MAKERERE UNIVERSITY

COLLEGE OF ENGINEERING, DESIGN, ART AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

FINAL YEAR PROJECT REPORT

DESIGN AND IMPLEMENTATION OF A PORTABLE IoT DEVICE FOR ASSESSING QUALITY OF GREEN COFFEE BEANS USING DEEP LEARNING

DENIS INNOCENT KIWEEWA
18/U/22542/PS

Supervisor: Dr. Edwin Mugume
Co-Supervisor: Mr. Frank Ssemakula

A final year project report submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Science in Electrical Engineering at Makerere University.

June 17 2024

Declaration

No portion of the work in this document has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

I have abided by the Makerere University academic integrity policy on this assignment.

DENIS INNOCENT KIWEEWA
18/U/22542/PS

Signature: ______________________ Date: 17-06-2024

Approval

This Report has been submitted for examination with the approval of the following supervisors.

Main Supervisor

Dr. Edwin Mugume

Lecturer

Department of Electrical and Computer Engineering

College of Engineering, Design, Art and Technology

Makerere University

Signature: .... Date: 14/06/2024

Co-Supervisor

Mr. Frank Ssemakula

Lecturer

Department of Electrical and Computer Engineering

College of Engineering. Design, Art and Technology

Makerere University

Signature: .... Date: 14/06/24.

Acknowledgement

I thank the Almighty God for giving me the resolve, guidance, and capability to accomplish this project effectively.

I would like to thank my supervisors, Dr. Edwin Mugume and Mr. Frank Ssemakula, for their exclusive guidance through execution of this project.

I also want to thank Mr. Oscar Ddembe, Mr. Rodgers Mukiibi and Mr. Ronald Omoding for their specialized counsel on this project and for permitting us to utilize Marconi Lab and its assets when essential.

My earnest appreciation goes out to Ms. Destiny Mulungi Sanyu, my project accomplice, for her energetic work and near coordination in coming to the various targets we set up at the commencement of this project.

I also want to thank my family and companions for all the bolster they have given me whereas I seek after my scholarly objectives.

Final but not slightest, I need to precise my appreciation to the Electrical and Computer department for giving me with the opportunity to work on this project since it empowers me to apply the hypothesis I have learnt in lesson.

Abstract

The existing system for assessing the quality of green coffee beans faces significant challenges. Currently, manual methods introduce delays and subjectivity, delaying the certification process for coffee beans exporters. These delays affect trade flow, and the variability in quality determination undermines the credibility of Ugandan coffee beans in international coffee markets.

This project addresses these issues by developing a portable IoT device with deep learning capabilities. This device automates the assessment process, reducing reliance on human judgment. By quickly certifying bean quality, it streamlines trade operations and ensures consistent grading.

A Raspberry Pi 4, a DHT11 sensor and a Picamera were integrated to create the portable IoT device. The methodology involves designing the schematic diagram, developing scripts, and interfacing the sensors. Additionally, a user-friendly web app interface is created using User-interface scripts ensuring that the interface is intuitive, efficient, and accessible for users during coffee bean quality assessment.

The hardware design includes the Raspberry Pi 4, Picamera module, and DHT 11 sensor. The implemented device is made portable and efficient for on-site quality assessment.

A deep learning model classifies the green coffee beans into quality categories according to the number of defects present in a sample of green coffee beans focusing on accuracy and speed, aiming to surpass manual methods.

In conclusion, this project offers an innovative solution for green coffee beans quality assessment. The project further recommends increasing on the number of defect classes of the green coffee beans at earlier processing stages and exploring more powerful microcontrollers for future enhancements.

Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
AWS	Amazon Web Services
CNN	Convolutional Neural Network
DHT	Digital Humidity and Temperature
DL	Deep Learning
EAGLE	Easily Applicable Graphical Layout Editor
ETG	Export Trading Group
GND	Ground
GPIO	General Purpose Input/Output
GPU	Graphics Processing Unit
IoT	Internet of Things
IoU	Intersection over Union
ISO	International Organization for Standardization
mAP	Mean Average Precision
OS	Operating System
PDF	Portable Document Format
RNN	Recurrent Neural Network
RAM	Random Acess Memory
SCA	Specialty Coffee Association
SMTPlib	Simple Mail Transfer Protocol Library
SSD	Single Shot MultiBox Detector
SSH	Secure Shell
UCDA	Uganda Coffee Development Authority
WiFi	Wireless Fidelity
YOLO	You Only Look Once

Chapter 1

INTRODUCTION

Despite coffee's immense popularity, evaluating its quality is subjective and complex. This project aimed to develop more objective methods for coffee quality assessment. This chapter overviews our project, addressing the background, problem statement, justification, project scope, and main and specific objectives.

1.1 Background

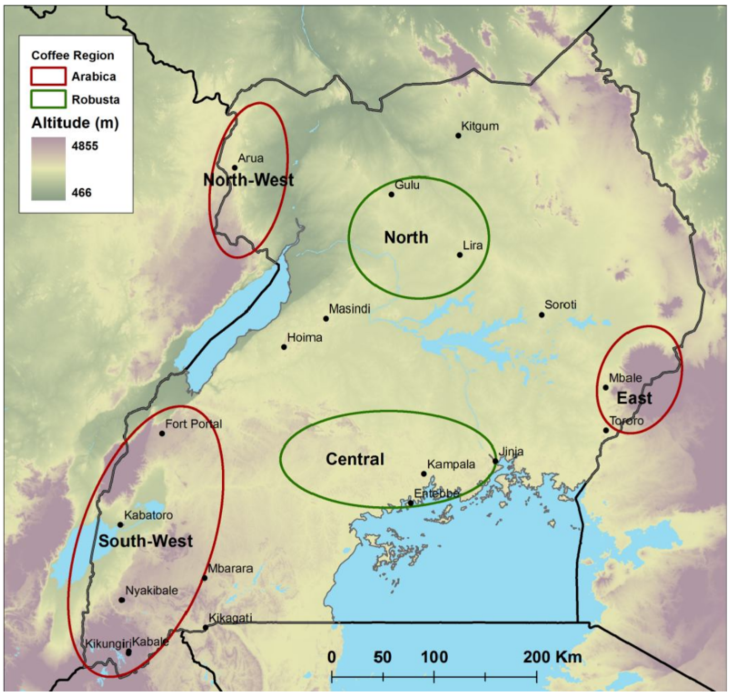

Figure 1.1: Coffee growing regions in Uganda [1]

In Uganda, coffee is cultivated across various regions of the country as illustrated in Figure 1.1, with Arabica and Robusta coffee varieties being the most prominent. As the second-largest export commodity after oil, coffee significantly contributes to the foreign exchange earnings of

Uganda accounting for about 20% of the total export earnings and supporting the livelihoods of more than 1.7 million households [4]. Despite its robust production practices, the coffee industry faces considerable challenges in maintaining consistent quality, particularly with Robusta coffee beans, which are widely grown across the country and highly valued on the global market. UCDA is instrumental in regulating and promoting the sector, striving to ensure that Ugandan coffee adheres to strict international standards. However, the current system for assessing the quality of green coffee beans, a critical step before export, faces significant challenges that impede its efficiency and reliability. The quality of green coffee beans is a paramount concern because it directly influences the flavour profile and overall quality of the final roasted product. Various factors such as growing conditions, harvesting techniques, and post-harvest processing all play critical roles in determining bean quality.

Figure 1.2: Manual Sorting of Green Coffee Beans at the UCDA lab

For quality control officers, accurately assessing these beans before they reach the roasting stage is essential to ensure they meet international standards. The traditional approach employed involves manual methods, where trained personnel, as shown in Figure 1.2, visually inspect the beans for various quality parameters such as size, shape, colour, and the presence of defects. These manual inspections are inherently subjective as they rely heavily on the expertise and judgment of the inspectors. This subjectivity can lead to inconsistencies in quality determinations, where different inspectors might have varying opinions on the same batch of beans. This poses a significant challenge for exporters who must meet the high standards demanded by international markets to command premium prices. In the global coffee trade, high-quality beans are highly sought after, and their assurance is vital for maintaining market competitiveness. Exporters therefore face substantial economic pressure to deliver the best green coffee beans as any compromise on quality can result in financial losses and damage to reputation. Furthermore, this approach is time-consuming leading to delays in the quality assessment process. These issues can significantly impact the certification process for exporters, introducing variability in quality determination that affects trade flow and undermines the credibility of Ugandan coffee in the global market.

Given this backdrop, there is an urgent need for more reliable, efficient, and objective methods of quality assessment that can enhance the credibility and consistency of Ugandan coffee on the international stage.

1.2 Problem Statement

The traditional methods of quality assessment for green coffee beans in Uganda are predominantly manual and labour-intensive. These methods rely heavily on visual inspection and physical sorting by trained personnel. While these inspectors are skilled, the process is inherently subjective and prone to human error, leading to inconsistencies in quality evaluations. This subjectivity can result in significant variability in the assessment of bean quality, where different inspectors may have differing opinions on the same batch of green coffee beans. Such inconsistencies can undermine the reliability and credibility of Ugandan coffee on the global market, where uniformity and adherence to quality standards are paramount.

Furthermore, the manual nature of these assessments is time-consuming, causing delays in the certification process that can disrupt the timely export of coffee beans. These delays are particularly detrimental for exporters who must meet strict shipping schedules to fulfil international market demands. The prolonged certification times not only affect the trade flow but also pose a financial burden on exporters, potentially reducing their competitiveness in the international market.

1.3 Objectives

1.3.1 Main Objective

To design and implement a portable Internet of Things device that assesses the quality of green coffee beans using deep learning.

1.3.2 Specific Objectives

- To identify the key parameters that influence green coffee beans quality and select the appropriate sensors for measuring them.
- To design the software components of the device, including the schematic designs.
- To develop deep learning models for image analysis of green coffee beans.
- To implement the hardware design and deploy the deep learning model.
- To integrate the software and hardware components and test the performance of the device.

1.4 Project Scope

The project aims to design, develop, and implement a portable Internet of Things (IoT) device integrated with deep learning algorithms to assess the quality of export-ready green Robusta coffee beans in Uganda. This is because at the export-ready level, Robusta beans often have more defects than Arabica beans due to the different processing techniques employed. The project therefore excludes assessments of other coffee types, large-scale manufacturing and distribution of the device. The intended end-users are quality control officers at the Uganda Coffee Development Authority (UCDA). Furthermore, this project is geographically scoped to Uganda.

1.5 Justification

This project addresses the critical need to modernize the quality assessment process for green coffee beans in Uganda. The goal is to improve consistency and reliability while reducing certification delays often caused by traditional manual methods. Given that high-quality beans command premium prices on the global market, improving the quality assessment process will result in steady economic growth and improve Uganda's reputation internationally as trade flow disruptions will be minimised thus enhancing the competitiveness of Ugandan coffee. Furthermore, this solution reduces human error and would standardize the quality evaluations thus improving the efficiency and reliability of the quality control system. Moreover, this project aligns perfectly with the strategic objectives of the Uganda Coffee Development Authority (UCDA) to elevate the country's coffee industry to meet and exceed international benchmarks. These improvements would contribute to the sustainable growth of Uganda.

Chapter 2

LITERATURE REVIEW

This chapter offers a comprehensive understanding of the theoretical and practical groundwork of this project, focusing on the integration of Internet of Things (IoT) and deep learning technologies in agricultural applications. It will also examine the critical importance of quality control in the coffee industry and how modern technologies can address existing challenges.

