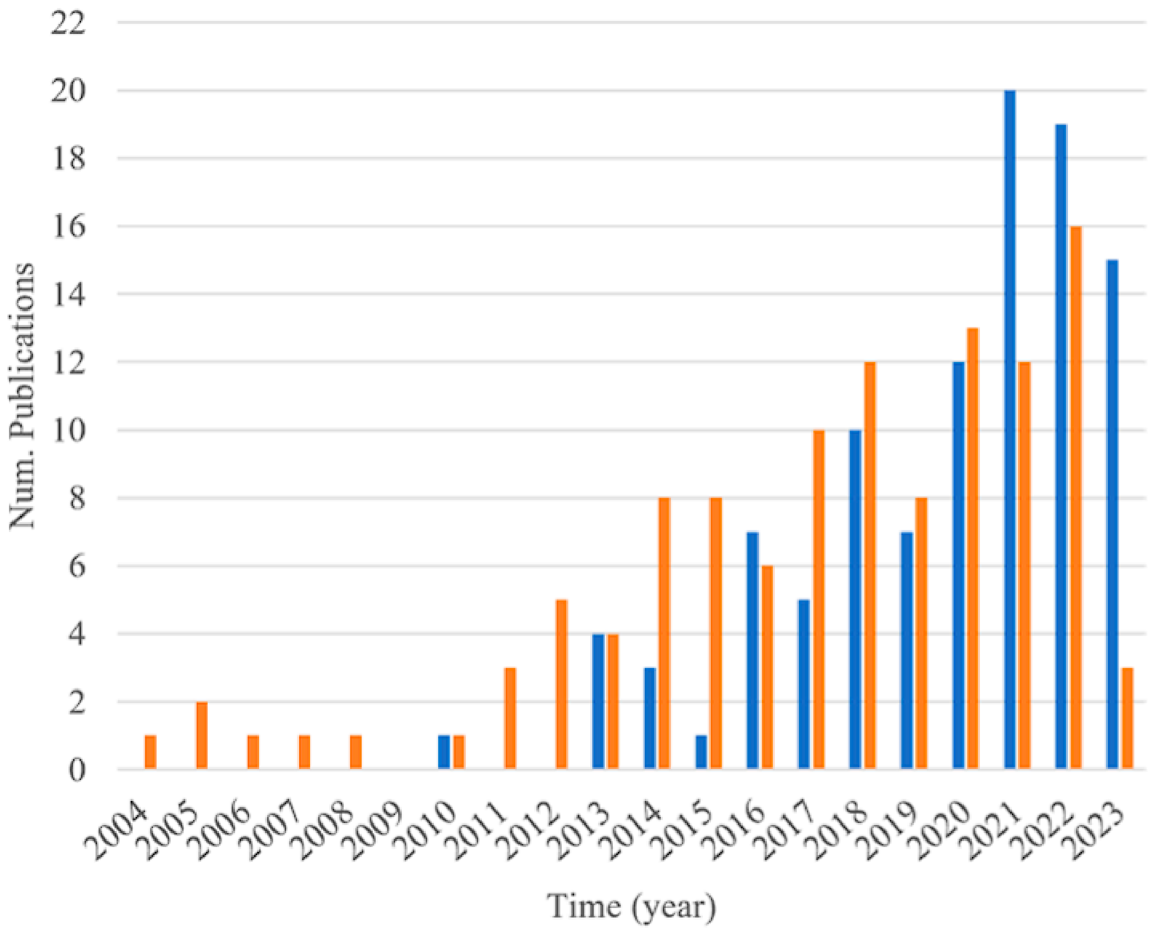


FIGURE 1.1: Literature review of the last 20 years using the keywords "hyperspectral imaging", "fruit and vegetables", "quality", "fresh-cut", and "postharvest" (Vignati et al., 2023)

2017). Over the past two decades, significant progress has been achieved in the development of non-destructive sensing technologies for the automated quality assessment and safety inspection of agricultural and food goods (Li et al., 2017, Lu et al., 2017). Although spectroscopy is regarded as a potent instrument in this context, it offers a punctual measuring approach that is incapable of accurately analysing a heterogeneous mass of materials (Lu et al., 2017). As a result, companies are looking for ways that combine the process benefits of spectroscopic techniques with the inclusion of spatial information. This method is also known as hyperspectral imaging. HSI has been one of the most researched and implemented methods in non-destructive monitoring systems for the fruit and vegetable supply chain during the past two decades.

1.2 Aim & Objectives

Many current applications utilise pre-built architectures such as ResNet and VGGNet to analyse HS images of fruits. However, these models are computationally expensive to train and require a large amount of data. This is an obstacle to be reckoned with in this field of research, given that procuring such data is laboursome and time-consuming. In addition, these models are too big to be deployed on embedded systems for real-time applications. The main aim of this dissertation is to develop lightweight models that can be deployed on embedded systems for real-time applications whilst maintaining the same performance as the pre-built models. This aim can be achieved through the following objectives:

1. Explore the current contribution of machine learning algorithms in HSI for fruits inspection.
2. Evaluating the performance of SVM, DT, RF, KNN, and CNN models on a dataset of hyperspectral images of Avocado and Kiwi.
3. Integrating and evaluating the concepts of multi-scale and multi-kernel convolutions on the deepHS-Fruit dataset.
4. Integrating and evaluating attention mechanisms on the deepHS-Fruit dataset.

The main contribution of this research can be summarised as follows. We develop two new architectures that implement multi-scale and multi-kernel concepts and integrate attention mechanisms. We evaluate these model's performances on the deepHS-Fruit dataset and compare them to the HSCNN model. From the results we discover the effectiveness of the MKDCNN and MKDCNNA on the deepHS dataset where it outperformed the HSCNN model on 2 out of 4 subsets and matched the performance on 1 subset. Similarly the MSDCNN model outperformed the HSCNN model on 2 out of 4 subsets and matched the performance on 1 subset, which is not a significant improvement. Adding an attention mechanism did not improve the performance of the MSDCNN model scoring lower than the HSCNN model on all subsets.

1.3 Organisation

The structure of the dissertation is as follows: Chapter 1 introduces the motivation, aim and objectives for the research problem. Chapter 2 covers in detail the basic concepts of HSI technology and related literature review. Methodology is presented in chapter 3. Chapter 4 documents the technical implementation of the project in detail. Chapter 5 presents the results and discussion. Finally, chapter 6 concludes the dissertation and provides recommendations for future work.

Chapter 2

Background Information & Literature Review

2.1 Introduction

This chapter will provide a brief yet detailed explanation on hyperspectral imaging. It will also discuss the spectral information of organic matter and the applications of hyperspectral imaging. This chapter also covers basic machine learning algorithms that are used in this dissertation. Furthermore, it will provide a critical analysis of the literature reviewed.

2.2 Hyper Spectral Images

Hyperspectral Imaging(HSI) is a technique that combines conventional imaging and spectroscopy to collect information from across the electromagnetic spectrum. HSI allows the collection of both spatial and spectral information simultaneously from an object (Guerri et al. (2023)). A Hyper-Spectral Image $I(x, y, \lambda)$, also known as a hypercube, is 3-dimensional, where the first two dimensions are the spatial dimensions and the third dimension is the spectral dimension (M. ElMasry and Nakauchi (2016)). The spatial aspect of the Image , denoted by $I(x, y)$, is called a voxel, while the spectral aspect of the image, denoted by $I(\lambda)$, is called a spectra. From a hypercube two views can be extracted, the spatial view, which is a voxel $I(x, y)$ at any given wavelength λ, and the spectral view, which is the spectra $I(\lambda)$ at any given pixel (x, y) (M. ElMasry and Nakauchi (2016)). The term spectral band, usually shortened to band, refers to a range of wavelengths that are grouped together, defined by the central wavelength and the bandwidth. The number of spectral bands in a hypercube is much larger than what is found in a typical RGB image, hence the term Hyperspectral.

2.2.1 Hyper Spectral Imaging Sensors and Methods

Fundamentally a hyperspectral camera works by capturing light though a lens and then splitting the light into its constituent wavelengths using a dispersive element, such as a prism or a grating. The split light is then focused onto a detector array, which is made up of a number of pixels. Each pixel is sensitive to a specific wavelength of light or multiple wavelengths depending on the sensor. The data is read from the detector array, and is then stored in a computer. There are 4 categories of hyperspectral imaging sensors, namely: Whisk-broom, Push-broom, Staring and Snapshot (Guerri et al. (2023)).

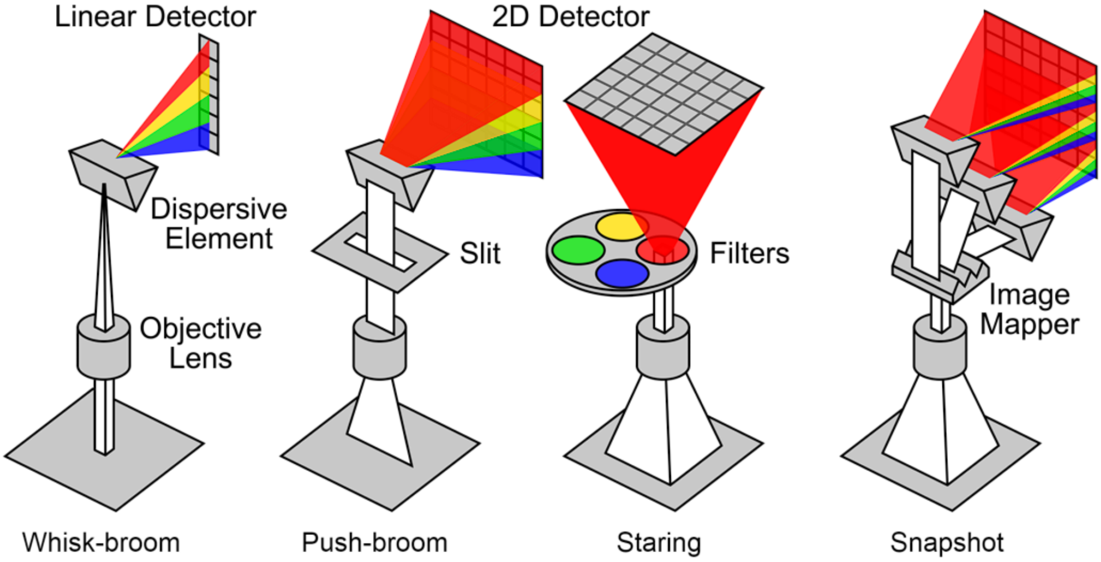


FIGURE 2.1: Hyperspectral Imaging Methods (Guerri et al. (2023))

1. **Whisk-broom** is a technique where the camera is mounted on a moving platform, such as a satellite, and scans the targeted area below it. The sensor captures a fraction of the targeted area at a time, whilst the sensor is moved to cover the targeted area in a series parallel lines. The area is captured in the form of narrow strips which are then combined to form a hypercube.

2. **Push-broom** is a technique similar to whisk-broom however the targeted area is captured in a continuous strip as the sensor moves along the targeted area. As the sensor moves across the area it continuously records the spatial information along one dimension which is perpindicular to the direction of motion, also recording the spectral information. The slices $I(y, \lambda)$ are combined to give the full hypercube.

3. **Staring** , also known as tuneable filter, is a technique where the sensor is kept stationary as it scans the entire targeted area. The sensor captures a series of images at different wavelengths to be later combined into a hypercube. This is achieved through a tuneable filter as seen in figure 2.1 which changes the spectral band that can be recorded. This technique requires the targeted area or object to be stationary and monitored for a long period of time to captue the full hypercube.

4. **Snapshot** , also known as single-shot, is a technique where the sensor captures all the spectral bands posssible for the targeted area in a single exposure, like a conventional camera. Snapshot sensors are the fastest and the most expensive out of all as they are designed to capture multiple spectral bands at once.

2.2.2 Spectral Information of Organic Matter

To proceed with this project it is important to understand the spectral signature of organic matter. What makes organic matter so different when viewed by a hyperspectral camera? **Note :** In the scope of this project, organic matter refers to plants, fruits or vegetation.

1. **Water Content :** no doubt that every fruit or vegetable contains water, however it varies from one fruit to another. The water content of a fruit has been shown to affect certain wavelengths based on the fruit in question. To further investigate the effect of water a correlation heatmap must be utilised to rule out the most fitting wavelengths. For example in a strawberry, the wavelengths 674 and 698 nm were observed to be highly correlated with the measured water content (Raj et al. (2022)).

2. **Fibre Content :** Fibres make up the structure of a fruit and directly affect its firmness and texture. pectin is a polysaccharide , a type of fibre, that exists in mulberries. Similar to the water content's effect on the wavelengths, the fibre

content also affects certain wavelengths. Due to an ovelap in the visible and NIR bands, there werent any highly correlated wavelengths to pectin only, multivariate analysis ruled out a set of wavelengths to determine the fibre content with accuracy (Yang et al. (2021)).

3. **Nutrient Content :** nitrogen, phosphorous, and potassium (NPK) are essential nutirents to the formation of the naturally complex compounds found in plants. A deficiency in any of these nutrients will present a plant with hindered growth and unfit for consumption. Taha et al. (2022) utilised a genetic algorithm (GA) to rule out 3 spectral bands correlating to each nutrient, where bands(measured in nm) 450-550, 590-700 and 780-850 correlated to nitrogen, bands 510-615, 700-820 and 850-900 correlated to phosphorous and bands 500-568, 700-780 and 900-120 correlated to potassium.

2.2.3 Hyperspectral Imaging Applications

HSI systems have the ability to capture a large amount of informtion hidden beyond the visible spectrum, and beyond the surface of an object. Due to this ability, HSI systems have been widely adapted in many industrial applications. HSI systems are fitted on food processing pipelines to detect unwanted produce or foreign material. HSI systems are also used in geological surveys to determine the possible profits from mining a specific area. Most importantly HSI systems have been used in the medical field to detect cancers. There are several applications for applying HSI in different sectors, especially in agriculture. Here we will include few applications.

Crop classification

Among the most important areas of agricultural research is the use of hyperspectral imaging to crop identification and classification. To identify distinct spectral signatures for different crop varieties and development stages, HSI technology collects precise spectral information in the visible and near-infrared portions of the electromagnetic spectrum. The analysis of hyperspectral data using machine learning algorithms makes it possible to classify crops according to their spectral reflectance properties. This makes it easier to create maps that show the different kinds of crops and their phases of development. This abundance of data provides insightful information for managing and making decisions in agriculture, such as identifying crop stress, identifying probable causes restricting production, and keeping an eye on the general health of the crop. Implementing DCNN, and KNN, Zhang et al. (2020c) investigated the Corn seed variety classification to get the results DCNN training accuracy=100% Testing accuracy rate=94.4%, 57 and validation accuracy rate= 93.3%

disease detection

Farmers face numerous challenges when different pests and diseases appear in crops. Common causes of these diseases and pests include nematodes, bacteria, viruses, and fungi. Nevertheless, in the past, farmers have avoided diagnosing or suspecting many infections due to a lack of knowledge and professional advice, underscoring the critical role that early detection, made possible by HSI technology, plays in preventing actual crop damage. For example Zhang et al. (2020b) applied the HSI for-variety identification of coated maize kernels using different ML & DL algorithms (LR, SVM, CNN, RNN and LSTM), where they got an accuracy of 90%.

soil analysis

Poor soil management poses a danger to the productivity and quality of soils, which are essential for rural livelihoods. Identifying soil erosion is crucial for addressing agricultural issues like crop quality and production. Soil erosion is a process that is heavily impacted by regional climatic elements including bright sunshine and heavy rain. With the use of hyperspectral imaging (HSI) technology, soil parameters such as moisture content, nutrient levels, and mineral composition may be analysed without causing damage to the soil. Table 4 provides information that makes it easier to map soil properties and helps with decision-making about agriculture, environmental management, and other applications using soil. As an important study, in Wei et al. (2021), they used the method of DNN-CARS to investigate Estimating the spatial distribution of soil total arsenic. The results show that R2CV = 0.69, RMSECV = 0.61, RECV = 6.56

2.3 Machine Learning Methods and Algorithms

There are three main types of machine learning algorithms that are used to model the relationship between the input and output data, namely: unsupervised learning, semi-supervised learning, supervised learning. The last approach is adopted in this dissertation.

