Disease Detection in Cassava Leaves

A study submitted in partial fulfilment
of the requirements for the degree of
M.Sc. Data Science

at

THE UNIVERSITY OF SHEFFIELD

by

Kamesh Kotwani

August 2024

Table of Contents

Abstract

Background: Cassava (Manihot esculenta) is a staple food crop for millions, particularly in sub-Saharan Africa, where it is a primary source of calories. Despite its importance, cassava cultivation faces significant challenges from various diseases, including Cassava Mosaic Disease (CMD) and Cassava Bacterial Blight (CBB), which can cause devastating yield losses and threaten food security. Early and accurate detection of these diseases is crucial to mitigate their impact and ensure stable food production. Traditional methods of disease identification, relying on visual inspection, are often inaccurate and time-consuming, making them inadequate for large-scale monitoring. Recent advancements in deep learning, specifically Convolutional Neural Networks (CNNs), offer a promising solution by enabling automated, rapid, and precise identification of plant diseases from images. These techniques can significantly enhance disease management strategies and improve agricultural productivity by providing timely information to farmers and agronomists.

Aims: The primary aim of this study was to evaluate the effectiveness of CNNs in detecting and classifying cassava leaf diseases using image data. The research also aimed to compare the performance of different CNN architectures, optimisers, and loss functions to identify the most effective model configuration for disease detection in cassava leaves.

Methods: A dataset of cassava leaf images, labelled with five classes (four disease classes and one healthy class), was used. Data augmentation techniques were applied to address class imbalance. The images were processed using CNNs with various configurations, including different optimisers (Adam, RMSprop), activation functions (ReLU, softmax), and loss functions (Categorical Cross Entropy, Focal Loss). The models were trained using a stratified 80-20 data split, ensuring a balanced representation of all classes in the training and validation sets. The best-performing models were evaluated based on accuracy, precision, recall, and F1 score.

Results: The CNN model using Categorical Cross Entropy with the Adam optimiser achieved the highest accuracy of 93%, demonstrating superior performance across all evaluation metrics, including precision (0.93), recall (0.94), and F1 score (0.94). Models using RMSprop and Focal Loss showed lower accuracy of 68% with lower metrics as well, highlighting the effectiveness of the Adam optimiser in this context. Despite efforts to address the class imbalance, the model showed a strong bias towards detecting CMD, indicating areas for further improvement in dataset balancing and model refinement.

Conclusions: This study underscores the importance of balanced datasets and appropriate model configurations in improving the accuracy and reliability of disease detection systems. The effectiveness of CNNs in detecting cassava leaf diseases, particularly with the Adam optimiser and Categorical Cross Entropy loss function, is demonstrated. The findings suggest that future work should focus on enhancing the model's ability to assess disease severity and provide treatment recommendations, as well as integrating more advanced machine learning techniques for comprehensive agricultural management solutions.

1. Introduction

Cassava (Manihot esculenta) is a crucial crop for food security and subsistence agriculture in various regions, particularly in East and Central Africa. The productivity of cassava is significantly impacted by several diseases, with the most notable ones being cassava mosaic disease (CMD) and cassava brown streak disease (CBSD) (Tomlinson et al., 2017). These viral diseases pose a severe threat to cassava cultivation, leading to substantial yield losses and food insecurity (Chavarriaga-Aguirre et al., 2016). The clonal propagation method commonly used for cassava planting, where stems are exchanged between farmers, contributes to the rapid spread of diseases such as bacteria, fungi, phytoplasmas, and viruses, including CMD and CBSD (Chavarriaga-Aguirre et al., 2016).

Figure 1: A healthy cassava leaf

In addition to CMD and CBSD, cassava is susceptible to various other diseases such as Cassava Vein Mosaic Disease (CVMD), Cassava Common Mosaic Disease (CCMD), Cassava Witches' Broom Disease (CWBD), Cassava Frog skin Disease (CFSD), and root rot (Carvajal-Yepes et al., 2014; Pardo et al., 2023; Pardo et al., 2022; Pham & Tran, 2021). These diseases affect different parts of the cassava plant, including leaves, stems, and roots, further exacerbating the challenges faced by cassava farmers (Onah, 2022; Maruthi et al., 2019). Moreover, the impact of Cassava Mosaic Disease (CMD) on cassava plants is significant and devastating. CMD causes various symptoms in cassava plants, including chlorosis in leaves, which reduces photosynthetic activity, leading to stunted growth, lowered yields, or even complete crop loss for the season. Studies have shown that CMD can reduce crop yields from 50% to 70% in Zambia, with some recent reports indicating even higher yield losses (Patrick et al., 2022). Furthermore, the economic impact of CMD is substantial, with annual losses in East and Central Africa estimated to be between $1.9 and $2.7 billion. CMD poses a significant threat to cassava production by affecting plant growth, reducing yields, and causing substantial economic losses in affected regions. Overall, cassava diseases threaten food security and have far-reaching environmental implications that can disrupt the delicate balance of agricultural ecosystems and natural resources.

The spread of these diseases is facilitated by factors such as the exchange of infected planting materials, the presence of insect vectors like whiteflies (Bemisia tabaci), and the movement of diseases across regions (Namuddu et al., 2023; Mugerwa et al., 2021). Efforts to combat these diseases include genetic mapping for resistance, generating virus-free planting materials, and implementing optimal control measures (Akano et al., 2002; Maruthi et al., 2019; Onah, 2022). Moreover, studies have highlighted the importance of disease management strategies, such as increasing planting densities to reduce disease incidence and utilising virus-free planting materials to mitigate the spread of diseases (Horowitz et al., 2011; Colvin et al., 2004).

Moreover, research has concentrated on comparing data from literature with field data to contribute to the existing knowledge of cassava disease control and prevention, highlighting the importance of data-driven approaches in disease management (Onah, 2022). Studies have also explored the use of molecular and serological methods for the detection and quantification of specific cassava diseases, showcasing the diverse applications of data science in disease diagnosis (Munganyinka et al., 2018). Optimal control measures, including the introduction of resistant cassava plants and the application of pesticides, have been suggested to prevent and eradicate diseases in cassava production (Onah, 2022). These measures aim to reduce disease spread and maintain optimal production levels while maximising yield and profitability.

Data science has been widely utilised in the field of disease detection in cassava plants. Several studies have emphasised the importance of machine learning algorithms, particularly deep learning techniques, in the early identification and detection of cassava diseases, leading to enhanced crop yields (Abayomi-Alli et al., 2021; Anitha & Saranya, 2022; Ramcharan et al., 2017). These technologies have been specifically applied to detect and measure various cassava diseases such as bacterial blight, brown streak, and mosaic disease (Elliott et al., 2022; Ramcharan et al., 2017). The utilisation of convolutional neural networks (CNNs) and image-based recognition systems has demonstrated promising results in automating the identification of cassava diseases based on leaf images (Ramcharan et al., 2017; Okokpujie et al., 2023; Oyewola et al., 2021; Ramcharan et al., 2019).

Furthermore, the integration of spectral data and machine learning algorithms has enabled the early detection of plant diseases in cassava crops even before visible symptoms appear, underscoring the significance of proactive disease management strategies (Owomugisha et al., 2020). Additionally, the development of web and mobile applications for cassava disease classification using CNNs has facilitated accessible and efficient disease diagnosis (Okokpujie et al., 2023). These advancements in data science have not only improved disease detection accuracy but have also provided cost-effective and rapid solutions for farmers and agricultural extension agents (Mrisho et al., 2020).

Detecting diseases in plants, such as in cassava plants, is crucial for ensuring agricultural productivity and food security. Artificial Intelligence (AI) techniques, particularly machine learning and deep learning, have shown significant promise in automating and enhancing disease detection in plants (Varshney et al., 2021). These technologies offer accurate and efficient methods for early identification of plant diseases, which is essential for timely intervention and effective management (Pranav Ranaware & Wagh, 2023).

Researchers have explored various approaches for plant disease detection, including transfer learning based on deep neural models (Hasan, 2023), deep convolutional neural networks utilising hybrid features (Kusumo et al., 2020), and soft computing-based systems using support vector machines (Sriram et al., 2023). These methods leverage advanced technologies to analyse images of plants and diagnose the presence of diseases accurately. The integration of deep learning techniques with image processing has enabled the development of automated systems for detecting plant diseases (Shoaib et al., 2022). By utilising computer vision and AI, these systems can provide early identification of plant illnesses, thereby preventing the negative consequences of diseases and overcoming the limitations of manual monitoring (Shoaib et al., 2022).

Recent studies have highlighted the importance of deep learning methodologies in addressing the pressing concern of plant diseases (Reddy, 2024). These cutting-edge approaches aim to ensure precise and effective disease identification, contributing to improved plant health and agricultural sustainability. Leveraging AI technologies, for disease detection in cassava plants can revolutionise agricultural practices. By automating the identification of plant diseases, these advanced systems enable early intervention, accurate diagnosis, and tailored treatment, ultimately enhancing crop yield and food security.

1.2 Research Aim, Objectives and Questions

Research Aim: The aim of this research is to develop, evaluate, and compare deep learning models for the accurate detection of diseases in cassava leaves, with a focus on Convolutional Neural Networks (CNN). This study seeks to identify the most effective model for classifying various cassava leaf diseases from images by assessing the performance of each approach in terms of accuracy, efficiency, and generalisability. The ultimate goal is to enhance early disease detection capabilities, thereby supporting improved crop management practices and contributing to increased cassava yield and quality, which are essential for food security in regions where cassava is a staple crop.

1.2.1 Research objective

RO1: To review and synthesise existing literature on the application of machine learning in the classification of diseases on cassava leaves.

RO2: To apply exploratory data analysis techniques to identify images and diseases infecting the leaves and description of the disease.

RO3: To develop and implement CNN model for the detection and classification of cassava leaf diseases using image data.

1.2.2 Research questions

RQ1: How can machine learning models be effectively utilised to detect and classify diseases in cassava leaves based on image data?

RQ2: What impact do the identified diseases have on cassava crop health and yield, and how can early detection improve agricultural outcomes?

RQ3: Which machine learning model of CNN demonstrates the highest accuracy and effectiveness in detecting and classifying diseases in cassava leaves?

1.3 Significance of the study

The significance of this study lies in its potential to enhance the early detection and management of cassava diseases through the application of advanced machine learning techniques. By leveraging models like CNNs this research aims to develop accurate, efficient, and scalable tools for the automated identification of cassava diseases. These tools can lead to improved crop management practices, reducing yield losses and supporting food security in cassava-growing regions (Ramcharan et al., 2017; Abayomi-Alli et al., 2021). The integration of these AI-driven approaches into agricultural practices can mitigate the negative impacts of diseases like CMD and CBSD, which have been responsible for significant economic losses in regions reliant on cassava production (Tomlinson et al., 2017; Patrick et al., 2022). The findings from this research will contribute to the broader field of precision agriculture, offering valuable insights into the practical application of AI technologies in disease detection and agricultural sustainability (Pranav Ranaware & Wagh, 2023; Varshney et al., 2021).

In conclusion, the application of data science, particularly deep learning techniques, has significantly advanced disease detection in cassava plants. By leveraging advanced algorithms and image recognition systems, researchers have made substantial progress in automating disease identification processes, enabling early intervention and improved agricultural outcomes.

2. Literature Review

Machine learning techniques are becoming increasingly popular for identifying plant diseases, and existing literature showcases various approaches to leaf disease detection using deep learning and machine learning algorithms. In the early days of artificial intelligence, traditional machine learning methods such as Support Vector Machines (SVM), k-Nearest Neighbours (k-NN), and Decision Trees were extensively used due to their simplicity and effectiveness with small datasets. However, with the advent of deep learning, there has been a significant shift towards Convolutional Neural Networks (CNNs) and transfer learning, which have demonstrated superior performance in image classification tasks. This section critically assesses the literature available on these methodologies, exploring the transition from traditional techniques to modern deep learning models, the adoption of hybrid approaches and ensemble methods, as well as advancements in data augmentation and synthetic data generation to enhance plant disease detection.

