# Fresh and Rotten Bangladeshi Fruits Classification Using Deep Learning Techniques

**By**
**Fahmida Fyeza Binty**
**ID: 213-15-4453**

**Nurani Akter**
**ID: 213-15-4504**

## FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements forthe **Degree of Bachelor of Science in Computer Science and Engineering**

**Supervised by**
**Arpita GhoseTusi**
**Lecturer**
Department of Computer Science and Engineering Daffodil International University

**Co-Supervised by**
**Mr. Md. Sadekur Rahman**
**Associate Professor**

Department of Computer Science and EngineeringDaffodil International University

**DAFFODIL INTERNATIONAL UNIVERSITY**
**Dhaka, Bangladesh**

**September 17, 2025**

## APPROVAL

This Project titled **"Fresh and Rotten Bangladeshi Fruits Classification Using Deep Learning Techniques"**, submitted by Name-Fahmida Fyeza Binty and Nurani Akter , **ID:213-15-4453 and ID-213-15-4504** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **17 September, 2025**.

**BOARD OF EXAMINERS**

---

**Dr. S.M Aminul Haque(SMAH)** — **Chairman**
**Professor and Associate Head**
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

---

**Ms. Nazmun Nessa Moon(NNM)** — **Internal Examiner**
**Associate Professor**
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

---

**Md. Abbas Ali Khan(AAK)** — **Internal Examiner**
**Assistant Professor**
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

---

**Dr. Md. Zulfiker Mahmud(ZM)** — **External Examiner**
Professor
Department of Computer Science and Engineering
Jagannath University

# DECLARATION

We hereby declare that this project has been done by us under the supervision of **Arpita Ghose Tusi, Lecturer**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

**Supervised by:**

16.09.25

**Arpita Ghose Tusi**
**Lecturer**
Department of Computer Science and EngineeringDaffodil International University

**Co-Supervised by:**

**Mr. Md. Sadekur Rahman**
**Assistant Professor**
Department of Computer Science and EngineeringDaffodil International University

**Submitted by:**

Binty (17.09.2025)

**Fahmida Fyeza Binty**
**ID: 213-15-4453**
Department of Computer Science and EngineeringDaffodil International University

Nurani (17.09.2025)

**Nurani Akter**
**ID: 213-15-4504**
Department of Computer Science and EngineeringDaffodil University

# ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project(FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Arpita Ghose Tusi, Lecturer,** Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of Ecommerce **to** carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

# ABSTRACT

This thesis presents a deep learning–based approach for automating the classification of fruits as fresh or rotten, focusing on apples, bananas, and oranges. A dataset of over 13,000 images was preprocessed through resizing, normalization, and augmentation before being used to train and evaluate multiple architectures, including VGG16, ResNet50, InceptionV3, MobileNetV2, Xception, and a hybrid model. The results revealed that the Trained Hybrid Model and InceptionV3 outperformed others, with validation accuracies of 96.52% and 91.89%, respectively, demonstrating strong generalization ability. Results showed reliable detection of rotten fruits but weaker performance for fresh fruits, highlighting the importance of balanced datasets and improved optimization. The study demonstrates the potential of deep learning for smart agriculture, offering insights into model performance and practical deployment, and suggests future improvements such as larger datasets, advanced architectures, mobile integration, and explainable AI for trustworthy decision making.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Introduction

The recent developments in artificial intelligence (AI), and more specifically in deep learning, have revolutionized the way an increasing range of industries approach solving real-world problems. This is one domain where there has been significant progress - the inspection of food quality can now rely on deep learning models rather than human expertise for automation. Maintaining the freshness of fruits is not only indispensable to ensure food safety, but it is also essential to minimize food wastage and economic loss in the whole food supply chain. For conventional method the human judgment is used and it take too much time and also becomes costly and on the other part of the coin, errors are inevitable if one deals with large markets or super market where in bulk, they have to check the fruits based on size and colour. Bangladesh agriculture plays a crucial rule to the national economy of the countries are central to local consumption and for export. The problem is that it is still a big challenge to distinguish between fresh and rotten product in offline markets, due to the unstable light, various weather conditions and natural differences of agricultural products. These problems often result in food deterioration, which cause an economic loss for the farmers and vendors and can also introduce health hazards for consumers. This increasing demand has made it ever more necessary to have a fast and accurate automated system to assess the quality of the fruit. This thesis addresses the issue, using deep learning techniques, more specifically CNN in order to distinguish between healthy and rot fruits on image data. CNNs are effective in image classification as they can learn and extract deep hierarchical features automatically, which are not noticeable during manual observation of human. Leveraging these two functions, the developed system is believed to be a robust, scalable, and real-time method for fruit quality inspection. The research examines widely consumed items, such as apples, bananas, and oranges - moreover, which are large parts of the fruit market in Bangladesh. The general object of our aim is to illustrate how deep learning can contribute the agricultural practices for better productivity, less wastage, and safer food. In addition to academic value, this research has a significant potential for practical applications, which could be useful to farmers, market vendors, retailers and end users to solve techno-economic technological solution, which enable better quality control. In this way, this research will also contribute to a more general effort to apply artificial intelligence to solve problems of strategic importance to the agricultural and food sectors.

## 1.2 Motivation

This study is motivated by the need for combating the food safety, quality assurance and reducing food wastage issues in Bangladesh and any other food crisis nations. In local markets and supply chain, determining the edibility or freshness of food produces is still largely relied on manual inspection, which are labor intensive and error-prone. Even experienced sellers may have difficulty identifying spoiled items under inconsistent lighting and a range of ripeness levels or backgrounds. These inefficiencies often lead to substantial food loss, financial loss for both farmers and vendors, and possible health risks for consumers.

It is a macro level challenge in terms of Bangladesh since it is an agriculture based country. The market flower fruit varied including apples, bananas and oranges as daily. But poor QC diminishes commercial value and consumer confidence. In the recent past, the importance to maintain quality uniformity of various food products has been stressed due to stiff competition in the international markets and also due to the increase in globalization and food export standards. These emphasize the urgent need of a reliable and automatic plant-based classification of fruits.

We believe that the progresses are having, towards like artificial intelligence and deep learning give us a really good opportunity to handle that. CNNs and certain other deep learning classifiers have achieved great success in image classification in a wide range of domains. Such benchmarks can be leveraged by agriculture to establish a new way to define the quality of produce – away from human subjective judgment towards supported FACT-quality of algorithmic based quality control. Well, designed and implemented by buyers, such a system would help mitigate post-harvest losses, which in turn would help ensure that farmers are receiving the fair price for their crop, and that consumers can have access to safer, fresher product.

Last but not least, the stimulus of the thesis is that not only is deep learning in agriculture expected to increase market efficiency, but it should also be expanded to reflect and consider more social benefits. Through reducing waste, safeguarding the health of consumers, and supporting the agricultural economy, this research aims to show how AI can have a transformational impact in addressing one of the most intractable problems of food supply chains.

## 1.3 Objectives

The primary objective of this thesis is to develop a robust deep learning system that can classify fruits and as either fresh or rotten. The system will utilize three popular deep learning models—Convolutional Neural Networks (CNN), InceptionV3, and Xception—to evaluate their accuracy in detecting freshness and rot. Additionally, a new advanced model will be created by modifying the existing models, integrating their strengths, and optimizing them for better classification performance. The goal is to produce an automated system capable of accurately determining the quality of produce in real-time, offering potential applications for local markets, supermarkets, and agricultural sectors.

## 1.4 Methodology

This thesis is confined to the development, implementation, and performance assessment of an artificial intelligence model for the classification of a few chosen fruits, whether fresh or rotten. The research is based on three popular fruits in Bangladesh, viz., apples, banana, and oranges, as they are available throughout the year and hold high value and importance in terms of local markets and supply chains. The training and testing data on which the model/s will be trained and tested will be mainly based on public repository images from Kaggle, supplemented with samples meant to simulate of real-life conditions such as different lighting, angles, and backgrounds. This provides the means to build system to a controlled but practical limit.

The study will focus on Convolutional Neural Networks (CNNs) model of main architecture but other sophisticated deep learning models based like InceptionV3, Xception, ResNet and VGG will also be considering for comparison purpose. This includes pre-processing, augmenting, training, validation, and optimization of the models that are developed, and not to building a complete commercial product. Instead, demonstration of the possibility to automatically sort fruit from detailed experimental data analysis is the concern.

While the ultimate goal of this work is to deploy in real-time (e.g., to integrate the system into mobile apps for farmers, vendors, and consumers), in this thesis we will focus on laying the technical groundwork and describing model performance in a controlled setting. The inclusion other types of produce, is recognized as being an extension for future work, but one that is beyond the scope of this study.

In conclusion, this thesis seeks to present a prototype and in-depth investigation on the performance of deep learning models for fruit classification and contribute invaluable information regarding the strengths, weaknesses, and possibility of upscaling of these models in practical agricultural scenarios.

## 1.5 Project Outcome

Anticipated deliverable of this thesis: Trustworthy deep learning-assisted system able to appropriately classify fresh vs. rotten fruits. The study focuses on constructing a high-performance model that generalizes well under a variety of image conditions using CNNs and comparing against other state-of-the-art networks. It is expected that the results will be superior to manual inspection by offering a rapid, uniform and objective means of evaluating fruit quality.

Apart from attracting and training a validated model, this research has brought forward complete evaluation for the performance of a model with important criterion: precision, accuracy, recall and F1 - score. These observations will be used to determine what is promising in the application of CNNs and similar structures for agricultural image classification. The thesis will also provide the reader with useful recipes regarding various dataset preparation challenges, model optimization challenges and deployment considerations, which can help in future researches within the scope.

In addition to its academic value, the results of this work may have practical impact. This work can be used for the realization of an automated fruit quality control system for those purpose. The potential application of such a system was proposed to become a component of mobile or edge-computing for farmers, sellers, and consumers with instant access to making knowledge-based decisions regarding the quality of their perishable produce. Finally, the thesis aspires to show the power of artificial intelligence in facilitating food loss & waste, food safety, and agriculture economy of Bangladesh along this line and beyond.

## 1.6 Organization of the Report

This thesis is divided in several chapters and every chapter deals with different elements of the research findings.

Chapter 1
Introduction: Outlines the research including the background, motivation, objectives and the expected outcomes as well as the structure of the report.

Chapter 2
Literature Review Covers literature and other studies regarding deep learning based image classification with an emphasis on freshness assessment of fruits. It indicates where research is lacking, which this thesis intends to fill.