Unsupervised Learning aims to uncover hidden patterns in the data without the need for labels. The most common unsupervised learning algorithms are clustering and dimensionality reduction. Clustering algorithms group similar data points together, while dimensionality reduction algorithms reduce the number of features in the data. There exists applications of unsupervised learning in the domain of hyperspectral imaging such as Murphy and Maggioni (2019)'s spectral-spatial diffusion learning

Semi-supervised Learning is combination of supervised and unsupervised learning. First a model is trained on a set of labelled data, then predictions are made on a set of unlabelled data. The most confident predictions are then added to the labelled data as "pseudo-labels". Giving the model a bigger labelled dataset to train on. In the domain of hyperspectral imaging, semi-supervised learning can be convenient given the expense of labelling data.

Supervised Learning is the most common type of machine learning approach where the model is trained on a set of labelled data to later make predictions on an unseen set of data.

In the context of supervised learning there are many algorithms that can model the relationship between features and labels. Namely, Support Vector Machines (SVM), Decision Trees (DT), Random Forest (RF), K-Nearest Neighbours (KNN), Convolutional Neural Networks (CNN).

2.4 Literature Review

2.4.1 Preprocessing Techniques

Reflectance calibration is a technique used to convert the raw data into reflectance values. Denoted by the following equation:

$$I_{c(x,y,\lambda)} = \frac{I_{r(x,y,\lambda)} - I_{d(x,y,\lambda)}}{I_{w(x,y,\lambda)} - I_{d(x,y,\lambda)}} \tag{2.1}$$

Where :

- I_c is the calibrated hypercube

- I_r is the raw hypercube
- I_d is the hypercube of a dark reference panel
- I_w is the hypercube of a white reference panel

It is important to note that sometimes the software used to capture the data performs the reflectance calibration automatically, however it is not always the case and the researcher explicitly states its application. This can be observed in Zhang et al. (2020a), Zhao et al. (2020), and Gao et al. (2020). Savitsky Golay (SG) filter is a preprocessing algorithm that is used to smooth out the data and remove noise without compromising the integrity of the data. It works by fitting a polynomial of a certain degree to a set of consecutive data points, then replacing the central data point with the value of the polynomial at that point. The application of a Savitsky-Golay filter is an essential step taken by many. Raj et al. (2022) leverages the benefits of utilising a Savitsky-Golay filter on hyperspectral images of strawberries to achieve a better prediction of the water content. Multiplicative Scatter Correction (MSC) is a preprocessing algorithm that eliminates light scattered by irregularites. The scatter can stand in the way of making a meaning of the data. Standard Normal Variate (SNV) is a preprocessing algorithm that removes the mean and scales the data to unit variance. SNV is used to remove the effect of the intensity of the light source. Taha et al. (2022) utilises MSC to decrease scatter and SNV to normalise the data, followed by a Savitsky-Golay filter to remove noise. Similarly, Yu et al. (2018) applies MSC however, they havent utilised a Savitsky-Golay filter. Zhao et al. (2020) segments the images automatically and utilises WT to remove noise from them, whilst Yang et al. (2021) manually selects the regions of interest (ROI) from the images. The histogram thresholding technique is an image segmentation technique based on the analysis of the histogram of the image. Rady et al. (2017) applies this technique to derive the ROI for each image, then the mean reflectance spectrum (MRS) was calculated based on the intensities within the ROI. Siedliska et al. (2018) applies automatic thresholding, which iterates over the layers in a hypercube and creates a threshold based on the layer with the highest contrast. The output is a hypercube containing only information on the strawberry and not the background. Moreover, computing the MRS for each image. In extreme contrast, Gao et al. (2020) selects 10 ROIs from each leaf, each ROI being 10x10 pixels in size. Following that is reflectance calibration, and normalisation.

2.4.2 Analysis and Wavelength Selection Approaches

Correlation heatmaps are a great way to rule out the most correlated wavelengths to the data. Raj et al. (2022) proposed an index, a function of reflectance of two specific wavelengths, named the Normalized Difference Strawberry Water Content Index (NDSWI):

$$Index_{mn} = \frac{R_m - R_n}{R_m + R_n} \tag{2.2}$$

where R_m and R_n are the reflectance values at wavelengths m and n respectively.The index is then analysed for correlation to the ground truth water content. Ruling out 674 and 698 nm as the most correlated wavelengths to the ground truth water content.

Correlation analysis is a fundamental tool in multivariate analysis, however there are more complex algorithms to analyze the data. Principle Component Analysis (PCA) is a dimensionality reduction algorithm that retains the variance of the original data. Taha et al. (2022) utilises PCA to find the optimal wavelengths by inspecting the peaks and valleys of the loading vector for each principle component (PC). In addition to PCA, a genetic algorithm (GA) was used to find the optimal wavelengths by selecting the wavelengths with the highest fitness score. Sequential forward selection (SFS) is a technique that iteratively adds a feature to the selection until the model trained on the selection no longer shows an increase in performance. Finally, SFS was applied, giving the team 9 different sets (3 algorithms for each of N P & K) of wavelengths to train their models on.

A similar approach is employed by Zhang et al. (2020a), where they used successive projections algorithm (SPA), comptetitive adaptive reweighted sampling, PCA, and wavelet transform (WT). SPA is an algorithm that gives a subset of the dataset containing the most informative features by iteratively removing the least informative features.On the other hand, CARS selects the most informative features by iteratively reweighting features based on their effect on the performnce of a predictive model. WT decomposes the dataset into a set of wavelets coefficients at different scales and positions. These coefficients represent the contribution of varios frequency components.

Yu et al. (2018) and Zhao et al. (2020) have decided on a simple approach to analyse the data, applying SPA only to find the optimal wavelengths.

In contrast to the previous approaches Sun et al. (2019) combines PCA, correlation coefficients (CC), and random forest (RF). First the PC loadings are inspected for peaks and valleys, then the correlation coefficients are calculated for each of the wavelengths. Finally, the final selection is taken from the feature importance scores found in a trained random forest model.

Rady et al. (2017) simply utilised only SFS to find the optimal wavelengths. A unique and unusual technique is used by Siedliska et al. (2018), the application of second derivative using the Savistky Golay method to enhance the spectral differences. The peaks and valleys of the second derivative are then inspected to find the optimal wavelengths. Gao et al. (2020) selects the wavelengths via Least Absolute Shrinkage and Selection Operator (LASSO), which is a regression technique. Furthermore, The analysis of variance (ANOVA) is utilized to identify wavelengths with statistically significant differences in reflectance values between healthy and GLD-infected leaves. "At last, six wavelengths (690, 715, 731, 1409, 1425 and 1582 nm) were determined as the salient wavelengths." (Gao et al. (2020))

2.4.3 Machine Learning-Based Approaches for HSI

Raj et al. (2022) has used a logarithmic model on the proposed NDSWI to achieve greater accuracy than utilising the NDSWI alone.

$$SFWC = \alpha \times ln(NDSWI) + \gamma \tag{2.3}$$

The logarithmic model was fitted using a k-fold cross validation approach and has achieved an RMSE of $0.0092 g/g$ (gram of water per gram of strawberry) and the output had a correlation coefficient of 0.82 with the actual water content whilst the NDSWI scored $0.012 g/g$ and 0.66 respectively. Partial Least Squares Regression (PLSR) is a regression model that is capable of handling multicollinearity and is robust to noisy data, making it a great tool for hyper spectral images. Taha et al. (2022), Zhao et al. (2020), Yang et al. (2021), and Zhang et al. (2020a) have all utilised PLSR on their datasets. Taha et al. (2022) compares the performance of PLSR to RF while the others compare PLSR to Least Squares Support Vector Machines (LS-SVM). Taha et al. (2022) achieved an RMSEP of 0.42, 1.92, and 0.34, for each of N P and K respectively, from a PLSR model trained on the output of GA, and an RMSEP of 0.55, 0.20, and 0.32 from a Random Forest (RF) trained on the wavelengths derived from the PC loadings. Moreover, Zhao et al. (2020) attempted to predict the soluble solids content (SSC) of jujubes cultivars using PLSR and LS-SVM. The PLSR model achieved an RMSEP of 1.127 on the spectral band 380 - 1030 nm and 1.146 on the spectral band 874 - 1734 nm, whilst the LS-SVM achieved an RMSEP of 0.930 and 0.974 respectively. Furthermore, Yang et al. (2021) attempted to predict total soluble pectin in mulberries achieving RMSEP(g/kg) of 24.487 and 18.484 from PLSR and LS-SVM respectively. Finally, Zhang et al. (2020a) has mixed and matched inputs (full, SPA, CARS, PCA, WT) with models (PLS, LS-SVM). Among these combinations, the best (based on RMSEP) for predicting total anthocyanins was achieved by CARS+LS-SVM, full+PLS for total flavonoids, and WT+LS-SVM for total phenolics. An uncommon model, adapted by Sun et al. (2019), is gaussian process regression (GPR). For N,P and K GPR achieved an RMSEP of 2.64, 0.62, and 1.79 , whereas SVM achieved an RMSEP of 3.71, 0.82, and 3.51, (trained on the images of the first angle of view).

Furthermore Raj et al. (2022) utlised an SVM and a decsision tree classifier to classify the strawberries into 3 classes, under-ripe, ripe and over-ripe. Trained on the 650-750nm band, the spectrum, and the spectrum plus the ground truth water content. The SVM achieved overall accuracy of 68.4%, 98.2%, and 98.3% respectively. Whilst the decision tree classifier achieved 70.1%, 93.8%, and 93.7% respectively. Given the task of detecting whether there are codling moths in the apple or not, Rady et al. (2017) compares the performance of linear discriminant analysis (LDA), k-nearest neighbours (KNN) and decision tree (DT). All the models were trained on all wavelengths and compared to when trained on the SFS selected wavelengths. The best performance with all the wavelengths was achieved by DT at 78% overall accuracy. However with the SFS selected wavelengths the DT was on par with the KNN at 78% overall accuracy. Given the great performance of the DT model, it was decided to seperate the images based on the angle of view and train a DT model on each set. The best performance was achieved by the DT model trained on the images angled at the calyx and at the stem, with the application of SFS, at 82% overall accuracy.

To classify whether a grape vine is diseased with a notorious virus Gao et al. (2020) employs a LS-SVM. The procured data consists of thre seasons each at 5 different growth stages. For the 2018 season the model achieves accuracies(%) 76.55, 70.65, 75.60, 83.05, and 95.85 for each of the 5 growth stages respectively. In comparison in 2019 the model achieves accuracies(%) 66.67, 60.74, 68.81, 74.15, and 97.26 for each of the 5 growth stages respectively. stage 5 being the most accurate for both, denoting the symptoms are more apparent at later stages.

Two pathogens affecting tomato crop cause two different differentiable lesions, target spot (TS) (caused by a fungus) and bacterial spot (BS). Abdulridha et al. (2020) utilises a step wise discriminant analysis model to classify the health status. Many scenarios within healthy vs BS, healthy vs TS, and TS vs BS, are tested for. The model performs consistently well for the first two scenarios but shies away from the third A.2.

Given a dataset of HS images of healthy, bruised, and infected strawberries Liu et al. (2018) aims to classify into these categories with PLS-DA and SVM. First training both of them on the full wavelengths, where they achieved 87.65% and 92.59% respectively. Then on the wavelengths selected by MNF we can see a slight downgrade in performance whereas the PLS-DA achieved 80.25% and the SVM achieved 90.12%. A similar dataset, has been constructed by Liu et al. (2022), however there are 4 classes, normaly, bruised, infected and contaminated. A baseline for the achievable performance was set by training a one class SVM (OCSVM). The OCSVM achieved an AUC score of 0.709(±0.014) and an F1 Score of 0.673(±0.011). In another instance, Liu et al. (2023) procures two datasets, one of strawberries and one of blueberries. Given the solid performance of the OCSVM in the previous study, it was the baseline performer once again. On the strawberry dataset it achieved an AUC score of 0.773 (±0.009) and F1 Score of 0.758 (±0.007).

2.4.4 Deep Learning-Based Approaches for HSI

2.4.4.1 MLP Implementations

Taha et al. (2022) and Sun et al. (2019) leverage the effectivness of backpropagation neural networks (BPNN) to predict the NPK nutrient content of lettuce and tomatoes respectively. Sun et al. (2019) used a BPNN with a common architecture of 3 layers (including input layer) with 15 neurons in the hidden layer, making use of tansig activation between the input and hidden layer and the purelin function in between the hidden and output layers. Taha et al. (2022) trained their model on 3 datasets being the output of GA, the output of PCA, and the output of SFS. Whilst Sun et al. (2019) only trained their model on the output of a combination of PCA,CC and RF. Taha et al. (2022) reports most consistent performance from the PCA dataset, achieveing an RMSEP of 0.6, 0.26, and 0.5 for N, P, and K respectively. On the other hand, Sun et al. (2019) reports the best performance from the application of PCA-CC-RF onto the 3D reconstructed images from 4 different angles, achieving an RMSEP of 1.17, 0.52, and 1.85 for N, P, and K respectively.

A different approach can be seen adapted by Yu et al. (2018), where the encoder segment of a sparse auto encoder was used to extract features from the dataset and feed them directly into a fully connected neural network optimised by back propagation (referred to as SAE-FNN). The model was used to predict the firmness and soluble solids content of korla pears. The model was altered and tested with different numers of input features from the SAE, the best performance at predicting firmness was achieved with 50 features, at a RMSEP of 1.81. In comparison to the best performances of PLSR and LS-SVM which achieved RMSEP of 2.17 and 2.21 respectively. On the other hand when predicting SSC the SAE-FNN performed best with 40 input features from the SAE, achieving an RMSEP(%) of 0.22. In comparison to the best performances of PLSR and LS-SVM which achieved RMSEP of 0.33 and 0.35 respectively. All of this was done on the entire dataset, the application of MSC seemed to increase the RMSEP of the PLSR models whilst decreasing that of the LS-SVM models. The application of SPA can be seen increasing the RMSEP of both models. The SAE-FNN was not tested on the MSC dataset nor the SPA dataset.

A simple fully connected artificial neural network can prove to be a great tool classification problems as demonstrated by Raj et al. (2022), using a fully connected neural network with 4 hidden layers of sizes 80, 40, 20 and 10 respectively. The activation function used was the rectified linear unit (ReLU) and the optimizer used was the Adam optimizer. The model when trained on the 650-750nm band, the spectrum, and the spectrum plus the ground truth water content achieved 67.8%, 96.4%, and 96.1% overall accuracy respectively.