Many research studies used traditional machine learning techniques in the early days of AI. Traditional machine learning techniques have been widely utilised for disease detection in cassava plants using images. Studies have shown that approaches such as image recognition combined with deep learning have been successful in classifying cassava leaf diseases (Ravi et al., 2021). These methods leverage machine learning algorithms to enhance the early identification and detection of cassava diseases, ultimately improving crop yields (Abayomi-Alli et al., 2021). Furthermore, the use of deep learning techniques in the identification and detection of cassava leaf diseases has been explored, emphasising the significance of these methods in plant health management (Anitha & Saranya, 2022). While deep learning has shown superior performance compared to traditional machine learning algorithms in plant disease detection, visible range images are commonly preferred over spectral images for this purpose (Bhagwat & Dandawate, 2021). Various machine learning strategies, including random forest, support vector machines, convolutional neural networks, and deep learning, have been employed to accurately detect and classify plant diseases, including those affecting cassava plants (Chimate, 2023).

2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are extended deep neural networks that have gained significant prominence in various fields due to their exceptional capabilities in image analysis and pattern recognition (Hoeser et al., 2020). CNNs have been extensively utilised in tasks such as image recognition, object detection, and image segmentation (Hoeser & Kuenzer, 2020). In the medical field, CNNs have been particularly valuable, with applications in medical imaging for tasks like cholangiocarcinoma diagnosis (Njei et al., 2023). The effectiveness of CNNs lies in their ability to extract intricate features from images, making them highly suitable for tasks requiring detailed analysis (Vidyasagar, 2024).

CNNs are essential in image data processing due to their exceptional capabilities in recognising, classifying, and extracting features from images. They have transformed the field of image processing by automatically learning abstractions from raw data, enabling precise image classification (Eftekhari et al., 2019). CNNs have been extensively utilised in various applications such as medical image analysis, disease classification, and detection tasks (Momeni et al., 2018; Garifulla et al., 2021). CNNs have demonstrated remarkable success in tasks like image classification, object detection, and segmentation in medical image analysis (Byra et al., 2020). Their proficiency in extracting complex patterns from images makes them suitable for tasks like classifying medical images, including high-resolution CT scans (Sreeja, 2024).

CNNs have shown superiority over traditional algorithms in various domains, especially in image interpretation applications like radiology (Chatterjee, 2019). Their success can be attributed to their capacity to learn complex representations from data, enabling them to excel in tasks that involve image-related applications (Shamshad et al., 2022). In neuroradiology, CNNs have been widely adopted for radiomics and deep learning tasks, showcasing their effectiveness in pattern recognition within imaging data (Park et al., 2020; Wagner et al., 2021).

Furthermore, CNNs have been employed in real-time image processing applications, underscoring their importance in addressing issues related to real-time image and video processing (Zhang & Yifang, 2010). Their capability to handle multiple labels in image classification tasks has been addressed through frameworks like CNN-RNN (Recurrent Neural Networks), providing a unified approach for multi-label image classification (Wang et al., 2016). Moreover, using CNNs in conjunction with transfer learning has optimised image processing tasks by enhancing feature extraction and improving image classification accuracy (Essa, 2023).

Plant disease classification using Convolutional Neural Networks (CNNs) has become a significant area of research, with various studies demonstrating the effectiveness of CNN models in accurately identifying and categorising different plant diseases. (Lu,2023) explored the application of popular CNN models such as EfficientNetB5, MobileNet, ResNet50, InceptionV3, and VGG16 for plant disease classification, highlighting the versatility and robustness of CNNs in this domain. Similarly, Lee et al. (2020) emphasised the promising outcomes achieved in plant disease classification through CNNs, underscoring the pivotal role of deep learning techniques in this field.

The use of CNNs for plant disease classification is further supported by Kamble (2024), who employed state-of-the-art CNNs on a large dataset of annotated leaf images to accurately detect various plant leaf diseases, including fungal, bacterial, and viral infections. Toda et al. (2019) also demonstrated the success of CNNs in diagnosing plant diseases, indicating significant advancements in leveraging deep learning for plant disease classification tasks.

A study by Burak Gülmez (2024) specifically applied multiple CNN algorithms such as VGG, ResNet, YOLOv3, RestNETV2 101, YOLOv5, and many more to classify diseases on rice plant leaves with an accuracy of over 98%, showcasing the adaptability of CNN architectures across different plant species. Guo (2023) provided an in-depth analysis of plant disease identification using deep learning methods, emphasising the remarkable performance of CNNs in accurately classifying plant diseases.

(Nerkar & Talbar, 2020) Proposed a hybrid CNN architecture to enhance the accuracy of plant leaf disease classification, showcasing continuous efforts to improve the efficiency and precision of CNN-based disease classification systems. (Panigrahi et al., 2020) Emphasised the superior accuracy of CNNs in classifying corn plant diseases based on leaf images, showcasing the robustness of CNNs in automatically extracting features for precise disease classification.

Deep learning methods, particularly those based on Convolutional Neural Networks (CNN), have gained significant traction in plant disease classification, including identifying and diagnosing diseases in cassava plants. Several studies have demonstrated the successful application of CNN models for classifying various diseases affecting cassava plants (Ali et al., 2021; Lu et al., 2021). These studies have leveraged transfer learning techniques, where pre-trained CNN models are fine-tuned to classify cassava diseases based on image datasets (Ramcharan et al., 2017; Anitha & Saranya, 2022). Transfer learning has been particularly effective in improving the accuracy of disease classification models (Liu et al., 2022).

Researchers have explored different approaches to enhance the performance of CNN models for cassava disease classification. For instance, integrating attention mechanisms and multi-scale fusion models has shown promising results in improving the recognition rates of cassava leaf diseases (Liu et al., 2022). Additionally, developing web and mobile applications based on CNN models has facilitated the creation of user-friendly interfaces for cassava disease classification (Okokpujie et al., 2023). These advancements have improved the accuracy of disease classification and enhanced the accessibility of diagnostic tools for farmers and researchers.

Moreover, studies have highlighted the importance of data augmentation and model optimisation in enhancing the robustness and efficiency of CNN-based disease classification systems for cassava plants (Abayomi-Alli et al., 2021; Oyewola et al., 2021). Researchers have detected cassava mosaic disease from low-quality images with a high accuracy of over 80% by employing deep residual CNN architectures and block processing techniques. Furthermore, lightweight transfer learning-based architectures have enabled the accurate detection of leaf diseases across multiple plants, including cassava, even with limited image data (Mazumder, 2024).

The effectiveness of CNN models in cassava disease classification has been further validated through comparative analyses of different CNN architectures. These studies have aimed to identify lower-capacity models suitable for deployment on mobile devices, ensuring that disease classification tools are accessible and practical for on-field applications. Additionally, the fusion of CNNs with other deep learning techniques, such as Generative Adversarial Networks (GANs), has shown promise in early prediction and improved classification of plant diseases, including those affecting cassava plants (Gomaa & El-Latif, 2021).

Several recent studies have provided empirical evidence supporting the effectiveness of CNNs in classifying cassava leaf diseases. (Pai et al., 2023) demonstrated high accuracy in classifying cassava leaf diseases using a modified CNN architecture with parallel pathways, multiple convolutions, and pooling. Their model achieved a training accuracy of 98.54% and a testing accuracy of 98.08%. (Mehta et al., 2023) explored a federated CNN model for cassava leaf disease classification, achieving precision values ranging from 0.74 to 0.85, recall values from 0.70 to 0.88, and an overall accuracy of 78.6%. It also examined a privacy-protecting federated learning CNN model for cassava leaf disease severity assessment. This model achieved an overall accuracy of 95% and an F1-score of 0.91875, highlighting high precision and recall for healthy and severe disease classes. (Sapre et al.,2023) utilised an Artificial Neural Network (ANN) model for cassava plant leaf disease classification into five classes, achieving a training data accuracy of 61.45% and a test data accuracy of 61.88%. (Guo, 2023) reported that a trained CNN model achieved 92.23% accuracy in disease classification, showcasing the potential of CNN technology in agricultural innovation.

These studies collectively underscore the efficacy of CNNs in cassava disease classification, illustrating various approaches and enhancements that have been successfully implemented to improve accuracy and robustness in disease detection.

2.2 Ethics and research gaps

Ethical considerations are crucial in applying machine learning and AI for disease detection in agriculture. Addressing bias, transparency, and privacy issues is essential to ensure fair and responsible technology use (Sreerama, 2022). Studies highlight the significance of detecting and mitigating bias for equitable outcomes and stress the importance of transparent decision-making processes to build stakeholder trust (Dara et al., 2022). Robust data privacy measures are vital to safeguard sensitive information and maintain the integrity of AI applications (Sreerama, 2022). Research has delved into ethical implications, including developing frameworks for transparent model evaluation, implementing bias mitigation techniques, and ensuring robust data privacy and security (Dara et al., 2022). By analysing extensive datasets with AI algorithms, actionable insights for early intervention and management in agriculture can be provided, empowering farmers with valuable information (Sahoo, 2024). AI algorithms can analyse satellite imagery and other data sources to detect anomalies, stress, and disease outbreaks in crops, aiding in optimising irrigation and nutrient management (Kumar, 2023).

Moreover, integrating AI and machine learning in agriculture underscores the need to address bias and fairness in machine learning models, emphasising the importance of ethical considerations in AI applications (Huang, 2024). Techniques such as fairness-aware machine learning, bias detection, and algorithmic auditing are crucial in identifying and mitigating biases in AI systems (Markan, 2024). Additionally, AI in agriculture focuses on identifying damaged crops and enhancing the yield of healthy crops, contributing positively to the industry (Das, 2024).

Despite the significant advancements in using machine learning and deep learning techniques for cassava plant disease detection, several research gaps need to be addressed to enhance the efficacy and applicability of these models in real-world scenarios. One major issue identified in the literature is the handling of unbalanced datasets. Many studies have noted the need for improved model accuracy and efficiency when dealing with imbalanced data, suggesting further fine-tuning and optimisation for better classification performance (Alford, 2024; Gopi, 2024; Yadav, 2022). This challenge is particularly pertinent given that unbalanced datasets can skew the model's predictions, leading to less reliable outcomes.

Another notable gap is the limited exploration of disease management beyond detection. While numerous studies have focused on detecting diseases, more work is needed to predict the recovery of affected plants and develop comprehensive disease management strategies (Omaye, 2024). This gap highlights the need for more integrated approaches that detect diseases and provide actionable insights for managing and mitigating their impact.

Moreover, the efficiency of transfer learning in disease detection has shown promise, yet research indicates gaps in automatic disease detection for cassava plants. Limited studies have focused on the effectiveness of transfer learning and the need for more research on this technique to enhance disease detection (Rajasree, 2023; "Hyperparameter Optimization for Transfer Learning-Based Disease Detection in Cassava Plants", 2023). The absence of comprehensive model performance evaluations and comparisons with other models further underscores the need for more rigorous and comparative studies (John, 2022).

In addition, the scalability and generalisation of disease detection models are critical areas that require more attention. Studies often need more discussions on the scalability of their models and how they can be generalised across different environments and cassava varieties (De Bock, 2022). This limitation hinders the broader application of these models in diverse agricultural settings.

Furthermore, many researchers have emphasised the challenges related to the manual identification of diseases, which is time-consuming and costly. The lack of timely disease recognition affects cassava plant yields significantly (Dharani, 2022). Addressing this challenge with automated systems could enhance the efficiency and productivity of cassava farming.

Lastly, the ethical considerations of AI and machine learning in agriculture are not extensively discussed. Issues such as bias, transparency, and privacy are crucial for the responsible deployment of these technologies. Addressing these ethical challenges is necessary to ensure that the benefits of AI and machine learning are equitably distributed and that the technologies are used responsibly (Omaye, 2024).

Addressing ethics and research gaps is crucial for effective AI in agriculture. Fairness, transparency, and privacy build trust and promote equitable technology use. Critical research gaps include improving model performance with unbalanced datasets, enhancing disease management strategies, transfer learning efficiency and model scalability. Automated systems for timely disease recognition can significantly boost cassava farming efficiency. Integrating ethical considerations with targeted research will lead to robust, accurate, and fair solutions for cassava disease detection, ultimately enhancing crop health, productivity, and food security.

This section aligns with the research objectives outlined in section 1.2, specifically objective one (RO1), which focuses on reviewing and synthesising existing literature on the application of machine learning in classifying diseases on cassava leaves (see section 1.2.1).