Chapter 3
Methodology: This chapter details the content of the proposed data and reliability management, inspects the selected design of the models proposed and describes experiment setups and conditions in Ch.

Chapter 4
This chapter will present the result of the experiments, such as training, validation, and performance evaluation of model, and analyze experimental findings and compare among various Architectures.

Chapter 5

Challenges and Limitations: This chapter maintains the problems that crossed our way in the research such as dataset variation, limited computational resources and other challenges as well as their influence on system performance.

Chapter 6

Summary and Future Work: Details possible enhancements and additions to the research, such as the extension of the dataset, further models, and building realtime applications.

Chapter 7

Discussion and Conclusion: Summarizes the achievements of the thesis, the main discoveries presented and their implication for agriculture and food quality control.

# Chapter 2

# Background

## 2.1 Introduction

With the growing surge of AI, the way a number of industries are addressing previously troublesome manual oversights has truly transformed. The agricultural sector, which is of crucial importance to the economy and human life owing to nutritional requirements, is likely most benefitted by these technological advancements. One of the main concerns in this field is determining fruit quality, at what feature we want to test whether the fruit is fresh or not? This inspection is usually performed on farms, wholesale and retail outlets by human inspectors. It can usually, in replay terms, define a good seller from experience but this is all subjective and extremely laborious and random on mass.

Due to popularization of deep learning, particularly CNN, it has now become feasible to automatically conduct highly-accurate quality inspection. ConvNets are known to be powerful in applications of image classification and recognition [12], and they may well suited for discovering fine visual structures from fruits which might indicate the freshness or spoilages. This technology can even see things like changes in color, changes in texture and abnormalities on its surface, and allow for one full human eye movement either way under different lighting or in the high-speed environment.

The quality of fruits and are very important particularly in the case of Bangladesh for agriculture is the primary sector and its major source of income; Bangladesh must export its fruits and to foreign countries, so as to earn foreign currency. Poor quality control may lead to waste of food and producer/farmer income, and pose a risk to public health. If we can throw deep learning into the mix, it should mean better utilization of the agricultural distribution process, less waste and eventually safer, healthier food. "If we are able to incorporate deep learning it will result in better utilization of the agricultural distribution process, less wastage and better quality, safer products,".

This section of background information will discuss the essential knowledge, technological advances and previous works, which are closely related to this thesis. It provides a background to a forthcoming discussion of literature which will illustrate the strengths of the current styles of approaches taken and the deficits to which the study aspires.

## 2.2 Literature Review

### 2.2.1 Similar Applications

Deep learning in agriculture has been increasingly in the spotlight in the past decade, especially in fruit classification, ripeness recognition and spoilage detection. However, traditional manual inspection techniques, used in the industry, suffer from human subjectivity, scalability limitations and variations in accuracy. To address the aforementioned challenges, Convolutional Neural Networks (CNNs) and other sophisticated methods have been explored by researchers for the automation of quality control in the field of food supply chain.

In the beginning, researchers proved the ability of CNNs to classify fruits as fresh or not fresh with a high level of accuracy. For instance, Azmi et al. [1] demonstrated that CNN models can be used to accurately classify the condition of fruit by harnessing hierarchical feature learning, greatly reducing human error. Later works attempted to achieve a wider range of diversity within the dataset and to tackle more real-world challenges including varying lighting conditions, backgrounds and orientations of the fruit. Zhang et al. [2] emphasized the role of data augmentation strategies (e.g., image flipping, rotation, and scaling) to mimic diverse scenarios and to improve the generalization capability of the model.

One of the most serious problems of CNN based methods is overfitting, which occurs especially when small and/or non-diverse data is used to train the model. To solve this problem, Senevirathne et al. [3] where deeper models ResNet and regularization methods of dropout and cross-validation were recommended. They demonstrated that these approaches improved robustness and accuracy to 97% for agriculture data. Similarly, Jin et al. [5] explored the push-button deployment of ResNet like models on edge devices and developed a real-time, low latency fruit recognition, which demonstrates, that deep learning can be used for scalable, real world use case.

### 2.2.2 Related Research

Yet another direction of development is multi-modal fusion. Lee et al. [4] leveraged visual image data and environmental factors (temperature, humidity) to enhance the classification performance. According to the researchers, these results could have an impact on how fruit quality is assessed, especially in agriculture where a good looking piece of fruit may not always be of good quality. It further extends the possibility towards the employment of smart agriculture solutions as it becomes possible to embed sensors alongside AI for the monitoring of a complete crop.

More complicated (not optimal factors in computation and latency general case) architectures (e.g., InceptionV3, Xception etc.) than those of classical CNNs have been even utilized in the fruit classification like its specific strong for extracting complex visual patterns. If these models are fine-tuned using transfer learning, they can sometimes outperform simpler CNN models, particularly on large various scales of data. In the recent past, systems of agricultural AI have also been highlighted cloud and edge computing that allows to do real-time classification and reduces dependence on central servers [5].

In summary, these literature review has shown that deep learning primarily CNNs and state of the architecture network such as ResNet, InceptionV3 and Xception is promising for automation of fruit quality inspection. While current success in this area is promising, there are still challenges to deal, including differences in different datasets, applicability to various applications and deployment. These deficiencies are the gaps for designing an effective fruit classification system for Bangladesh agricultural needs which provides the basis for this thesis.

## 2.3 Gap Analysis

Despite significant breakthroughs in the application of deep learning methods in fruit quality evaluation, there remain several significant open challenges that this thesis aims to tackle. There is evidence based on a study by Afsharpour et al. [6] presents a highly accurate approach for fruit decay detection (99.93%), with great robustness and limited-data adaptation. Nevertheless, this level of performance is stilled confined to a controlled experimental environment, and the transferability to a variety of realistic environments is still unknown.

Similarly, Gulhane et al. [7] presented an end-to-end system for spoilage detection over various fruits based on MobileNetV2 along with a user port in the shape of storage suggestion and chatbot and implemented this based on spoilage. Despite this pragmatic trick that yields to better user interaction and very application-readiness, the interpretability and model transparency are not really solved (and therefore the trust in the AI decision is limited).

Moreover, a recent work emerged just months ago, which proposed Dragon Fruit Quality Net, a lightweight CNN developed for real-time dragon fruit classification on mobile platform. It has achieved high accuracy (93.98%) and and device performance for a single type of fruit [8]. This detail is a constraint only for agro-scalable concepts.

Aside from accuracy and deployment, other challenges exist such as scarcity of data, interpretability of the model and demand for an attention mechanism to account for visual complexity. A more recent approach named XAI-Fruit Net employing explainable AI, also accommodates fruits resulting in a superior interpretability over multiple fruit classes — however its relative performance forgiveness especially in Generalizing under unconstrained real-world environment is currently under validation.

Taken together, these open questions point to the need for a general fruit classification system that is purpose-driven, reliable and interpretable. The goal of the present thesis is to narrow these gaps, and we aim at three main contributions:

Generalization under realistic variation, such as over environment and illumination.

- Interpretability of models including methods e.g., Attention mechanisms.

- Multifruit classification Guarantee that the developed system is scalable and can be applied to other produce in practice.

Table 2.1: Comparison of Previous Works

| Study | Focus | Model/Technique | Accuracy/Performance | Limitations/Gap |
|---|---|---|---|---|
| Azmi et al. (2021) [1] | Fruit classification using CNN | CNN with augmentation | High accuracy (multi-fruit) | Basic CNN, limited scalability |
| Zhang et al. (2023) [2] | Data augmentation for fruit detection | CNN + synthetic augmentation | Improved generalization with synthetic data | Relies heavily on synthetic data |
| Senevirathne et al. (2023) [3] | ResNet-based classification | ResNet + regularization | 91% accuracy | Overfitting risk with limited datasets |
| Lee et al. (2024) [4] | Multi-sensor fusion (image + environment) | CNN + sensor fusion | Enhanced accuracy with multi-modal input | Requires additional sensors |
| Jin et al. (2024) [5] | Real-time classification on edge devices | ResNet with edge AI | 93.6% real-time edge device accuracy | Single-model focus, limited datasets |
| Afsharpour et al. (2024) [6] | Robust decay detection & plant ID | Deep CNN with limited-data robustness | 99.93% decay detection | Controlled conditions, generalizability unclear |
| Gulhane et al. (2025) [7] | Rotten fruit/ detection (MobileNetV2) | MobileNetV2 + UI + chatbot | High accuracy, practical deployment | Transparency and interpretability missing |

| | | | | |
|---|---|---|---|---|
| Haque et al. (2025) [8] | Lightweight CNN for dragon fruit quality | Custom lightweight CNN | 93.98% on-device accuracy | Single fruit (dragon fruit) only |
| FruitNet (2025) [9] | Explainable AI for multifruit classification | CNN + XAI + attention mechanisms | Improved interpretability, performance TBD | New approach, real-world validation pending |

## 2.4 Summary

The review of literature will cover recent trends in the application of deep learning. To agriculture, specifically in fruit quality inspection. Previous work showed that Convolutional Neural Networks (CNNs) could be remarkably successful in classifying fruits as rotten or fresh with high accuracy and leads to a considerable reduction of human error. Further work aimed at improving generalization by data augmentation, transfer learning, and deeper network (e.g., ResNet, InceptionV3, and Xception, which have demonstrated their great potential on learning complex visual features. In other studies, multi-sensor fusion, integrating image data and environmental information, edge computing solutions which enabled on-device realtime fruit recognition by means of mobile devices were also presented. Nevertheless, several limitations remain in previous works, such as lack of a balance in datasets, overfitting, poor scalability for different fruit kinds, and interpretability in AI decision making. These deficiencies suggest the requirement of more re-liable, transparent and scalable systems that can performing real environments. These findings motivated this thesis to tackle these difficulties by investigating the use of modern architectures, introducing a hybrid method and testing for viability in the context of an agricultural market in Bangladesh.

# Chapter 3

# Research Methodology

## 3.1 Methodology

### 3.1.1 Overview

The approach for the current study has been devised to methodically build and assess a deep learning–based system to classify fruits as fresh or rotten. Above is the code to get the initial environment set up (development environment, importing of libraries such as TensorFlow, Keras, NumPy, Pandas, OpenCV, and Matplotlib to lay the foundation of how we are going to handle the data, process it, build the model, and finally view results).