In the search for greater performance, Abdulridha et al. (2020) uses a MLP based neural network optimised by gradient descent to classify the health status. In each scenario tested for the MLP achieved the higher accuracy, however the difference was not significant. see A.2.

Liu et al. (2018) utilises a BPNN to classify strawberries into 3 classes, healthy, infected and bruised. The model was trained on the full dataset and the SPA dataset. The model achieved 91.36% and 87.65% overall accuracy respectively.

2.4.4.2 CNN Implementations

Zhang et al. (2020a) trained a convolutional neural network (CNN) to predict the chemical compositions of goji berries. The CNN was trained on the full dataset, the SPA dataset, the CARS dataset, the PCA dataset, and the WT dataset. The best prediction (lowest RMSEP) for total phenolics was achieved by the CNN trained on the WT dataset at 279.658, whilst the best prediction for total anthocyanins and total flavonoids was achieved by the CNN trained on the CARS dataset at 110.602 and 43.963 respectively. Most notably they did not stop there, they decided to use a CNN and a DAE as feature extractors, then feed the extracted features into PLS and LS-SVM models. The best prediction for total anthocyanins was achieved by DAE-LS-SVM at 119.308, total flavonoids by CNN-PLS at 43.721, and total phenolics by DAE-LS-SVM at 287.190.

Varga et al. (2021) has created a large dataset of avocados and kiwis as of 2021, and persimmons, mangoes, and papayas as of 2023. The dataset for each fruit was created by taking a sample each day, noting the days it was kept in storage, weight, and firmness, before manually testing the ripeness. The ultimate goal of Varga et al. (2021) is to classify a fruit as unripe, ripe, or overripe based on the hyperspectral image. It was decided by them to take a specific approach to a CNN suiting the requirements of a HS dataset and minimising the number of parameters. Initially, three convolutional layers responsible for extracting features. The most notable design decision is the use of average pooling layers instead of the max pooling layers. This neural network was compared to the following: SVM, kNN, RES-NET18, AlexNet. The results of this sophisticated architecture speak for themselves, achieving rarely paralleled accuracies on the avocado dataset. However with the kiwis, althought it remains the best performing model, its performance can be seen on par with the other models throughout the categories.

Qiao et al. (2020) developed a deep residual 3DCNN to detect and classify early decay on blueberries. In their dataset they had 4 classes, sound, slight-decayed, moderate-decayed and severe-decayed. They compared their architecture to AlexNet, GoogleNet, a 3D-CNN, and ResNet. Their architecture outperformed all the others in terms of accuracy scoring 96.69% while the others scored 89.12%, 91.88%, 92.15% and 95.42% respectively.

Zhao et al. (2022) proposed a deep learning-based extended morphology-nonlocal capsule network (EMP-NLCapsNet) algorithm to classify the duration a fruit has been stored for at different conditions. Their dataset contained hyperspectral images of kiwis stored in a low temperature environment and others in a room temperature environment. The images were classified according to the duration of their storage in days, ”D0”, ”D2”, ”D4”, and ”D6”. EMP-NLCapsNet has outperformed all the other models in terms of accuracy, achieving 100% on both the low temperature and room temperature subsets. However the most notable comparison is to the original NLCapsNet architecture which was able to achieve 99.6%, and 98.47% respectively whilst having a fraction of the parameters. Whereas NLCapsNet had 34,734,464 parameters and EMP-NLCapsNet had 6,562,664.

Davur et al. (2023) developed a spectral-spatial residual network (SSRN) using convolutional and batch normalisation layers to classify and regress the duration to ripeness of avocado fruit. In contrast to other implementations Davur et al. (2023) did not pass the whole image as input instead they segmented their images into sub-images, effectively making a dataset of 551 images turn into approximately 44k ”sub-images”. This approach circumvents the lack of training data. Their SSRN scored an RMSE of 1.32 (days) and a classification accuracy of 51.43%.

2.4.4.3 Multi-Scale/Multi-Kernel Implementations

Due to some limitations in hyperspectral imaging of fruits, there will be some articles that are not fully within this field but share the same key aims and challenges this project will have.

Ilyas et al. (2021) proposed a multi-scale context aggregation network to classify strawberry images (RGB) into 4 classes, ”Unripe”, ”Partially Ripe”, ”Ripe”, and ”Overgrown/Disease”. The model achieved an F1-score of 82.92%.

He et al. (2017) developed a multi-scale 3D deep convolutional neural network by branching the input into different convolutional layers with different kernel sizes. It must be noted that this article developed this architecture to deal satellite acquired hyperspectral imagery. The core difference in this implementation is the way that the classification occurs which is pixel-wise for He et al. (2017) but object-wise for this project. Nevertheless this model was tested on the following datasets: Indian Pines, Pavia University and Salinas. This model was compared to a non multi-scale version similar to itself, outperforming it on every dataset by at least 0.45%.

Ren et al. (2022), similar to He et al. (2017) in the context of datasets choice they used and the pixel-wise classification, developed a multi-scale adaptive convolutional neural

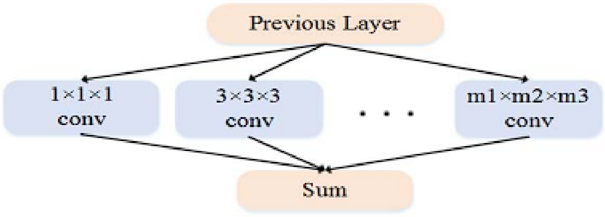


Figure 2.2: Multi-Scale approch taken by He et al. (2017)

network by inputting the image at 3 different scales into 3 identical branches consisting of adaptive convolutional layers. Each branch ends with a softmax layer giving each scale its set of output probabilities which are then put through a voting mechanism to give the final classification. The model was pitched against an SVM,SSRN, MLP, 1D-CNN, 2D-CNN, 3D-CNN, RNN, GRU and a content guid CNN (CGCNN). There are 16 classes in the dataset, in which 10 of them this model outperforms the rest.
Another implementation of pixel wise classification done on satellite acquired hyperspectral imagery by Lee and Kwon (2017) proposes their ”Contextual Deep CNN” which starts off with a multi-scale feature extraction followed by deep convolutional layers. The model was tested on the Indian Pines, Pavia University and Salinas datasets. Where it scored 94.24%, 96.73%, and 95.42% respectively in comparison an SVM scored 87.6%, 90.52% and 91.66%.

2.4.4.4 Attension Mechanism Implementations

Within the context of deep learning models, attention mechanism (Dou et al., 2021) is reffered to as a mechanism that is used in CNN to focus on specific parts of data. This attention layer (or module based) in CNN architecture enhances the classification performance of HSI. Hence the benefit of including the attention mechanism in the CNN architecture is to allow the model to focus on the features that carry more information and pay less weight for unimportant ones (Weng et al., 2021). In other words, attention mechanism weighs the features based on their information. Attention in networks is for Squeeze-and-Excitation Networks (Hu et al., 2017). In our model, the inclusion of the attention is an easy to implement and costs not much computation. One of the very few studies that integrates attention mechanism and multi-scale deep learning network to classify HSI is the work of Qing and Liu (2021). They used three different sets (Pavia

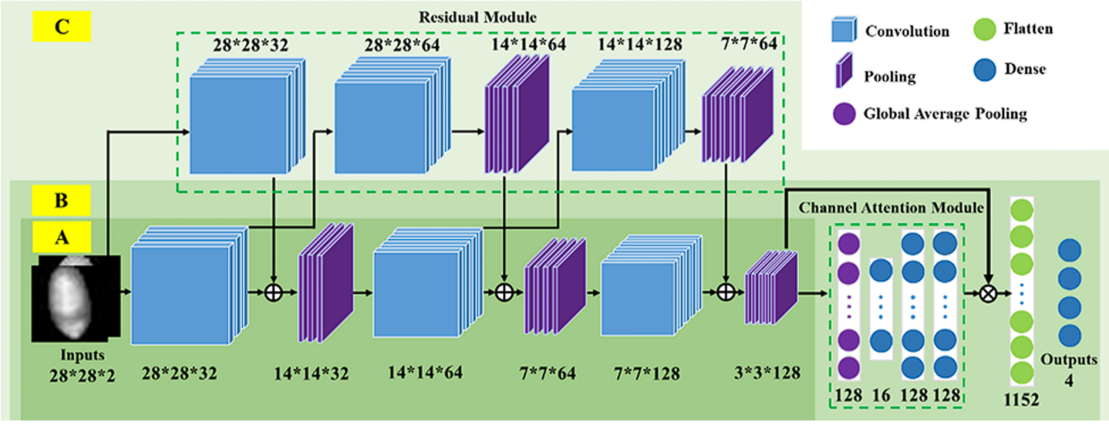


Figure 2.3: Residual Attention CNN (Weng et al. (2021))

University Dataset, Indian Pines and KSC Datasets) of HSI public data sets to test the classification accuracy of their proposed model. Their model includes multi-staged architecture, where in first stage they applied PCA to reduce the dimensions of the spectral information of the HSI to a 2-dimentions. In the next stage they considered the images with reduced dimensions as input for the integrated model. The model performance is evaluated using 3 data sets which are available publicly. The accuracy rate for the 3 data sets ars 99.82%, 99.81%, and 99.37%, respectively. This performance is better than the traditional CNN.

Weng et al. (2021), implemented attention mechanism in CNN to classify the degree of infection of wheat kernels. Authors considered the Fusarium head blight (FHB) to be identified. They proposed an CNN using the HSI for the infected wheat kernels (classified as healthy, mildly, moderately and severely FHB). HSI is used to extract the spectra of the FHB classes. Moreover, of these spectra they selected 5 “effective wavelengths” by random fog. They noted that the reflectance images increases with the severity degree of the FHB infection. They adopted LeNet-5 to screen the selected reflectance images. They expanded the CNN by including the attention and the residual convolution to get the RACNN (residual attention convolution neural network) to classify the infection degree of FHB in wheat kernels. The RACNN performance achecived a 100% accuracy calibration, 98.60% accuracy for validation and 98.13% accuracy for prediction set.

In another application for implementing attention in CNN, Zhang et al. (2022) investigated the combination of HSI and attention in DL to predict the oil content in single maize kernel. They proposed the ACNNR (Attention Convolution Neural Network Regression) model in addition to CNNR (without attention) and PLSR (Partial Lest Square Regression) for comparing the performances of the attention. HSI is used to get the reflectance images of the embryo side and non-embryo side of single maize kernel. For data preprocessing for the PLSR, the authors adopted series of spectral analysis in addition to PCA for dimensionality reduction to get the optimal hybrid PLSR model. On the other hand, HSI spectra are used as inputs for the ACNNR and CNNR. The authors adjusted all the three models in a small-scaled data set. Moreover, they utilised the SPA, CARS, and PCA to select the most effective wavelengths to get rid of the redundant information. Their results revealed that the coefficient of determination is 0.918 for ACNNR is outperformed, which confirms that the attention has enhanced the performance of the CNN. Moreover, the regression model is performing better for the embryo side.

Also, Multi-Scale and Attention with different machine learning are used in recognition is different sectors, for example see: Yang et al. (2022) and Yang et al. (2024).

2.5 Critical Analysis

CNN is a deep learning, that was distinguished among others as on of the best object recognition, especially after the ImageNet Large Scale Visual Recognition Competition (Li et al., 2022, Taye, 2023). A type of NN that proved great performance in image classification (Blaschke et al., 2014) is the convolutional neural network CNN. Researchers on many applications showed that CNN have capability to extract the detailed image features with a high accuracy (Taye, 2023). There are (Swapna et al., 2020) many architectures for CNN, (for example AlexNet, VGG, ResNet). These variants of CNN include huge number of parameters. AlexNet includes 60M, VGG has 138M and ResNet 60.1 M parameters. Hence designing a CNN architecture that take these parameters with high performance is still challenge task, which is that main ain of this dissertation, as it will be explained latter. Hyperspectral imaging (HSI) is used heavily as main non-destructive fruits assessment (Li et al., 2017, Lu et al., 2017). In the last two decades CNN have been utilised in HSI classification in different sectors. A good review for using the CNN in HSI in agriculture can be found in (Guerri et al., 2023). The overall accuracy performance of CNN in HSI fruits and vegetables classification was outperform. This is because the CNN can learn the hundreds of wavelength features of the HSI and extract them with high accuracy. This research has been done for many fruits like avocado (Varga et al., 2021), kiwi (Varga et al., 2021, Zhao et al., 2022), blueberries (Qiao et al., 2020). To consider the most important information from inputs, the concept of “attention mechanism” (which is one of the most important concepts in deep learning) was introduced to focus on and weigh the important information that can be extracted from the input data (Niu et al., 2021). The lake of getting information from the input is due to using a fixed length for the encoding vector. In this case the decoder will get a limited access to the input information. This led to introducing the Bahdanau et al. (2014) attention model, that was used in machine translation. Expanding the CNN to include the attention mechanism was a good idea to be implemented to study image classification (see for example, Qing and Liu (2021)). Paoletti et al. (2023) proposed a model to handle the large information for the spectral and spatial HSI and in same time to reduce the computational cost for the architecture. As over all evaluation, many researchers found that combing the attention and CNN is a state- of- the-art. In many real situations, the fruits images have variation in size, scale, and resolution. A technique that can handle these issues is the mutli-scale (see for example: Ilyas et al. (2021), He et al. (2017)). This approach combined with CNN can learn the spatial features (2D) of HSI as well as the spectral features (1D) (He et al., 2017). Based on the above, as a research gap it is interesting to design an architectures that does not exceed the computational cost attainable by embedded systems, and in the same time to have a high performance in the classification of the HSI. This is the main aim of this dissertation.

Chapter 3

Methodology

This chapter explains the methodology of the project, explaining the steps taken with justifications. Moreover, defining the requirements the implementation must fulfil.

3.1 Overview

Due to the limitations found in this specific field of research, datasets of sufficient size are scarce and hard to find.

3.2 Data Augmentation

The objective of this step is to artificially increase the size of the dataset given its size is not enough to sufficiently train deep neural networks. This step is rarely implemented by other researchers. Deep learning models like autoencoders and generative adversarial networks are used to generate new data. The data augmentation process is crucial to the success of the model. The main focus of this step will be to investigate the performance of GANs in this specific field of research, and to compare them to the primitive methods of data augmentation. Their effectiveness will be measured by the performance of the model trained on the augmented data.

3.3 Classification and Expirements

The classification step will involve training a range of deep neural networks. A train-test split will be performed on the dataset as per the standard practice. For the train-test split approach the models will be trained on augmented and balanced inputs to guarantee the best outcome possible, and tested on. Furthermore due to unforseen circumstaces regarding the dataset Leave One Out Cross Validation (LOOCV) will be performed using the models.

The following machine learning algorithms will serve as a baseline model to measure the performance of the deep learning models.

- Support Vector Machine (SVM)
- Decision Tree (DT)
- Random Forest (RF)
- K-Nearest Neighbours (KNN)

A deep learning model will be taken from the literature as a starting point. To improve on the starting point deep learning model, new architectures will be designed by incorporating new concepts which have shown promising results in the literature.

3.4 Evaluation Strategy

The evaluation strategy for classification will be based on the accuracy obtained by the model via train-test split and LOOCV. Whereas for the GAN model, the evaluation will be based on the loss and the quality of the generated images.

3.5 Requirement Analysis

This section will define the requirements of the project. The requirements are divided into functional and non-functional requirements. Priorities are defined as per the MoSCow convention.