3. Methodology

3.1 Libraries used

1. **Jupyter Notebook:** It is an open-source web application that allows for the creation and sharing of documents containing live code, equations, visualizations, and narrative text. It is widely used in data science for interactive data analysis and visualization.
2. **Keras:** It is a high-level neural networks API written in Python that runs on top of TensorFlow. It enables fast experimentation and development of deep learning models by providing an intuitive and user-friendly interface.
3. **MLflow:** It is an open-source platform designed to manage the machine learning lifecycle. It facilitates experiment tracking, model versioning, and deployment, providing a seamless way to manage and reproduce machine learning experiments.
4. **OpenCV:** Open-Source Computer Vision Library is an open-source computer vision and machine learning software library. It contains numerous functions for image processing, computer vision tasks, and real-time operations, making it widely used for image and video analysis.
5. **Pandas:** It is a powerful data analysis and manipulation library for Python that provides data structures like Data Frames to efficiently handle, analyse, and process structured data. It supports various functions for data cleaning, transformation, and aggregation, making it essential for data-driven research and exploration.
6. **Python**: It is a high-level programming language known for its readability and versatility. It is widely used in data science, machine learning, and scientific computing due to its extensive libraries and community support.
7. **Scikit-learn:** It is a machine learning library that implements various machine learning algorithms and data pre-processing techniques that can be leveraged for effective data splitting and evaluation of models based on regression and classification.
8. **TensorFlow:** It is an open-source machine learning framework developed by Google. It provides a comprehensive ecosystem for building and deploying machine learning models, supporting both deep learning and traditional machine learning algorithms.
9. **Visual Studio Code:** It is a widely-used, open-source code editor developed by Microsoft that provides a versatile and customizable environment for programming. It supports numerous programming languages, debugging tools, and extensions, making it ideal for developing, testing, and maintaining code in data science and software development projects.

Download links for all these are at appendix 8.5.

The methodology for disease detection in cassava leaves involves a comparative analysis of Convolutional Neural Networks (CNNs) with different parameters. The goal is to evaluate the performance of each model in accurately identifying and classifying diseases affecting cassava plants. The process begins with collecting and preparing a robust dataset of labelled images of cassava leaves, each representing a specific disease. These images undergo standard preprocessing steps, including resizing, normalisation, and data augmentation, to ensure consistency and enhance the model's generalisation ability across diverse data samples.

Once the dataset is prepared, model of CNN trained and optimized with different parameters. CNNs are utilized for their strength in handling image data, allowing the model to learn complex features from the images automatically using kernels. After training, each model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. The results from these evaluations are then compared to determine which model offers the best performance for cassava leaf disease detection (Hong et al., 2023). This comparative approach ensures that the most effective model is identified, providing valuable insights into the suitability of each technique for this specific application.

3.2 Workflow

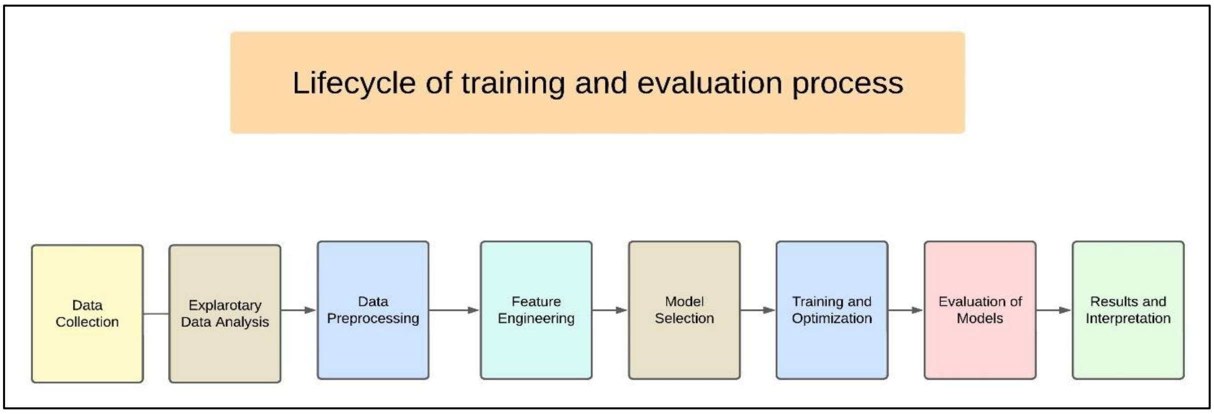

Lifecycle of training and evaluation process
Data Collection
Explarotary Data Analysis
Data Preprocessing
Feature Engineering
Model Selection
Training and Optimization
Evaluation of Models
Results and Interpretation

Figure 2: machine learning workflow

Visual Studio Code, an extensive and feature-packed Integrated Development Environment that helps to properly structure directories and provides an impressive development environment for programmatical computations, was leveraged to carry out the whole machine learning process.

1. Firstly, the dataset is downloaded from Kaggle, and the pre-processing is done to make it more accessible and easier to load, as explained in section 3.5.
2. It is then split into 80-20 ratio using a stratified sampling technique to ensure imbalanced class distribution as explained in section 3.5.1
3. The dataset is then loaded into the Keras library and converted into tensors; TensorFlow internally handles the image data to convert it into tensors, as explained in section 3.5.2
4. The training data is then pre-processed by normalising it and using data augmentation techniques such as rotation, flipping, and zooming to increase the diversity of the training data and improve model generalisation, as explained in sections 3.5.3 and 3.5.4.
5. The Keras library, which provides a high-level API for Deep Learning is used to program the CNN architecture using layers such as Conv2D, MaxPooling, Dense, Flatten, Dropout, and Normalisation, as explained in section 3.6.
6. The CNN model was compiled using optimisers such as Adam and RMSProp. The loss functions used for this study are binary cross-entropy and focal loss, which are explained in section 3.7.
7. The model was then trained using the GPU for accelerated training and Nvidia CUDA to leverage multicore parallelism.
8. All the parameters were logged using the MLflow library, which helped in experimentation tracking, artefact logging, and metrics logging. This allowed for easy comparisons for multiple runs, which reduced manual work significantly.
9. The model created was then tested on the test split to check the model's performance using metrics such as accuracy, F1 score, recall and precision, as explained in section 3.8.
10. An iterative process of hyperparameter tuning, including multiple epochs, different batch sizes, and different optimisers, was used to tune and optimise the CNN model, all of which was logged using the MLflow library.
11. Generated and visualised the confusion matrix to assess model performance across different classes. Analysed these metrics to understand the model's strengths and weaknesses and documented the findings; a detailed discussion is in section 4.

All the models are trained using an RTX 3060 Laptop GPU.

3.3 Data collection

Kaggle was utilised as a valuable resource for the project's data collection phase to acquire a comprehensive dataset. The dataset, "Cassava Leaf Disease Classification," was sourced from Kaggle's repository, a well-established platform for data science and machine learning competitions. The keywords for searching the dataset were “Cassava Leaves Dataset” and “CNN Leaves Dataset”. This dataset includes over 20,000 images of cassava leaves with a shape of 800x600 pixels in RGB format, each labelled with one of the five possible disease classes or a healthy class (refer to appendix 8.2 for dataset link). This dataset was selected due to its extensive collection of images and multiple diseases in the plant. Moreover, its availability in the public domain made it a great choice for this research. The Exploratory Data Analysis of the data is further explained in section 3.4.

3.2.1 Distribution of the classes

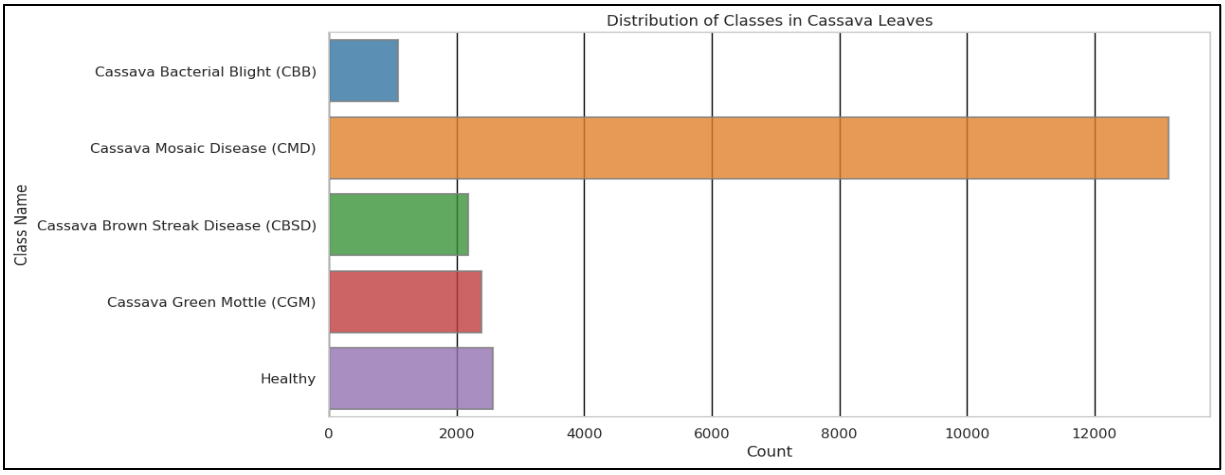


Figure 3: Distribution of classes in the dataset

Figure 3 shows that the distribution of classes within the cassava leaf dataset highlights a significant imbalance, with Cassava Mosaic Disease (CMD) being vastly overrepresented compared to other classes like Cassava Bacterial Blight (CBB), which is underrepresented. This imbalance presents challenges for training machine learning models, as the model may become biased toward the CMD class, leading to poor performance in detecting less frequent diseases like CBB. Such imbalances can result in high false negatives for minority classes, which is problematic for accurate disease detection in real-world applications. To mitigate these issues, data augmentation, oversampling, or class weighting may be necessary to ensure balanced model performance across all classes refer section 3.5.4.

3.4 Exploratory Data Analysis

The dataset contains over 20,000 images of the leaves with five types, which consist of the categories as follows:

3.4.1 Healthy

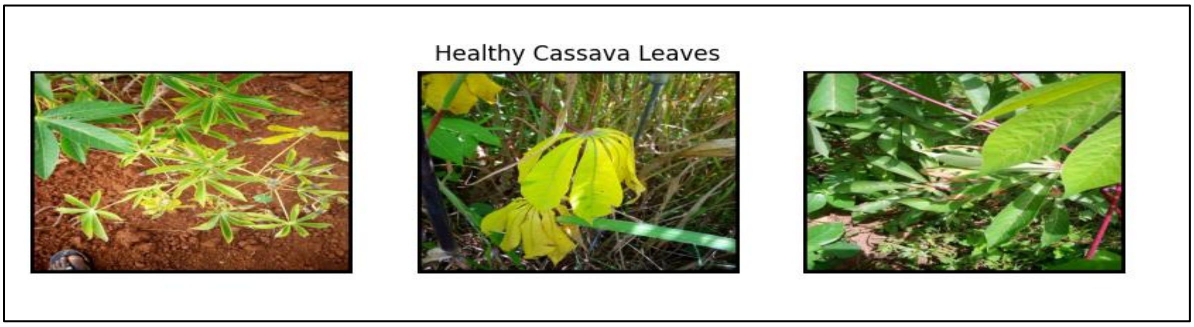


Figure 4: Healthy cassava leaves sample

Figure 4 depicts a healthy cassava leaf characterized by its vibrant green colour and uniform pigmentation across the surface. The leaf displays the typical structure of a cassava plant, with multiple elongated lobes radiating symmetrically from a central point, resembling the shape of a palm. Each lobe is smooth and free from any spots, discolouration, or lesions, indicating that the plant is healthy. The leaf's overall appearance, with its lush and unblemished texture, suggests that the cassava plant is thriving without signs of common diseases like Cassava Mosaic Disease (CMD) or Cassava Brown Streak Disease (CBSD).