2.1 Internet of Things

2.1.1 Overview

The Internet of Things (IoT) refers to a vast network of interconnected devices that communicate and exchange data with each other through the internet. These devices are often embedded with sensors, software, and other technologies to connect and exchange data with other devices and systems over the internet. The concept of IoT hinges on the ability to connect everyday objects, ranging from household appliances to industrial machinery, creating smart environments that can operate more efficiently and autonomously

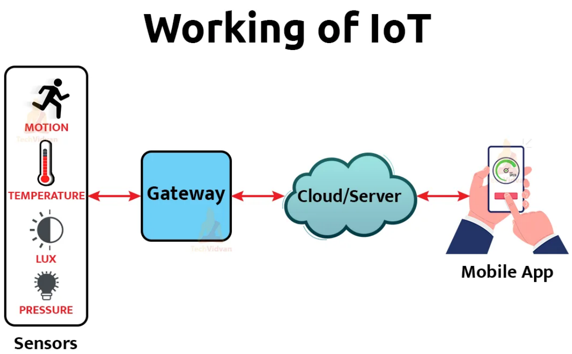

Figure 2.1: How IoT works [2]

In a nutshell, IoT works by having devices with hardware sensors that collect data, which is then shared via the cloud, integrated with software, and analyzed before being transmitted to users through an app or website as seen in Figure 2.1. Sensors embedded in IoT devices gather data from their surroundings, such as temperature, humidity, light levels, and motion. This data is then transmitted through a network, often using wireless communication protocols like Wi-Fi, Bluetooth, or cellular networks, to centralized servers or cloud-based platforms. Once collected, the data is analyzed providing actionable insights and enabling automated responses to specific conditions. For example, a smart thermostat can adjust home temperatures based on the homeowner's preferences and real-time weather conditions.

In the realm of agriculture, IoT has been increasingly utilized to enhance farming practices and productivity. By deploying IoT devices, farmers can monitor environmental conditions such as soil moisture, temperature, humidity, and light levels in real time. This allows for more precise and data-driven decisions, consequently optimizing the use of resources like water, fertilizers, and pesticides. As a result, farmers can achieve better crop yields and quality.

2.1.2 IoT in Agriculture

The Raspberry Pi, seen in Figure 2.2, is a small and affordable single-board computer, that has become a cornerstone in the development of Internet of Things (IoT) applications due to its versatility, low cost, and robust community support. Initially designed as an educational tool to promote computer science learning, the Raspberry Pi has evolved into a powerful platform for prototyping and deploying IoT solutions. Its compact size and low power consumption make it ideal for integration into various environments, from smart homes to industrial settings. The device's GPIO pins enable it to interface with a wide range of sensors and actuators, facilitating the collection and processing of data from the physical world. This capability is crucial for IoT applications, which rely on real-time data to monitor and control devices.

Figure 2.2: Raspberry Pi 4 [3]

One of the significant advantages of using Raspberry Pi in IoT applications is its support for various programming languages and operating systems, including Python, which is particularly popular for IoT projects due to its simplicity and extensive libraries. The Raspberry Pi can run a full Linux operating system, providing a flexible and familiar environment for developers to create and test their applications. Additionally, its compatibility with numerous IoT platforms and services, such as AWS IoT, Azure IoT, and Google Cloud IoT, allows for seamless integration with cloud-based solutions, enabling advanced data analytics, remote monitoring, and automated control.

In agriculture, Raspberry Pi has been used to develop smart farming solutions that enhance productivity and sustainability. For instance, a Raspberry Pi-based system can monitor soil moisture, temperature, and humidity, providing farmers with real-time data to optimize irrigation and other agricultural practices. By connecting to IoT sensors placed in the fields, the Raspberry Pi can collect environmental data, process it locally or send it to the cloud for further analysis. This data-driven approach enables precision farming, reducing resource waste and improving crop yields. Moreover, Raspberry Pi can control automated systems such as irrigation pumps or greenhouse ventilation, responding dynamically to the sensed conditions and ensuring optimal growing environments.

2.2 Deep Learning

Agriculture faces many challenges due to the increase in demand and the presence of fewer workers in the fields. In this context, smart farming can be used to address issues such as food security, sustainability, productivity, and environmental impact [8]. As is known, agriculture plays a vital role in Uganda's economy, particularly coffee framing [17]. This is because it ensures food security for countries, and most companies rely on it for their external trade.

In the world today, most home appliances, travel means, and other commonly used services are becoming automated through the adoption of artificial intelligence (AI). Farming practices should also adopt AI, as they are the backbone of a country. Continuous monitoring, measuring, and analysis of different physical aspects and phenomena can provide valuable data. Achieving understanding and making quick responses based on this data would help overcome the complex, multivariate, and unpredictable challenges of agricultural ecosystems. [18]. This would require the analysis of huge amounts of agricultural data and the use of new information and communication technologies (ICTs) and would be necessary for both small-scale farms and large-scale ecosystem monitoring [18].

It could be achieved using Deep Learning with a large network. Deep Learning is an aspect of machine learning that aims to build neural networks that can analytically learn by simulating the human brain. It acts like the human brain in that it works by reading data, such as pictures, videos, text, and sounds. With its continued development, Deep Learning has already been implemented in various complex tasks, such as image segmentation, image recognition, natural language processing, object detection, and image classification [18]. However, Deep Learning requires a huge dataset since the quality of the Deep Learning results entirely depends on the size of the dataset, and the model tends to learn from that data and then respond accordingly. Some computer and industrial advancements, such as image processing, IoT technologies, robotics, machine learning, deep learning, and computer vision, are very useful in the agricultural industry and for local farmers. High-quality image processing makes AI based on drone technology a very helpful asset for farmers since they can identify the progress of the crops and determine whether they are ready to harvest or not while sitting in one place rather than having to move long distances. This success has been achieved using AI and a drone system. One can only imagine the benefits of implementing deep learning in agriculture [19]. Figure 2.3 lists numerous advantages of using deep learning in agriculture.

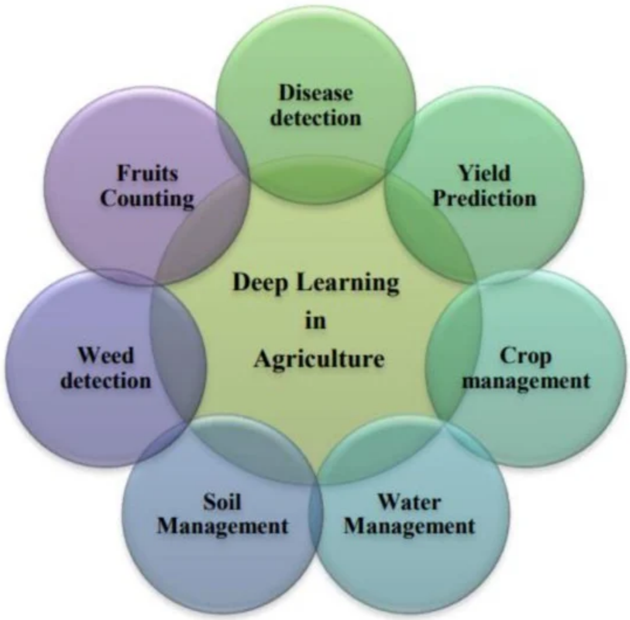

Figure 2.3: Deep learning applications in Agriculture [4]

2.2.1 Deep Learning Architecture

Several nonlinear transformations are used to model higher-level abstractions in data, and these are the foundations of deep learning [20]. One of the main benefits of deep learning is the automatic extraction of features from raw data or feature learning. Producing features in lower-level components yields features in higher-level components [21]. Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) are two types of deep learning networks that are often used in agriculture.

2.2.1.1 Convolutional Neural Networks (CNNs)

The CNN is a type of DL algorithm [22] composed of multiple convolutional layers, pooling layers, and fully connected layers. Two of the most common applications for CNNs are the recognition of handwritten characters and image processing. In the domain of computer vision, CNNs have been used for a variety of tasks, including object detection, image classification, voice recognition, image fragmentation, medical image analysis, and text and video processing.

Convolutional, pooling, and fully connected layers are the typical architectural components of a CNN [23]. Figure 2.4 depicts the architecture of a CNN, and brief descriptions of each layer are provided below.

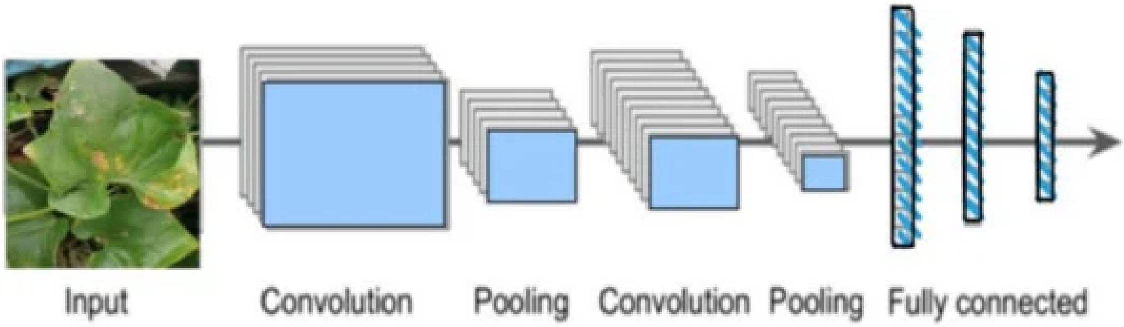

Figure 2.4: Convolutional neural network architecture. [4]

In a CNN, the convolutional layer is the most fundamental and significant. It stores all of the images' distinguishing characteristics while making it possible to limit the amount of data that must be simultaneously processed. Then, pooling enables a CNN to aggregate all the different dimensions of an image and recognize the object, even if its form is distorted or it is positioned at an angle. Thus, the number of learnable features in the model is reduced, helping to address the overfitting issue. Pooling can be accomplished in a variety of ways, including average pooling, maximum pooling, and stochastic pooling. The fully connected layer is the final layer, which is used to feed the neural network [24].

2.2.1.2 Recurrent Neural Networks (RNNs)

An RNN is a type of neural network model capable of performing exceptionally well in fundamental tasks such as machine translation, language modelling, and speech recognition [25]. Unlike traditional neural networks, RNNs use the network's sequential information. This feature is essential in many applications because the data sequence's inherent structure contains valuable information that can be extracted from it. Figure 2.5 depicts the fundamental structure of a recurrent neural network [24].

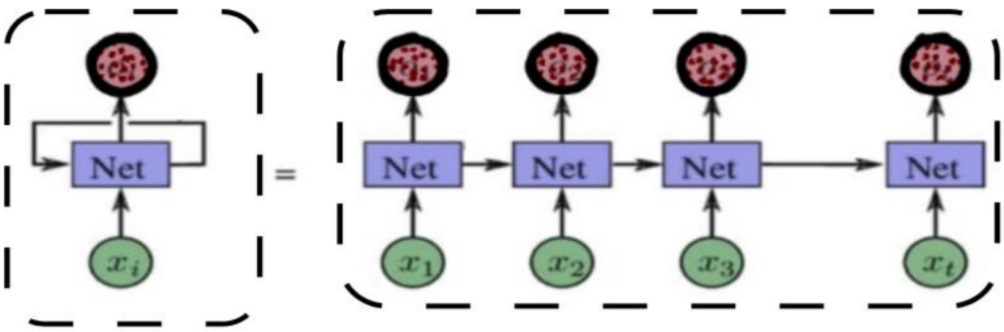

Figure 2.5: Recurrent neural network generic structure. [4]

2.2.2 Deep learning Frameworks

These are extensions of existing frameworks or specialized frameworks developed to create deep learning algorithms.