3.5.1 Functional Requirements

Requirement number	Requirement Description	Priority
1	Procuring the dataset	Must
2	Preprocessing the data	Must
3	Augmenting the data	Must
4	Feature extraction	Must
5	Create new model architecture	Must
6	Model training, testing, evaluation	Must
7	Results comparison	Must
8	Computational cost reduction	Could
9	Model performance optimisation	Should

TABLE 3.1: Functional Requirements

3.5.2 Non-Functional Requirements

Requirement number	Requirement Description	Priority
1	The model's performance is better than state of the art	Could
2	The code developed in this project is open source	Could
3	Clear and concise documentation of code and design choices	Should
4	The model adapts well to bigger datasets	Could

TABLE 3.2: Non-Functional Requirements

Chapter 4

Technical Implementation

4.1 Development Environment

In this section, we discuss the hardware and software aspects of this project.

4.1.1 Hardware Specifications

It is important to recognise the difficulties of working with hyperspectral images. Whilst the spatial dimensions may not be of a notable scale, the spectral dimension compensates for the lack of spatial information. This attribute of hyperspectral images makes them computationally expensive to work with. Initially the project was developed on a personal laptop with a single gpu. The laptop specifications are as follows:

- CPU: AMD Ryzen 7 4800H
- GPU: NVIDIA GeForce GTX 3060
- RAM: 16GB
- VRAM: 6GB
- Storage: 1TB SSD
- OS: Windows 11

However due to a change in the evaluation strategy the project was moved to a HPC cluster. The HPC cluster specifications are as follows:

- CPU: 32 cores
- GPU: 1x NVIDIA A40
- RAM: 64GB
- VRAM: 48GB
- OS: CentOS 7

4.1.2 Software Specifications

Due to the nature of this project being a deep learning project, Python was the language of choice, and tensorflow was the deep learning framework of choice due to its ease of use and prior experience. In addition to that, Spectral (sPy) was used to read the hyperspectral images due to its compatibility with the data format and its use of numpy arrays to hold the data. The software specifications are as follows:

- Python 3.9.19
- Tensorflow 2.10.1
- Keras 2.10
- Numpy 1.26.4
- Scipy 1.12.0
- Scikit-learn 1.2.2
- Pandas 2.0.1
- Spectral 0.23.1

4.2 Data Acquisition

During the data acquisition phase, multiple datasets were collected from various sources. Upon further inspection of the datasets, it was found that most of them were not suitable for the project. The datasets were either reduced in size through dimensionality reduction algorithms like PCA, or they were pure and unlabelled. This left out only one dataset that was suitable for the project. The dataset was used and published in Varga et al. (2021). The dataset is published to a website hosted by the university of Tubingen, available here.

4.3 Data Insights

The total size of the data is 209 GB of which the subsets used account up to 160gb. The data is stored in the form of header files and binary files which are usually accessed by proprietary software. However there exists a python library (sPy) that can read the data. The subsets used for the project are the kiwi subset and the avocado subset. From each subset there are NIR and VIS images. The NIR images have 252 bands ranging from 919nm to 1727nm, while the VIS images have 224 bands ranging from 397nm to 1003nm. It was an important observation that the images in this dataset were not of the same spatial dimensions (see B). This presented an issue such that there would be data losses when resizing the images to a common size. The class distribution of the dataset is shown in figure 4.1.

Fruit	Bands	Number of samples			
		Train set	Validation set	Test set	Total
Avocado	NIR	40	9	9	58
	VIS	92	15	83	190
Kiwi	NIR	58	9	9	186
	VIS	138	24	24	76

Table 4.1: Train, Validation and Test set sizes

4.4 Data Generator

A custom data generator was created to read the data from the header files and binary files. The data generator was a necessary step for multiple reasons, first is to account for

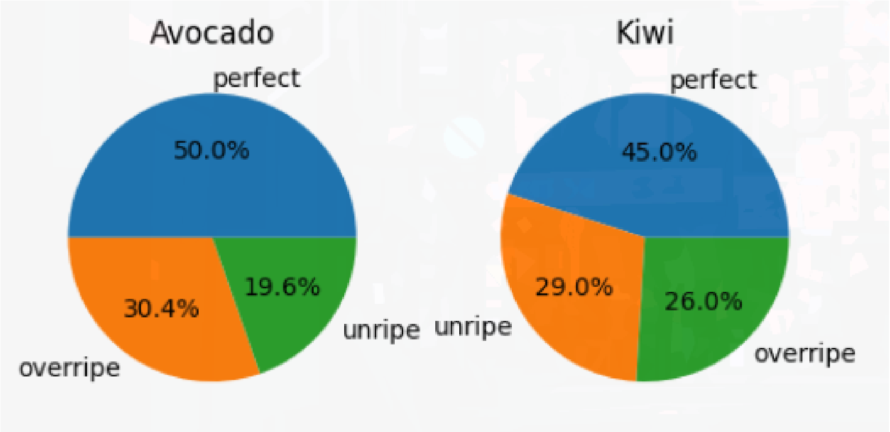


Figure 4.1: Class Distribution of the fruits in the dataset

hardware limitations due to the data size, second is due to lack of existing implementations that allow augmenting and balancing of hyperspectral images. The data generator was implemented as a subclass of keras.utils.Sequence to ensure its compatibility with the keras API. The following data augmentation techniques were implemented: random flip, random rotation, random noise. The balancing algorithm on the other hand would work by undersampling a class if its count is greater, and oversampling by augmenting the class if its count is less than the other classes.

4.5 Simple ML algorithms

These implementations have their limitations whereas the image array (3 dimensional) had to be flattened to be fed into the classifier, losing all spectral and spatial information. All algorithms used were implemented using the scikit-learn library.The models were trained on the training set plus the validation set and evaluated on the test set in order to be compared with the model replicated in the next section. Furthermore, the model was trained and evaluated using the LOOCV given the change in evaluation strategy. The models used were the decision tree classifier (DT), random forest classifier (RF), k-nearest neighbors (KNN), and support vector classifier (SVC). A summary of the models and their parameters is shown in table .

Model	Parameters
DT	criterion='entropy', max_features='sqrt'
RF	criterion='entropy', max_features='sqrt'
KNN	n_neighbors=10, weights="distance", algorithm="ball_tree"
SVM	kernel='rbf'

TABLE 4.2: Simple ML models and their parameters

4.6 Replicating Results

Initially the HSCNN model developed in Varga et al. (2021) was implemented in tensorflow and its performance and results were attempted to be replicated. The images were resized to 64x64x252 and 64x64x224 for the NIR and VIS images respectively, and augmented and balanced. The model was trained on all subsets seperately for 100 epochs each using the adabound optimizer with initial learning rate of $1e-2$. The HS images were augmented and balanced using the data generator. It was observed that the model would converge multiple times, and the results were not as expected. It must also be noted that the test set was quite small to be appropriate for a train-test split approach (see table 4.1). Using model checkpoints at every 5 epochs it is observed that the accuracy of the model would fluctuate between every checkpoint on the test set. This is a clear indication that the model is unreliable and that the train-test split is not an appropriate approach for this dataset.

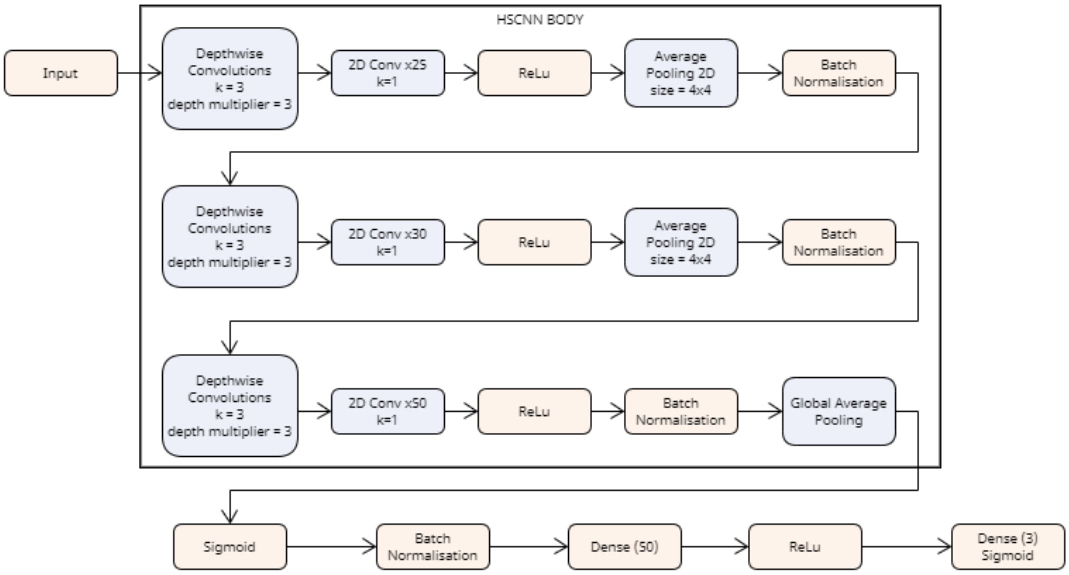


FIGURE 4.2: Replication of the HSCNN model

4.7 GAN Expirement

Due to the limitations of the dataset, a GAN model was attempted to artificially increase the size of the dataset. The GAN model was created by heavily altering the CGAN python script found in a repository of GAN implementations made by Linder-Norén (2020). The first attempt was to train the GAN on the kiwi near infrared subset, it was trained for 10,000 epochs. Observing the output images it was clear that my generator architecture was limited and that it wasnt capable of comprehending the spatial and spectral information of the images. In addition, we decided to integrate the HSCNN architecture as the disciminator and it was observed to be very accurate at identifying the fake images from the real images. An attempt was made to further improve the generator architecture however this approach was abandoned due to the computational cost and the approaching deadline, for which the results were not promising.

4.8 Architecture Expirements and Improvements

Once the results were replicated, it was clear that the model was not as performant as it was expected to be, hence many impovements had to be attempted on the model. The core concept of having depthwise convolutions was kept, whilst multi-scale and multi-kernel concepts were introduced. Furthermore, the attention mechanism was introduced to both of the improved models. Due to the quantity of the images the train-test split was an improper evaluation technique. To fully understand the models behavior, the models were trained on the entire subset for 60 epochs to observe when the models would converge. The models were trained with the RMSprop optimizer with initial learning rate of $1e-2$, and the sparse-categorical-cross-entropy loss function. The models were later properly evaluated using the leave one out cross validation strategy.

4.8.1 Multi-Scale Model

The multi-scale concept is a fairly novel approach in the domain of hyperspectral image analysis that aims to improve the performance and generalization capabilities of existing deep CNN models. This approach introduces the concept of multi-scale convolutions, which allows the network to effectively capture fine-grained details as well as high-level contextual information. This multi-scale architecture enhances the model's ability to discriminate between different classes and improves its overall classification accuracy. In conclusion, the introduction of a multi-scale approach presents a significant advancement in hyperspectral image analysis and demonstrates the potential for further improvements in this field. This architecture was inspired by the approach Ren et al. (2022) takes to implementing a multi-scale model.

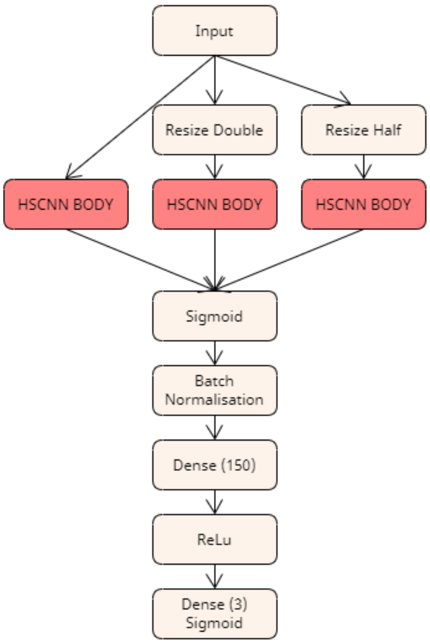


FIGURE 4.3: Multi-Scale Model Architecture

4.8.2 Multi-Kernel Model

The multi-kernel concept is similar to the multi-scale concept, however instead of resizing the images, the images are branching off to multiple convolutional layers with different kernel sizes, combining the features at the end. This approach allows the nueral network to capture different features just like a multiscale model but with less computational cost. The multi-kernel model was implemented to reduce the computational cost of the multiscale model, and to observe if the model would perform better than the multiscale model.

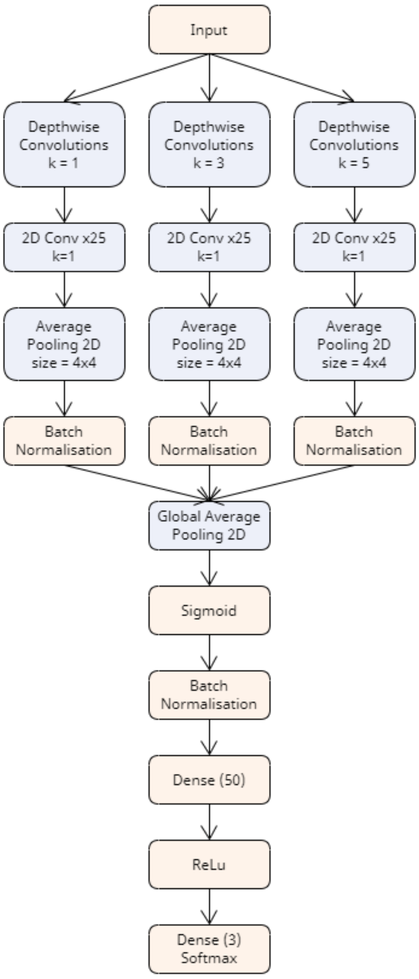

Input
Depthwise Convolutions k = 1
Depthwise Convolutions k = 3
Depthwise Convolutions k = 5
2D Conv x25 k=1
2D Conv x25 k=1
2D Conv x25 k=1
Average Pooling 2D size = 4x4
Average Pooling 2D size = 4x4
Average Pooling 2D size = 4x4
Batch Normalisation
Batch Normalisation
Batch Normalisation
Global Average Pooling 2D
Sigmoid
Batch Normalisation
Dense (50)
ReLu
Dense (3) Softmax

FIGURE 4.4: Multi-Kernel Model Architecture

4.8.3 Introducing the Attention Mechanism

So far the improved models would combine the features from the different branches instead of weighing them. The aim of introducing the attention mechanism is to give more importance to the more informative features. In other words this allows our models to learn which kernel or scale is more informative to aid the task at hand. The attention mechanism implemented is a dot product attention layer, also known as Luong-style attention (Keras, 2023). This layer was added to both the multi-scale and multi-kernel models to observe if the models would perform better.