3.4.2 Cassava Bacterial Blight

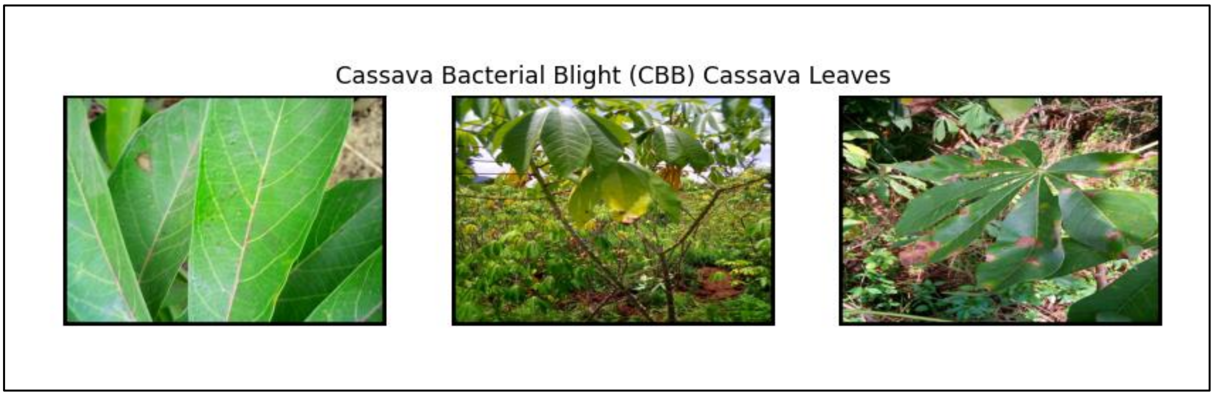


Figure 5: Cassava Bacterial Blight infected leaves sample

Figure 5 depicts cassava leaves exhibiting symptoms of Cassava Bacterial Blight (CBB), a significant bacterial disease affecting cassava plants. The leaves show characteristic signs of infection, including yellowing (chlorosis) and brown to dark brown necrotic spots. These spots are often irregular in shape and are typically surrounded by a yellow halo, indicative of the bacterial infection spreading through the leaf tissue. The affected areas may also display a wilted appearance, particularly around the margins of the necrotic spots.

3.4.3 Cassava Brown Streak Disease (CBSD)

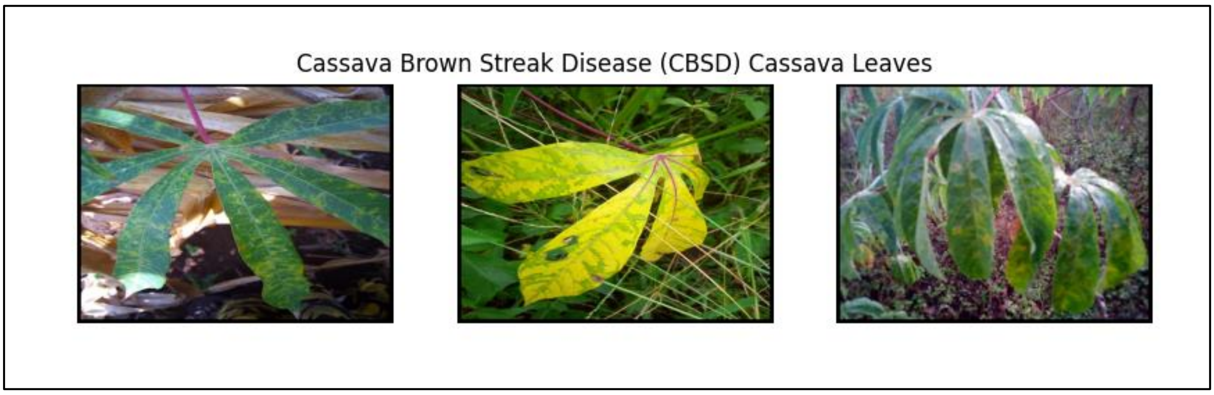


Figure 6: Cassava Brown Streak Disease (CBSD) infected leaves sample

Figure 6 depicts cassava leaves displaying symptoms characteristic of Cassava Brown Streak Disease (CBSD), a viral disease severely impacting cassava crops. The leaves exhibit a distinct yellow chlorotic pattern, with mottling and vein banding that are typical of CBSD infection. The yellowing is irregular and dispersed, often concentrated around the secondary veins, while the primary veins may remain green, creating a striking contrast.

3.4.4 Cassava Green Mottle (CGM)

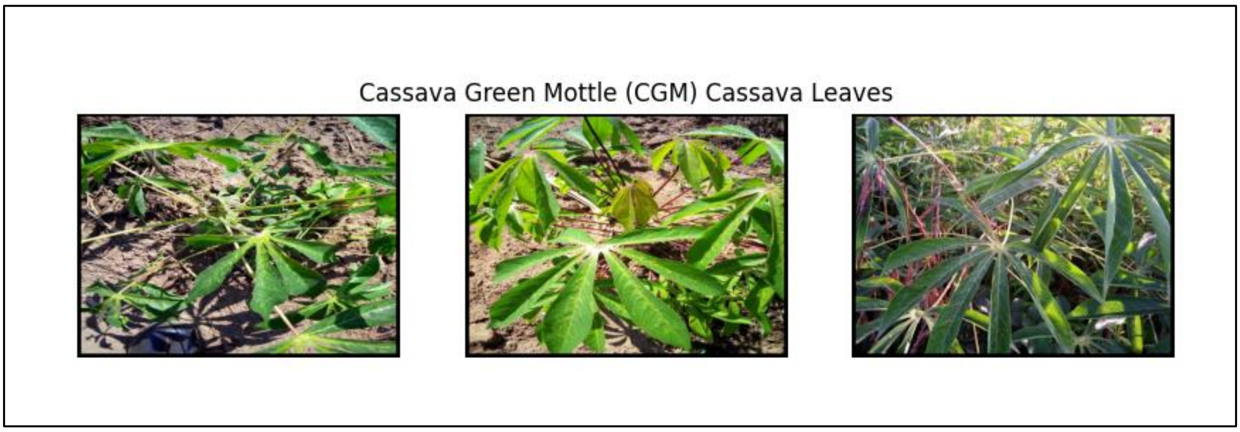


Figure 7: Cassava Green Mottle (CGM) infected leaves sample

Figure 7 showcases cassava leaves affected by the Cassava Green Mottling (CGM), which results in a distinct mottling pattern. The leaves exhibit a combination of dark green and light green patches, giving them a mottled appearance. This irregular discolouration is a hallmark of CGM, which typically manifests as scattered patches of healthy green tissue interspersed with lighter, chlorotic areas. The condition can affect the plant's overall vigour, leading to reduced photosynthetic efficiency and potentially impacting crop yield.

3.4.5 Cassava Mosaic Disease (CMD)

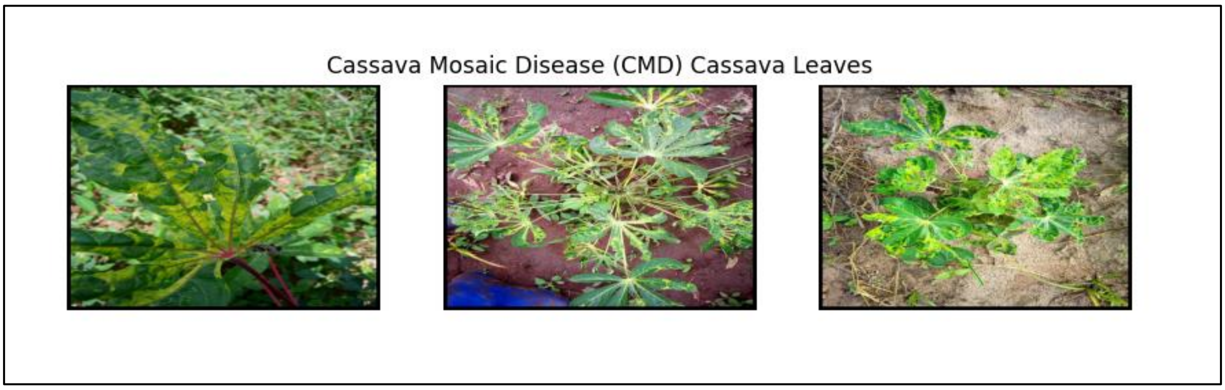


Figure 8: Cassava Mosaic Disease (CMD) infected leaves sample

Figure 8 depicts cassava leaves affected by Cassava Mosaic Disease (CMD), one of the most devastating viral diseases impacting cassava crops. The leaves in the images display the characteristic symptoms of CMD, including a pronounced mosaic pattern with a mix of light green, yellow, and dark green areas. This mottling effect is caused by the virus disrupting the normal chlorophyll production in the leaf tissues, leading to irregular patches of discolouration. Additionally, the leaves may show signs of distortion, reduced size, and a crinkled appearance, all typical indicators of CMD.

These insights align with the research objective outlined in section 1.2, specifically objective two (RO2). The objective aims to apply exploratory data analysis techniques to identify images and diseases infecting the leaves and describe the diseases (see section 1.2.1).

3.5 Data Preprocessing and Feature Engineering

For the data preprocessing phase, a custom script was developed to organise the dataset into a more structured format suitable for machine learning tasks. The original dataset from Kaggle consisted of images stored in a single folder, with their corresponding disease labels provided in a CSV file. The script systematically processed each image using the pandas library based on its label to streamline the training process and ensure that the data was appropriately organised.

The script first reads the CSV file, which contains the mapping between each image file and its associated disease label. Using the mapping, the script then categorises the images into their respective disease classes. A separate folder is created for each disease class. The script subsequently moves each image into the corresponding folder based on its label, resulting in a directory structure where each disease class has its folder containing all relevant images. Creating a distinct directory for each disease prepares the data for efficient loading, augmentation, and training for the CNN model, allowing the use of Keras and TensorFlow inbuilt functions.

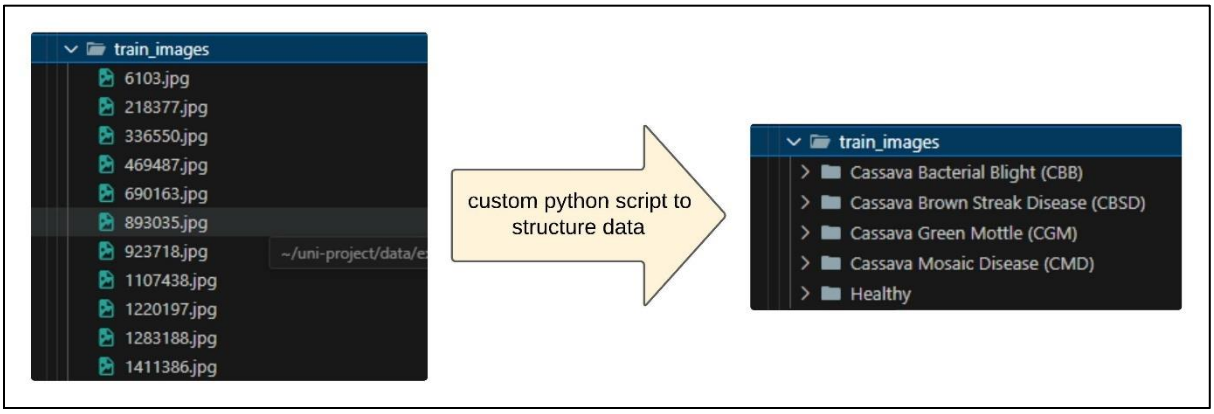


Figure 9: Data pre-process script output

3.5.1 Train test split

The Cassava leaf dataset was split into training and testing subsets using an 80-20 ratio with stratified sampling. Stratified sampling ensures that each class is proportionally represented in both the training and testing sets, preserving the original class distribution (Kohavi, 1995). Train data contains 20,897 images and testing data contains 4179 images, see figure 10 and 11 for distribution of split data.

3.5.1.1 Benefits of stratified sampling

- **Maintains class distribution:** This method keeps the same proportion of each class in both the training and test sets, ensuring balanced representation across all classes (He & Garcia, 2009).
- **Reduces bias:** Stratified sampling minimizes bias towards more frequent classes, providing a fair evaluation of the model's performance (Japkowicz & Stephen, 2002).
- **Improves model evaluation:** By ensuring all classes are represented, stratified sampling offers a more reliable assessment of the model's ability to generalize to new data (Sokolova & Lapalme, 2009).

Stratified sampling is essential for the cassava dataset to avoid over-representation of certain classes and under-representation of others, ensuring that the model learns effectively from all types of disease classes and can accurately evaluate its performance (He & Garcia, 2009).