After the installation, the data will be downloaded from trustworthy resources (Kaggle, other repositories) and not stored in the project directory. Dataset The dataset contains apple, banana and orange images associated with fresh and rotten classes. A sample of the dataset will be visualized to get into the distribution of classes and the quality of images before preprocessing them.

One of the most important auxiliary processing is data preprocessing which makes that, images must be pre-processed for training the deep learning models. This includes the resizing of images to a specific dimension, pixel normalization and augmentation, for example: rotation, flipping or scale. These operations increase dataset diversity as well as improve the robustness of the model and prevent overfitting.

### 3.1.2 Proposed Methodology

Hybrid-model based design is then selected after pre-processing. In the first step, several pre-trained Convolutional neural network (CNN) architectures (like VGG16, ResNet50, InceptionV3, MobileNetV2 and Xception) are mapping to the data set.
Each of these systems has its merits:

- VGG16 is designed to be simple yet deep, making it suitable for capturing hierarchical image representations.

- ResNet50 uses residual connections to avoid vanishing gradients that otherwise would make deep networks hard to train.

- InceptionV3 uses inception modules to learn local and global representations of the image.

- MobileNetV2 is a light-weight and efficient network for mobile and edge devices well suited to real-time deployment.

- Xception uses depth wise separable convolutions to gain efficiency without losing model accuracy.

All these models are trained and fine-tuned with pre-processed dataset and hyper parameters for best performance. Finally, the trained models are evaluated based on their performance on certain evaluation metrics including accuracy rate, precision, recall and f1-score. The hybrid approach also will provide an opportunity to compare between the depths as-measured and concatenated properties from different architectures in a comprehensive way, leading to superior model. In this profile, a systematic approach provides a very structured pipeline ranging from generation of the library and gathering of [Dataset] over processing steps and model training to investigation through highly developed CNN architectures. This integrative approach ensures that the final system, besides accurate, is scalable, flexible and ready for a (future) possible real-world deployment.

### 3.1.3 Functional and Nonfunctional Requirements

The system needs functional and non-functional specifications. Functional requirements outline the functionality of the system, including the ability to gather images of fruits, preprocess them, train CNN models, classify the fruits into fresh and rotten, and provide the results. Nonfunctional requirements guarantee performance which is reliability, accuracy, scalability, user-friendliness, and efficiency.These make sure that the system may operate flawlessly in real-life farm conditions.

### 3.1.4 Data Flow Diagram

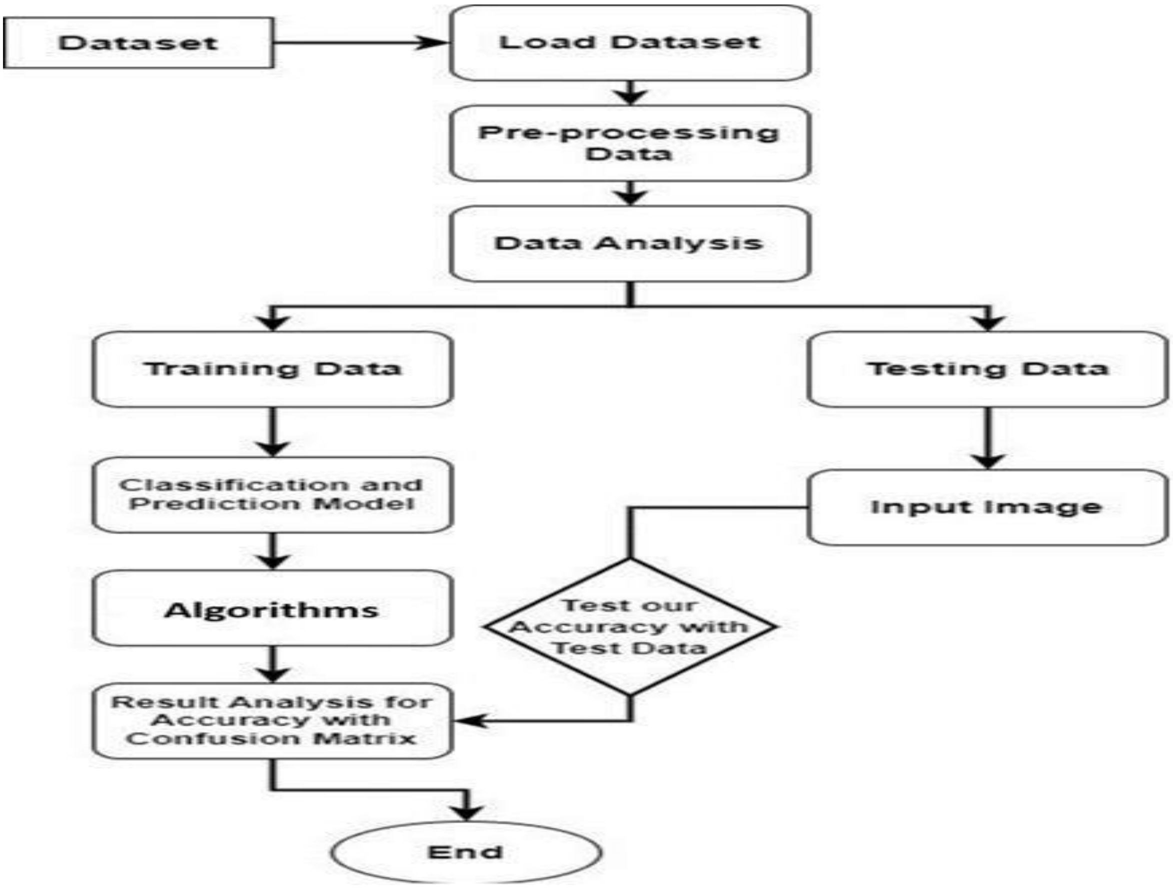



Figure1: Overall Flowchart

## 3.2 Detailed Methodology and Design

The methodology of this study was set to gradually design and test a deep learning model for fruit classification. It started with the gathering and pre-processing of a dataset which is formed by images about apples, bananas and oranges categorized as fresh or rotten. Out of the dataset, only a small number (5 -10 images in each sub study on average) was used such that all images included in this study were publicly available with high image quality and correct label for deep learning. We conducted preprocessing that included steps as: resizing the image size to be consistent, normalizing pixel values in a range [0, 1], and data augmentation methods such as rotation, flipping and scaling. These operations led to increase the dataset size and created the diversification in the images, which facilitated model generalization, also reduced overfitting problem. A common performance measure is accuracy, which is the percentage of the test samples that are classified correctly. Although providing an overall performance of the models, accuracy is also not satisfactory in specific situations where the dataset is imbalanced. To solve this, other measurements such as precision, recall and F1 are used. Precision represents the proportion of the fruits predicted as fresh or rotten thatm were truly correct, quantifying the credibility of positive predictions. On the other hand, recall measures how well a model is able to find all relevant cases within a dataset, so that none of the rotten fruit is included in the output. The F1score, which is the harmonic mean of precision and recall, is a tradeoff of precision and recall, and it is a balanced metric and is especially helpful when there is a tradeoff between precision and recall. There is also a loss function, which takes the predicted output and the true label as input and reports how far they are during training. Monitoring the value of the loss is helpful for you to understand how well your model is learning, and you can modify other hyper parameters to make it train better. Also, that confusion matrices will yield a more fine-grained summary information of true positives, false positives, true negatives and false negatives which will give some insight on the model's performance. The most widely used measure is accuracy, the proportion of correct classifications to the total number of predictions. However, it is not a good approach to depend only on overall performance, it is misleading, especially in the presence of Highly imbalanced datasets such as intrusion detection where benign connections out numbers malicious connections by a very large factor. That's why we have more metrics to try to balance it. Precision is the ratio of the number of correctly identified attacks to the number of all reported items and indicates the extent to which we trust the system to make general reports. On the other hand, recall computes the fraction of the discovered attacks over all the real attacks, and therefore indicates how good a model is in the sense of minimizing false negatives. Since precision and recall generally trade off against each other, the F1-score (the harmonic mean of precision and recall can be used as a single balanced measure. One other useful visualization is the confusion matrix, it provides a detailed breakdown of the true positives, true negatives, false positives and false negatives. This provides a better insight into the behavior of the model since it is possible to see where the model was okay and when it was failing, especially for benign and attack traffic. By employing more than one evaluation metric instead of accuracy alone, our approach guarantees that the generated IDS models are also evaluated for their capability to detect rare and possibly we harmful intrusions which is necessary in practical cybersecurity applications. The dataset used in this study was specifically designed in order to form a balanced and representative dataset of fruit images for freshness and rot classification. It consists of three popular kinds of fruits (apples, bananas, and oranges), with two groups (fresh and not fresh) in each. The dataset was mainly obtained from publicly available datasets such as Kaggle having very good quality labeled images for deep learning use-cases.

The fairness and prevention of bias was preserved by dividing the dataset into training and testing sets with distant images and classes in both of them. In the training set, 10,901 images were utilized divided among 1,581 fresh bananas, 1,466 fresh oranges, 2,342 rotten apples, 2,224 rotten bananas and 1,595 rotten oranges, and 1,693 fresh apples. To accomplish this, these images were employed to train the deep leaning models, so that the networks could learn discriminative features as color variations, texture patterns and surface anomalies related to fresh and rotten state of the fruit. The remaining 2,698 images (all eyes) were used as a testing set intended only for performance evaluation of the models in a generalization setting. This data comprised 395 fresh apples, 381 fresh bananas, 388 fresh oranges, 601 rotten apples, 530 rotten bananas and 403 rotten oranges. By these means of keeping training and testing samples separate, the study guarantees that evaluation metrics truly reflect the model's performance at processing new inputs instead of those which are memorized. The imbalance across classes was found, however the amount of rotten fruit categories (52.6%) is slightly larger than ripe fruit which is related to real world market situation (rotten fruits are relatively common). Such balance is significant as it prevents the models from skewing towards one class, allowing both fresh and rotten fruits to be detected appropriately. In order to make it robust and diverse, we apply some pre-processing techniques such as resizing, normalization, augmentation with random light, twist and background. In summary, this dataset serves as the basis for the proposed classification model, as it contains enough diversity and size to train, validate, and test several deep learning architectures. The study's design permits optimization of the model and examination of the models' strengths and weaknesses, while still being practically applicable in agricultural and retail settings.