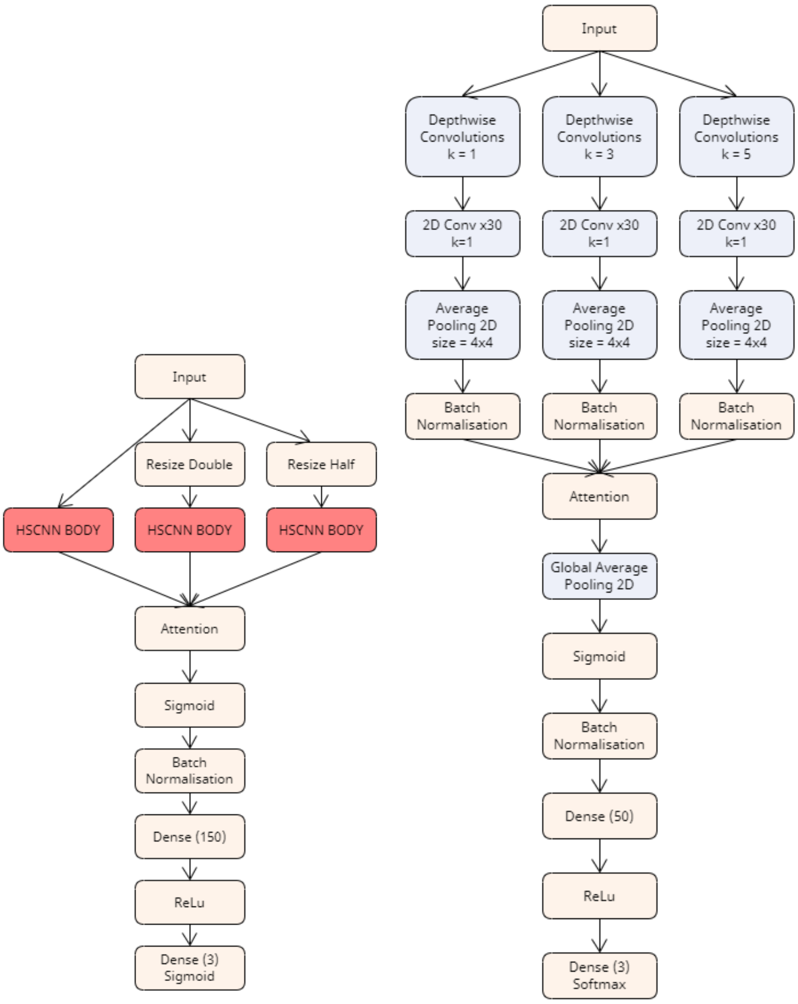


FIGURE 4.5: Multi-Scale Model and Multi-Kernel Model with Attention Mechanism

4.9 Model Parameters

Since one of the aims of this dissertation is to develop a more efficient model to accomodate embedded systems, the models developed were kept as light as possible (see table 4.3).

Model	Parameter count for each band	
	VIS	NIR
HSCNN	34,511	37,367
MKDCNN	29,268	32,348
MKDCNN w/ Attention	30,273	33,773
MSDCNN	97,727	106,295
MSDCNN w/ Attention	118,527	127,095

TABLE 4.3: Model Parameters

4.10 Model Evaluation

Given that the dataset is insufficient to continue carrying out the traditional train-test split approach, the models were evaluated using the LOOCV strategy. This strategy was chosen to ensure that the models were evaluated on all the data available. This strategy has also shed light on the true models performance and their ability to generalise.

Chapter 5

Results and Discussion

5.1 Replicated Results

Whilst training HSCNN on each subset, every 5 epochs a checkpoint was saved and later recalled to be evaluated on the test set. The best checkpoint was selected based on the highest accuracy on the test set. Whilst generating the confusion matrices for all the checkpoints it was found that the model's performance was not consistent as the epochs progressed. The test accuracy was always fluctuating, and the best checkpoint was not always the last checkpoint. This is supported by the learning curves in figure B.1. This leads to doubt in the model's ability to generalise to hyperspectral data, justifying the need to switch the evaluation strategy to LOOCV. The confusion matrices of the best checkpoints are shown in figure 5.1. Where (a) shows an accuracy of 77.8% on the NIR band of the Avocado subset, (b) shows an accuracy of 77.8% on the NIR band of the Kiwi subset, (c) shows an accuracy of 85.5% on the VIS band of the Avocado subset, and (d) shows an accuracy of 54.16% on the VIS band of the Kiwi subset.

For further comparison, the simple machine learning models were trained on the same subsets. See table 5.1. For the first subset (Avocado VIS), the HSCNN model achieved an accuracy of 85.5% with 40 epochs of training, compared to the second best, the decision tree (DT) model which achieved an accuracy of 45.78%. This is a huge gap in performance, and the HSCNN model is clearly superior. For the second subset (Avocado NIR) we can observe the same trend with the HSCNN model achieving 77.8% accuracy compared to the second best performances achieved by DT, RF, and KNN which all

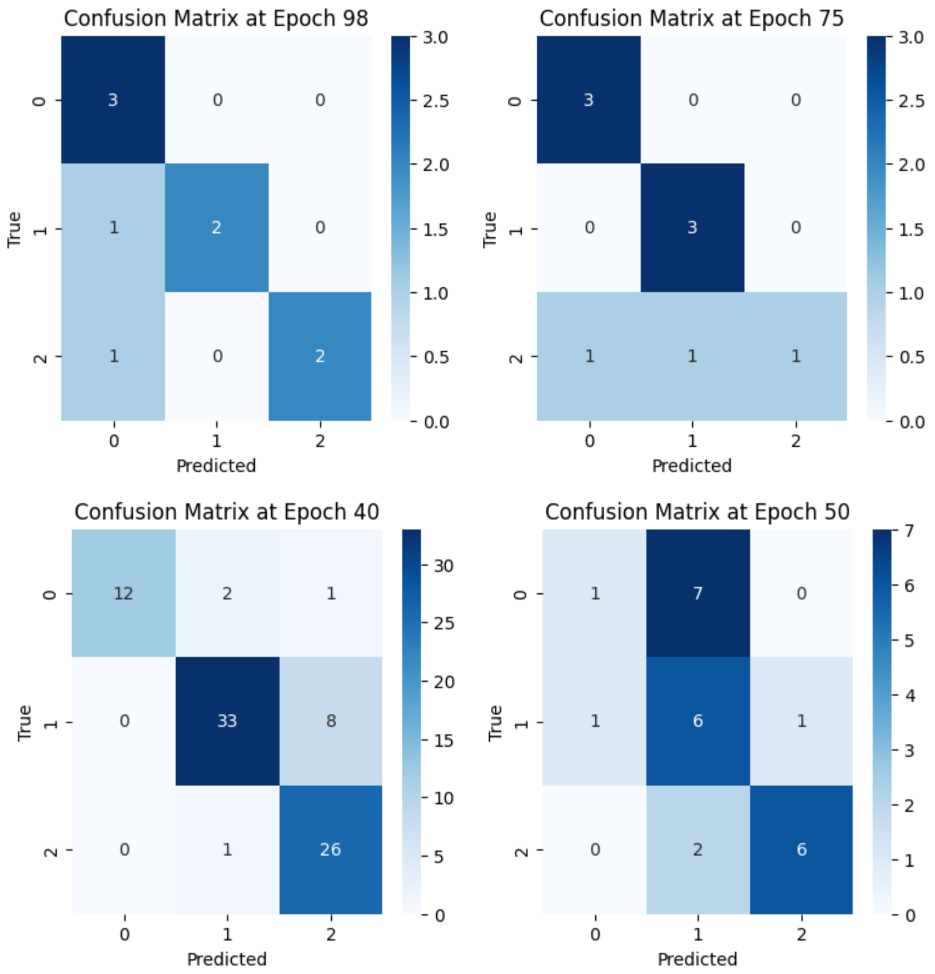


Figure 5.1: Confusion Matrices of the HSCNN model
The confusion matrices of the models trained on (a, top left) the NIR band of the Avocado subset, (b, top right) the NIR band of the Kiwi subset, (c, bottom left) the VIS band of the Avocado subset, (d, bottom right) the VIS band of the Kiwi subset.

achieved 33.33% accuracy. From this point onwards we see a decline in the performance gap between the HSCNN model and the simple machine learning models. For the third subset (Kiwi VIS) It can be seen that the HSCNN model did not generalise as well as the previous subsets achieving an accuracy of 54.16% compared to the second best performance achieved by the KNN model which achieved 50% accuracy. Most interestingly, for the fourth subset (Kiwi NIR) the HSCNN model achieved an accuracy of 77.8% which was on par with the RF model.

Fruit	Avocado		Kiwi	
Bands	VIS	NIR	VIS	NIR
HSCNN	85.5%	77.8%	54.16%	77.8%
(& epoch)	40	98	50	75
DT	45.78%	33.33%	37.5%	33.33%
RF	38.55%	33.33%	45.83%	77.78%
KNN	37.35%	33.33%	50%	44.44%
SVM	36.14%	22.22%	33.33%	33.33%

Table 5.1: Results of HSCNN and the simple machine learning models

5.2 Learning Curves

To procure the learning curves of these models on equal footing, the models were optimised by RMSprop with a learning rate of $1e-2$. The learning curves of the HSCNN, MSDCNN, and MKDCNN models are shown in figures B.1, B.2, and B.3 respectively.

During the process of exploring the domain, an alarming amount of papers did not provide learning curves for their models. This is a critical aspect of the model's characteristics and should be provided to the reader.

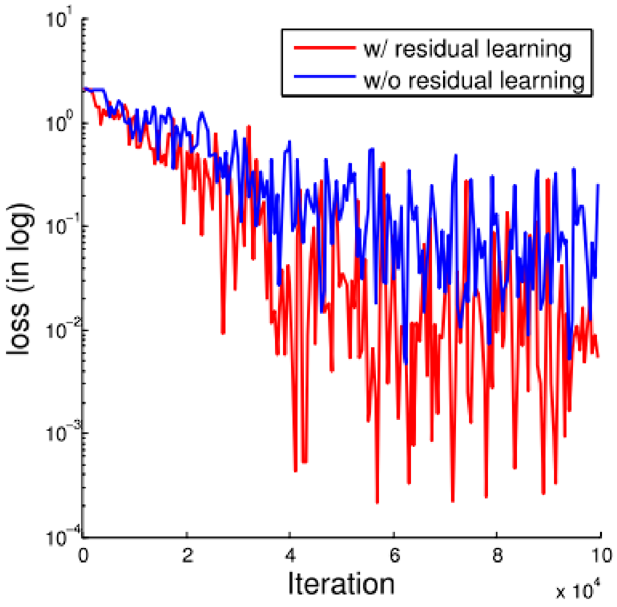


Figure 5.2: Learning curve from Lee and Kwon (2017)

The learning curves in figure 5.2 show the training loss and validation accuracy from their contextual deep cnn model which was trained on the Indian Pines dataset. Their model was trained for 10000 epochs. It is noticed that their training loss values are always fluctuating between 1 and $1e - 4$ which may or may not be a side effect of the model's complex architecture.

From the curve we can observe many fluctuations, which is normal behavior for a model that is learning however it is not the frequency of the fluctuations that matters more than the amplitude of the fluctuations.

In comparison to the learning curves of the models in this study, we can see that the HSCNN model follows the same behavior as the contextual deep cnn model in Lee and Kwon (2017).

The MSDCNN model shows slightly more stable learning curves than HSCNN, indicating a better learning process. However, adding the attention mechanism seems to throw off the learning process as the learning curves show higher losses and lower accuracies.

Similarly the MKDCNN model also has a slightly more stable learning curve than the HSCNN model. When the attention mechanism is added this time the learning curves indicate that the model can keep up with its non-attention counterpart.

5.2.1 LOOCV Results

Based on the learning curve, the models were trained for 40 epochs for each iteration of the LOOCV, giving each model enough epochs to generalise and not overfit whilst also not prolonging a time lengthy process for little to no reward. This strategy was adapted to ensure that the models' ability to generalise to hyperspectral data is not greatly hindered by the lack of data.

Fruit	Avocado		Kiwi	
Bands	VIS	NIR	VIS	NIR
DT	63.15%	55.17%	46.24%	60.53%
RF	74.74%	70.69%	56.45%	61.84%
KNN	66.32%	32.75%	56.99%	55.26%
SVM	60%	48.83%	43%	48.68%
HSCNN	48.42%	44.82%	38.71%	42.11%
MKDCNN	66.32%	44.82%	35.48%	44.74%
MKDCNN w/ Attention	64.21%	44.82%	37.63%	43.42%
MSDCNN	55.8%	44.82%	37.63%	42.11%
MSDCNN w/ Attention	37.37%	24.14%	29.03%	28.95%

TABLE 5.2: Results of the models using LOOCV

For the first subset (Avocado VIS), the RF model achieved the highest accuracy of 74.74%, second highest being a match between KNN and our MKDCNN at 66.32%, and third highest being our MKDCNN with attention at 64.21%. For the second subset (Avocado NIR), the RF model achieves the highest accuracy of 70.69% (again), second highest being the DT model at 55.17%, and third highest being the SVM model at 48.83%. On the third subset (Kiwi VIS), the KNN model scores highest at 56.99%, second highest being the RF model at 56.45%, and third highest being the DT model at 46.24%. And on the last subset (Kiwi NIR), the RF model scores highest at 61.84%, second highest being the DT model at 60.53%, and third highest being the KNN model at 55.26%.

These simple machine learning models are outperforming the deep learning models on every subset. However for all subsets our main focus is how the proposed models perform. Let the HSCNN model serve as the baseline for comparison. For the first subset (Avocado VIS), the HSCNN model achieved an accuracy of 48.42%, the MKDCNN model achieved an accuracy of 66.32%, the MKDCNN with attention model achieved an accuracy of 64.21%, and the MSDCNN model achieved an accuracy of 55.8%. The MSDCNN with attention model performed the worst with an accuracy of 37.37%. On the second subset (Avocado NIR), the HSCNN model achieved an accuracy of 44.82%, which is on par with the MKDCNN, MKDCNN with attention, and the MSDCNN models. Again, the MSDCNN with attention model performed the worst with an accuracy of 24.14%. On the third subset (Kiwi VIS), the HSCNN model achieved an accuracy of 38.71%, the MKDCNN model achieved an accuracy of 35.48%, the MKDCNN with attention and MSDCNN model achieved accuracies of 37.63%. For the last subset (Kiwi NIR), the HSCNN and MSDCNN models achieved an accuracy of 42.11%, the MKDCNN model achieved an accuracy of 44.74%, the MKDCNN with attention model achieved an accuracy of 43.42%. Unsurprisingly, the MSDCNN with attention model performed the worst for the last two subsets at 29.03% and 28.95% respectively. This indicates that none of the deep learning models are generalising well to this dataset. There are many conclusions to be made from these results, as well as theories to be explored. Introducing multi-scale and multi-kernel concepts to an architecture can improve or match it's performance whilst having less parameters like in the case of the MKDCNN and MKDCNN with attention. The reason behind the efficient performance of the MKDCNN model is that it combines multiple feature extraction from different kernel sizes, instead of trickeling the features down the network like in the HSCNN model. This allows it to capture more information from the input data, whilst haveing less parameters than HSCNN. In contrast the MSDCNN model which takes the image at different scales and trickles the features down the network is not as efficient as the MKDCNN model yet has approximately the same performance. This may be due to the fact that the MSDCNN model has three HSCNN models in parallel. The unpredicted downfall in performance of MSDCNN when the attention mechanism is integrated can be explained by the model's learning curve (B.2). Other than the less desirable loss and accuracy curves, the attention mechanism appears to aid to the hesitance of the model at learn the data. This is supported by the fact that the MSDCNN with attention model is the worst performing model in all subsets. The MSDCNN model is seen to have the biggest performance gap when integrating the attention mechanism.