- **TensorFlow:** TensorFlow is an open-source library for numerical computation based on the computational graph. A computational graph is a directed graph where nodes represent mathematical operations and edges represent the flow of data between nodes. TensorFlow is written with a Python API over a C/C++ engine, this makes it run faster.
- **PyTorch:** PyTorch is TensorFlow's primary competitor in the deep learning framework market. Developed and created by Facebook, PyTorch is an open-source deep-learning framework that works with Python.
- **Keras:** Keras is a high-level nonmainstream neural network library. It is written in Python and can run on top of either TensorFlow. It also allows fast and easy prototyping, supports **CNNs** and recurrent networks, and runs seamlessly on GPU.

2.3 Coffee Beans quality assessment

Coffee bean quality assessment is a critical step in ensuring the production of high-quality coffee, which is essential for maintaining market standards and consumer satisfaction. Traditionally, this process has been largely manual and subjective, but modern technological advancements have introduced automated and objective methods. This section details traditional methods of coffee bean quality assessment, with a specific focus on the Ugandan context, and compares these methods with the approach used in our project.

2.3.1 Traditional Methods of Coffee Bean Quality Assessment

Traditional methods of coffee bean quality assessment are often manual and rely heavily on human expertise. These methods include:

1. **Visual Inspection:**

   Process: Trained quality inspectors manually sort and grade coffee beans based on visual characteristics such as size, shape, colour, and the presence of defects.

   Tools: Simple tools like sieves, color charts, and magnifying glasses.

   Limitations: Subjective and inconsistent due to human error and fatigue.

2. **Sensory Evaluation (Cupping):**

   Process: Professional cuppers evaluate the aroma, flavor, acidity, body, and aftertaste of brewed coffee samples. This process involves a standardized procedure where coffee beans are ground, brewed, and then tasted in a controlled environment. The evaluation is typically carried out using standardized cupping protocols and scoring sheets to ensure consistency and comparability across different samples. Figure 2.6 depicts a cupping session at the UCDA Lab, where expert cuppers meticulously analyze each coffee sample. The cupping process includes sniffing the dry grounds, breaking the crust of the brewed coffee, and then slurping the coffee to assess its characteristics. Each of these steps is crucial in identifying the quality and potential defects in the coffee beans.

Tools: Standardized cupping protocols and scoring sheets.

Limitations: Subjective, time-consuming, and requires highly trained personnel.

Figure 2.6: Cupping Session at the UCDA Lab

3. **Moisture Content Measurement:**

   Process: Moisture content is measured using moisture meters to ensure beans are dried to the optimal level. Figure 2.7 illustrates the use of a handheld moisture meter to assess the moisture content of green coffee beans.

   Tools: Handheld moisture meters.

   Limitations: Accuracy can vary depending on the device and method used.

Figure 2.7: Getting moisture content of green coffee beans using a moisture meter

Table 2.1 shows the three main current traditional methods for the quality assessment of green coffee beans.

Method	**Advantages**	**Disadvantages**
Manual sorting	Accurate for obvious defects	1. Inconsistent results due to human fatigue and variability. 2.Slow process
Visual inspection	Simple and cost effective	1. Subjective and inconsistent results due to human error. 2. Time consuming and labour intensive 3. Limited ability to detect subtle defects
Cupping (Sensory Evaluation)	Recognized standard in the coffee industry	1. Highly subjective and dependent on cuppers' experience and skill 2. Time-consuming and requires trained personnel 3. Happens too far down the value chain

Table 2.1: The current quality assessment methods

2.3.2 Comparison of Traditional and Technological Methods

The traditional methods of assessing coffee bean quality, as discussed, rely heavily on manual processes such as visual inspection and sensory evaluation (cupping). While these methods have been the cornerstone of coffee quality assessment for decades, they come with several limitations, including subjectivity, inconsistency, and the need for highly trained personnel. These limitations can lead to variability in quality assessment and affect the credibility and marketability of coffee beans on an international scale.

Technological advancements, particularly in the fields of IoT and deep learning, offer promising solutions to these challenges. IoT devices can automate data collection and analysis, providing real-time insights into various quality parameters. Deep learning models can analyze large datasets of coffee bean images to classify and grade them with high accuracy, reducing human error and subjectivity.

The integration of IoT and deep learning in coffee quality assessment aims to enhance the precision, speed, and reliability of the process. By leveraging these technologies, the coffee industry can standardize quality assessment, improve traceability, and ensure consistent product quality, ultimately boosting consumer confidence and market competitiveness. The implemented device utilizes a combination of modern methods, particularly digital imaging and machine learning, to assess the quality of green coffee beans. Specifically, it uses a Raspberry Pi 4, a Picamera, a DHT 11 sensor, and a YOLO v8 deep learning model to automate and enhance the accuracy of the quality assessment process.

Table 2.2 shows the Comparison of Traditional and Modern Methods of Coffee Bean Quality Assessment in Uganda.

Aspect	**Traditional Methods**	**Implemented IoT Device**
Visual Inspection	Manual sorting by quality inspectors	Uses Picamera and YOLO v8 model for image analysis
Subjectivity	High (depends on inspector's expertise)	Low (objective, based on trained model)
Consistency	Variable (human fatigue and error)	High (consistent model performance)
Speed	Slow (manual process)	Fast (real-time analysis on Raspberry Pi)
Cost	High (labour-intensive)	Moderate (cost-effective Raspberry Pi setup)
Scalability	Low (limited by human resources)	High (portable and scalable system)
Moisture Measurement	Handheld moisture meters	DHT 11 sensor for temperature and humidity
Data Storage	Manual record-keeping	Local storage and email distribution
Training Requirement	Extensive (training inspectors and cuppers)	Minimal (automated system reduces need for training)
Adaptability	Low (hard to adapt to new standards quickly)	High (software updates and model retraining)

Table 2.2: Comparison of implemented solution with current assessment methods

2.3.3 Previous Work

Previous efforts to automate the coffee bean quality screening process have employed various machine learning and computer vision techniques, leading to notable advancements in accuracy and efficiency. Several studies have explored the use of image processing and machine learning algorithms to classify coffee beans based on visual attributes such as size, color, and defect presence. These methods have contributed to enhancing the quality control processes in the coffee industry, ensuring that only the best green coffee beans reach the consumer market.

For instance, a significant body of work has focused on using convolutional neural networks (CNNs) to analyze images of coffee beans. CNNs are particularly effective in feature extraction and classification tasks, making them suitable for identifying defects and categorizing beans into quality grades. Studies have demonstrated that CNN-based models can achieve high accuracy in distinguishing between defective and non-defective beans, thereby reducing the reliance on manual inspection and minimizing human error.

Additionally, there has been progress in integrating these advanced algorithms into portable devices and IoT systems. Such integration allows for on-site quality assessment, providing immediate feedback to farmers and exporters. This approach not only improves efficiency but also ensures that quality control measures are maintained throughout the supply chain.

2.3.3.1 CNN-Based Classifiers

One notable approach involved the development of a Convolutional Neural Network (CNN) based classifier. This classifier was trained to categorize coffee beans into multiple categories such as sour, black, broken, mouldy, shell, insect-damaged, and good beans. The CNN employed several convolutional layers, pooling layers, fully connected layers, and a Softmax classifier. The system achieved a classification accuracy of over 90% for most categories, except for shell beans, which had an accuracy of 88% [5].

2.3.3.2 System Implementation

The developed system, seen in Figure 2.8, consisted of both mechanical and control components. The mechanical part included a vibrating feeder, buffer, DC motor, buffer plate, tub, storage tanks, solenoid valve, and silo. The control part was implemented using a camera and an embedded microcomputer equipped with the CNN-based classifier to identify the type of coffee beans [5]. The workflow was as follows:

1. Coffee beans were fed into the system through a vibrating feeder.
2. A camera captured images of the coffee beans as they passed through a synchronous belt wheel bin.
3. The captured images were processed by the CNN-based classifier.
4. Based on the classification results, the microcontroller controlled an air spray valve to separate defective beans from the good ones.

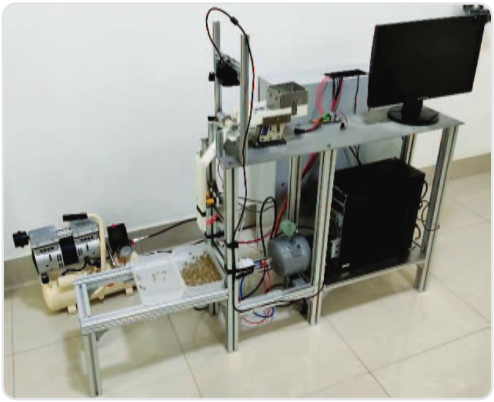

Figure 2.8: Coffee Beans Screening System [5]

2.3.3.2.1 Image Pre-Processing

Given the distortions in images captured during the screening process, an image pre-processing algorithm was developed. The pre-processing steps, illustrated in Figure 2.9, included extracting images from the camera system, dividing them into smaller segments, and resizing them to ensure accurate classification by the CNN. This step was crucial in enhancing the classifier's performance by providing high-quality input images.

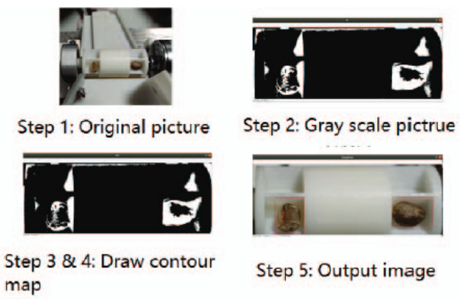

Figure 2.9: Steps involved in the image pre-processing stage [5]

2.3.4 Summary

The shift from traditional to modern methods of coffee bean quality assessment represents a significant advancement in the industry. While traditional methods are tried and tested, they are often limited by subjectivity, inconsistency, and scalability issues. Modern methods, including the approach used in this project, offer objective, consistent, and scalable solutions that can enhance the quality assessment process, particularly in the Ugandan context where maintaining high standards is crucial for the export market. This project, by integrating modern technology, aims to streamline and improve the accuracy of coffee bean quality assessments, aligning with the industry's move towards digital transformation.

Chapter 3

METHODOLOGY

The methodology encompasses field consultation, software design, model development, hardware implementation, and prototype assembly and testing. Each step was crucial in ensuring the project met its objectives and delivered a functional and reliable system.

3.1 Field Consultation

Field consultation involved collaborating with industry experts and visiting coffee processing companies to gain practical insights into the coffee quality assessment process.

3.1.1 Collaboration with Mr. Fidel Bakomeza

Mr. Fidel Bakomeza, a seasoned quality assurance officer at the Uganda Coffee Development Authority (UCDA), provided invaluable guidance during the consultation phase. Over two weeks, he offered insights into the quality assessment process and facilitated visits to various coffee processing companies. Practical knowledge was acquired on post-harvest processing of green coffee beans through observing primary and secondary processing techniques at various processing plants and factories. This assisted in the identification of the most relevant parameters that influence the green coffee bean quality enabling appropriate sensor selection. Through the consultation, it was revealed that UCDA quality control officers evaluate parameters such as temperature, humidity, moisture content and the number of defects to determine the quality of coffee beans, revealing these to be the critical parameters for determining the bean quality at this stage. The quality officers acquire samples from various batches slated for export. It is on these samples that the quality of the entire batch is assessed. For every 100 bags to be exported, a 100-gram sample is extracted, and for every 300 bags, a 300-gram sample is used and thereafter the assessment results are extrapolated to the entire export batch.

For a particular sample, its weight is taken, temperature and humidity measured and the defects separated. The sound beans are then weighed again and a percentage by weight is generated.

Hereafter, the sound beans are counted as well as the defects twice and an average is calculated. A percentage by count is generated. On average, it takes a professional quality officer with years of experience at least 20 minutes to assess a 100-gram sample of coffee beans.

All the analysis activities are performed under controlled laboratory conditions of temperature; 22.0 ± 6 degrees Celsius and relative humidity; 35.0 - 75.0 %. At this point, the quality officer assigns an assessment to the sample which is reflected on the entire batch from which it was extracted. The sample is then sent for roasting in order to assess the bean quality after roast as well as the cup quality, assessed by cupping experts in order to determine the overall quality of the batch destined for export.