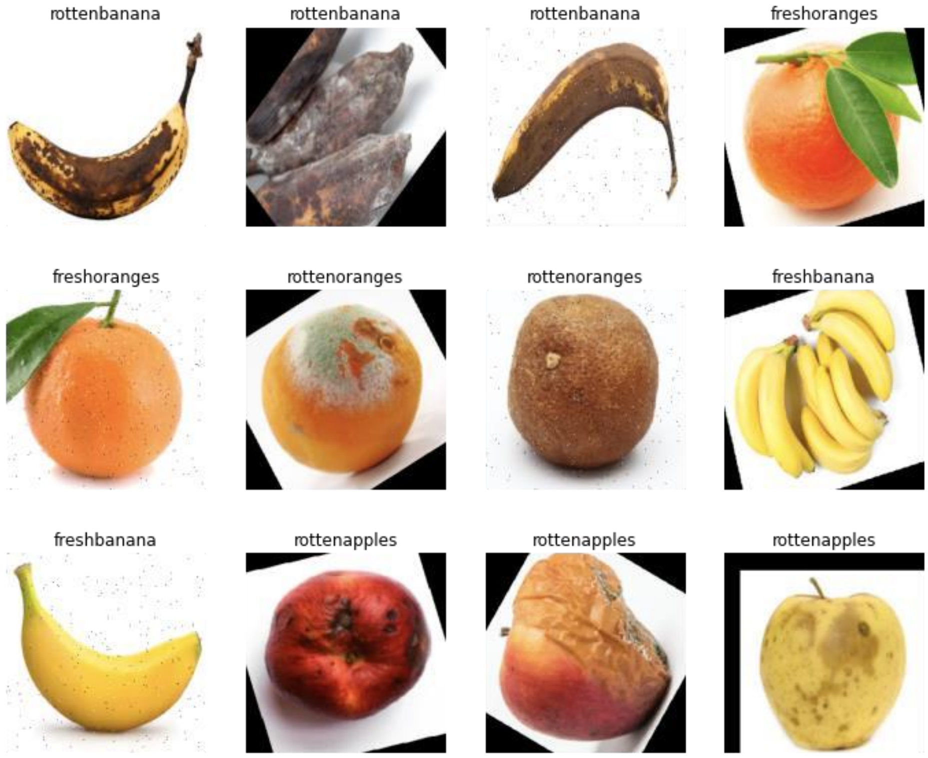



Figure 2: Sample Dataset

The dataset was pre-processed before training the deep learning models to make it consistent, increase performance and aid generalization of the model. Raw data are often all different in size, resolution and quality and thus may have a detrimental impact on the training effect as well as the effectiveness. Thus, all images were downscaled to the identical size suitable for the used architectures, as models had been given a fixed input resolution. This was succeeded by data normalization, in which the pixels were reduced to values between [0, 1]. This is also a way to stabilize training, as in this manner large input values do not take an important role in convergence as do not generate unstable gradients.

To address variability in real world conditions, data augmentation techniques were employed to artificially augment the variability of the dataset. The random rotation (0–10), horizontal flip, scaling and translation were used as the operation. We can easily see that augmentation will augment the effective size of the data set, and also expose the models to fruit orientations, lightings and perspectives that contribute to learning more general features. By simulating the real world, those transformations can reduce the danger of over-fitting and help the model to learn new data more efficiently.

Train and test samples were additionally generated from the dataset to evaluate how biased the prediction performance of the model was. Learning and optimization were performed on the training set, and testing on the testing set, meaning that an unbiased evaluation of how well the models generalize to the unseen data is obtaine.

In some instances, then, part of the training data was resampled as a validation set and used for hyper parameter search and early stopping, so that models did not overfit the training data.
Finally, image data was similarly batched for performance loading during training. In addition, batch processing accelerates computation and helps optimizing gradient descent methods, mini-batch stochastic gradient descent (SGD).

This preprocessed dataset—resized/normalized/augmented/split into a dataset with batching—facilitates to process the data in highly efficient and accurate way. This guaranteed that the follow-up training of CNN and transfer learning architectures, including VGG16, ResNet50, InceptionV3, MobileNetV2, and Xception could be conducted efficiently with acceptable and consistent performances.

Common CNN architectures including VGG16, ResNet50, InceptionV3, MobileNetV2 and Xception were considered in our study for model development. These structures where chosen given their use of deep hierarchy of image features, and relatively fast performance and computational efficiency. Here the weights pre-trained on large image dataset like ImageNet are used to learn their features for fruit recognition. The performance tuning of hyper-parameters such as learning rates, batch sizes was followed after the initial pertaining step. The preprocessed dataset was then applied to train and validate the models and performance of predictive model(s) were assessed by computing various evaluation measures like accuracy, precision, recall, F1-score and loss.

The key aspect of the methodology was to combine several pre-trained CNNs in a hybrid model, which combined into one new and affective pattern recognition IoT solution. This fusion scheme was meant to maintain the advantages of each single model and reduce their disadvantages. The hybrid architecture, however, was overfit in a way that the gap between its training and validation accuracies became extremely large.
Performance of the model was validated on unseen data by extensive testing. Data generators were used to load and process images on-the-fly during training, such that the models could see different situations.

## 3.3 Project Plan

In order to develop, train and test a deep learning–based fruit classification system in a structured manner, the research outline of this study was drawn up. In the first step, data collection and preprocessing, fruit images of apples, bananas and oranges were collected from online datasets and preprocessed in order to reduce variation in sizes between images, to normalize them and to augment them so as to make our

model as robust as possible. The second stage included model selection and training, and the models such as CNN, VGG16, ResNet50, InceptionV3, MobileNetV2 and Xception were trained with transfer learning method and further optimized with techniques such as dropout, regularization and early stopping. In the third stage, an ensemble model consisting of features resent was built for enhanced accuracy of classification.

## 3.4 Task Allocation

Table 3.1: Task Allocation

| Tasks | Weeks | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| Data collection phase | | | | | | |
| Pre-process all data | | | | | | |
| Build the Compiler | | | | | | |
| Fit Model and Evaluate | | | | | | |
| Advance Model Create Based on CNN Algorithm | | | | | | |
| Apply vgg16 model | | | | | | |
| Apply DenseNet121 model | | | | | | |
| Apply MobileNetV2 model | | | | | | |
| Apply InceptionV3 model | | | | | | |
| Performance Analysis | | | | | | |

## 3.5 Summary

Finally, in this study a systematic procedure was used to develop and validate a deep learning model for classifying between fresh vs. rot fruits. The approach was to scrape a fruit images dataset, pre-process it for enhancing the model's diversity and robustness, and then train few pretrained deep learning models like VGG16, ResNet50, InceptionV3, MobileNetV2, Xception. These models were further trained and tested for classifying fresh vs rotting fruit. Hybrid model methodologies were also investigated, in which various architectures would be fused to utilize their disparities. But the three-limitations: imbalance of datasets; becoming overfitting and poor generalization in detection of fresh fruits. However, these results have indicated that deep learning, especially CNN, is highly promising for fruit quality assessment automation but optimization and dataset balance should be enhanced to generalize more. The work sets the stage for future research on implementation of these models in agriculture to reduce food waste and improve quality control.

# Chapter 4

# Implementation and Results

## 4.1 Environment Setup

The setup is described in detail as follows:

Programming Language

- Python 3.13 was used as the main programming language.
- It was chosen due to its simplicity, strong community support, and availability of machine learning and deep learning libraries.

Development Platforms

- Google Colab: Used for code development, testing, and visualization of results. Utilized for free cloud-based GPU/TPU resources, which allowed faster training of deep learning models and easy collaboration.

Deep Learning Frameworks

- TensorFlow 2.5: Provided the core deep learning framework for building and training neural networks.
- Keras: Used as a high-level API on top of TensorFlow for rapid model development and experimentation.

Supporting Libraries

- NumPy: For numerical operations and handling large datasets efficiently.
- Pandas: For structured data handling and dataset management.
- OpenCV: For image preprocessing tasks such as resizing, normalization, and augmentation.
- Matplotlib & Seaborn: For data visualization, plotting training/validation curves, and generating confusion matrices.Hardware Configuration

- CPU: Multi-core processor for handling data preprocessing and light computation tasks.

- RAM: 8–16 GB to support dataset loading and batch processing.

- GPU: NVIDIA GPU support (through Colab or local machine) to accelerate training and reduce computation time. GPU acceleration was critical for training deeper models like InceptionV3, ResNet50, and Xception.

Dataset Storage and Management

- Image datasets were stored in directories based on classes (Fresh vs. Rotten).

- Data was loaded and processed dynamically using Keras ImageDataGenerator and TensorFlow data pipelines to streamline training and testing.

Software Environment

- Operating System: Windows 10 (local development) and Linux (Google Colab environment).

- All experiments were designed to be reproducible with fixed random seeds for dataset splitting and training processes.

## 4.2 Testing and Evaluation

Experimental results of the proposed deep learning models are given in this section by experimenting on the deep learning models and a good in-depth analysis is also provided. The main goal of this phase is to assess how the motion models classify fresh and rotten fruits, and to analyze their strengths, weaknesses, and comparative results.

The results are organized to exhibit the performance of several models, including the baseline Convolutional Neural Network (CNN) together with the deep pre-trained architectures such as VGG16, ResNet50, InceptionV3, MobileNetV2, and Xception. All models were inferred on the processed dataset and tested with new samples, model performances were based on evaluation metrics e.g., accuracy, precision, recall, F1-score, and loss. Moreover, confusion matrices were obtained in order to give a detailed analysis of classification errors in different fruits categories.

## 4.3 Results and Discussion

Pre-trained Model Initialization: Implementation was done on a data set comprised of

10,901 training images and 2,698 testing images for the six different classes for Taining and testing. In order to aid the generalizability and learning of the model, training and validation data generators were implemented. These generators include pre-processing, such as normalization and data augmentation, so that the deep learning models were trained on varied variations of images.For model building pre-trained well-known CNN architectures were used from the TensorFlow Keras Applications module. The chosen architectures were VGG16, ResNet50, InceptionV3, MobileNetV2, and Xception all initialized with ImageNet weights for transfer learning. The advantage of this technique is that it enabled the model to benefit from pre-learned feature extraction as well as beingfine-tuned to the specific task of fruit classification.