Chapter 6

Conclusion

In this study we expirement with architecture changes to: 1. improve performance ofthe model regardless computation cost and 2. to reduce computational cost whilst maintaining or improving performance. We bring to light the issue in chapter 1, followed by exploring the domain in chapter 2. We then present the methodology in chapter 3, and the documented implementation in chapter 4. We finally show the results in chapter 5, and discuss the limitations and further works in this chapter.

6.1 Contributions

During the project's implementation stage, we develop two new architectures that implement multi-scale and multi-kernel concepts and integrate attention mechanisms. We evaluate these model's performances on the deepHS-Fruit dataset and compare them to the HSCNN model. From the results we discover the effectiveness of the MKDCNN and MKDCNNA on the deepHS dataset where it outperformed the HSCNN model on 2 out of 4 subsets and matched the performance on 1 subset. Similarly the MSDCNN model outperformed the HSCNN model on 2 out of 4 subsets and matched the performance on 1 subset, which is not a significant improvement. Adding an attention mechanism did not improve the performance of the MSDCNN model scoring lower than the HSCNN model on all subsets.

6.2 Limitations

There were many obstacles to be reckoned with in this project. The worst being the lack of puclicly available datasets. Another limitation was the due to the nature of the dataset and the computational resources available, leading to the development of a tailor made data generator. The most significant limitation were the number of images and the inconsistent spatial dimensions of the images in this dataset (see B).

6.3 Further Works

The learning curves of models trained on hyperspectral data show highly fluctuating loss values, this is an issue that needs to be resolved by re-evaluating and improving the model's architecture and hyperparameters. Changes to the architecture could include adding batch normalisation and/or dropout layers, and utilising weight regularisations. All throughout the literature review, it is most common to see an object-wise classification approach to hyperspectral image classification of fruits and other produce. Adopting a pixel-wise classification approach can help demonstrate the non uniformity of a fruit's spectral signature. This is a more realistic approach to the problem as it is more likely that a hyperspectral image of a fruit will contain multiple classes of pixels, for example a partially ripe fruit will have a different spectral signature to a fully ripe fruit. Zhang et al. (2020d) and Liu et al. (2022) have demonstrated this approach for the task of anomaly detection. Said approch can also be more convenient to use on the dataset used in this project as it would not require significant changes such as resizing but rather a simple addition of padding.

Appendix A

Literature Review Figures and Tables

Input	InputSize	MLMethod	Ripe%	Unripe%	Overripe%	Overall%
NDSWI	1	DecisionTree	76.6	55	100	71.2
NDSWI	1	SVM(rbf)	94.1	22.4	0	68
NDSWI	1	MLP	98.2	9.2	0	66.6
NDSWI+WC	2	DecisionTree	92	75.6	100	87.4
NDSWI+WC	2	SVM(rbf)	100	0	0	65
NDSWI+WC	2	MLP	100	0	0	65
670–750nm band	81	DecisionTree	77.9	53.3	69	70.1
670–750nm band	81	SVM(rbf)	93.1	26	0	68.4
670–750nm band	81	MLP	90.7	29.2	0	67.8
Spectrum	1640	DecisionTree	97.7	90.4	88	93.8
Spectrum	1640	SVM(linear)	99.7	94.9	100	98.2
Spectrum	1640	MLP	97.9	92.7	99	96.4
Spectrum+WC	1641	DecisionTree	95.6	90.7	87	93.7
Spectrum+WC	1641	SVM(linear)	99.8	94.9	100	98.3
Spectrum+WC	1641	MLP	97.2	93.3	98	96.1

Table A.1: Strawberry Classification Results (Raj et al. (2022))

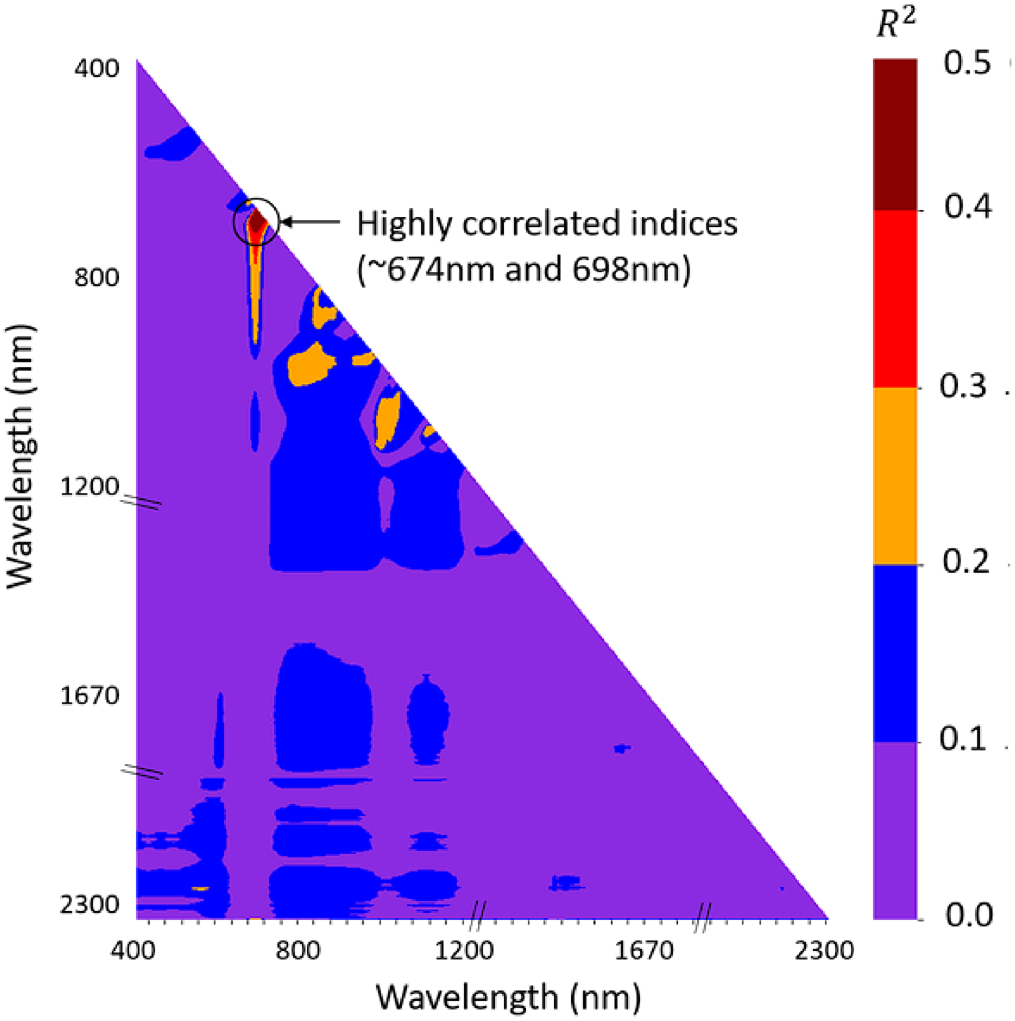


Figure A.1: Correlation Heatmap (Raj et al. (2022))

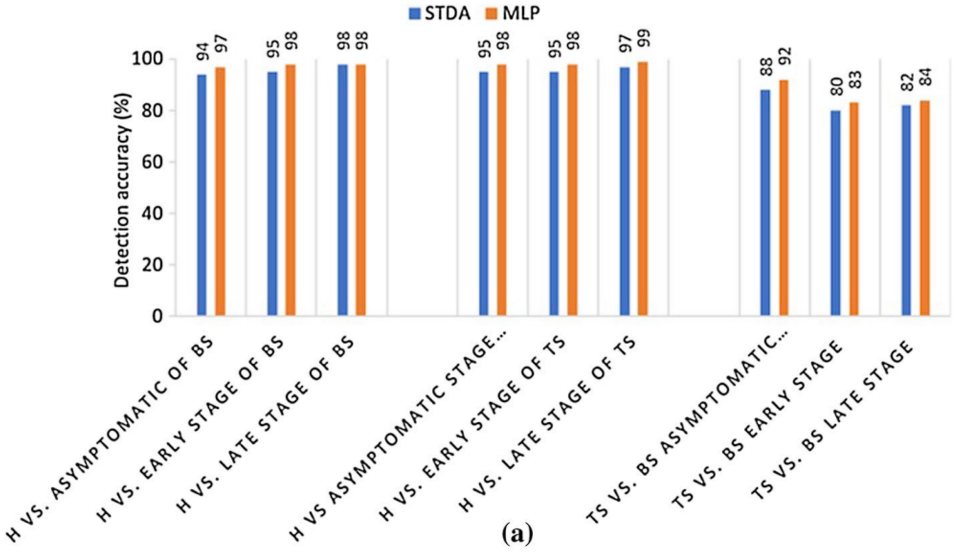


Figure A.2: STDA vs MLP on different scenarios (Abdulridha et al. (2020))

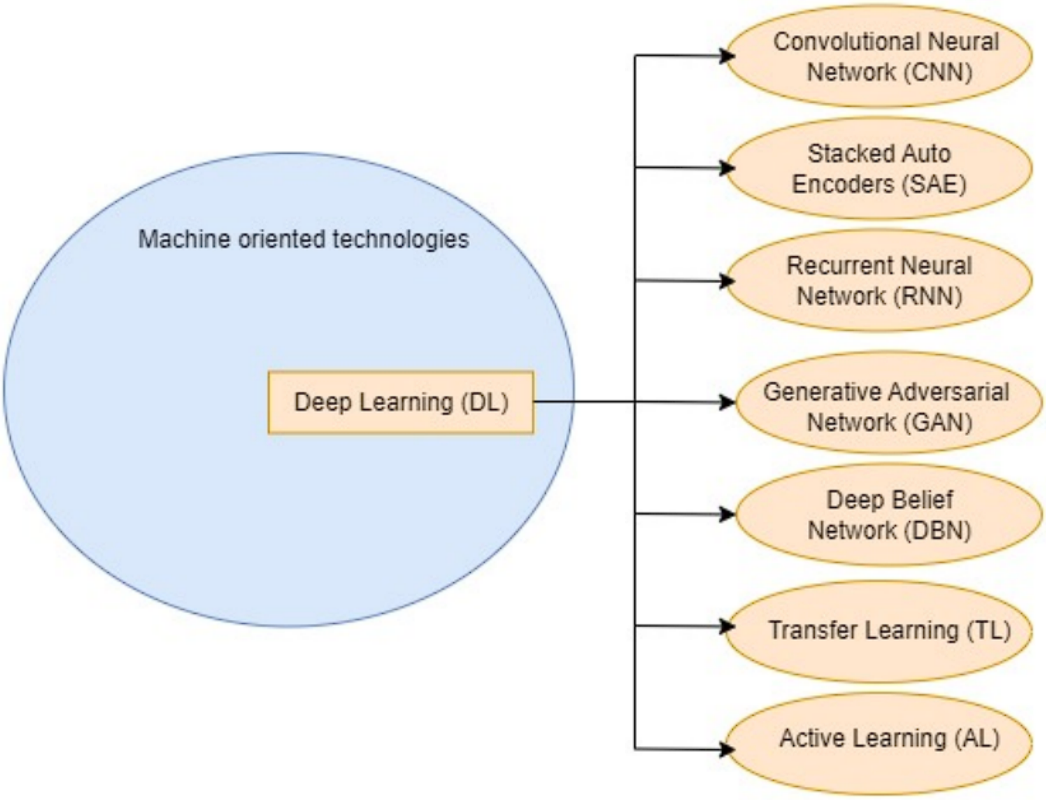

Machine oriented technologies
Deep Learning (DL)
Convolutional Neural Network (CNN)
Stacked Auto Encoders (SAE)
Recurrent Neural Network (RNN)
Generative Adversarial Network (GAN)
Deep Belief Network (DBN)
Transfer Learning (TL)
Active Learning (AL)

FIGURE A.3: Deep Learning Methods (Guerri et al. (2023))

Appendix B

Results

For Avocado and NIR camera, the unique image shapes are: [(219, 94, 252) (220, 94, 252) (278, 93, 252) (277, 93, 252) (245, 182, 252) (241, 182, 252) (243, 212, 252) (240, 209, 252) (329, 189, 252) (308, 197, 252) (281, 210, 252) (277, 210, 252) (316, 185, 252) (319, 190, 252) (299, 192, 252) (311, 191, 252) (337, 190, 252) (329, 190, 252) (363, 178, 252) (361, 186, 252) (302, 193, 252) (307, 191, 252) (312, 197, 252) (305, 194, 252) (303, 191, 252) (281, 208, 252) (279, 209, 252) (296, 202, 252) (287, 201, 252) (289, 200, 252) (315, 188, 252) (311, 186, 252) (291, 190, 252) (307, 182, 252) (308, 180, 252) (293, 190, 252) (286, 177, 252) (286, 174, 252) (268, 194, 252) (305, 188, 252)] For Avocado and VIS camera, the unique image shapes are: [(286, 294, 224) (291, 299, 224) (360, 295, 224) (362, 293, 224) (311, 265, 224) (307, 268, 224) (306, 311, 224) (307, 309, 224) (402, 281, 224) (369, 282, 224) (342, 305, 224) (334, 302, 224) (320, 310, 224) (321, 313, 224) (363, 271, 224) (366, 272, 224) (350, 281, 224) (386, 284, 224) (419, 264, 224) (415, 270, 224) (347, 283, 224) (352, 280, 224) (345, 283, 224) (309, 267, 224) (310, 266, 224) (324, 305, 224) (319, 306, 224) (335, 291, 224) (328, 294, 224) (360, 277, 224) (356, 277, 224) (326, 294, 224) (325, 296, 224) (319, 280, 224) (333, 292, 224) (335, 289, 224) (331, 271, 224) (334, 282, 224) (328, 260, 224) (333, 260, 224) (308, 285, 224) (352, 279, 224) (351, 280, 224) (460, 238, 224) (462, 234, 224) (429, 240, 224) (367, 246, 224) (362, 249, 224) (431, 239, 224) (408, 227, 224) (413, 228, 224) (385, 236, 224) (390, 232, 224) (454, 233, 224) (418, 238, 224) (419, 235, 224) (409, 226, 224) (409, 223, 224) (364, 241, 224) (372, 233, 224) (449, 234, 224) (438, 233, 224) (395, 237, 224) (383, 233, 224) (409, 239, 224) (407, 240, 224) (438, 222, 224) (436, 223, 224) (390, 242, 224) (388, 235, 224) (388, 234, 224) (483, 239, 224) (480, 236, 224) (433, 242, 224) (439, 244, 224) (474, 213, 224) (441, 231, 224) (447, 232, 224) (453, 224, 224) (398, 221, 224) (394, 223, 224) (378, 237, 224) (422, 220, 224) (416, 225, 224) (409, 228, 224) (413, 227, 224) (449, 220, 224) (432, 227, 224) (428, 227, 224) (458, 228, 224) (450, 220, 224) (455, 226, 224)] For Kiwi and NIR camera, the unique image shapes are: [(155, 77, 252) (153, 76, 252) (162, 73, 252) (161, 73, 252) (173, 160, 252) (169, 166, 252) (184, 161, 252) (185, 167, 252) (175, 168, 252) (174, 165, 252) (178, 160, 252) (181, 152, 252) (186, 145, 252) (187, 155, 252) (206, 116, 252) (182, 172, 252) (184, 174, 252) (196, 175, 252) (200, 173, 252) (209, 157, 252) (208, 156, 252) (196, 165, 252) (197, 165, 252) (205, 163, 252) (192, 166, 252) (191, 186, 252) (199, 162, 252) (202, 165, 252) (193, 177, 252) (190, 177, 252) (205, 176, 252) (202, 175, 252) (185, 162, 252) (182, 165, 252) (183, 163, 252) (196, 157, 252) (204, 158, 252) (200, 152, 252) (199, 161, 252) (201, 156, 252) (196, 170, 252) (195, 170, 252) (196, 160, 252) (197, 158, 252) (194, 173, 252) (195, 174, 252) (204, 159, 252) (200, 158, 252) (178, 173, 252) (177, 179, 252) (186, 157, 252) (201, 158, 252) (197, 172, 252) (198, 167, 252)] For Kiwi and VIS camera, the unique image shapes are: [(218, 245, 224) (210, 244, 224) (224, 234, 224) (225, 233, 224) (220, 236, 224) (224, 249, 224) (240, 242, 224) (241, 244, 224) (228, 243, 224) (230, 246, 224) (239, 238, 224) (238, 238, 224) (243, 241, 224) (242, 244, 224) (237, 227, 224) (236, 228, 224) (219, 258, 224) (236, 259, 224) (243, 231, 224) (229, 243, 224) (236, 240, 224) (230, 272, 224) (230, 242, 224) (232, 242, 224) (226, 264, 224) (238, 258, 224) (217, 245, 224) (216, 243, 224) (214, 245, 224) (239, 234, 224) (240, 231, 224) (221, 256, 224) (237, 232, 224) (237, 228, 224) (230, 240, 224) (230, 236, 224) (233, 228, 224) (231, 250, 224) (230, 255, 224) (231, 258, 224) (226, 237, 224) (232, 258, 224) (244, 232, 224) (243, 233, 224) (205, 258, 224) (210, 259, 224) (215, 235, 224) (216, 230, 224) (236, 239, 224) (231, 253, 224) (238, 251, 224) (238, 239, 224) (231, 246, 224) (231, 245, 224) (220, 244, 224) (223, 244, 224) (231, 190, 224) (236, 189, 224) (272, 202, 224) (270, 202, 224) (268, 201, 224) (291, 185, 224) (288, 186, 224) (330, 201, 224) (331, 199, 224) (282, 193, 224) (276, 197, 224) (262, 187, 224) (279, 201, 224) (277, 200, 224) (288, 185, 224) (286, 185, 224) (299, 184, 224) (295, 189, 224) (306, 204, 224) (307, 207, 224) (290, 214, 224) (287, 216, 224) (285, 187, 224) (278, 205, 224) (279, 208, 224) (271, 197, 224) (271, 193, 224) (274, 192, 224) (276, 191, 224) (291, 212, 224) (290, 183, 224) (313, 191, 224) (315, 187, 224) (250, 174, 224) (249, 177, 224) (245, 209, 224) (246, 208, 224) (259, 192, 224) (258, 194, 224) (275, 204, 224) (276, 187, 224) (274, 188, 224) (293, 206, 224) (296, 188, 224) (297, 182, 224) (278, 191, 224) (266, 189, 224) (256, 213, 224) (252, 216, 224) (248, 216, 224) (248, 217, 224) (276, 182, 224) (278, 183, 224) (274, 195, 224) (274, 194, 224) (262, 193, 224) (256, 189, 224) (279, 199, 224) (253, 178, 224) (254, 179, 224) (254, 184, 224) (283, 194, 224) (289, 206, 224) (257, 191, 224) (254, 191, 224) (294, 186, 224) (254, 193, 224) (280, 185, 224) (280, 184, 224) (283, 206, 224) (284, 210, 224) (269, 185, 224) (290, 194, 224) (278, 184, 224) (254, 195, 224) (253, 185, 224) (252, 185, 224)]