3.1.2 Industrial Visits

During the consultation period, we conducted industrial visits to several coffee processing companies located in the industrial area of Namanve and Bugolobi. These visits were instrumental in understanding the different stages of coffee processing and quality assessment. Companies visited were;

- KAWACOM
- ETG
- Kyagalanyi
- Qualicoff

3.2 Stages of Coffee Processing

Mr. Bakomeza guided us through the entire coffee processing pipeline, highlighting the following stages:

Primary Processing: Initial cleaning, pulping, and drying of coffee cherries as illustrated in Figure 3.1.

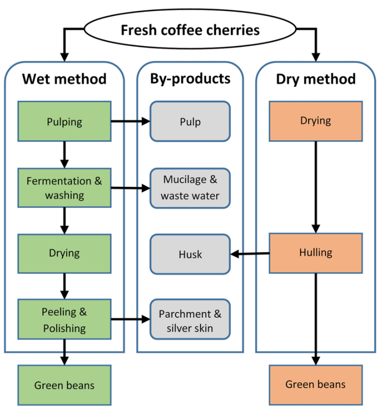

Figure 3.1: Stages of Primary processing of coffee [6]

Secondary Processing: Hulling, as shown in Figure 3.2, polishing, grading, and sorting of dried coffee beans.

Figure 3.2: Hulling of Coffee Beans

Quality Assessment: Detailed inspection and evaluation of bean quality in the UCDA lab.

Exportation: Packaging, as illustrated in Figure 3.3 and shipping of quality-certified green coffee beans in jute gunny bags.

Figure 3.3: Packing Export-ready green coffee beans in Jute Gunny Bags at KAWACOM

3.3 Software Design

The software design phase focused on developing the system architecture, creating the web application interface, and integrating the necessary software components for data collection, processing, and report generation.

3.3.1 System Architecture

The system architecture for our portable IoT device designed to assess the quality of green coffee beans integrates several hardware components with a robust software framework. The core components include a Raspberry Pi 4, a DHT 11 sensor, a Picamera module, and a web application interface. This architecture ensures seamless interaction between data collection, processing, and user interface, enabling real-time quality assessment and reporting.

3.3.1.1 Components and Interaction

- **Raspberry Pi 4:** The Raspberry Pi 4 serves as the central processing unit of our system. It hosts the web application interface, processes data from the DHT 11 sensor and Picamera, and performs inference using the YOLO v8 model. Figure 3.4 shows the model version of the Raspberry Pi 4 that was used as a microcontroller for the project implementation.

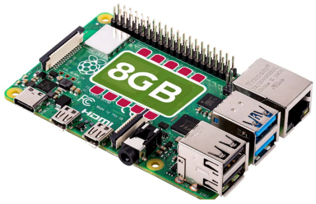

Figure 3.4: Raspberry Pi 4 model B [7]

- **DHT 11 Sensor:** The DHT 11 sensor is responsible for measuring the temperature and humidity of the coffee bean sample. When the web application interface is launched, the sensor takes readings, which are then displayed on the interface. These environmental readings are crucial as they influence the quality of coffee beans. Figure 3.5 shows the DHT 11 sensor that captures temperature and humidity readings of the green coffee beans sample.

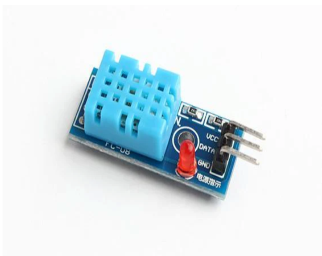

Figure 3.5: DHT 11 Sensor [8]

- **Picamera Module:** The Picamera module shown in Figure 3.6 is used to capture high-resolution images of the green coffee beans sample. Upon clicking the capture button on the web application interface, the Picamera takes an image, which is then processed by the YOLO v8 model for defect detection.

Figure 3.6: Picamera Module [9]

- **Web Application Interface** The web application interface is the user interaction point of the system. It allows users to view sensor readings, capture images, and initiate quality assessment. The interface displays the results of the assessment, including the number of defects detected by the YOLO v8 model and the sensor readings to determine whether they fall within acceptable limits.

3.3.1.2 Interaction Workflow

1. Launching the Interface: When the web application interface is launched, it initializes communication with the Raspberry Pi 4.
2. Reading Sensor Data: The DHT 11 sensor takes temperature and humidity readings of the coffee beans sample, and these readings are displayed on the web interface.
3. Capturing an Image: When the user clicks the capture button on the web application, the Picamera module takes a high-resolution image of the green coffee beans sample.
4. Inference and Quality Assessment: The captured image is processed by the YOLO v8 model, which performs inference to detect defects in the coffee beans. The quality assessment is based on the number and types of defects identified, in conjunction with the sensor readings.
5. Displaying Results: The results of the quality assessment, including defect counts and sensor readings, are displayed on the web application interface. If the readings and defect counts are within acceptable limits, the beans are deemed of good quality.

3.3.1.3 Block Diagram of System Architecture

The block diagram of the system architecture visually represents the interaction between the hardware and software components, providing a clear overview of the data flow and integration points. Figure 3.7 shows the block diagram of the system architecture illustrating the interaction between the Raspberry Pi 4, DHT 11 sensor, Picamera module, and the web application interface.

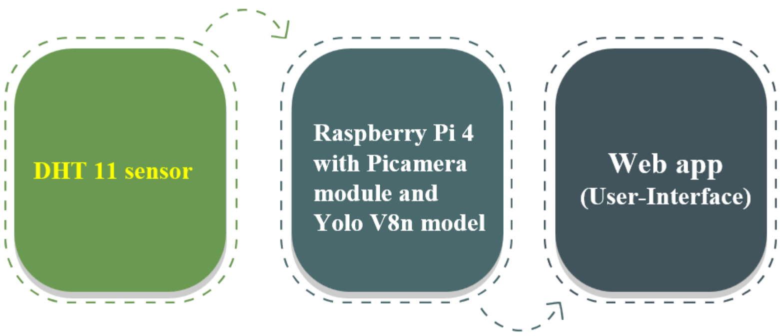

Figure 3.7: System Architecture

3.3.2 Web Application Interface

The web application interface for the IoT device was designed using the Streamlit Python library, which provides a simple and intuitive way to create interactive web applications. The interface is user-friendly and offers several key features that facilitate the quality assessment process of green coffee beans.

The design of the web application interface focuses on simplicity and ease of use. Key elements of the interface include:

- **Sidebar:** The sidebar contains options for image selection and report generation.
- **Main Display Area:** The main area displays sensor readings, captured images, and the results of the quality assessment.
- **Buttons:** Clearly labelled buttons for uploading images, capturing images, and sending reports to ensure intuitive navigation.

Figure 3.8 shows the screenshot of the web application interface showing the upload and capture buttons, sensor readings, and quality assessment results.

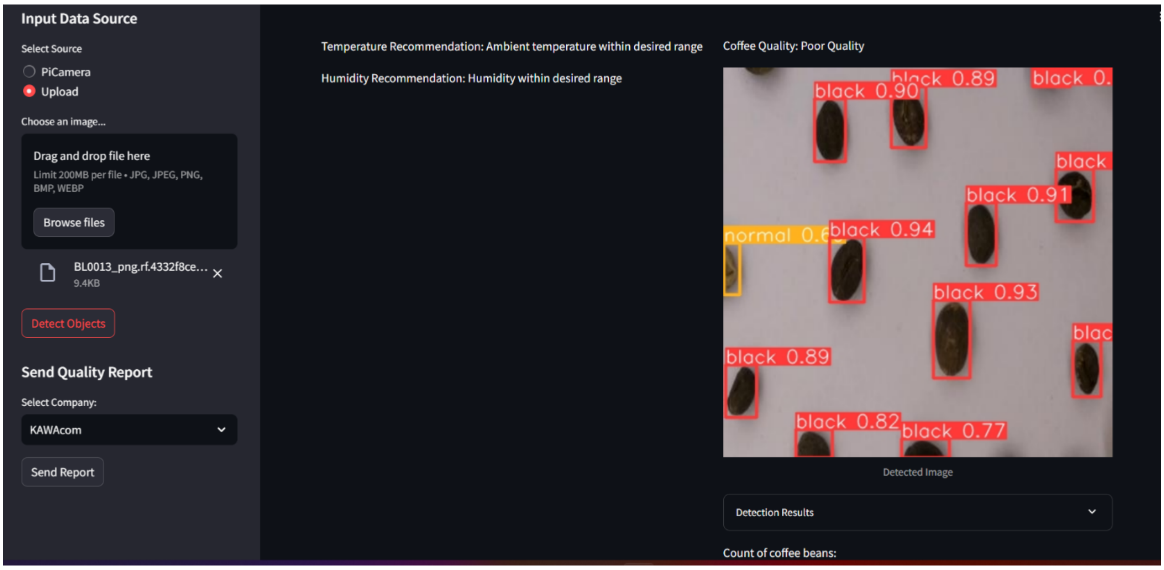

Figure 3.8: Web App Interface

Key Features of the web application interface are;

- **Image Selection Options:**

  Option 1: Upload Image; Users can upload an image directly from the Raspberry Pi 4's storage by clicking the upload button on the sidebar. This feature is useful for analyzing previously captured images.
  Option 2: Capture Image: Users can capture a new image using the Picamera module by clicking the capture button located just below the upload button on the sidebar. This allows for real-time image capturing and analysis.

- **Sensor Readings Display:**

  The interface displays the temperature and humidity readings from the DHT 11 sensor in real time. This information is crucial for assessing the environmental conditions affecting the coffee beans.

- **Quality Assessment Results:**

  After an image is uploaded or captured, the YOLO v8 model performs inference on the image to detect defects. The results, including the number and types of defects, are displayed on the interface, providing immediate feedback on the quality of the coffee beans.

- **Report Generation and Distribution:**

  1. **Send Report Button:** Users can generate a PDF report of the quality assessment results by clicking the Send Report button.

  2. **Email Distribution:** The interface includes a drop-down list of clients' email addresses in the sidebar. Users can select an email address and send the report via email. This feature uses the SMTPlib library to facilitate email sending.

  3. **Local Storage**: A copy of the report is also stored locally on the Raspberry Pi 4 for record-keeping and future reference.

To support the functionality of the web application interface and the hardware components, several libraries were installed on the Raspberry Pi 4:

- **Adafruit Circuit-Python:** Used for interfacing with the DHT 11 sensor to read temperature and humidity data.
- **Picamera2:** Provides the necessary tools to control the Picamera module for capturing high-resolution images.
- **Streamlit:** The main library used to create the interactive web application interface.
- **Ultralytics:** Used to run the YOLO v8 model for image inference and defect detection.
- **ReportLab:** Facilitates the creation of PDF reports based on the quality assessment results.
- **SMTPlib:** Enables sending emails directly from the Raspberry Pi 4, ensuring that clients receive their reports promptly.

Figure 3.9 shows a block diagram illustrating the flow of data from image capture to quality assessment and report generation.

By integrating these libraries, the system ensures seamless interaction between the hardware components and the web application interface, providing a comprehensive and efficient tool for

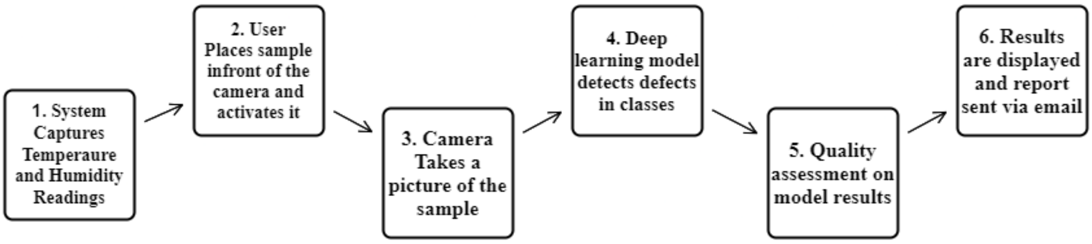

Figure 3.9: Block diagram for the system flow

green coffee bean quality assessment.