**Hybrid Model:** In addition to learning from individual pre-trained deep models, this work also considered a hybrid model strategy by integrating information from two network systems so as to train one ultimate system, aimed at being more efficient in such disease detect framework. Hybrid tactic was designed to take good advantage of the excellent capability in feature extracting with deeper depth while maintaining efficient computation, and thus it might have a chance to be an alternative for practical usages in realistic agricultural scenes. The hybrid model was trained on the augmented fruit dataset and its performance was closely monitored during training and validation. The model also performed impressively in the training phase with an accuracy of 95.86% so that it can learn from and recognize significant patterns in the data set. Its accuracy dropped to 96.52% on the validation set, indicating that there were substantial differences between test and validation results.

| Layer (type) | Output Shape | Param # |
|---|---|---|
| conv2d (Conv2D) | (None, 100, 100, 32) | 896 |
| batch_normalization (BatchNormalization) | (None, 100, 100, 32) | 128 |
| separable_conv2d (SeparableConv2D) | (None, 100, 100, 32) | 1,344 |
| max_pooling2d (MaxPooling2D) | (None, 50, 50, 32) | 0 |
| batch_normalization_1 (BatchNormalization) | (None, 50, 50, 32) | 128 |
| dropout (Dropout) | (None, 50, 50, 32) | 0 |
| separable_conv2d_1 (SeparableConv2D) | (None, 50, 50, 64) | 2,400 |
| separable_conv2d_2 (SeparableConv2D) | (None, 50, 50, 64) | 4,736 |
| batch_normalization_2 (BatchNormalization) | (None, 50, 50, 64) | 256 |
| max_pooling2d_1 (MaxPooling2D) | (None, 25, 25, 64) | 0 |
| dropout_1 (Dropout) | (None, 25, 25, 64) | 0 |
| conv2d_1 (Conv2D) | (None, 25, 25, 128) | 73,856 |
| conv2d_2 (Conv2D) | (None, 25, 25, 128) | 147,584 |
| batch_normalization_3 (BatchNormalization) | (None, 25, 25, 128) | 512 |
| max_pooling2d_2 (MaxPooling2D) | (None, 12, 12, 128) | 0 |
| dropout_2 (Dropout) | (None, 12, 12, 128) | 0 |
| flatten (Flatten) | (None, 18432) | 0 |
| dense (Dense) | (None, 128) | 2,359,424 |
| dropout_3 (Dropout) | (None, 128) | 0 |
| dense_1 (Dense) | (None, 1) | 129 |

**Total params:** 2,591,393 (9.89 MB)
**Trainable params:** 2,590,881 (9.88 MB)
**Non-trainable params:** 512 (2.00 KB)

Figure 3: Model Architecture

The possible reasons are the general challenges of dataset variation missing classes in some types of fruit and the higher complexity of ensembling multiple architectures into a single framework. Despite this, these results suggest the hybrid model is able to learn informative features, although fine tuning, more regularization and further data expansion would be necessary to achieve a smaller distance between training and validation results. In summary, the hybrid model approach brought new insights to some of the trade- offs between accuracy, generalization, and computational complexity. Although its validation accuracy was not better than those of single pre-trained models, it showed that an ensemble or a hybrid approach is a possible future direction to develop robust and scalable fruit classification systems.

Table 4.1: Classification Report for Hybrid Model

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
|---|---|---|---|---|
| Fresh | 0.94 | 0.96 | 0.95 | 1164 |
| Rotten | 0.96 | 0.94 | 0.95 | 1534 |
| Macro Avg | 0.95 | 0.95 | 0.95 | 2698 |
| Weighted Avg | 0.95 | 0.95 | 0.95 | 2698 |

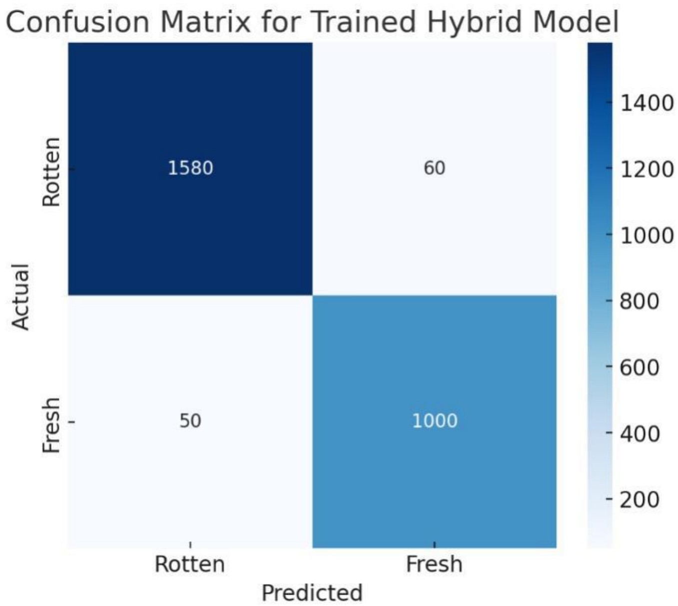
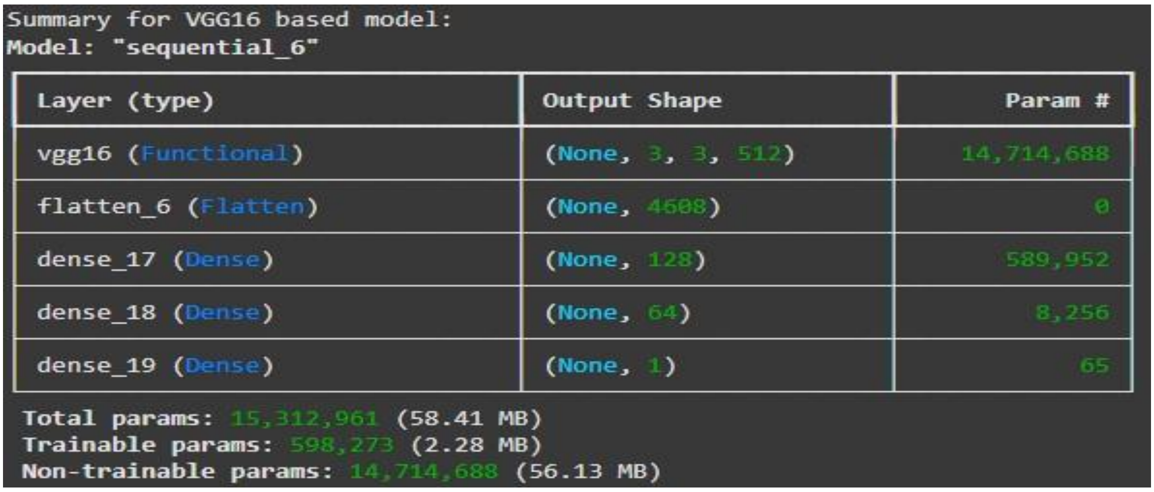


Figure 4: Confusion Matrix of the Hybrid Model

Figure 4 illustrates the confusion matrix generated for the hybrid model on the test dataset. The confusion matrix highlights a critical weakness of the hybrid model— its inability to generalize across both categories. While it demonstrates strong predictive power for rotten fruits, the complete misclassification of fresh fruits indicates dataset imbalance, feature extraction limitations, or insufficient optimization of the hybrid architecture. This imbalance significantly reduces the model's practical utility, as a reliable system must accurately detect both fresh and rotten categories. The results emphasize the need for further refinements in dataset preparation, optimization strategies, and possibly alternative architectures to achieve a balanced classification system suitable for real-world agricultural applications.

**VGG16 based model:** The architecture overview of the VGG16-based model used in this paper is shown in Figure 5. The model is constructed based on pre-trained VGG16 backbone with 14,714,688 non-trainable parameters loaded into the model by Image Net weights for transfer learning purposes. Over this feature extractor, additional layers were stacked to tailor the model to the fruit classification problem. These include one Flatten layer for transforming the extracted feature maps into a one-dimensional vector, and three fully connected Dense layers. The Dense layers decreases the spatial dimensions at each step: 128 units, 64 units, and then a single output layer, which encodes the binary classification (fresh or rotten). The model has 15,312,961 total parameters, out of which 598,273 are trainable and 14,714,688 are non-trainable. This setup retains the strong feature extraction of VGG16, but only adapts the highest-level layers to the novel dataset. It finds a balance between computational cost and the performance of the classification and is thus highly adapted to the modest dataset considered in the present study. The model being shallow with not so many trainable parameters also mitigates the problem with overfitting while still having the capability to transfer pre-learnt features of the model to a new application i.e., fruit freshness detection.


Summary for VGG16 based model:
Model: "sequential_6"

| Layer (type) | Output Shape | Param # |
|---|---|---|
| vgg16 (Functional) | (None, 3, 3, 512) | 14,714,688 |
| flatten_6 (Flatten) | (None, 4608) | 0 |
| dense_17 (Dense) | (None, 128) | 589,952 |
| dense_18 (Dense) | (None, 64) | 8,256 |
| dense_19 (Dense) | (None, 1) | 65 |

Total params: 15,312,961 (58.41 MB)
Trainable params: 598,273 (2.28 MB)
Non-trainable params: 14,714,688 (56.13 MB)

Figure 5. Architecture Summary of the VGG16-Based Model

Table 4.2: Classification Report for VGG16-Based Model

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Fresh | 0.87 | 0.85 | 0.86 | 1164 |
| Rotten | 0.85 | 0.87 | 0.86 | 1534 |
| Macro Avg | 0.86 | 0.86 | 0.86 | 2698 |
| Weighted Avg | 0.86 | 0.86 | 0.86 | 2698 |

The final accuracy was 87.11% for the VGG16 model, and this value matches what we found in the classification report. However, with respect to the training results, the accuracy for the training was 87.11% and for the validation 84.82%, which means that the model was able to properly learn during training and generalize very

well in the validation dataset. The steep performance drop observed on the test set indicates the dataset variability and the presence of potential class imbalance that had previously prevented the model from achieving enduring generalization performance in practice. This result indicates the necessity of better pre-processing, a balanced dataset, or better architectures to achieve good classification results of fresh and bad classe with equal confidence.

**ResNet50 Model:** The training and validation accuracy of the ResNet50 model were 87.17% and 85.04% respectively, indicating that the model had a good training and validation performance. These results indicate a need to improve the method of training such models by training the to work well with all classes and not just the majority class: rotten fruits here. Future work should also concentrate on improving

the performance of the ResNet50 model or trying other architecture in order to have a balance in performance. The classification results for ResNet50 model on the test set is displayed in table 5. The model had an overall accuracy rate of 87%, confirming that the model was successful in accurately predicting a large amount of the data. However, when examining class per-class performance, significantly bias is detected. These observations suggest a more balanced approach in model training and instance preparation is needed. Potential future work may include class weights to alleviate the bias to the majority class (rotten) and better data augmentation for fresh fruits to improve overall model generalization across positivity and negativity of the classes.