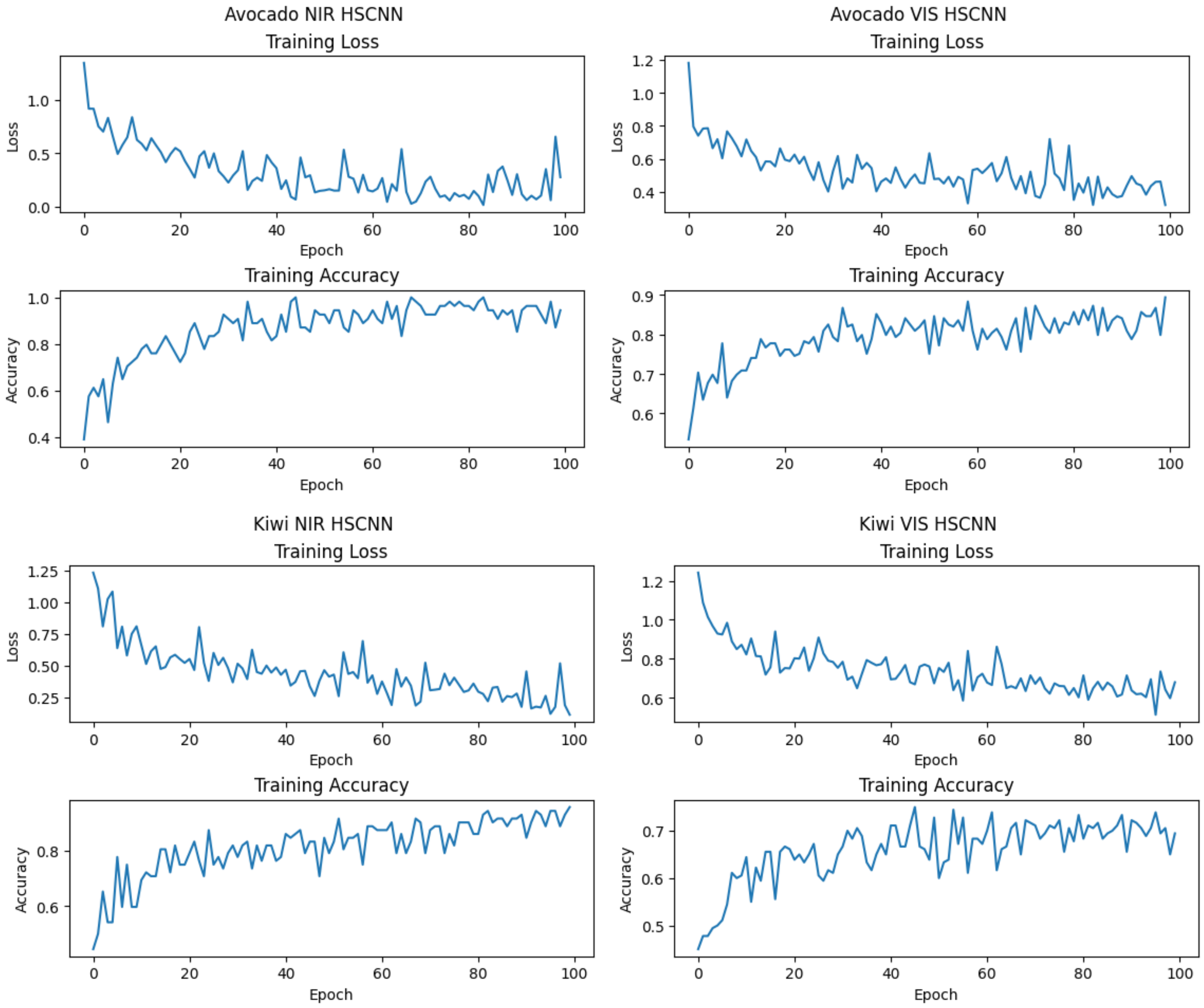


FIGURE B.1: Learning Curves of the HSCNN model

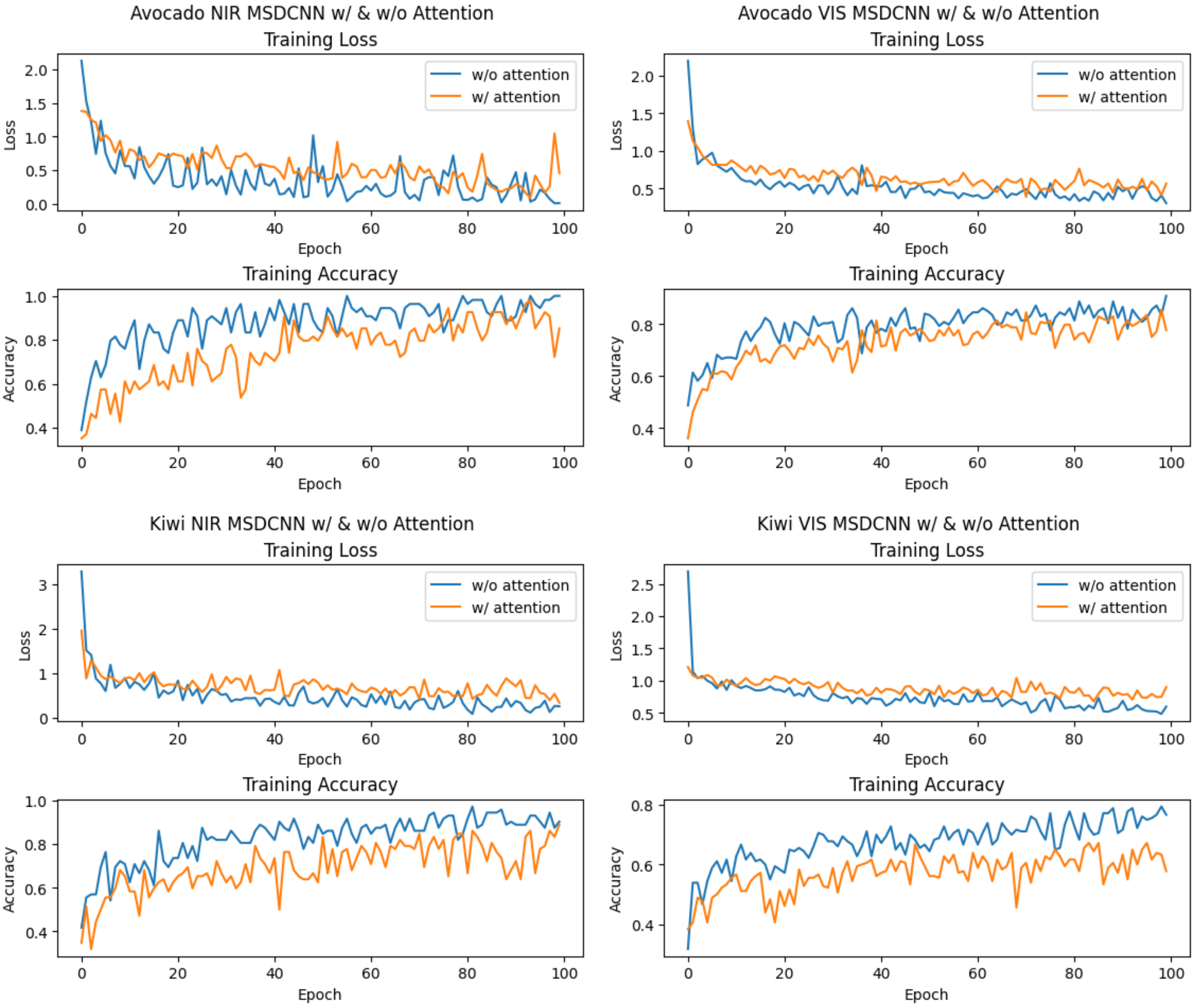


FIGURE B.2: Learning Curves of the MSDCNN model

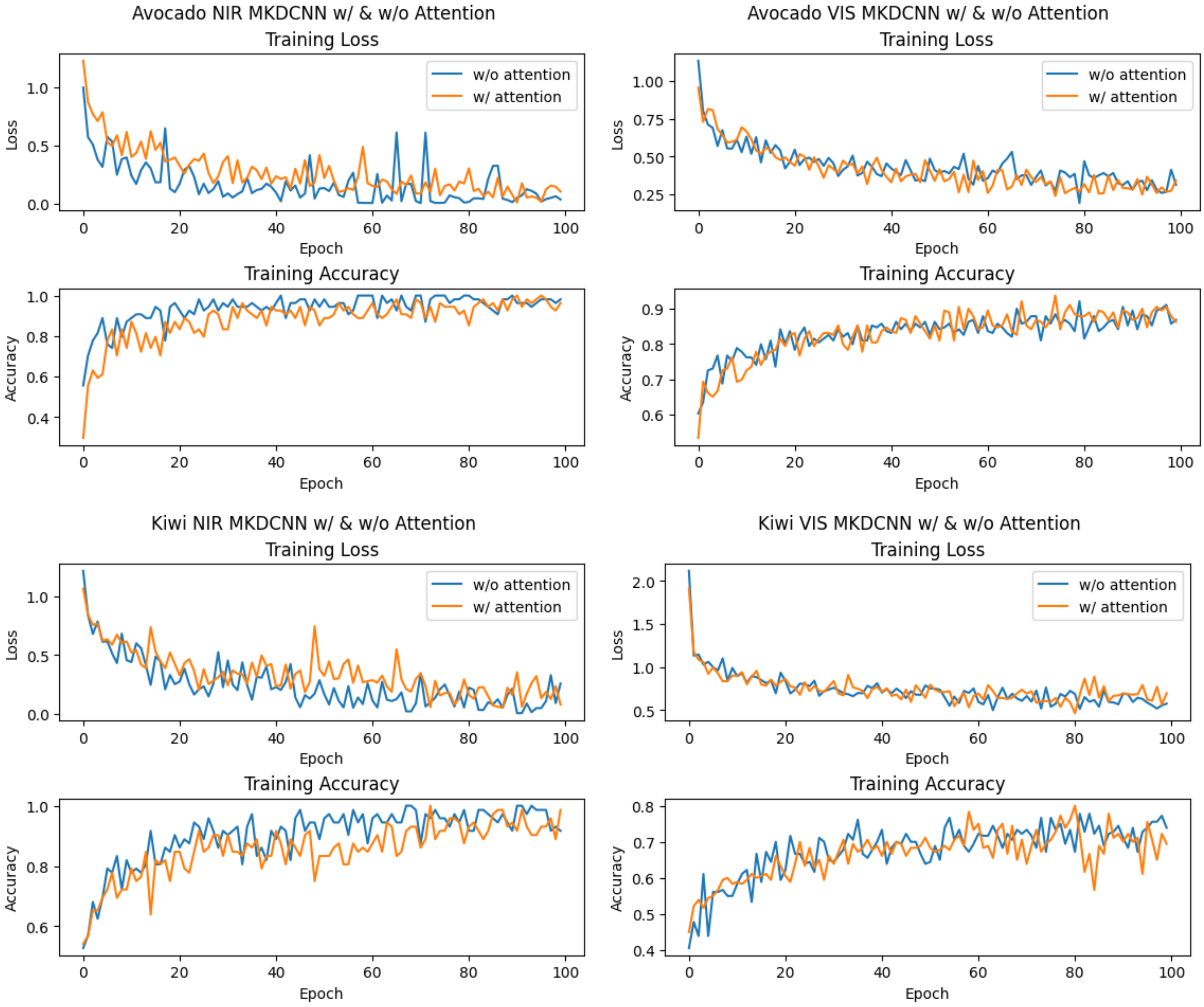


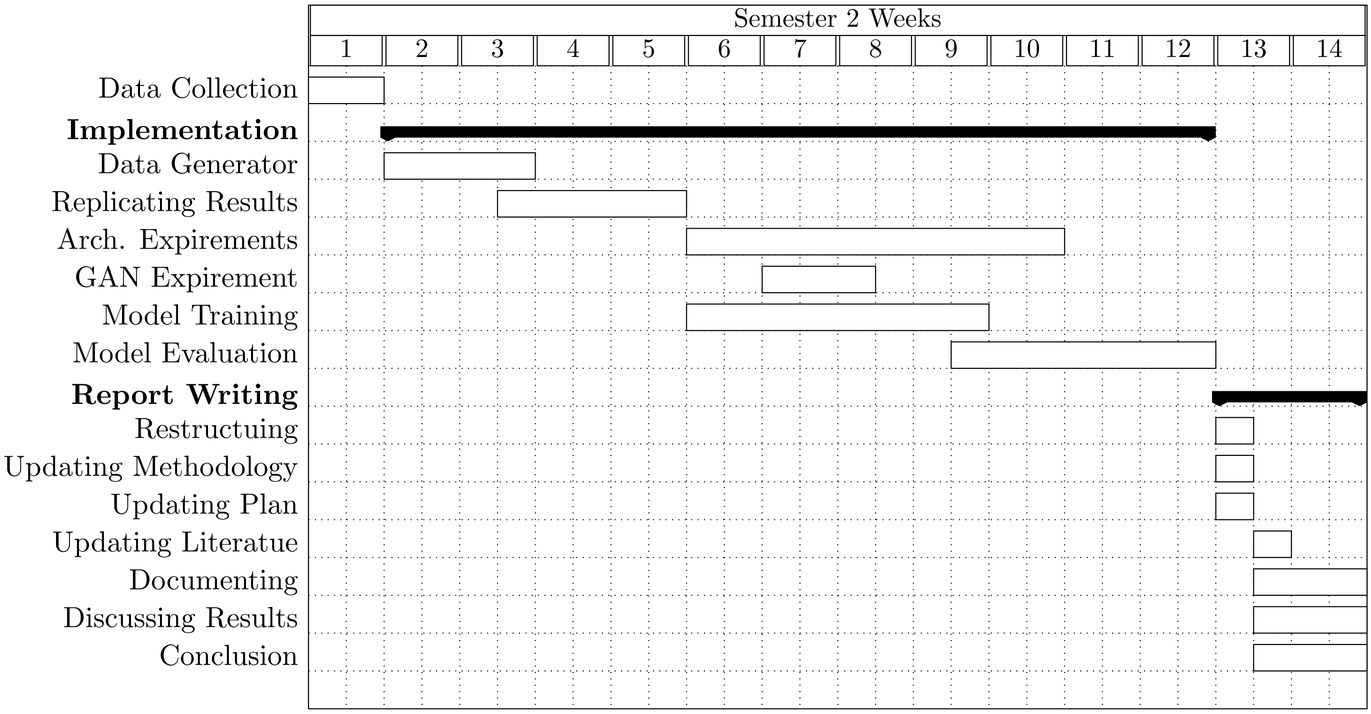
Figure B.3: Learning Curves of the MKDCNN model

Appendix C

Project Plan

C.1 Plan

FIGURE C.1: Gantt Chart



C.2 Risk Analysis

The following is a list of risks that may occur during the project. The risks are divided into three likelihoods and impacts: Low, Medium and High. The risks are also divided into two categories: Technical and Non-Technical. The likelihood is the probability of the risk occurring. The impact is the effect of the risk on the project. For each risk there is a plan of prevention or mitigation depenfing on the risk.

Table C.1: Risk Analysis

Risk	**Likelihood**	**Impact**	**Prevention/Mitigation**
Technical			
Dataset is not large enough	High	High	Use data augmentation techniques to increase the size of the dataset.
Insufficient computational resources	Medium	High	Use cloud services to train models.
Network architecture collapse	Medium	High	Use a pre-trained network architecture.
Tools &/or libraries are not available or outdated	Low	High	Use alternative tools &/or libraries. Or create ones own
Non-Technical			
Loss of code &/or data	Medium	High	Use a version control system to keep track of changes.
Loss of hardware	Low	High	Use cloud services to store data and code.
Lack of time to finish project	High	High	re-evaluate objectives importance and manage priorities
Health	Medium	High	Consult medical professionals and submit an MC form.

C.3 PLES

C.3.1 Professional and Legal Issues

All research works, books, journals, source codes used in this project will be cited and properly referenced. All the code written for this project will be written by me. For this project, we will obtain publicly available datasets, and make my code publicly available. However, in the case we am presented with the chance to procure a better dataset from a private source, we will not make the dataset available to the public, respecting the terms and conditions of the source.

C.3.2 Ethical and Social Issues

There are no ethical issues nor any social issues associated with this project.

Bibliography

Abdulridha, J., Ampatzidis, Y., Kakarla, S. C., and Roberts, P. (2020). Detection of target spot and bacterial spot diseases in tomato using uav-based and benchtop-based hyperspectral imaging techniques. *Precision Agriculture*, 21:1–24.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *ArXiv*, 1409.

Blaschke, T., Hay, G. J., Kelly, M., Lang, S., Hofmann, P., Addink, E., Queiroz Feitosa, R., van der Meer, F., van der Werff, H., van Coillie, F., and Tiede, D. (2014). Geographic object-based image analysis – towards a new paradigm. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87:180–191.

Davur, Y. J., Kämper, W., Khoshelham, K., Trueman, S. J., and Bai, S. H. (2023). Estimating the ripeness of hass avocado fruit using deep learning with hyperspectral imaging. *Horticulturae*, 9(5).

Dou, Z., Gao, K., Zhang, X., Wang, H., and Han, L. (2021). Band selection of hyperspectral images using attention-based autoencoders. *IEEE Geoscience and Remote Sensing Letters*, 18(1):147–151.

Gao, Z., Khot, L. R., Naidu, R. A., and Zhang, Q. (2020). Early detection of grapevine leafroll disease in a red-berried wine grape cultivar using hyperspectral imaging. *Computers and Electronics in Agriculture*, 179:105807.

Gowen, A., O'Donnell, C., Cullen, P., Downey, G., and Frias, J. (2007). Hyperspectral imaging – an emerging process analytical tool for food quality and safety control. *Trends in Food Science & Technology*, 18(12):590–598.

Guerri, M. F., Distante, C., Spagnolo, P., Bougourzi, F., and Taleb-Ahmed, A. (2023). Deep learning techniques for hyperspectral image analysis in agriculture: A review. *arXiv preprint arXiv:2304.13880*.

He, M., Li, B., and Chen, H. (2017). Multi-scale 3d deep convolutional neural network for hyperspectral image classification. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3904–3908.

Hu, J., Shen, L., and Sun, G. (2017). Squeeze-and-excitation networks. *CoRR*, abs/1709.01507.

Ilyas, T., Khan, A., Umraiz, M., Jeong, Y., and Kim, H. (2021). Multi-scale context aggregation for strawberry fruit recognition and disease phenotyping. *IEEE Access*, 9:124491–124504.

Keras (2023). Attention layer. Keras 2.15 API documentation.

Lee, H. and Kwon, H. (2017). Going deeper with contextual cnn for hyperspectral image classification. *IEEE Transactions on Image Processing*, 26(10):4843–4855.

Li, X., Li, R., Wang, M., Liu, Y., Zhang, B., and Zhou, J. (2017). Hyperspectral imaging and their applications in the nondestructive quality assessment of fruits and vegetables. In Maldonado, A. I. L., Fuentes, H. R., and Contreras, J. A. V., editors, *Hyperspectral Imaging in Agriculture, Food and Environment*, chapter 3. IntechOpen, Rijeka.

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2022). A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019.

Linder-Norén, E. (2020). Keras-gan: Keras implementations of generative adversarial networks. last accessed: 2024-02.

Liu, Q., Sun, K., Peng, J., Xing, M., Pan, L., and Tu, K. (2018). Identification of bruise and fungi contamination in strawberries using hyperspectral imaging technology and multivariate analysis. *Food Analytical Methods*, 11(5):1518–1527.

Liu, Y., Zhou, S., Wan, Z., Qiu, Z., Zhao, L., Pang, K., Li, C., and Yin, Z. (2023). A self-supervised anomaly detector of fruits based on hyperspectral imaging. *Foods*, 12(14).

Liu, Y., Zhou, S., Wu, H., Han, W., Li, C., and Chen, H. (2022). Joint optimization of autoencoder and self-supervised classifier: Anomaly detection of strawberries using hyperspectral imaging. *Computers and Electronics in Agriculture*, 198:107007.

Lu, B., Dao, P. D., Liu, J., He, Y., and Shang, J. (2020). Recent advances of hyperspectral imaging technology and applications in agriculture. *Remote Sensing*, 12(16).

Lu, Y., Huang, Y., and Lu, R. (2017). Innovative hyperspectral imaging-based techniques for quality evaluation of fruits and vegetables: A review. *Applied Sciences*, 7(2).