3.4 Model Development

The model development phase involved selecting appropriate models, training them, and evaluating their performance to identify the best model for the detection task.

3.4.1 Model Selection

For the task of assessing the quality of green coffee beans, we evaluated three deep-learning models: SSD-MobileNet v2, YOLO v7, and YOLO v8. These models were selected based on their strong performance in object detection tasks and their suitability for deployment on resource-constrained devices such as the Raspberry Pi 4.

- **SSD-MobileNet v2:** SSD with MobileNet v2 is a popular model known for its balance between accuracy and computational efficiency. MobileNet v2 serves as the backbone of the SSD architecture, providing feature extraction capabilities with reduced computational overhead, making it suitable for real-time applications on embedded devices.
- **YOLO v7:** YOLO v7 is an iteration of the YOLO family, renowned for its high-speed performance and accuracy in object detection. YOLO v7 introduces several optimizations over its predecessors, making it a strong candidate for tasks requiring real-time processing and high detection accuracy.
- **YOLO v8:** YOLO v8 is the latest version in the YOLO series, offering further improvements in detection accuracy and speed. YOLO v8 incorporates advanced architectural enhancements and training techniques, making it the most accurate and efficient model among the three evaluated for our specific use case.

The selection of these models was driven by their demonstrated performance in object detection tasks similar to our project requirements. The key considerations included:

1. **Accuracy:** The ability to accurately detect and classify defects in green coffee beans was paramount. YOLO v8, being the latest iteration, showed superior accuracy compared to

SSD-MobileNet v2 and YOLO v7 in benchmark tests.

2. **Speed:** Real-time processing capability was essential for our application. YOLO models, particularly YOLO v7 and v8, are optimized for high-speed inference, making them suitable for deployment on a Raspberry Pi 4.

3. **Resource Efficiency:** The computational resources available on the Raspberry Pi 4 are limited. SSD-MobileNet v2 is known for its efficiency on resource-constrained devices, while YOLO v8, despite being more complex, offers optimized performance that can be handled by the Raspberry Pi 4 with careful implementation.

3.4.1.1 Model Architectures

- **SSD-MobileNet v2 Architecture:** The SSD-MobileNet v2 architecture combines the SSD framework with the MobileNet v2 backbone. As illustrated in Figure 3.10, MobileNet v2 utilizes depth-wise separable convolutions to reduce the number of parameters and computational cost, enabling efficient feature extraction. SSD then applies multiple convolutional filters at different scales to detect objects of varying sizes.

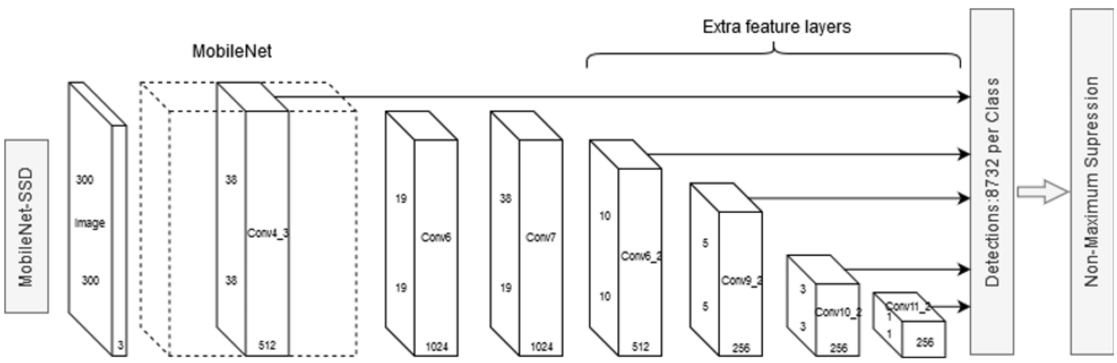

Figure 3.10: Architecture of the SSD MobileNet v2 deep learning model [10]

- **YOLO v5 Architecture:** YOLO v5 breaks down into three parts: the backbone, neck, and head. The backbone, a modified Darknet structure called CSPDarknet53, extracts features from the image at different resolutions as shown in Figure 3.11. The neck, consisting of SPPF and Path Aggregation Network (PAN), fuses these features to create a rich understanding of the image. Finally, the head, based on the YOLOv3 design, interprets this information to predict bounding boxes and class labels for objects detected within the image. This combination allows YOLOv5 to achieve good accuracy while maintaining real-time processing speeds.

- **YOLO v7 Architecture:** YOLO v7 builds on the principles of the original YOLO model, employing a single neural network to predict bounding boxes and class probabilities directly from full images in one evaluation. As illustrated in Figure 3.12, YOLO v7

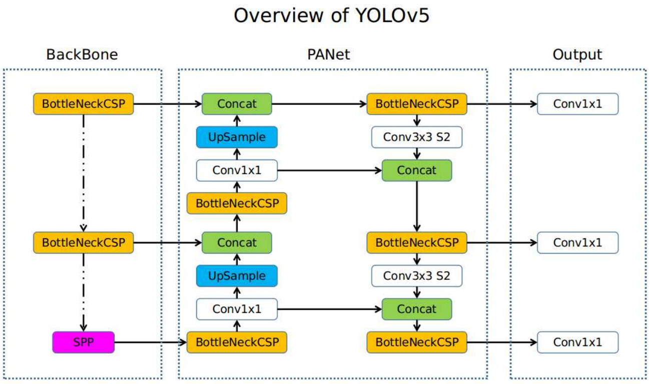

Figure 3.11: Architecture of YOLO v5 deep learning model [11]

introduces architectural improvements such as better backbone networks and anchor box design, enhancing its accuracy and speed.

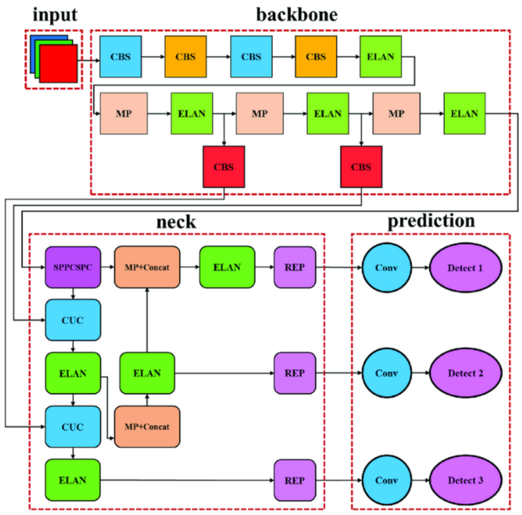

Figure 3.12: Architecture of YOLO v7 deep learning model [12]

- **YOLO v8 Architecture:** A modified version of the CSPDarknet53 architecture forms the backbone of YOLOv8 as illustrated in Figure 3.13. This architecture consists of 53 convolutional layers and employs cross-stage partial connections to improve information flow between the different layers.

  YOLO v8 further refines the YOLO architecture by incorporating advanced techniques like CSPNet (Cross Stage Partial Networks) to enhance feature learning and reduce computational load. It also uses more sophisticated anchor box mechanisms and loss functions to improve detection accuracy. YOLO v8's architecture is designed to balance speed and accuracy, making it ideal for real-time applications.

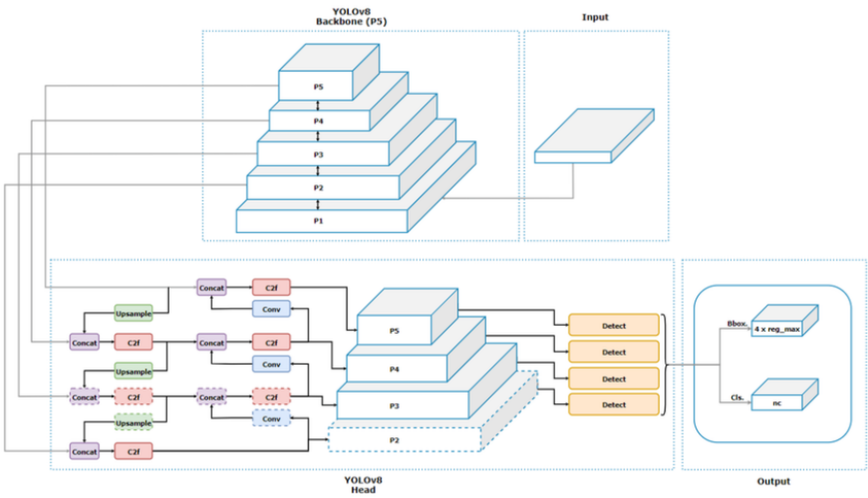

Figure 3.13: Architecture of YOLO v8 deep learning model [9]

3.5 Training Process

3.5.1 Dataset

The dataset used for training the deep learning models consisted of images of green coffee beans sourced from two primary origins:

1. **Roboflow Universe Dataset:** We sourced a dataset of 4,265 images from Roboflow Universe. This dataset provided a diverse collection of green coffee bean images annotated with four classes: Black, Broken, Cherry, and Normal.

2. **Local Dataset from UCDA:** To ensure the dataset was representative of local conditions, additional samples of green coffee beans were collected from UCDA. A total of 500 images were captured and then annotated under the strict guidance of UCDA experts, ensuring accurate and consistent labelling.

After combining both datasets, the total number of images were 7,020. To enhance the dataset's robustness and variability, we applied various augmentation techniques using Roboflow Universe's tools.

3.5.2 Preprocessing and Augmentation Steps

Processing step	**Action Taken**
Orientation	auto-orient
Resolution	416x416
Augmentation	Flip, Horizontal, vertical
Splitting	Train, validation, test sets

Table 3.1: Preprocessing and Augmentation steps

Table 3.1 shows some of the steps and actions taken during data preprocessing.

The dataset was then split into training, validation, and test sets with the following proportions:

Training Set: 70% (4,914 images)

Validation Set: 15% (1,053 images)

Test Set: 15% (1,053 images)

Sample Images:

Figure 3.14: Sample image from the Roboflow dataset [13]

Figure 3.15: Sample image from the local dataset

3.5.3 Training Setup

3.5.3.1 Hardware Specifications

The training process was conducted on Google Colaboratory, which provided access to powerful virtual resources, including A100 GPUs. These high-performance GPUs enabled efficient training of our models within the maximum runtime of three hours per session.

3.5.3.2 Software Tools

Several software tools were employed to facilitate the training process:

- **Google Colaboratory:** For accessing GPU resources and running the training scripts.
- **Roboflow:** For dataset management, preprocessing, and augmentation.
- **TensorFlow and PyTorch:** Frameworks used to implement and train the models.
- **Ultralytics:** For utilizing the YOLO v8 model.

3.5.3.3 Hyperparameters

The following hyperparameters were used during the training of the models:

- **Learning Rate:** 0.001
- **Batch Size:** 16
- **Number of Epochs for YOLO models:** 100
- **Training Steps for SSD MobileNet v2:** 50,000
- **Optimizers:** Adam, AdamW

These hyperparameters were chosen based on preliminary experiments and literature recommendations to ensure optimal performance and convergence of the models.