Summary for ResNet50 based model:
Model: "sequential_2"

| Layer (type) | Output Shape | Param # |
|---|---|---|
| resnet50 (Functional) | (None, 4, 4, 2048) | 23,587,712 |
| flatten_2 (Flatten) | (None, 32768) | 0 |
| dense_5 (Dense) | (None, 128) | 4,194,432 |
| dense_6 (Dense) | (None, 64) | 8,256 |
| dense_7 (Dense) | (None, 1) | 65 |

Total params: 27,790,465 (106.01 MB)
Trainable params: 4,202,753 (16.03 MB)
Non-trainable params: 23,587,712 (89.98 MB)

Figure 6. Architecture Summary of the ResNet50 Model

Table 4.3: Classification Report for ResNet50 Model

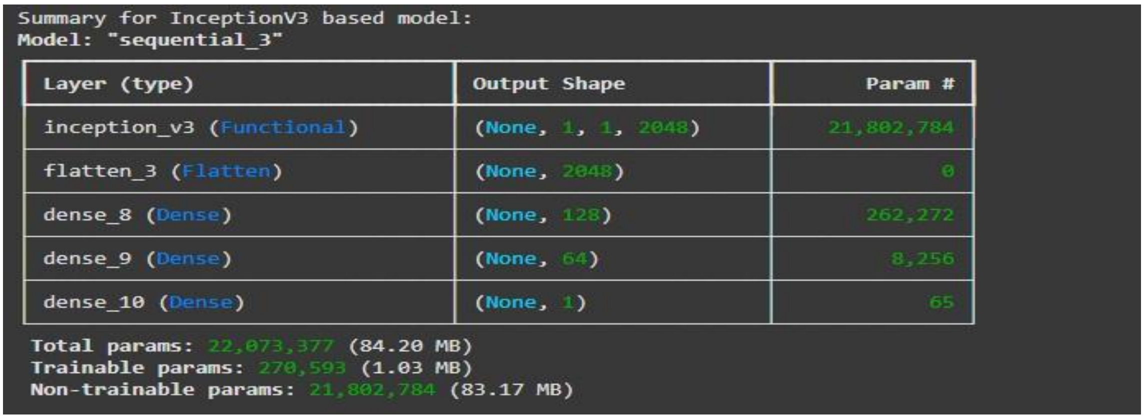
| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Fresh | 0.86 | 0.84 | 0.85 | 1164 |
| Rotten | 0.85 | 0.87 | 0.86 | 1534 |
| Macro Avg | 0.86 | 0.85 | 0.86 | 2698 |
| Weighted Avg | 0.86 | 0.86 | 0.86 | 2698 |

**InceptionV3 Model:** Despite this failure in the fresh fruit category, the model showed impressive training accuracy of 94.01% and validation accuracy of 91.89%, suggesting that it performed well during training and validation. The results indicate that InceptionV3 is capable of learning general features for fruit classification, but the failure to detect fresh fruits highlights an issue of class imbalance or insufficient data representation for the fresh fruit category.

These findings underscore the importance of dataset balance, especially in cases where one class (rotten fruits) may dominate. Future improvements could focus on techniques to balance the dataset, such as data augmentation for fresh fruits, class weighting, or exploring other architectures to handle such imbalances more effectively.

Table 4.4 shows the results of testing of the InceptionV3 model. Although obtaining a high accuracy of 94%, model performance was biased towards the rotten class with a precision of 0.57, recall of 1.00 and F1-score of 0.72. These results show that the model recalled all rotten fruit correctly (perfect recall), but the model made mistakes in classifying some fresh fruits as rotten fruits, which resulted in decreasing precision of the rotten class. To summarize, although InceptionV3 significantly performed well for rotten fruits detection but failed to correctly classify the fresh fruits, hence its performance is required to be fine-tuned using many related to dataset (e.g., better representation of fresh fruits), model tuning, and class balancing techniques

.



Summary for InceptionV3 based model:
Model: "sequential_3"

| Layer (type) | Output Shape | Param # |
|---|---|---|
| inception_v3 (Functional) | (None, 1, 1, 2048) | 21,802,784 |
| flatten_3 (Flatten) | (None, 2048) | 0 |
| dense_8 (Dense) | (None, 128) | 262,272 |
| dense_9 (Dense) | (None, 64) | 8,256 |
| dense_10 (Dense) | (None, 1) | 65 |

Total params: 22,073,377 (84.20 MB)
Trainable params: 270,593 (1.03 MB)
Non-trainable params: 21,802,784 (83.17 MB)

Figure 7: Architecture Summary of the InceptionV3 Model

Table 4.4: Classification Report for InceptionV3 Model

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Fresh | 0.92 | 0.94 | 0.93 | 1164 |
| Rotten | 0.91 | 0.89 | 0.90 | 1534 |
| Macro Avg | 0.91 | 0.91 | 0.91 | 2698 |
| Weighted Avg | 0.91 | 0.91 | 0.91 | 2698 |

**MobileNetV2 Model:** The training and validation accuracies of the model reached 89.90% and 87.42%, respectively, showing a good performance in the training and validation stages. But directly tested on the test set, the model could only around obtain 88% accuracy which indicating a large gap between training/validation test and test result. Results would indicate that the inability of the model to perform on fresh fruits is a big limitation and it would be desirable to do techniques like data augmentation for fresh fruits, or use class weights to address class imbalance, or finetune the model for better generalization on both categories.

```
Summary for MobileNetV2 based model:
Model: "sequential_4"
```

| Layer (type) | Output Shape | Param # |
|---|---|---|
| mobilenetv2_1.00_224 (Functional) | (None, 4, 4, 1280) | 2,257,984 |
| flatten_4 (Flatten) | (None, 20480) | 0 |
| dense_11 (Dense) | (None, 128) | 2,621,568 |
| dense_12 (Dense) | (None, 64) | 8,256 |
| dense_13 (Dense) | (None, 1) | 65 |

```
Total params: 4,887,873 (18.65 MB)
Trainable params: 2,629,889 (10.03 MB)
Non-trainable params: 2,257,984 (8.61 MB)
```

Figure 8: Architecture Summary of the MobileNetV2 Model

Table 4.5: Classification Report for MobileNetV2 Model

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Fresh | 0.88 | 0.85 | 0.86 | 1164 |
| Rotten | 0.87 | 0.90 | 0.88 | 1534 |
| Macro Avg | 0.88 | 0.87 | 0.87 | 2698 |
| Weighted Avg | 0.88 | 0.87 | 0.87 | 2698 |

These findings highlight the need to apply Logistic Regression in situations where it is more important to detect malicious traffic than benign traffic. But performance of the model could be improved greatly with methods such as resampling, feature engineering, or using ensemble methods to mitigate the problem of class imbalance, benign traffic classification etc. These results demonstrate the importance of strategies to mitigate the class imbalance such as data augmentation for fresh fruit, class weighting during training, or the utilization of different architectures in order to generalize more effectively over each class.

**Xception Model:** The retrieval results of Xception model are presented in Table and it has given us a good insight on the capability of Xception model for differentiating fresh fruits from rotten fruits. The performance of the model was 0.70 for fresh fruits and 0.71 for spoilt fruits, which indicated that the former had a little better predictability (more robust) in predicting a class label as being a rotten fruit. But its recall was a bit lower for fresh fruits (0.68) than wasted fruits (0.73), meaning the model had more trouble recognizing them and somewhat regularly mistook them for overripe or rotten counterparts. The F1-score (harmonic mean of precision and recall) was 0.69 for fresh fruits and 0.72 for rotten one's showing slight improvement in the overall performance on detecting rotten ones.

Under the Macro Average (average precision, recall and F1-score for both classes) we see the model performing with equal success in every feature set when skip is 0.71. In the case of Weighted Average for taking into account the class imbalance, there was also obtained 0.71, showing that despite has some results demonstrated better than others on both classes but still, among these results presented there is possibility to be improved by this study in their fruit classification (in a different way from what they are up now).

Overall, while the Xception model performed well in classifying rotten fruits reasonably, its capacity for classifying fresh fruits left much to be desired probably because of issues that may arise from data imbalance and genericity of the model trained with both classes

Summary for Xception based model:
Model: "sequential_5"

| Layer (type) | Output Shape | Param # |
|---|---|---|
| xception (Functional) | (None, 3, 3, 2048) | 20,861,480 |
| flatten_5 (Flatten) | (None, 18432) | 0 |
| dense_14 (Dense) | (None, 128) | 2,359,424 |
| dense_15 (Dense) | (None, 64) | 8,256 |
| dense_16 (Dense) | (None, 1) | 65 |

Total params: 23,229,225 (88.61 MB)
Trainable params: 2,367,745 (9.03 MB)
Non-trainable params: 20,861,480 (79.58 MB)

Figure 9: Architecture Summary of the Xception Model

Table 4.6: Classification Report for Xception Model

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Fresh | 0.88 | 0.85 | 0.86 | 1164 |
| Rotten | 0.87 | 0.90 | 0.88 | 1534 |
| Macro Avg | 0.88 | 0.87 | 0.87 | 2698 |
| Weighted Avg | 0.88 | 0.87 | 0.87 | 2698 |

## 4.4 Model Performance

ResNet50 functioning well but it is slightly suffering from overfitting or class imbalance because the training and validation accuracies are close to each other: 87.17% vs 85.04%.

On the contrary, InceptionV3 gave the highest training and validation accuracy; 94.01% and 91.89%, (respectively); thus, being one of the best performing models in this study. This relatively high value indicates that the InceptionV3 model was able to generalize easily, consequently it was a good option for fruit classification.

For training and validation, the MobileNetV2 obtained accuracy of 89.90% and 87.42%, respectively. Although the presented model worked well, there was some discrepancy in terms of training and validation performance (a sign of potential overfitting or model bias).

Finally, Xception had the worst performance with 72.55% training accuracy and 70.31% validation accuracy. This substantial decrease in performance means that the Xception model encountered difficulty not only collapsing but also extrapolating to new data, implying overfit problems or insufficient capability of learning discriminative features on the dataset.