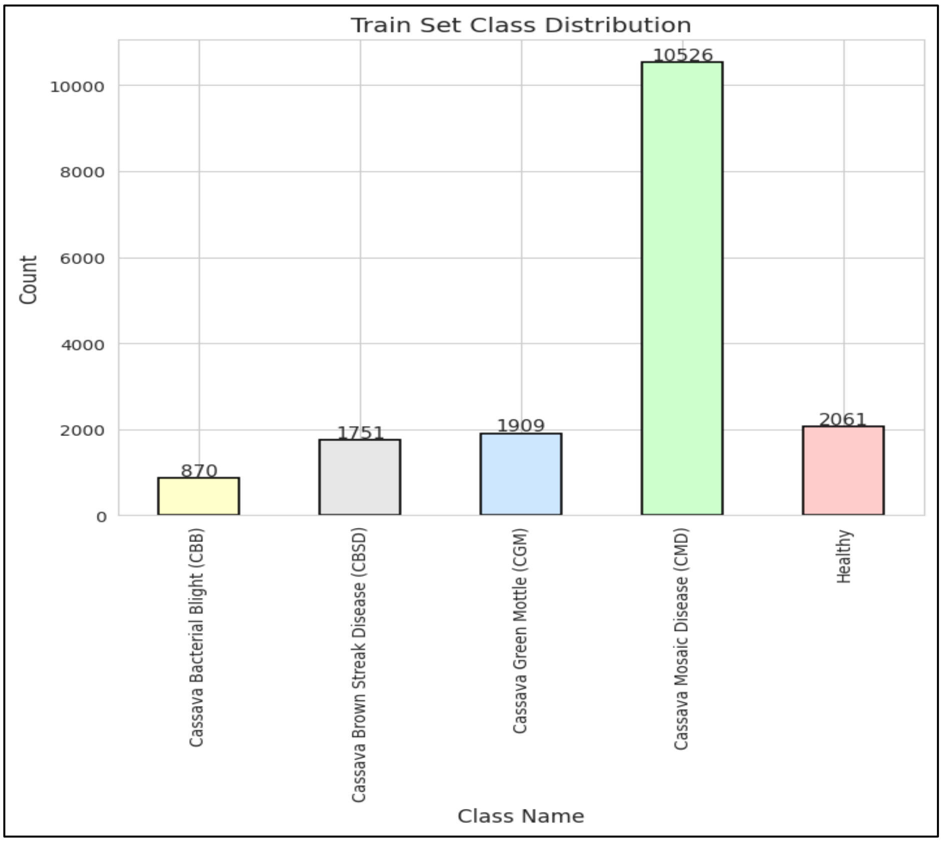


Figure 10: Distribution of training data

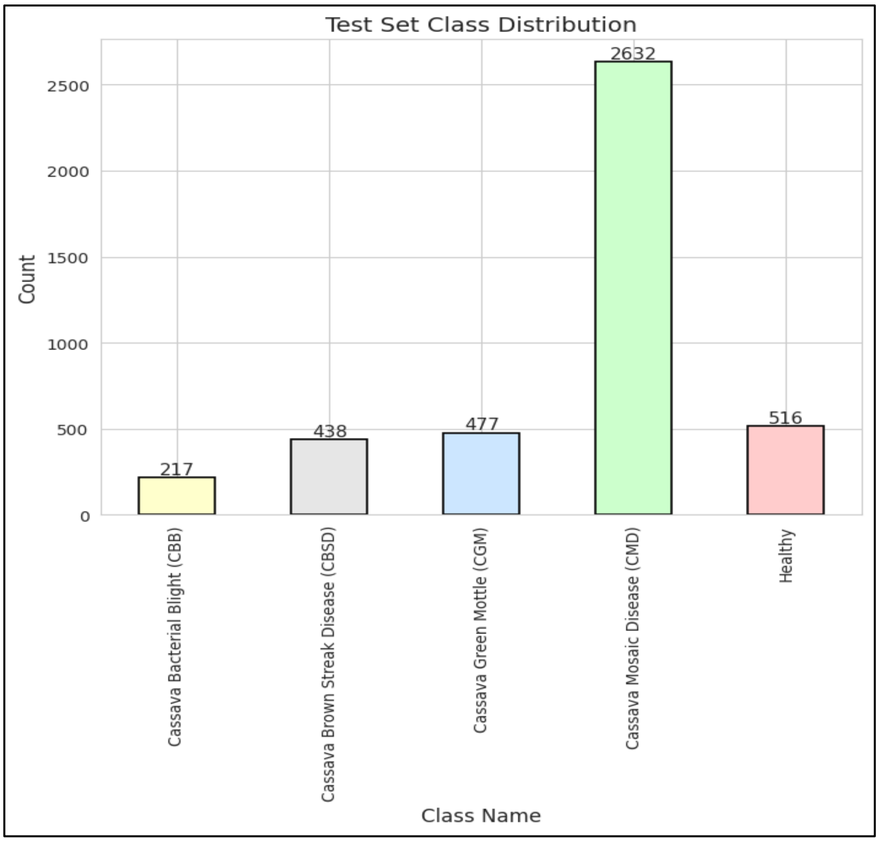
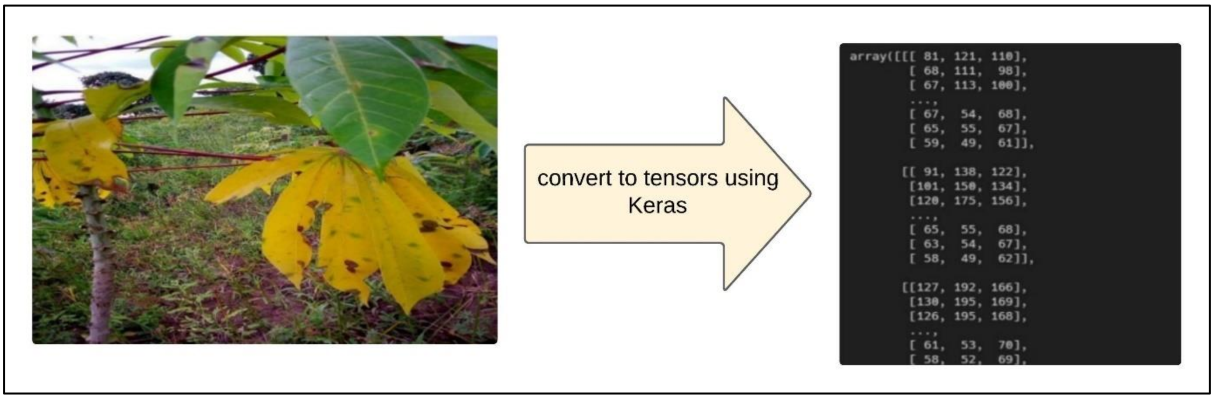


Figure 11: Distribution of test data

3.5.2 Loading of images

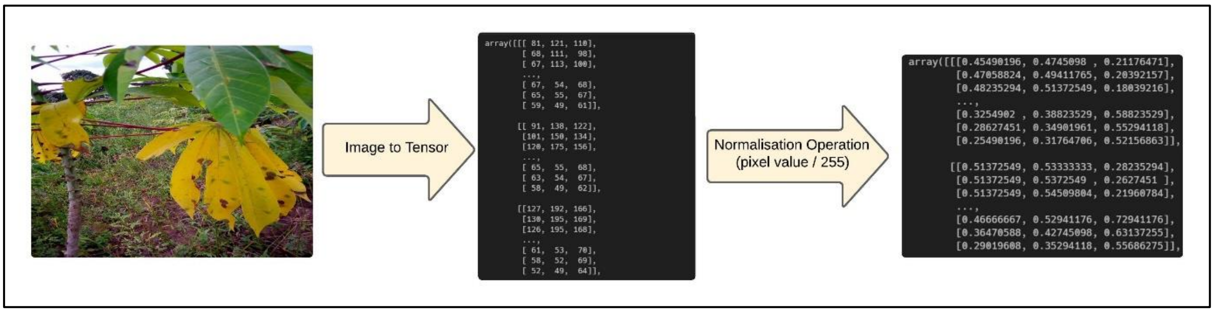


```
array([[[ 81, 121, 110],
        [ 68, 111,  98],
        [ 67, 113, 100],
        ...,
        [ 67,  54,  68],
        [ 65,  55,  67],
        [ 59,  49,  61]],

       [[ 91, 138, 122],
        [101, 150, 134],
        [120, 175, 156],
        ...,
        [ 65,  55,  68],
        [ 63,  54,  67],
        [ 58,  49,  62]],

       [[127, 192, 166],
        [130, 195, 169],
        [126, 195, 168],
        ...,
        [ 61,  53,  70],
        [ 58,  52,  69],
```

Figure 12: Conversion from image to tensors

To load the images into a data frame for training, the Keras, TensorFlow, and OpenCV libraries were used to convert the images into tensors for the machine learning algorithm to train on. As in Figure 12, all the images are converted into a tensor dataset. The number represents the intensity of the RGB (Red, Green, Blue) channel for each pixel in an 800x600-pixel image.

3.5.3 Normalisation of images



```
array([[[ 81, 121, 110],
        [ 68, 111,  98],
        [ 67, 113, 100],
        ...,
        [ 67,  54,  68],
        [ 65,  55,  67],
        [ 59,  49,  61]],

       [[ 91, 138, 122],
        [101, 150, 134],
        [120, 175, 156],
        ...,
        [ 65,  55,  68],
        [ 63,  54,  67],
        [ 58,  49,  62]],

       [[127, 192, 166],
        [130, 195, 169],
        [126, 195, 168],
        ...,
        [ 61,  53,  70],
        [ 58,  52,  69],
        [ 52,  49,  64]],
```

```
array([[[0.45490196, 0.4745098 , 0.21176471],
        [0.47058824, 0.49411765, 0.20392157],
        [0.48235294, 0.51372549, 0.18039216],
        ...,
        [0.3254902 , 0.38823529, 0.58823529],
        [0.28627451, 0.34901961, 0.55294118],
        [0.25490196, 0.31764706, 0.52156863]],

       [[0.51372549, 0.53333333, 0.28235294],
        [0.51372549, 0.5372549 , 0.2627451 ],
        [0.51372549, 0.54509804, 0.21960784],
        ...,
        [0.46666667, 0.52941176, 0.72941176],
        [0.36470588, 0.42745098, 0.63137255],
        [0.29019608, 0.35294118, 0.55686275]],
```

Figure 13: Normalising tensors for efficient training

For several reasons, the normalisation of images is a crucial preprocessing step in developing Convolutional Neural Networks (CNNs), as shown in Figure 13. Primarily, normalisation helps to standardise the input data, which can significantly improve the training efficiency and overall performance of the model. (LeCun et al. (1998). Moreover, it also helps accelerate the training and prevent internal covariate shifts (Ioffe & Szegedy, 2015). For the efficient training of a CNN model, every image is normalised before training the model by dividing each pixel value by 255, as RGB values range from 0-255.

3.5.4 Data augmentation

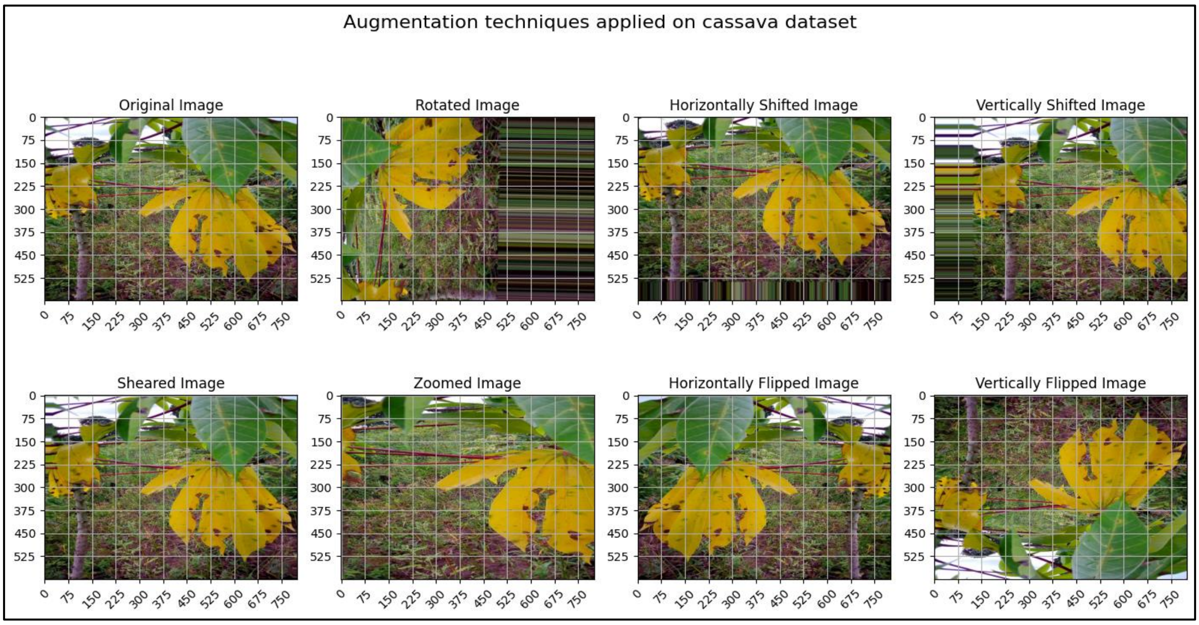


Figure 14: Various data augmentation techniques

Data augmentation is a powerful technique used in deep learning to artificially expand a dataset's size and variability by applying various transformations to the training data. These transformations include rotations, translations, zooming, flipping, and shearing, as shown in Figure 14. Data augmentation's primary goal is to enhance models' generalisation ability by exposing them to a broader range of scenarios, thereby reducing overfitting to the training data (Shorten & Khoshgoftaar, 2019). By increasing the diversity of training examples, data augmentation helps models handle better input data variations, such as changes in orientation, scale, or lighting conditions, which are common in real-world applications.