3.5.3.4 Training Process work flow

1. **Data Loading:** The pre-processed and augmented dataset was loaded into the training pipeline on Google Colaboratory.
2. **Model Initialization:** Each model (SSD-MobileNet v2, YOLO v7, YOLO v8) was initialized with the specified architectures and pre-trained weights.
3. **Training Loop:** The training loop involved forward passes through the models, loss computation, backpropagation, and parameter updates using the Adam optimizer.
4. **Validation:** After each epoch, the models were evaluated on the validation set to monitor performance and prevent overfitting.
5. **Testing:** Once training was completed, the models were tested on the test set to assess their final performance.

The performance of each model was evaluated based on standard metrics such as accuracy, precision, recall, and F1-score. YOLO v8 emerged as the best-performing model, achieving the highest accuracy and most reliable defect detection.

3.5.4 Model Evaluation

3.5.4.1 Metrics

To evaluate the performance of the models, we used several standard metrics commonly employed in object detection tasks. These metrics include precision, recall, F1-score, and mean Average Precision (mAP). Each metric provides valuable insights into different aspects of the model's performance.

Precision: Precision measures the proportion of true positive detections among all positive detections made by the model. It is defined as:

$$\text{Precision} = \frac{TP}{(TP + FP)} \tag{3.1}$$

where:

- TP = True Positives (correctly detected defects)
- FP = False Positives (incorrectly detected defects)

Recall: Recall measures the proportion of true positive detections among all actual positive instances in the dataset. It is defined as:

$$\text{Recall} = \frac{TP}{(TP + FN)} \tag{3.2}$$

where:

• FN = False Negatives (missed defects) F1-score. The F1-score is the harmonic mean of precision and recall, providing a single metric that balances both precision and recall. It is defined as:

$$\text{F1-score} = 2 \times \frac{(Precision \times \text{Recall})}{(Precision + \text{Recall})} \tag{3.3}$$

Mean Average Precision (mAP): mAP evaluates the precision-recall trade-off at different threshold levels, providing an overall measure of the model's detection accuracy.

3.5.4.2 Results

For the YOLO models (YOLO v5, YOLO v7, and YOLO v8), detailed evaluation metrics were captured, including precision, recall, F1-score, and mAP. Curves and visualizations were generated to better understand the performance dynamics during training and evaluation.

1. YOLO v8 Results:

Training Loss Curve: Figure 3.26 shows the training loss curve for the YOLO V8n model. It demonstrates a consistent decrease in loss over the 100 training epochs. This indicates that the model is learning effectively from the dataset, improving its predictions with each iteration. A lower training loss signifies that the model is minimizing the error between its predictions and the actual labels in the dataset.

Precision-Recall Curve: Demonstrates the precision and recall trade-off across different confidence thresholds.

F1 Score Curve: Figure 3.17 shows the F1-score progression during model training.

Precision Curve: Figure 3.18 displays the precision values over the training epochs.

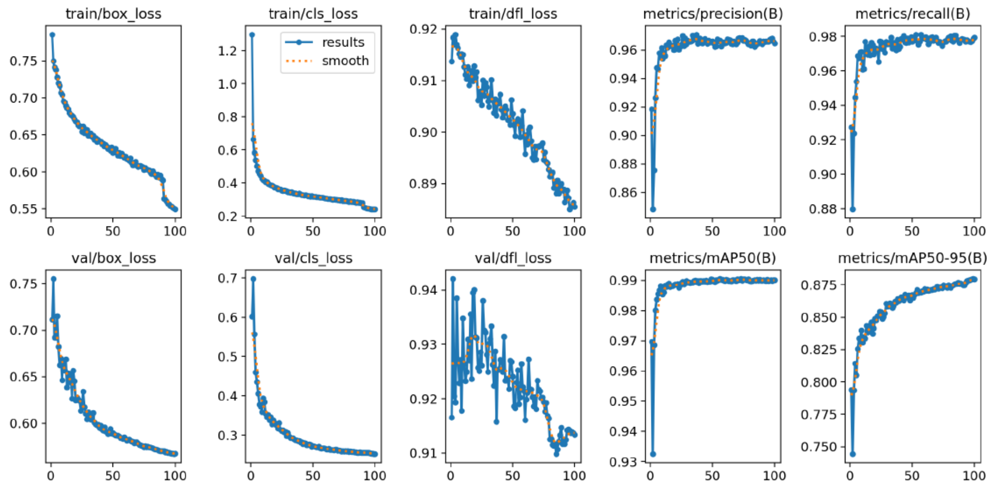

Figure 3.16: Training Loss curve

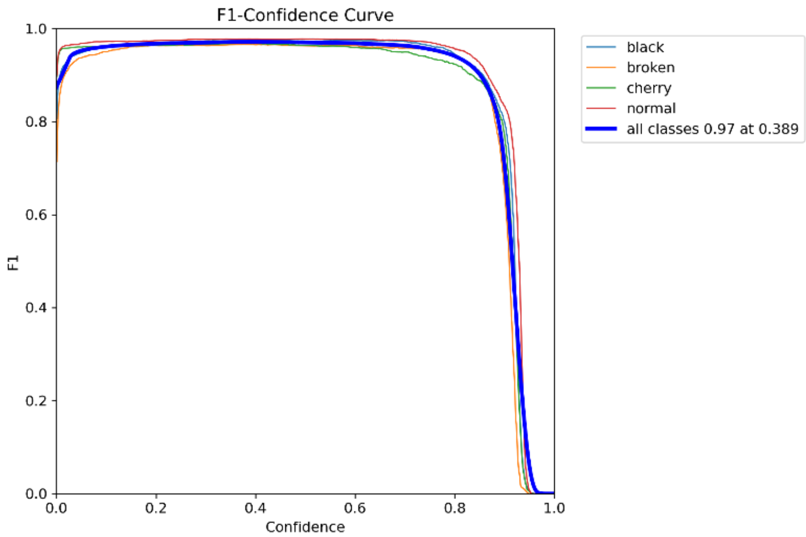

Figure 3.17: F1 confidence curve

Recall Curve: Figure 3.19 displays the recall values over the training epochs.

Confusion Matrix: Figure 3.20 illustrates the true positive, false positive, and false negative counts for each class.

2. YOLO v5 and YOLO v7 Results:

Similar evaluation metrics and curves were generated for YOLO v5 and YOLO v7, though the performance was slightly lower compared to YOLO v8.

3. SSD-MobileNet v2 Results

For the SSD-MobileNet v2 model, we primarily focused on the mAP score due to the lack of detailed evaluation curves and metrics. The mAP score provides a comprehensive measure of the model's performance across all classes, but without additional metrics, it's challenging to

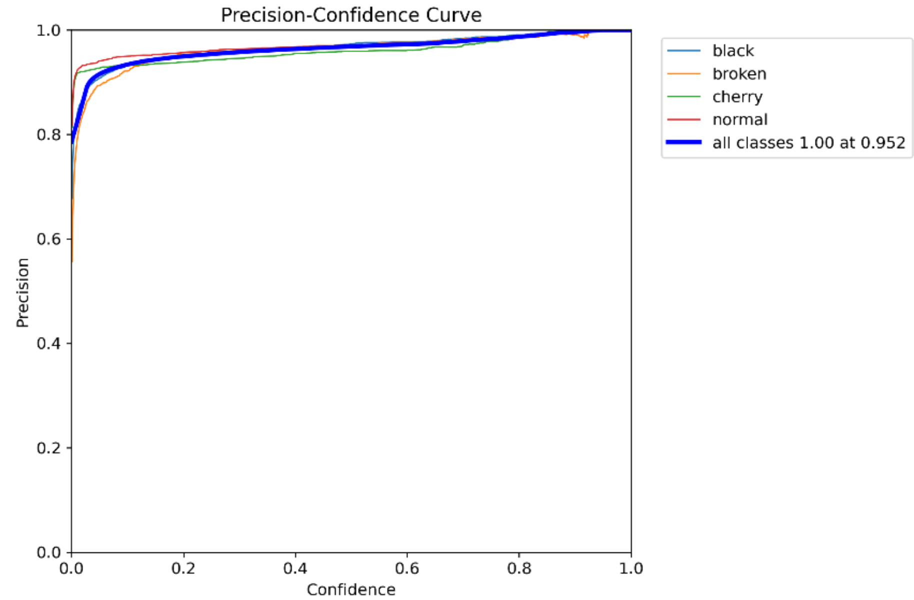

Figure 3.18: Precision Confidence Curve

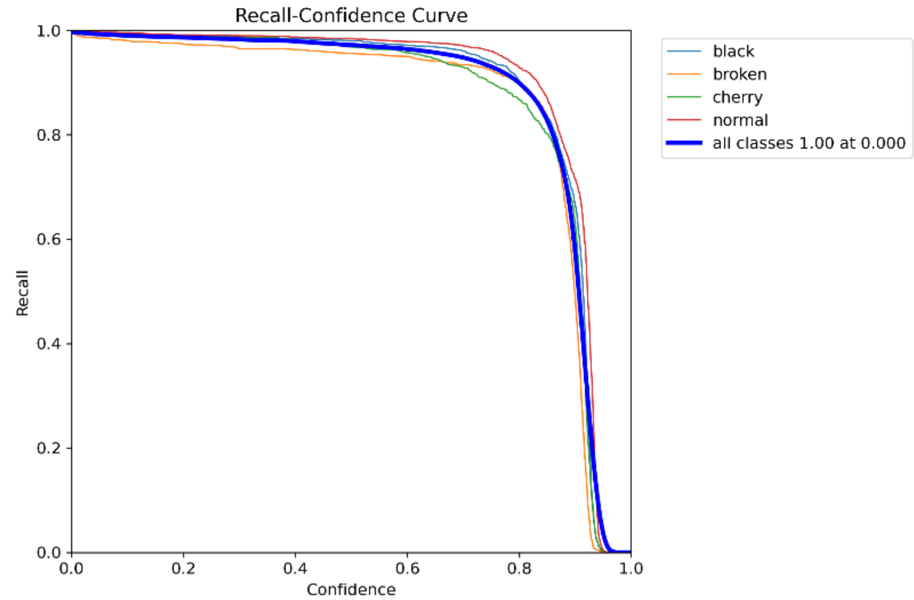

Figure 3.19: Recall Confidence Curve

perform a detailed analysis.

mAP Score for SSD-MobileNet v2: The model achieved a mAP score of 68%.

While the SSD-MobileNet v2 model provided a baseline performance, the detailed metrics and visualizations available for the YOLO models, especially YOLO v8, demonstrated its superior performance in detecting defects in green coffee beans.

3.5.5 Comparative Analysis

The comparative analysis of the models highlighted the following key points:

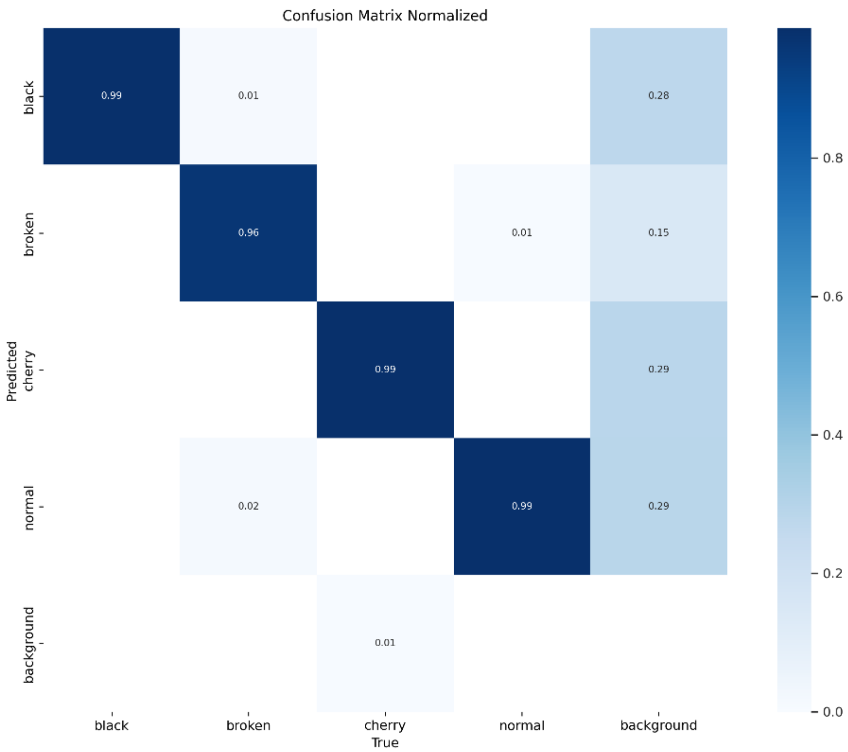

Figure 3.20: Confusion Matrix

Accuracy: YOLO v8 achieved the highest precision, recall, and F1-score among the evaluated models, indicating superior defect detection capabilities.