M. ElMasry, G. and Nakauchi, S. (2016). Image analysis operations applied to hyperspectral images for non-invasive sensing of food quality - a comprehensive review. *Biosystems Engineering*, 142:53–82.

Murphy, J. M. and Maggioni, M. (2019). Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion. *IEEE Transactions on Geoscience and Remote Sensing*, 57(3):1829–1845.

Nicolaï, B. M., Beullens, K., Bobelyn, E., Peirs, A., Saeys, W., Theron, K. I., and Lammertyn, J. (2007). Nondestructive measurement of fruit and vegetable quality by means of nir spectroscopy: A review. *Postharvest Biology and Technology*, 46(2):99–118.

Niu, Z., Zhong, G., and Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62.

Paoletti, M., Moreno-Álvarez, S., Xue, Y., Haut, J., and Plaza, A. (2023). Aatt-cnn: Automatic attention-based convolutional neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, PP:1–1.

Qiao, S., Wang, Q., Zhang, J., Pei, Z., and Huang, C. (2020). Detection and classification of early decay on blueberry based on improved deep residual 3d convolutional neural network in hyperspectral images. *Scientific Programming*, 2020:8895875.

Qing, Y. and Liu, W. (2021). Hyperspectral image classification based on multi-scale residual network with attention mechanism. *Remote Sensing*, 13(3).

Rady, A., Ekramirad, N., Adedeji, A., Li, M., and Alimardani, R. (2017). Hyperspectral imaging for detection of codling moth infestation in goldrush apples. *Postharvest Biology and Technology*, 129:37–44.

Raj, R., Cosgun, A., and Kulić, D. (2022). Strawberry water content estimation and ripeness classification using hyperspectral sensing. *Agronomy*, 12(2).

Ren, Q., Tu, B., Li, Q., He, W., and Peng, Y. (2022). Multiscale adaptive convolution for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:5115–5130.

Siedliska, A., Baranowski, P., Zubik, M., Mazurek, W., and Sosnowska, B. (2018). Detection of fungal infections in strawberry fruit by vnir/swir hyperspectral imaging. *Postharvest Biology and Technology*, 139:115–126.

Sun, G., Ding, Y., Wang, X., Lu, W., Sun, Y., and Yu, H. (2019). Nondestructive determination of nitrogen, phosphorus and potassium contents in greenhouse tomato plants based on multispectral three-dimensional imaging. *Sensors*, 19(23).

Swapna, M., Sharma, D. K., and Prasad, D. B. (2020). Cnn architectures: Alex net, le net, vgg, google net, res net.

Taha, M. F., ElManawy, A. I., Alshallash, K. S., ElMasry, G., Alharbi, K., Zhou, L., Liang, N., and Qiu, Z. (2022). Using machine learning for nutrient content detection of aquaponics-grown plants based on spectral data. *Sustainability*, 14(19):12318.

Taye, M. M. (2023). Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. *Computation*, 11(3).

Varga, L. A., Makowski, J., and Zell, A. (2021). Measuring the ripeness of fruit with hyperspectral imaging and deep learning.

Vignati, S., Tugnolo, A., Giovenzana, V., Pampuri, A., Casson, A., Guidetti, R., and Beghi, R. (2023). Hyperspectral imaging for fresh-cut fruit and vegetable quality assessment: Basic concepts and applications. *Applied Sciences*, 13(17).

Wei, L., Zhang, Y., Lu, Q., Yuan, Z., Li, H., and Huang, Q. (2021). Estimating the spatial distribution of soil total arsenic in the suspected contaminated area using uav-borne hyperspectral imagery and deep learning. *Ecological Indicators*, 133:108384.

Weng, S., Han, K., Chu, Z., Zhu, G., Liu, C., Zhu, Z., Zhang, Z., Zheng, L., and Huang, L. (2021). Reflectance images of effective wavelengths from hyperspectral imaging for identification of fusarium head blight-infected wheat kernels combined with a residual attention convolution neural network. *Computers and Electronics in Agriculture*, 190:106483.

Yang, L., Gao, H., Meng, L., Fu, X., Du, X., Wu, D., and Huang, L. (2021). Nondestructive measurement of pectin polysaccharides using hyperspectral imaging in mulberry fruit. *Food Chemistry*, 334:127614.

Yang, T., Wang, Y., and Lian, J. (2024). Plant diseased lesion image segmentation and recognition based on improved multi-scale attention net. *Applied Sciences*, 14(5).

Yang, W., Ma, X., Hu, W., and Tang, P. (2022). Lightweight blueberry fruit recognition based on multi-scale and attention fusion ncbam. *Agronomy*, 12(10).

Yu, X., Lu, H., and Wu, D. (2018). Development of deep learning method for predicting firmness and soluble solid content of postharvest korla fragrant pear using vis/nir hyperspectral reflectance imaging. *Postharvest biology and technology*, 141:39–49.

Zhang, C., Wu, W., Zhou, L., Cheng, H., Ye, X., and He, Y. (2020a). Developing deep learning based regression approaches for determination of chemical compositions in dry black goji berries (lycium ruthenicum murr.) using near-infrared hyperspectral imaging. *Food Chemistry*, 319:126536.

Zhang, C., Zhao, Y., Yan, T., Bai, X., Xiao, Q., Gao, P., Li, M., Huang, W., Bao, Y., He, Y., and Liu, F. (2020b). Application of near-infrared hyperspectral imaging for variety identification of coated maize kernels with deep learning. *Infrared Physics & Technology*, 111:103550.

Zhang, J., Dai, L., and Cheng, F. (2020c). Corn seed variety classification based on hyperspectral reflectance imaging and deep convolutional neural network. *Journal of Food Measurement and Characterization*, 15:484–494.

Zhang, L., An, D., Wei, Y., Liu, J., and Wu, J. (2022). Prediction of oil content in single maize kernel based on hyperspectral imaging and attention convolution neural network. *Food Chemistry*, 395:133563.

Zhang, M., Jiang, Y., Li, C., and Yang, F. (2020d). Fully convolutional networks for blueberry bruising and calyx segmentation using hyperspectral transmittance imaging. *Biosystems Engineering*, 192:159–175.

Zhao, Y., Kang, Z., Chen, L., Guo, Y., Mu, Q., Wang, S., Zhao, B., and Feng, C. (2022). Quality classification of kiwifruit under different storage conditions based on deep learning and hyperspectral imaging technology. *Journal of Food Measurement and Characterization*, 17:1–17.

Zhao, Y., Zhang, C., Zhu, S., Li, Y., He, Y., and Liu, F. (2020). Shape induced reflectance correction for non-destructive determination and visualization of soluble solids content in winter jujubes using hyperspectral imaging in two different spectral ranges. *Postharvest Biology and Technology*, 161:111080.