3.5.6 Image resizing

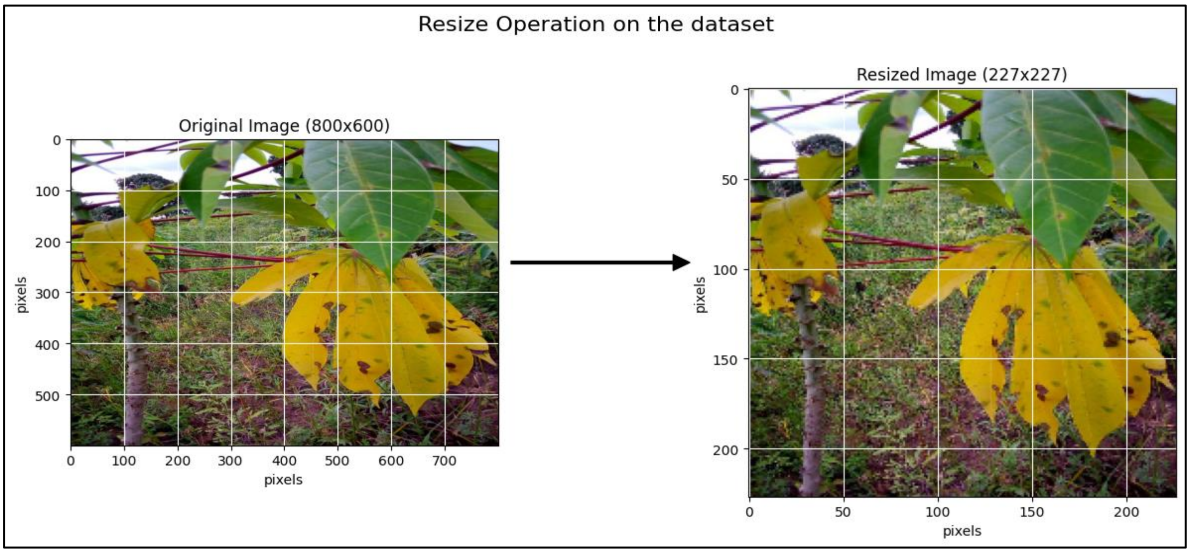


Figure 15: Image resizing operation

Every image is resized to 227x227 pixels using OpenCV and Keras Figure 15. Resizing images before training a CNN model is a critical preprocessing step for several purposes. First, CNNs typically require input images to be of a consistent size to ensure that the network architecture, particularly the fully connected layers, can be applied uniformly across all images. Inconsistent image sizes would necessitate varying network architectures or lead to errors during the forward pass (Krizhevsky et al., 2012). It also helps to reduce computational complexity. High-resolution images contain more pixels and, consequently, more data, which can significantly increase the computational load and memory requirements during training. By resizing images to a smaller, fixed size, it is possible to balance maintaining essential features and reducing the data's dimensionality, accelerating the training process and utilising less computation power without significantly compromising model performance (Simonyan & Zisserman, 2014).

3.6 Model selection and training

The Convolution Neural Network consists of numerous parameters and layers to design the architecture and classify the image data. These layers are organised into multiple learning stages, incorporating convolutional layers, non-linear processing units, and subsampling layers (Khan et al., 2020). The architecture is composed of repeatedly stacked feature stages. Each stage contains a convolution module, a pooling/subsampling module, a normalisation module, and a dropout layer to prevent overfitting (Nitish Srivastava et al., 2014). After these feature extraction stages, the output is passed through a flattened layer to convert the multidimensional feature maps into a one-dimensional vector. This vector is then fed into one or more dense layers for further processing and feature interpretation. Finally, the output layer, a fully connected module with a softmax activation function, converts the network's outputs into a probability distribution across the target classes. Then, the class that has the highest probability is selected.

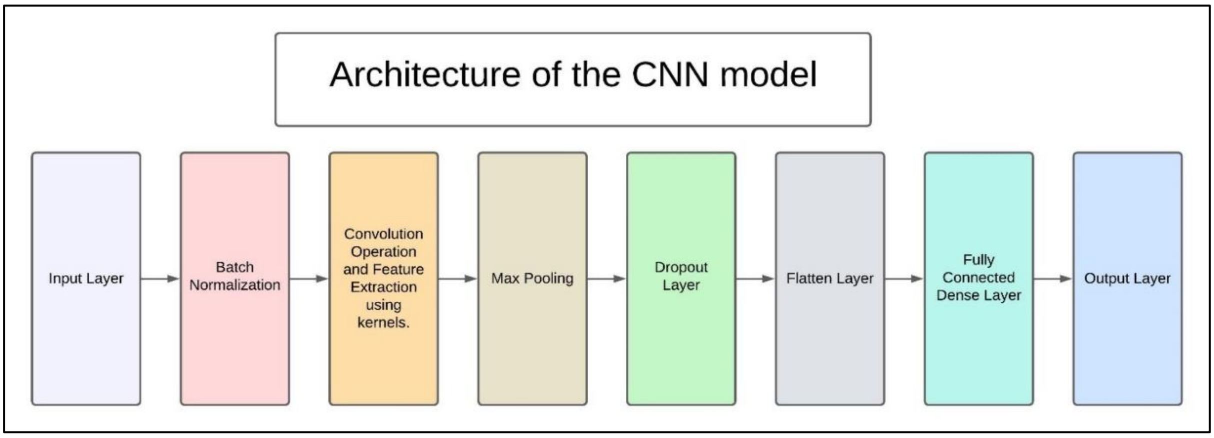
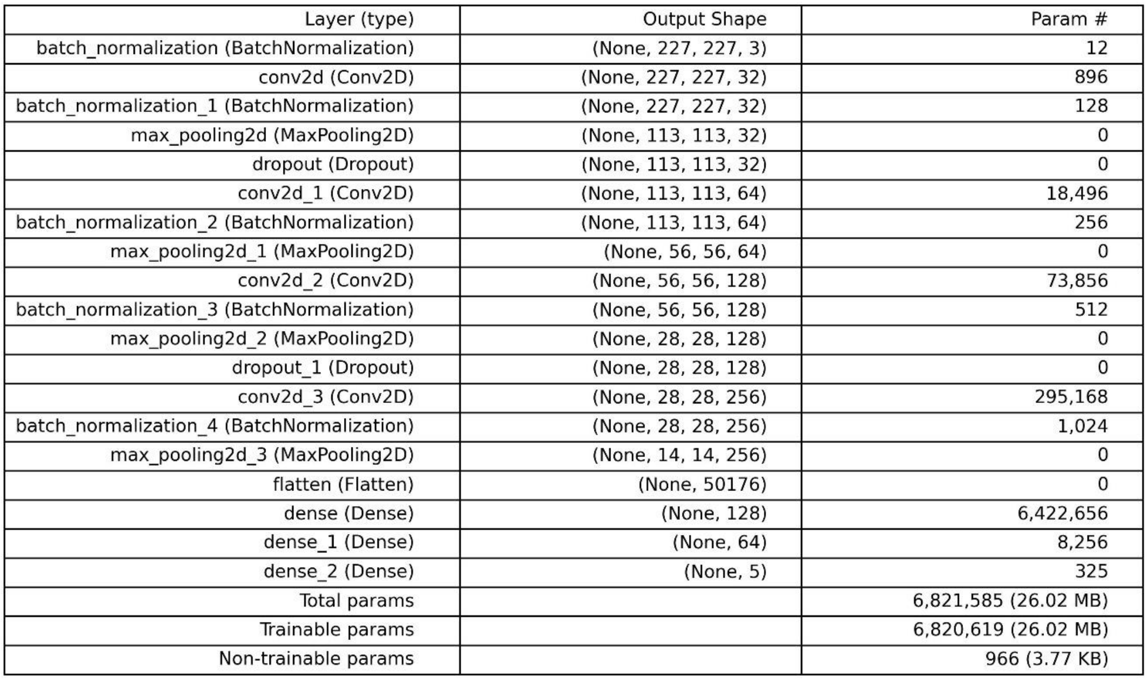

Architecture of the CNN model
Input Layer
Batch Normalization
Convolution Operation and Feature Extraction using kernels.
Max Pooling
Dropout Layer
Flatten Layer
Fully Connected Dense Layer
Output Layer

Figure 16: Layers in a typical CNN model

3.6.1 Layers

The CNN architecture consists of a wide array of layers created using Keras and TensorFlow. All the layers are explained below, and the architecture that showed the best results is Figure 17. The deep learning CNN model has approximately 7 million trainable parameters.



Layer (type)	Output Shape	Param #
batch_normalization (BatchNormalization)	(None, 227, 227, 3)	12
conv2d (Conv2D)	(None, 227, 227, 32)	896
batch_normalization_1 (BatchNormalization)	(None, 227, 227, 32)	128
max_pooling2d (MaxPooling2D)	(None, 113, 113, 32)	0
dropout (Dropout)	(None, 113, 113, 32)	0
conv2d_1 (Conv2D)	(None, 113, 113, 64)	18,496
batch_normalization_2 (BatchNormalization)	(None, 113, 113, 64)	256
max_pooling2d_1 (MaxPooling2D)	(None, 56, 56, 64)	0
conv2d_2 (Conv2D)	(None, 56, 56, 128)	73,856
batch_normalization_3 (BatchNormalization)	(None, 56, 56, 128)	512
max_pooling2d_2 (MaxPooling2D)	(None, 28, 28, 128)	0
dropout_1 (Dropout)	(None, 28, 28, 128)	0
conv2d_3 (Conv2D)	(None, 28, 28, 256)	295,168
batch_normalization_4 (BatchNormalization)	(None, 28, 28, 256)	1,024
max_pooling2d_3 (MaxPooling2D)	(None, 14, 14, 256)	0
flatten (Flatten)	(None, 50176)	0
dense (Dense)	(None, 128)	6,422,656
dense_1 (Dense)	(None, 64)	8,256
dense_2 (Dense)	(None, 5)	325
Total params		6,821,585 (26.02 MB)
Trainable params		6,820,619 (26.02 MB)
Non-trainable params		966 (3.77 KB)

Figure 17: CNN model summary

3.6.2 Input Layer

The input layer is configured to receive images of 227x227 pixels with three colour channels (RGB). This layer is crucial for ensuring that the image data is appropriately formatted before being processed by the subsequent layers. The specified dimensions balance the need for computational efficiency and the retention of critical image details. At the same time, the three colour channels allow the model to utilise colour information effectively, which is often vital in distinguishing between different classes.

3.6.3 Batch Normalisation

Batch Normalisation is applied to normalise the input layer by adjusting and scaling the activations (Krizhevsky et al., 2012), as explained in section 3.5.3.

3.6.4 Convolution Layers

The convolution operation in Convolutional Neural Networks (CNNs) involves applying a filter (or kernel) across an input image to produce a feature map. This process detects local patterns, such as edges or textures, by multiplying element-wise between the input's filter and overlapping regions and then summing the results (LeCun et al., 1998). Key aspects include using padding to maintain spatial dimensions and striding to control the movement of the filter across the image. The resulting feature maps retain the spatial hierarchy of features, making CNNs particularly effective for image recognition tasks (Goodfellow et al., 2016). Figure 18 showcases the convolution operation on an image.