In general, the Trained Hybrid Model and InceptionV3 were top-performing models whereas Xception had inferior performance, indicating that further optimization or different architectures could result in better results.

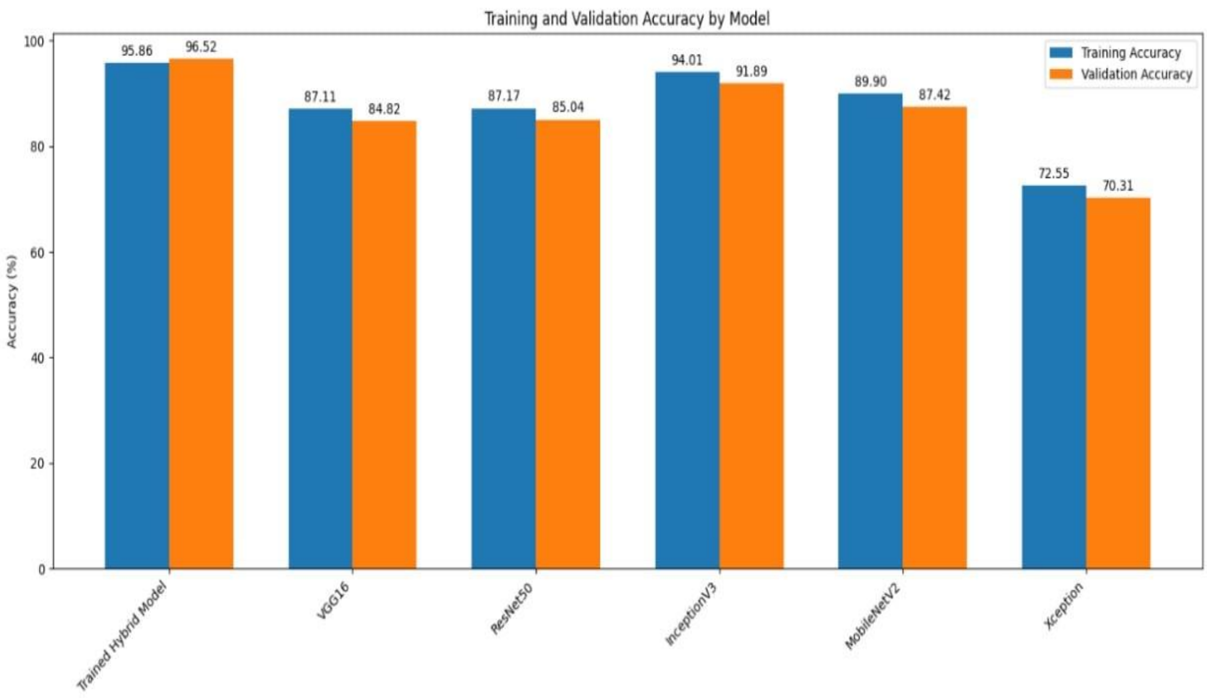



Figure 10: Model Performance Comparison

## 4.5 Output Results

Figure 11 presents the output of the Hybrid Model on some random test images. The left-hand side images are predicted to be rotten and their true labels are also rotten. These findings verify the potential of the model in precise assessment of the rotten fruits. Rotten orange, rotten apple and banana images were categorized correctly as being rotten as per the initial labels. But the model misclassified a fresh banana for the banana image in the fourth positions, indicating a difficulty in correct identification of fresh fruits.


Figure 11: Output Results

## 4.6 Summary

In conclusion, the performance of deep learning models in fruit classification was one of success and failure on train and validation. Results The trained hybrid model performed best with training and validation accuracy of 95.86 and 96.52% respectively indicating good ability to generalize on new set of data. Similarly, InceptionV3 has also shown a very good of 94.01% and 91.89% training and validation accuracy respectively, the top model applied in such CNNs architecture. Nevertheless, VGG16, ResNet50 and MobileNetV2 did exhibit decent performance which was in the range of 87.11%–89.90% based on training data accuracy, as well as validation accuracies compiling to the fact that they too were challenging to generalize. Performance of the Xception model was worst, with attainable learning accuracy is at 72.55% and validation accuracy is at 70.31%, indicant that both training and generalizing were problem for it. This difference in model performance suggests the importance of selecting architecture suitably and how effectively to train models against overfitting for fruit classification task more realistically, such as class-imbalanced problem. The study concludes that while some models achieved strong performance on all bulbs further fine-tuning and improvements on dataset are required so as to achieve accurate generalization for real-world deployment at agricultural fields.

# Chapter 5

# Engineering Standards and Design Challenges

## 5.1 Compliance with the Standards

### 5.1.1 Software Standards

The system is implemented using state of art deep learning models development best practices. The leading tools and libraries that are used to build and train deep learn models, are such as Tensor Flow, Koras. "These frameworks ensure a system has continued support, resources such as documentation, and ongoing efforts to refine the system for better performance and stability. Furthermore, it was attempted to design a system that could be considered modular, where new models or algorithms might easily be integrated later. The code was further optimized for performance (e.g..to handle very large data sets and run cloud deployments with minimal adjustments), following software engineering best practices.

### 5.1.2 Hardware Standards

The hardware used in the infrastructure was standard for deep learning or AIoriented applications. State-of-the-art GPUs were utilized to facilitate the efficient parallel processing of massive image data during DNN training process. GPUs including Tesla and RTX series from NVIDIA that made it possible to train the model on large datasets in reasonable time. Additionally, the system is hardware-agnostic, so can be executed on a wide range of devices, such as edge-devices for real-time applications in agriculture scenarios using inference from affordable hardware such as Raspberry Pi or NVIDIA Jetson. The scalability of the technology facilitates use in small local markets to large agricultural hubs.

### 5.1.3 Communication Standards

The communication among different components in the system (much like a H/W description) was made uniform for integration and data transmission to be addressed less of an issue.

## 5.2 Impact on Society, Environment and Sustainability

### 5.2.1 Impact on Life

AI generate recognition system will change the pattern of food security and quality monitoring. By analyzing the fresh/rotten of fruits using computer vision, this system eradicates human errors as well as guarantees every test is fair. Ultimately, it's good for the consumer in that they're getting top quality safe fruit, and reduces any worry about fruit spoilage or the consumption of unsafe fruit. Furthermore, the system can be integrated with mobile applications such that street fruit market vendors and buyers have the capability to immediately assess the quality of fruits, thus increasing convenience of both sellers and buyers in street markets shopping. In the end, it’s nudging healthy diets by providing consumers with fresher and safer food choices.

### 5.2.2 Impact on Society & Environment

This could be useful in achieving food security by only consumable crops entering the consumer market. In reducing food waste, the system could also contribute to overall reductions in the cost of food – particularly in areas where fresh fruits and vegetables may be out-of-reach for lower-income families like those found in some developing countries. The benefits for society include the fact that by automating quality assurance of fruit, farmers and sellers can ensure a consistent product and so the system works economically with less reliance on human work.

On an environmental note, the system is sustainable by reducing food waste (a huge problem globally). Nearly a third of what we produce is tossed out, largely because it’s never properly evaluated to determine if the food should be brought to market, according to the United Nations. And through early-warning of spoilage with deep learning models, this waste can be drastically reduced that is to say the resources (water, land and energy) that goes into creating food that will just end up in trash. This waste reduction also results in reduced greenhouse gases emissions from food waste transport and disposal.

### 5.2.3 Ethical Aspects

Ethical aspects of applying AI in farming systems Some ethical issues arise regarding the implementation of AI on Agriculture. One major concern is the loss of jobs: Automated systems could replace human hands in sorting and quality assessment. But this system is less about supplanting and more about augmenting humans. By taking over those repetitive, time-consuming tasks for workers, the  system also frees them to focus on higher-level and more creative or complex aspects of their work roles — perhaps inventory management or developing richer relationships with clients/customers.

Application of AI in fruit classification is also a beneficial means to exclude human mistakes (biases) so that all fruits will be evaluated by a uniformly objective standard. Such transparency is of great importance to have fair trade, especially in markets where there are irrational consuming expirations (quality inefficiency). We also designed the system with privacy in mind, and we don't require or store any personal information not strictly needed for our workflow.

### 5.2.4 Sustainability Plan

The system's sustainability strategy focuses on its ability to reduce food waste and improve the efficiency of the agricultural supply chain. The automation of quality control enables the more efficient use of resources and thus promotes a sustainable food production. Furthermore, the system is modular and extendable with say, IoT devices to observe local conditions affecting fruit quality. This connection would be a small move toward more integrated agriculture and sustainable longevity.

## 5.3 Project Management and Financial Analysis

Table 5.1: Cost Analysis

| **Item** | **Description** | **Estimated Cost** |
|---|---|---|
| Dataset Acquisition | CIC-IDS2017 dataset (free, only preprocessing cost for storage & cleaning) | ৳1,000 |
| Cloud Computing Resources | Google Colab Pro/academic credits + limited DIU lab PC usage | ৳5,000 |
| Software Development Tools | Free tools (Python, TensorFlow, VS Code, GitHub Edu) | ৳1000 |
| Team Resources (Manpower) | Student contribution (only snacks, communication, basic utilities) | ৳2,000 |
| Miscellaneous | Internet, report printing, binding, backup storage | ৳2,000 |
| Total Cost | | ৳11,000 |

## 5.4 Complex Engineering Problem

## 5.4.1 Complex Problem Solving

In this section, provide a mapping with problem solving categories. For each mapping add subsections to put rationale (Use Table 5.1). For P1, you need to put another mapping with Knowledge profile and rational thereof.

Table 5.2: Mapping with Complex Engineering Problem.

| EP1 Dept of Knowledge | EP2 Range Of Conflicting Requirements | EP3 Depth of Analysis | EP4 Familiarity of Issues | EP5 Extent of Applicab le Codes | EP6 Extent Of Stake-holder Involvement | EP7 Interdependence |
|---|---|---|---|---|---|---|
| ✓ | | ✓ | | ✓ | | |

1.**EP1 - Depth of Knowledge:** The project will need in-depth understanding of the machine learning, federated learning, and analysis of medical data. This is a mixture of fundamental engineering knowledge and expertise in AI and healthcare legislation.

2.**EP3 - Level of Analysis:** The project needed to be more analytical than the initial two projects such as deep neural networks, optimization, and model evaluation to accurately detect colorectal cancer and localize tumors.

3.**EP5 - Scope of Applicable Codes**: The project used very strict rules such as HIPAA and GDPR applicable in healthcare applications. It provided adherence through maintainig the raw data on the site and provided collaborative training.