Speed: Both YOLO v7 and YOLO v8 demonstrated high-speed inference, essential for real-time applications.

Robustness: The detailed evaluation metrics and curves for YOLO v8 provided deeper insights into the model's robustness and reliability compared to SSD-MobileNet v2.

Based on the comprehensive evaluation metrics, YOLO v8 emerged as the best-performing model, offering the highest accuracy and reliability for assessing the quality of green coffee beans. The detailed visualizations further reinforced YOLO v8's suitability for deployment in the IoT device, ensuring accurate and efficient quality assessment.

3.6 Hardware Implementation

This phase involved setting up the hardware components required for the IoT device.

3.6.1 Raspberry Pi 4

For this project, a Raspberry Pi 4 with the 64-bit Desktop version of its operating system was used. This was installed via the Raspberry Pi Imager on a Windows 11 machine. The initial setup involved setting up Wi-Fi, changing the default password, and enabling interfaces for the DHT 11 sensor and Picamera module. This was followed by updating and upgrading the preinstalled libraries and installing additional ones like Adafruit Circuit Python for the DHT 11 sensor, Picamera2 for the camera module, Streamlit for the web interface, Ultralytics for the YOLO v8 model, and ReportLab and SMTPlib for generating and sending PDF reports. The Picamera was connected via the camera interface, and the DHT 11 sensor was connected to the GPIO pins, ensuring the hardware setup was ready for data collection and image capture.

3.7 Prototype Assembly and Testing

The final phase involved assembling the prototype and conducting testing to ensure its functionality and reliability.

3.7.1 Assembly

3.7.1.1 Physical Assembly

The physical assembly of the prototype involved creating a compact and functional housing for all the hardware components. As illustrated in Figure 3.21, the schematic design of the prototype was created using EAGLE software. Using SolidWorks, a 3D casing shown in Figure 3.22 was designed that securely houses the Raspberry Pi 4, Picamera module, and DHT 11 sensor. The casing was designed to be lightweight and portable, ensuring ease of use in different settings, such as the quality assessment labs at UCDA. The layout was meticulously planned to provide adequate ventilation and easy access to the ports and connectors of the Raspberry Pi 4.

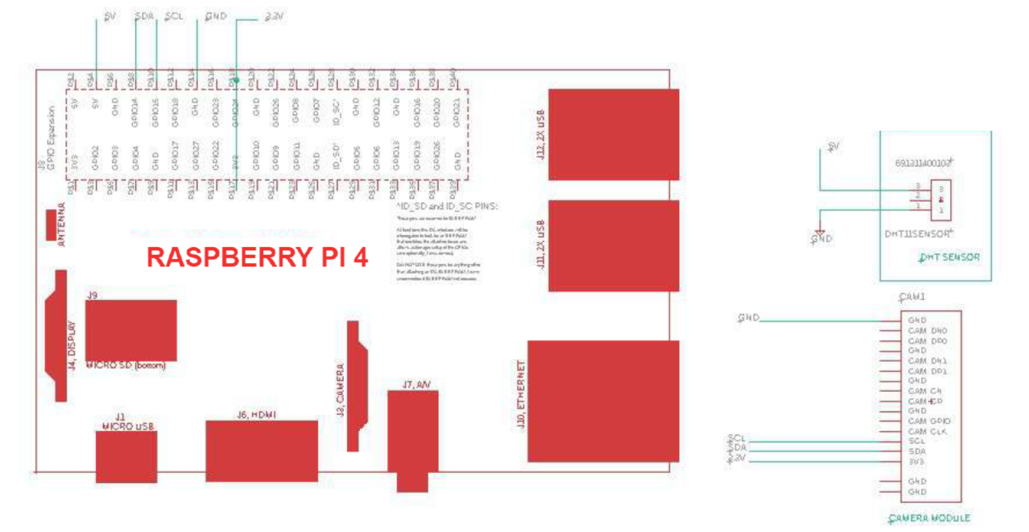

Figure 3.21: Schematic Design of the Prototype

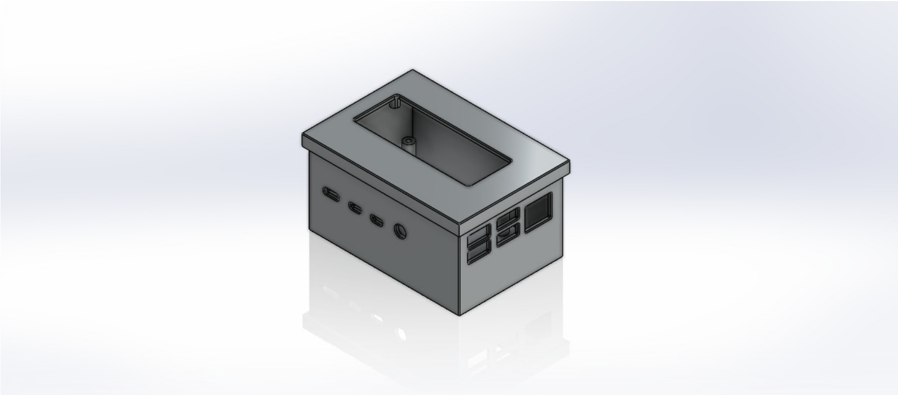

Figure 3.22: Prototype 3D case design in Solid Works.

3.7.1.2 Wiring and Connections

The wiring and connections between the components were carefully planned to ensure reliable data transmission and power supply. The DHT 11 sensor was connected to the Raspberry Pi 4 via the GPIO pins, with the data pin linked to a designated GPIO pin and the power and ground pins connected to the respective 3.3V and GND pins. The Picamera module was connected through the dedicated camera interface, with the ribbon cable securely attached to both the camera and the Raspberry Pi. All connections were tested for stability and functionality, ensuring that the sensor readings and image captures were accurately transmitted to the Raspberry Pi for processing.

3.7.1.3 Field Testing

3.7.1.3.1 Real-World Testing

To evaluate the device's performance in a real-world setting, we conducted field tests in collaboration with the Uganda Coffee Development Authority (UCDA). The tests were performed on various samples of green coffee beans in the UCDA lab. The device successfully assessed the quality of robusta coffee beans, generating accurate reports based on predefined metrics. However, the device encountered challenges with samples containing a large number of coffee beans that were too close to each other. This limitation arose because the training images primarily featured coffee beans spaced apart. Additionally, the Raspberry Pi camera module experienced issues with light balance and exposure, affecting image quality under varying lighting conditions.

3.7.1.3.2 Feedback

The feedback from UCDA was instrumental in identifying areas for improvement. The UCDA quality assurance officers noted the device's accurate performance on well-spaced samples but highlighted the difficulties with densely packed beans.

Chapter 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents the results obtained from the implementation and testing of the portable IoT device for assessing the quality of green coffee beans. The discussion includes an analysis of model performance, quality assessment results and feedback from prototype testing.

4.2 Model Performance

The performance of the models—SSD-MobileNet v2, YOLO v5, YOLO v7, and YOLO v8—was evaluated using mean Average Precision (mAP) metrics as illustrated in Table 4.1. YOLO v8 exhibited superior performance with the highest mean Average Precision (mAP). These results underscore YOLO v8's robustness and accuracy in detecting defects in green coffee beans.

Model	**mAP@0.5**	**mAP@0.5:0.95**
SSD-MobileNet-V2	0.68	0.677
YOLO v5	0.94	0.85
YOLO v7	0.962	0.867
YOLO v8	**0.99**	**0.879**

Table 4.1: Model results

4.3 Quality Assessment Metrics and Results

The quality assessment results, derived from the YOLO v8 model, indicated a high accuracy in identifying and categorizing defects such as black, broken, and cherry beans. The quality assessment metrics used by UCDA are based on a strict count of defects;

Quality grades are:

Good quality - less than 7% defects in sample

Moderate quality- between 7% and 10% defects in sample

Poor quality- greater than 10% defects in the sample

Desired relative humidity: 35-75%

Desired temperature: $(22.0 \pm 6)°C$

These assessments were validated against the metrics provided by UCDA and were found to be consistent with manual evaluations performed by UCDA experts. The detailed results, including temperature and humidity readings from the DHT 11 sensor, were accurately recorded and displayed on the web application interface as illustrated in Figure 4.1.

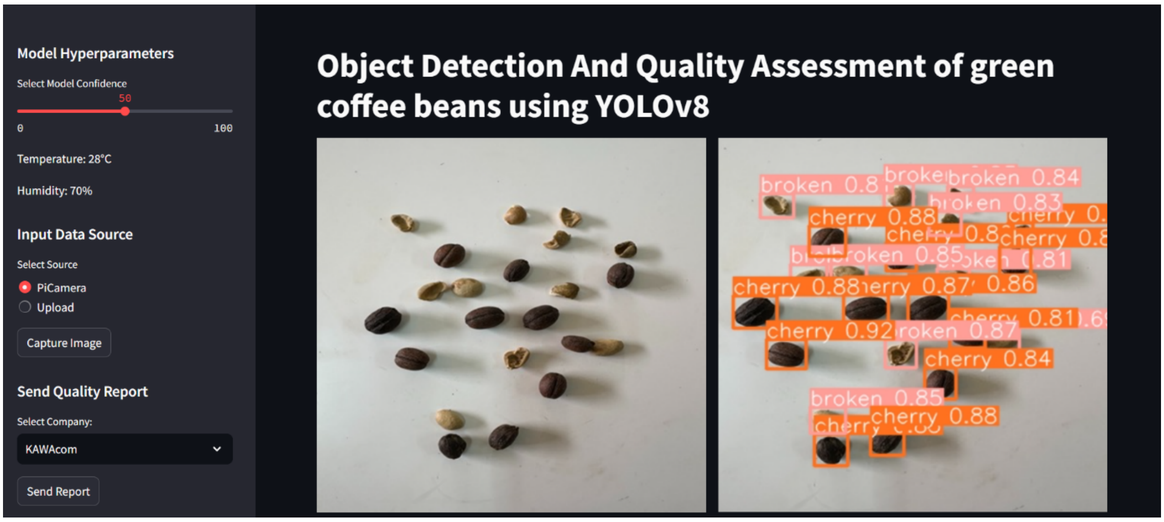

Figure 4.1: Web App Interface with DHT readings

4.4 Prototype Testing and Feedback

Field tests conducted at the UCDA lab demonstrated the prototype's effectiveness in assessing coffee bean quality. The device performed well on samples with well-spaced beans, accurately identifying defects and generating quality assessments. However, it faced challenges with densely packed beans due to the limitations of the training dataset and the camera module's resolution and light balance issues. Feedback from UCDA prompted iterative improvements, including expanding the training dataset to include images of densely packed beans and enhancing the camera settings for better light balance and exposure.