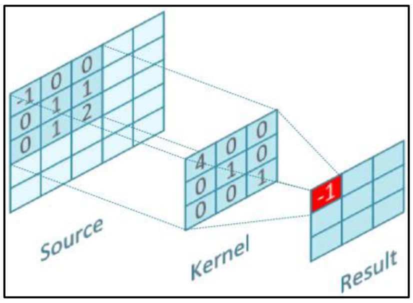
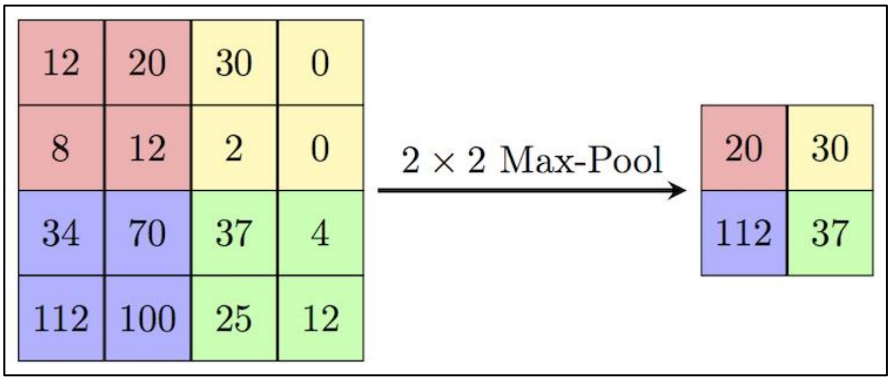


Figure 18: Convolution operation on tensor to extract features (Wicht, 2018)

3.6.5 MaxPooling2D: Pooling Layer

MaxPooling layers reduce the feature maps' spatial dimensions (height and width) while retaining the most essential information. This is achieved by taking the maximum value from a window (e.g., 3x3) within the feature map. By doing so, MaxPooling reduces the computational complexity and controls overfitting by progressively reducing the spatial size of the representation, allowing the network to focus on the most critical features (Zafar et al., 2022). Figure 19 depicts an example of a MaxPooling operation on a tensor using a 2x2 filter.



12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

20	30
112	37

Figure 19: Example of MaxPooling operation

3.6.6 Dropout layer

Dropout is a regularization technique used to prevent overfitting in neural networks. During training, Dropout randomly sets a fraction of the input units to zero at each update during the forward pass, which helps to prevent the model from becoming too dependent on any one feature. (Nitish Srivastava et al., 2014).

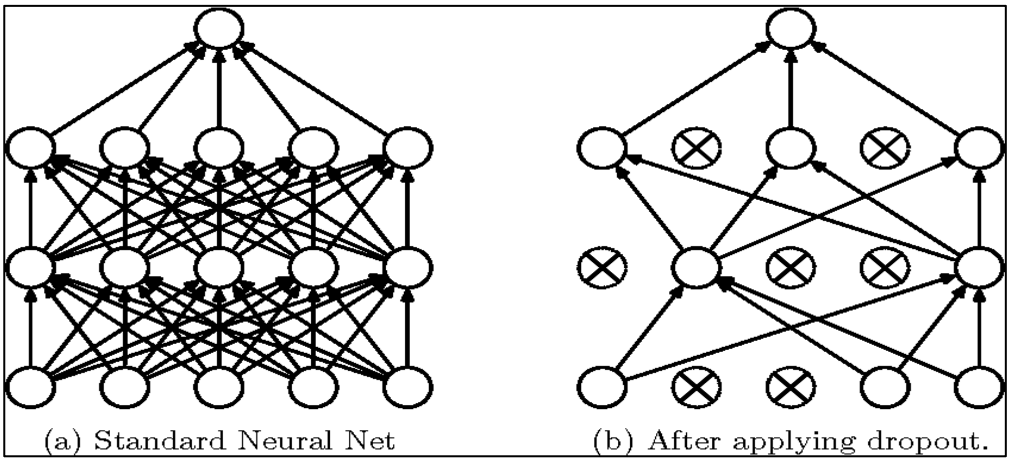


Figure 20: Dropout in fully connected dense layer (Nitish Srivastava et al., 2014)

3.6.7 Flatten layer

The flatten layer transforms the 2D matrix of the feature map into a 1D vector that can be fed into fully connected layers. This step is crucial as it bridges the convolutional and fully connected layers, allowing the extracted features to be used for classification tasks (Simonyan & Zisserman, 2014). Figure 21 shows the flattening operation on a multidimensional tensor to convert it into a one-dimensional vector.

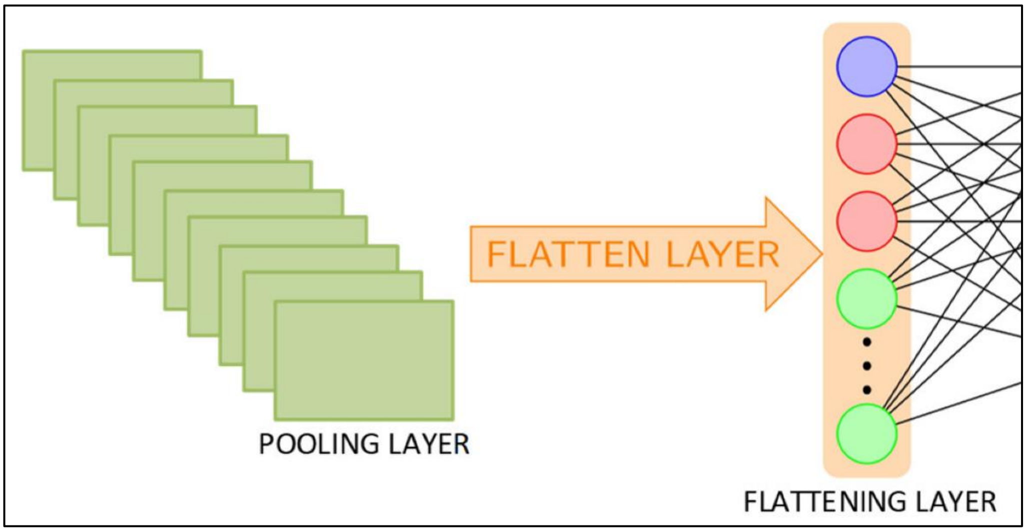


Figure 21: Flattening of multi tensors to transfer to dense layer (Khan et al., 2022)

3.6.8 Dense layer

A dense layer, also known as a fully connected layer, is a fundamental component of deep learning architectures, particularly in neural networks. In this layer, each neuron is connected to every neuron in the preceding layer, allowing for the comprehensive integration of features extracted from previous layers (Goodfellow et al., 2016). This full connectivity enables the network to learn complex patterns by combining features in various ways, which is crucial for image recognition and classification (LeCun et al., 2015). The dense layer applies a linear transformation to the input, followed by an activation function that introduces non-linearity, thus helping the model learn intricate relationships within the data (Schmidhuber, 2015). Figure 22, showcases a fully connected dense layer.

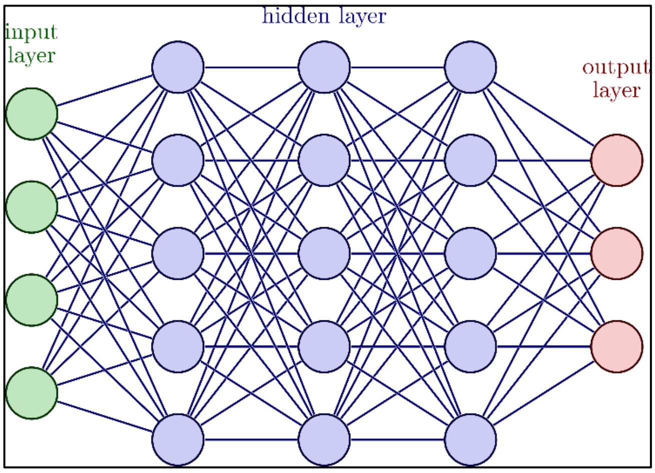


Figure 22: A fully connected dense layer (Neutelings, n.d.)

Output layer

The output layer comprises five neurons, each representing a class in the classification task. It employs the softmax activation function explained in section 3.7.3 to convert the network's raw outputs into a probability distribution, ensuring that the sum of the probabilities equals 1. This facilitates a clear interpretation of the model's predictions, with the class assigned the highest probability being identified as the predicted label.

3.7 Parameters

3.7.1 Batch size

Batch size refers to the number of training examples processed in one iteration during a Convolutional Neural Network (CNN) training. Large batch sizes can speed up training and provide stable gradient estimates but may require significant memory and lead to poorer generalization. Conversely, a small batch size uses less memory and can improve generalization through noisier gradient estimates but may slow down training due to more frequent updates (LeCun et al., 2012).

3.7.2 ReLU

Rectified Linear Unit is used as the activation function in the layers due to its ability to introduce non-linearity while being computationally efficient. Unlike sigmoid or tanh, ReLU helps mitigate the vanishing gradient problem by setting negative values to zero, which accelerates the training process and improves convergence without adding significant computational overhead (Nair & Hinton, 2010; Glorot et al., 2011). It is a de facto standard for deep learning applications; hence, for this study, it is used for all the experiments. Figure 23 illustrates the geometrical intuition of ReLU.

The ReLU function is defined as

$$f(x) \; = \max{(0, x)}$$

It outputs the x if it's positive, and zero otherwise.

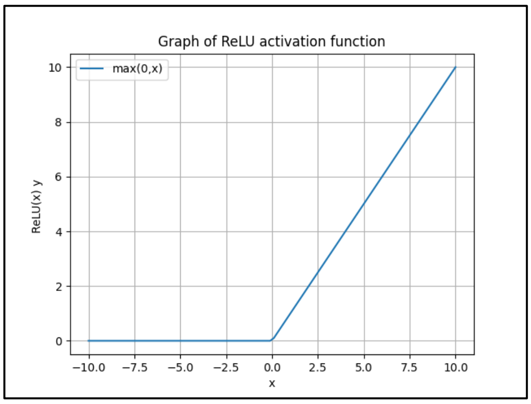


Figure 23: Graph of ReLU activation function

3.7.3 Softmax

The Softmax function is the final output layer for multi-class classification tasks. Here, the output probabilities are five. It calculates the probability distribution of the classes in the output layer, allowing the network to make a decision based on the class with the highest probability (Jing et al., 2022). The Softmax function is crucial for converting the neural network's raw output into probabilities, enabling the model to make accurate predictions (Polat & Mehr, 2019).

$$\sigma(x) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{j}}, \forall\ i\ is\ the\ current\ class, k\ is\ total\ number\ of\ classes$$

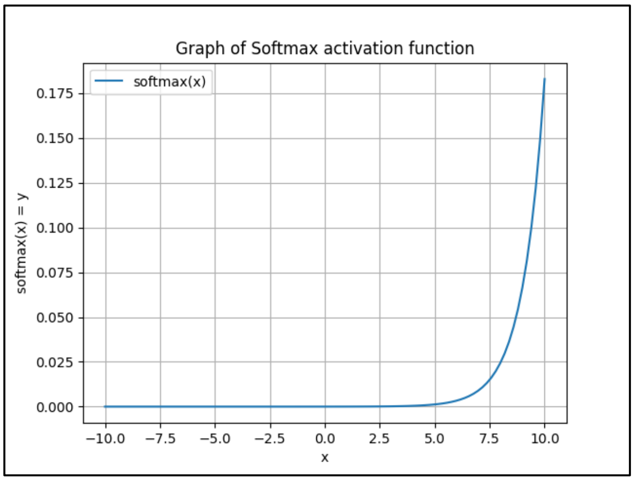


Figure 24: Graph of softmax function

3.7.4 Padding

Padding in CNNs involves adding extra pixels around the borders of an image to preserve spatial dimensions during convolution operations, which get lost in convolution operations see section 3.6.4. This technique ensures that spatial information, especially at the edges, is retained, enhancing the network's ability to learn features across the entire image. Additionally, padding aids in position encoding, which is crucial for tasks requiring spatial understanding, such as object localisation (Innamorati et al., 2019; Islam, 2021; Liu et al., 2018). For example, if a 227x227 image is passed, and if padding is not applied after the convolution operation, the output image will be of shape less than the original image; utilising padding, the output shape will be the same as the input image see Figure 25, preserving the spatial dimension which is crucial for feature extraction.