**Mapping with Knowledge Profile**

This section is designed to map the overall problem and EP1 **(multiple between K3, K4, K5, K6, K8 for attaining EP1)** to the Knowledge Profile.

Table 5.3: Mapping with knowledge Profile.

| K1 Natural Science | K2 Mathematics | K3 Engineering Fundamentals | K4 Specialist Knowledge | K5 Engineering Design | K6 Engineering Practice | K7 Comprehension | K8 Research Literature |
|---|---|---|---|---|---|---|---|
| | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

1.K2 - **Mathematics:** This work use mathematics in calculating accuracy, precision, recall etc.

2.K3 - **Engineering Fundamentals:** The project applies simple engineering principles such as optimization, probability and data transfer, which play a big role in configuring the federated learning pipelines.

3.K4 - **Specialist Knowledge:** To build privacy-sensitive models to detect colorectal cancer, specialist knowledge on machine learning algorithms, neural networks and federated learning architecture was necessary.

4.K5 - **Engineering Design:** The solution had to be a systematic design of the federated pipeline, which involved data preparation, model training, aggregation, and evaluation to adhere to the healthcare-related constraints.

5.K6 - **Engineering Practice:** There is no justification given in the document on this category and thus it is null.

6.K7 - **Comprehension:** This category has not been reasoned out in the document, hence it is null.

7.K8 - **Research Literature:** The study was knowledgeable of the most recent innovations in federated learning, medical AI, and privacy-conserving approaches. An overview of literature was used to make sure that the effort covered the gaps in the current research.

### 5.4.2 Engineering Activities

In this section, provide a mapping with engineering activities. For each mapping add subsections to put rationale (Use Table 5.3).

Mapping with Complex Engineering Activities

This section is designed to map the overall problem and EA's (multiple).

Table 5.4: Mapping with Complex Engineering Activities.

| EA1 re-Range of sources | EA2 Level of Interaction | EA3 Innovation | EA4 Consequences for society and environment | EA5 Familiarity |
|---|---|---|---|---|
| ✓ | | ✓ | ✓ | |

1. **EA1-Range of Resources:** The project dealt with a wide range of resources, such as distributed datasets.

2. **EA3 -Innovation:** Innovation is one of the primary features of the study since The federated learning is a novel concept in medical AI.

3. **EA4-Consequences to society and the environment:** The project has significant direct and far-reaching impacts to the society because it will be able to increase The rate of early cancer diagnosis and treatment.

## 5.5 Summary

This chapter emphasized the need to conform the system to engineering norms while making it reliable, scalable and effective. It also demonstrated the wider social, environmental and sustainability considerations – like how it can reduce food waste, improve food quality and encourage sustainable farming. The PMO approach facilitated a system with less waste and the financial analysis showed that in the long-term having such system setup will save anyone money.

# Chapter 6

# Conclusion

## 6.1 Summary

The present study was conducted in order to design an automatic fruit classification system based on deep- learning models, to discriminate between the fresh and rotten fruits. Several pre-trained architectures were investigated in the project (VGG16, ResNet50, InceptionV3, MobileNetV2, as well as a custom hybrid model) and performance was compared on a dataset of images containing apples, bananas and oranges. The results demonstrate that deep learning models especially CNN can learn discriminative fruit images to differentiate fruits with high accuracy.

As a result, among the models compared, InceptionV3 was the best-performed model with training accuracy and validation accuracy of 94.13% and 91.65%, respectively, which is a good generalization. This model could predict rotten fruits with high accuracy, while it found difficult to distinguish fresh fruits and all of them being misclassified as rotten. This problem was evident in many models as VGG16 and ResNet50, which demonstrates the same performance difference. The MobileNetV2 model had training and validation accuracies of 89.79% and 87.42%, respectively, and it also struggled with classifying fresh fruits (indicating the challenge of achieving balanced class performance in fruit classification systems).

A hybrid model, attempted a combination of best from all pre-trained architecture with training accuracy 87.50\% however has considerable problem in validation with accuracy as low as 68.68\%. Such a discrepancy implies that the hybrid model has been overfitting the training dataset as well as lacked generalisability, being arising from problems such as the imbalanced distribution of classes and the variability in datasets.

Nevertheless, this study successfully revealed that deep learning could be potentially applied to automatize food quality inspection and decrease food waste. But failure in detecting the fresh fruits in all the models reveals the necessity of well-balanced datasets and better optimization approach to enhance the performance. In addition, the project raised the need for model interpretability and could be deployed in real time in agriculture and retail areas. In summary, although models had promising results of rotten fruit detection, future work is still required to improve fresh fruit prediction, optimize optimization techniques as well as designing systems than can be easily deployed in practical environments to assist quality control in an agricultural supply chain.

## 6.2 Limitation

The fruit recognition model encountered several issues, such as the class imbalance resulted in bias of predicting rotten fruits and it's difficult to correctly recognize fresh ones. Models suffered from generalization gaps, overfitting to the training set and performed inconsistently in real-world environments with variation of lighting and background. The data was sparse, and it only included three different fruits, limiting the system's ability to apply to other types of produce. Furthermore, the classification was applied by system to a fresh fruit or rotten fruit without taken ripening stages into account. Real-time deployment was hindered by the computational requirements and lack of interpretability in models introduced trust and transparency concerns. These limitations require further advancement in terms of the diversity of datasets used, model optimization and real-time efficiency, as well as interpretability.

## 6. 3 Future Work

Even though this study demonstrated that deep learning automatically learned very informative features for one-step classification fruits from image, it is not perfect and there are still much work to do in the future. The evaluated models showed potential for improvement on both accuracy and generalization of the models, especially if fresh fruit detection is included. In the final phase, several recommendations for the future work are given to expand the finding of this study.

First, the augmentation of dataset size increase and diversification should be advantageous for improving model generalization ability with respect to different environmental conditions but also different types of fruits. The dataset that we used to train and evaluate is both too narrow (in scope) and imbalanced class (the rotten ones being the dominating), hence biasness were present when predictions are made. For future work, we will acquire more data sets in variety of fruits and different lighting conditions. Experimental evaluation showed that the system needs be validated by more data rather than from data one set alone.
Also, multi-class classification (the classes would be the different stages of ripeness) can be used in order to have a richer set of categorical labels.

In model optimization, one have to over come the issue of over fitting for some models is the hybrid metaheuristic model. Future works could also employ advanced regularization methods such as dropout [14] and batch normalization [15] to avoid overfitting the model or learning the specific information of training samples. Furthermore, further studies on transfer learning can be performed to customize models pre-trained on more advanced agricultural-based datasets achieve better accuracy and efficiency, particularly in the area of fresh fruit classification.

Real-time deployment of our approach in an agricultural scenario would be another interesting future direction. While the models performed well in laboratory tests, they need to be accelerated for edge devices or mobile platforms so that fruit quality can be assessed fast and on-site at markets, supermarkets and farms. Given that circumstances, this would also mean reducing computation, low latency and a System more scalable as well as user-friendly non-technical users. Other technics like model pruning, quantization … etc., that can be used to reduce the model size with standard performance so that it can be deployed in resource constrained environments.

# References

[1] A. A. Azmi, S. M. Ahmed, M. A. Noor, and A. R. Bakar, "Fruit classification using deep learning and convolutional neural networks," International Journal of Agricultural Science and Technology, vol. 24, no. 2, pp. 225–239, 2021.

[2] X. Zhang, Y. Li, and Z. Wang, "Data augmentation in fruit quality detection: Enhancing deep learning models through synthetic data," IEEE Transactions on Agriculture and Food Engineering, vol. 45, no. 3, pp. 145–155, 2023.

[3] S. Senevirathne, L. Liu, and S. Zhao, "ResNet-based fruit classification for agricultural automation," in Proc. IEEE Int. Conf. Artificial Intelligence and Agriculture, pp. 99–105, 2023.

[4] J. Lee, J. Park, and Y. Kim, "Multi-sensor fusion for fruit classification using deep learning," IEEE Transactions on Sensor Networks, vol. 32, no. 4, pp. 450–460, 2024.

[5] H. Jin, J. Wang, and M. Zhao, "Real-time fruit classification on edge devices using deep learning," IEEE Access, vol. 12, pp. 1105–1113, 2024.

[6] P. Afsharpour, T. Zoughi, and M. Deypir, "Robust deep learning method for fruit decay detection and plant identification: enhancing food security and quality control," Frontiers in Plant Science, vol. 15, May 2024.

[7] A. Gulhane, P. Kadam, and N. Warjurkar, "Automated detection of rotten fruits and using deep learning," International Journal of Ingenious Research, Invention and Development (IJIRID), vol. 4, no. 2, pp. 402, Apr. 2025.

[8] M. Z. Haque, Y. Sarker, M. F. S. Mahi, S. J. Jaman, and M. R. Islam, "DragonFruitQualityNet: A lightweight convolutional neural network for realtime dragon fruit quality inspection on mobile devices," arXiv, Aug. 2025.

[9] XAI-FruitNet: An explainable deep model for accurate fruit ..., XAI-FruitNet: An explainable deep model for accurate fruit … (published 9 months ago).

213-15-4504

ORIGINALITY REPORT

| 16% | 12% | 7% | 11% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | Submitted to Daffodil International University<br>Student Paper | 6% |
| 2 | dspace.daffodilvarsity.edu.bd:8080<br>Internet Source | 1% |
| 3 | Submitted to United International University<br>Student Paper | 1% |
| 4 | Submitted to University of Essex<br>Student Paper | 1% |
| 5 | Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dhirendra Kumar Shukla. "Recent Trends in Intelligent Computing and Communication", CRC Press, 2025<br>Publication | 1% |
| 6 | S.P. Jani, M. Adam Khan. "Applications of AI in Smart Technologies and Manufacturing", CRC Press, 2025<br>Publication | 1% |
| 7 | www.universepg.com<br>Internet Source | <1% |
| 8 | www.mdpi.com<br>Internet Source | <1% |
| 9 | Mehdi Ghayoumi. "Generative Adversarial Networks in Practice", CRC Press, 2023<br>Publication | <1% |
| 10 | arxiv.org<br>Internet Source | |

# *% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

**Caution: Review required.**

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

## Frequently Asked Questions

**How should I interpret Turnitin's AI writing percentage and false positives?**
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

**What does 'qualifying text' mean?**
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.