For each sample assessed, the prototype generates a detailed PDF report. This report includes an image of the coffee beans with bounding boxes around detected defects as shown in Figure 4.2, the number of detected classes (e.g., black, broken, cherry, and normal beans), and temperature and humidity readings from the DHT 11 sensor. Additionally, the report as shown in Figure 4.3, provides comments on the quality of the sample based on the UCDA's assessment metrics. A copy of this report is stored locally on the Raspberry Pi 4, and another copy is emailed to the client's email address for their records.

Coffee Beans Quality Report

Quality Assessment: Poor quality

Reason: High proportion of black, broken, or cherry beans. Unsuitable for premium coffee.

Humidity : 70

Humidity recommendation: Humidity within desired range

Temperature : 28

Temperature recommendation: Ambient temperature within desired range

Detected Classes:

- normal
- black
- normal
- normal
- normal
- normal
- black
- black
- black
- normal
- black

Total Count: 11

Image with Bounding Boxes:

Figure 4.2: Sample PDF Report Generated by the Prototype

Figure 4.3: Image with detected defects.

4.5 Comparative Analysis

A comparative analysis between traditional manual sorting methods and the automated IoT device highlights significant improvements in efficiency and accuracy. Traditional methods are labour-intensive and subject to human error, while the developed device offers quick, consistent, and objective assessments. Table 4.2 compares the key aspects of traditional and developed IoT devices, emphasizing the advantages of the IoT-based approach:

Aspect	**Traditional Methods**	**IoT Device (Developed Prototype)**
Speed	Slow	Fast
Accuracy	Variable (subject to human error)	High (consistent, model-driven)
Labor-Requirement	High	Low
Data Logging	Manual	Automated (digital storage and reports)
Objectivity	Low (subjective assessment)	High (objective and repeatable)

Table 4.2: Comparative Analysis of Traditional Methods and IoT Device for Coffee Beans Quality Assessment

4.6 Summary

The evaluation presented in this chapter underscores the effectiveness of the YOLO v8 model in defect detection and quality assessment using UCDA metrics. Practical testing feedback guided iterative improvements, affirming the prototype's real-world applicability in field tests at the UCDA lab. Comparative analysis highlighted the efficiency and accuracy of technological methods over traditional approaches, significantly enhancing green coffee bean quality assessment through deep learning and IoT integration.

Chapter 5

CHALLENGES, RECOMMENDATIONS, FUTURE WORK AND CONCLUSION

This chapter outlines the challenges faced by the project, offers recommendations, and provides conclusions and suggestions for future research.

5.1 Conclusion

The design and implementation of the portable IoT device for assessing the quality of green coffee beans using deep learning models yielded impressive results. The device effectively identified and categorized defects in coffee beans, producing comprehensive reports that align with the quality metrics of UCDA. This makes it an invaluable tool for quality assessment labs focused on green coffee beans.

5.2 Challenges

Several challenges were encountered during the project:

- **Limited Availability of Local Datasets:** Scarcity of locally relevant data for training the deep learning models, which necessitated additional data collection efforts.
- **Model Training Resources:** The available computational resources for deep learning model training were limited, with Google Colaboratory providing only three runtime hours per day.
- **Library Conflicts:** The Raspberry Pi 4 faced compatibility issues with certain libraries, such as conflicts between the Firebase library and the Streamlit library.

- **Camera Resolution:** The Picamera module struggled with light balance and exposure issues, affecting the quality of the captured images.

5.3 Recommendations

To enhance the functionality and performance of the device, the following recommendations are made:

- **Cloud Integration:** Connecting the device to a cloud platform for secure storage and to facilitate remote access and management of data.
- **Improved Hardware:** Considering a more powerful microcontroller, such as the NVIDIA Jetson Nano, which offers superior computational capabilities and better support for deep learning applications.
- **Enhanced Camera Module:** Upgrading to a higher resolution camera with better light balance and exposure control to improve image quality under various lighting conditions.

5.4 Future Work

Future work should focus on expanding the application and capabilities of the device:

- **Broader Value Chain Adaptation:** Adapting the device for use in earlier stages of the coffee value chain, including farm-level quality assessments and processing stages.
- **Agricultural Product Assessment:** Extending the device's functionality to assess the quality of other agricultural products, such as maize and beans, using tailored deep learning models and relevant quality metrics.
- **Automated Calibration:** Develop automated calibration routines for sensors and camera settings to ensure consistent and reliable operation in diverse environments.
- **User Interface Enhancements:** Improve the web application interface to provide more intuitive controls and better visualization of assessment results.

By addressing these challenges and incorporating the recommendations and future work, the IoT device can be further refined to provide even more accurate, reliable, and versatile quality assessments for a wide range of agricultural products.

Bibliography

[1] P. J. V. Asten, “Coffee growing regions in uganda,” September 2012, online. [Online]. Available: https://www.researchgate.net/figure/Study-area-and-the-five-main-coffee-growing-regions-in-Uganda_fig1_269338210

[2] Techvidian, “How iot works,” online. [Online]. Available: https://techvidvan.com/tutorials/how-iot-works/

[3] S. l. E. India, “Raspberry pi 4,” online. [Online]. Available: https://www.silverlineelectronics.in/products/raspberry-pi-4-model-b-2gb

[4] M. Albahar, “A survey on deep learning and its impact on agriculture: Challenges and opportunities,” online. [Online]. Available: https://www.mdpi.com/2077-0472/13/3/540

[5] Y. H. B. Shao, “Deep learning based coffee beans quality,” in *IEEE International Conference on e-Business Engineering (ICEBE)*, 2022.

[6] B. L. Chala, “Hulling of coffee beans,” online. [Online]. Available: https://www.researchgate.net/figure/Primary-coffee-processing-pathways-to-produce-green-beans-following-the-dry-and-wet_fig1_326711035

[7] “Raspberry pi 4 model b,” online. [Online]. Available: https://thepihut.com/products/raspberry-pi-4-model-b

[8] L. Electronics, “Dht 11 sensor,” online. [Online]. Available: https://www.olelectronics.in/product/dht11-temperature-and-humidity-sensor

[9] “Yolo v8 architecture,” online. [Online]. Available: https://www.researchgate.net/figure/The-improved-YOLOv8-network-architecture-includes-an-additional-module-for-the-head_fig2_372207753

[10] “Ssd mobilenet architecture,” online. [Online]. Available: https://www.researchgate.net/figure/The-architecture-of-SSD-MobileNet-v2-model-9_fig4_367812032

[11] seekFire, “Architecture of yolo v5,” online. [Online]. Available: https://github.com/ultralytics/yolov5/issues/280

[12] R. Gate, "Yolo v7 architecture," online. [Online]. Available: https://www.researchgate.net/figure/YOLOv7-architecture-Different-color-means-different-function-performed-in-a-single_fig1_370669942

[13] CPE, "Roboflow universe," online. [Online]. Available: https://app.roboflow.com/green-coffee-beans/classified-coffee-beans/3

[14] seedstudio, "Raspberry pi 4 model b," May 20 2020, online. [Online]. Available: https://www.seeedstudio.com/blog/2020/05/28/meet-the-brand-new-raspberry-pi-4-8gb-ram/

[15] T. E. Projects, "Dht 11 sensor specifications," online. [Online]. Available: https://www.theengineeringprojects.com/2019/03/introduction-to-dht11.html

[16] "Picamera module specifications," online. [Online]. Available: https://components101.com/misc/pi-camera-module

[17] R. Pi, "Picamera module," online. [Online]. Available: https://www.raspberrypi.com/products/camera-module-v2/

[18] N. Thai-Nghe, N. T. Tri, and N. H. Hoa, "Deep learning for rice leaf disease detection in smart agriculture," in *Artificial Intelligence in Data and Big Data Processing*. Berlin/Heidelberg, Germany: Springer International Publishing, 2022, pp. 659–670.

[19] M. Shaila and N. Begum, "Ancient farming methods of seed storage and pest management practices in india—a review," *Plant Archives*, vol. 21, pp. 499–509, 2021.

[20] S. Dargan, M. Kumar, M. Ayyagari, and G. Kumar, "A survey of deep learning and its applications: A new paradigm to machine learning," *Archives of Computational Methods in Engineering*, vol. 27, pp. 1071–1092, 2020.

[21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[22] H. Abdullahi, R. Sheriff, and F. Mahieddine, "Convolution neural network in precision agriculture for plant image recognition and classification," in *Proceedings of the 2017 Seventh International Conference on Innovative Computing Technology (INTECH)*. Luton, UK: IEEE, 2017, pp. 256–272.

[23] A. Ajit, K. Acharya, and A. Samanta, "A review of convolutional neural networks," in *Proceedings of the 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Vellore, India, 2020.

[24] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *arXiv*, vol. arXiv:1409.2329, 2014, in press.

Appendix

A.1 Hardware Specifications

- Detailed specifications of all hardware components used.

Raspberry Pi 4 model B

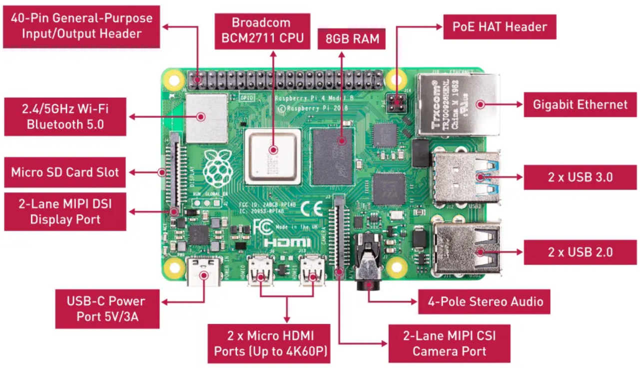

Figure 5.1: Specifications of the raspberry pi 4 model B that was used [14]

DHT 11 sensor

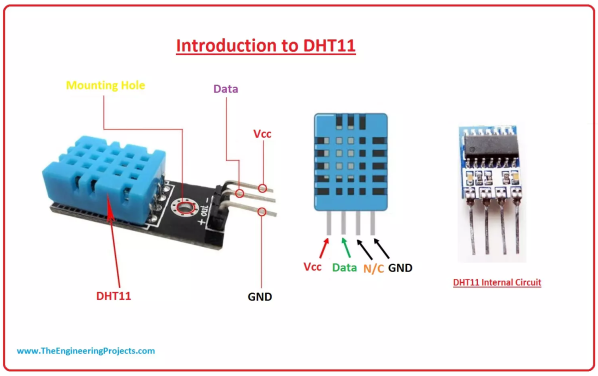

Figure 5.2: Specifications of DHT 11 sensor that was used [15]

DHT11 Pinout consists of 4 Pins in total, listed below from left to right:

- **Vcc:** Need to provide +5V at this pinout.
- **Data:** It's the digital output pin, that gives either 0V or 5V.

- **NC:** Not Connected. (It's left open for future design)
- **GND:** Need to provide Ground at this pinout.

Picamera module

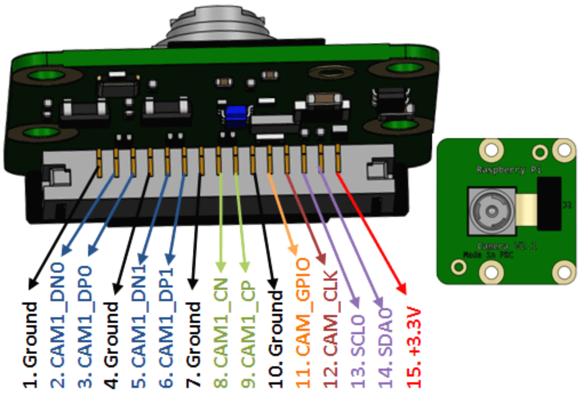

Figure 5.3: Picamera Module Pinout [16]

A.2 Field Images

Figure 5.4: Prototype testing session at the UCDA lab with my project partner.

Figure 5.5: Coffee Hulling at the UCDA Lab