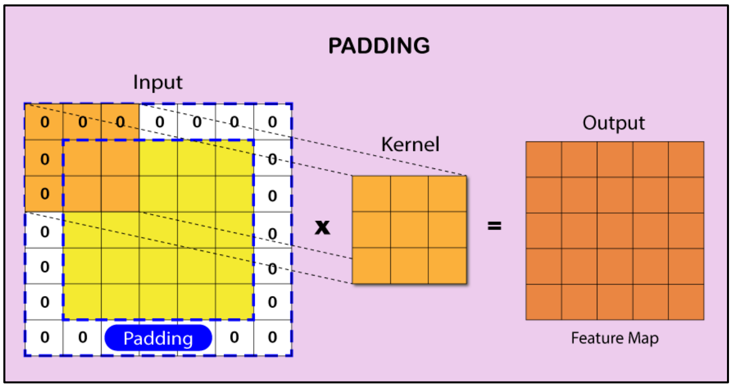


Figure 25: Padding operation on an image tensor (K E, n.d.)

3.7.5 Epochs

An epoch in machine learning is one complete pass through the entire training dataset during which the model's parameters are updated to minimise the loss function (Goodfellow et al., 2016). Multiple epochs are typically required to allow the model to improve its performance iteratively.

3.7.6 Optimizers (Adam and RMSProp)

Deep Learning uses a concept called gradient descent, which is a fundamental optimisation technique used to minimize the loss function by iteratively updating model parameters in the opposite direction of the gradient, general equation of gradient update is described below.

$$\theta_{\{t+1\}} = \theta_t - \eta \cdot \nabla_{\theta J(\theta_t)}$$

where,

- θ is loss function.
- t is the current gradient.
- ∇ is partial derivate of θ w.r.t j, j is the current weight.
- η is the learning rate.

For this study, two state-of-art optimizers, namely Adam and RMSProp, are tested, as explained below.

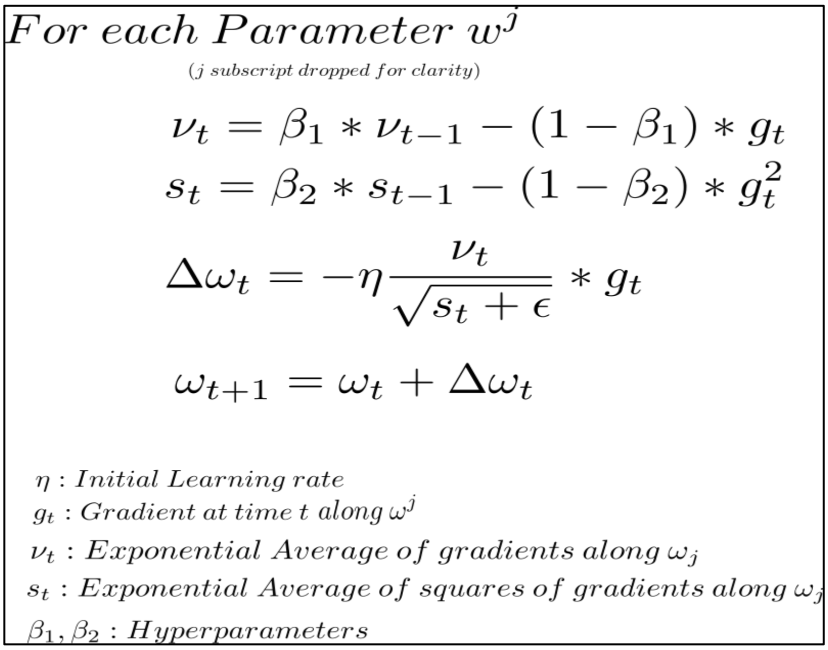
3.7.6.1 RMSProp

RMSProp (Root Mean Squared Propagation) is an optimization algorithm that builds on AdaGrad by adaptively adjusting the learning rate during training. It does this by using an exponentially decaying average of past squared gradients, allowing it to dynamically adjust the learning rate for each parameter. This helps prevent the learning rate from decreasing too rapidly, enabling more stable and efficient training of neural networks compared to standard Gradient Descent. Algorithm on how it works is explained below.

- The gradient is calculated, $g_t = \nabla_\theta$, where $g_t$is the gradient of the loss function θ.
- Accumulate squared gradients: E[g ²]t = βE[g²]t-1 + (1-β) gt², where β is the decay rate, typically set to 0.9.
- Compute the adaptive learning rate: ηt = η / √(E[g^2]t + ε), where η is the initial learning rate and ε is a small constant to prevent division by zero, often set to 1e-8.
- Update the parameters: θt+1 = θt - ηt x g_t

3.7.6.2 Adam

The Adaptive Moment Estimation optimizer, an extension of gradient descent, combines the advantages of both momentum and RMSprop by adapting the learning rate for each parameter, improving the convergence speed and performance stability of deep neural networks (Kingma & Ba, 2015). It is an adaptive learning rate optimization algorithm that combines the advantages of two other popular methods: AdaGrad and RMSprop. It computes individual adaptive learning rates for different parameters based on the estimates of the first and second moments of the gradients, making it particularly effective for training deep neural networks (Kingma & Ba, 2014). The mathematical formulation is explained in Figure 26.



$For\ each\ Parameter\ w^j$

$(j\ subscript\ dropped\ for\ clarity)$

$$\nu_t = \beta_1 * \nu_{t-1} - (1 - \beta_1) * g_t$$

$$s_t = \beta_2 * s_{t-1} - (1 - \beta_2) * g_t^2$$

$$\Delta\omega_t = -\eta \frac{\nu_t}{\sqrt{s_t + \epsilon}} * g_t$$

$$\omega_{t+1} = \omega_t + \Delta\omega_t$$

$\eta : Initial\ Learning\ rate$

$g_t : Gradient\ at\ time\ t\ along\ \omega^j$

$\nu_t : Exponential\ Average\ of\ gradients\ along\ \omega_j$

$s_t : Exponential\ Average\ of\ squares\ of\ gradients\ along\ \omega_j$

$\beta_1, \beta_2 : Hyperparameters$

Figure 26: Adam optimizer mathematical concept (Kathuria, 2018)

3.7.7 Loss Functions

3.7.7.1 Categorical Cross Entropy

The categorical cross-entropy loss function is commonly used in multi-class classification tasks because it penalises incorrect predictions, encouraging the model to assign high probabilities to the correct class. This loss function is particularly appropriate for this study, which involves a dataset with more than two classes, as it ensures that the model's predictions closely align with the actual class labels, thereby enhancing classification accuracy (Goodfellow et al., 2016). The sigmoid function is used for binary classification since this dataset contains five classes; using sigmoid is futile. For mathematical intuition, see Figure 27.

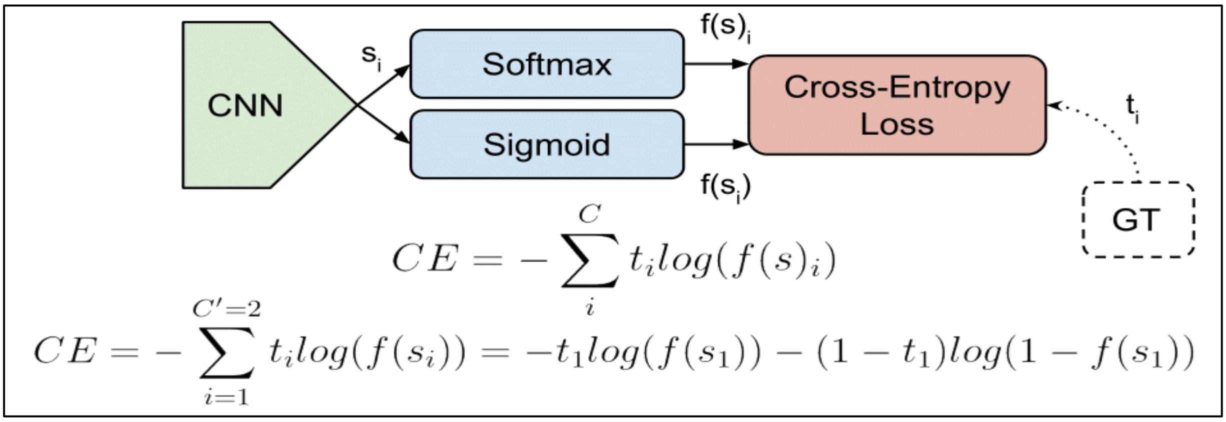

CNN
s_i
Softmax
Sigmoid
$f(s)_i$
$f(s_i)$
Cross-Entropy Loss
t_i
GT
$CE = -\sum_{i}^{C} t_i log(f(s)_i)$
$CE = -\sum_{i=1}^{C'=2} t_i log(f(s_i)) = -t_1 log(f(s_1)) - (1 - t_1) log(1 - f(s_1))$

Figure 27: Categorical Cross Entropy mathematical concept (allglenn, 2023)

3.7.7.2 Focal loss

Focal Loss is a specialised function designed to address the challenge of class imbalance in classification problems, a common issue in computer vision tasks such as object detection and multiclass classification using CNNs. Lin et al. (2017) proposed that focal Loss modifies the standard cross-entropy Loss by introducing a scaling factor that adjusts the loss contribution from easy-to-classify examples.

$$Focal\ Loss = FL(pt) = -\alpha t(1 - pt)\gamma log(pt),$$

Where:

- Pt_i: The predicted probability for the true class label.
- α_t : A weighting factor that balances the importance of positive and negative examples (optional).
- γ: The focusing parameter that reduces the relative loss for well-classified examples, placing more focus on hard-to-classify ones. Commonly, γ is set to 2.

This method focuses the learning process on hard-to-classify, minority-class instances, thereby improving the model's ability to handle imbalanced datasets effectively (Lin et al., 2017). Since this study uses an imbalanced dataset, it is an exciting loss function for optimising the model.

3.8 Evaluation metrics

- True Positive: It refers to the instances that are correctly predicted as belonging to a specific positive class. Also referred to as TP.
- True Negative: It refers to the instances that are correctly predicted as belonging to a specific negative class. Also referred to as TN.
- False Positive: It refers to the instances that are incorrectly predicted as belonging to a specific positive class. Also referred to as FP.
- False Negative: It refers to the instances that are incorrectly predicted as belonging to a specific negative class. Also referred to as FN.

3.8.1 Accuracy

It is defined as the ratio of correctly predicted instances to the total instances in the dataset. It provides a straightforward metric for overall model performance, indicating the proportion of correct classifications out of all classifications made (Sokolova & Lapalme, 2009).

$$Accuracy = \frac{(Total\ Number\ of\ Correct\ Predictions)}{(Total\ Number\ of\ Predictions)} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

3.8.2 Recall

Also known as sensitivity or true positive rate, measures the ability of a model to correctly identify all relevant instances of a particular class. In the context of multiclass classification, recall is calculated for each class individually. It indicates the proportion of actual positive instances that were correctly classified by the model, making it a critical metric in scenarios where missing a positive instance has a high cost (Powers, 2020).

$$Recall = \frac{(TP_i)}{(TP_i +\ FN_i)}, \qquad \forall\ i\ is\ the\ numer\ of\ classes$$

3.8.3 Precision

It is a measure of a model's accuracy in identifying true positive instances out of all instances it predicted as positive. In multiclass classification, precision is calculated for each class individually, indicating the proportion of instances predicted as a particular class that actually belong to that class. Precision is crucial when the cost of false positives is high, as it reflects the model's ability to minimise incorrect positive predictions (Powers, 2020).

$$Precision = \frac{(TP_i)}{(TP_i +\ FP_i)}, \qquad \forall\ i\ is\ the\ numer\ of\ classes$$

3.8.4 F1-Score

It is a metric that combines precision and recall into a single value, providing a balanced measure of a model's accuracy. The harmonic mean of precision and recall ensures that both false positives and false negatives are considered in the evaluation. In multiclass classification, the F1 score is calculated for each class individually and can be averaged to provide an overall performance measure (Sokolova & Lapalme, 2009). The F1 score is instrumental when the dataset has a class imbalance, which is true in this case, as it provides a more comprehensive view of model performance than accuracy alone or when the costs of false positives and false negatives are both high. Focusing on precision and recall, the F1 score provides a more holistic view of a model's accuracy (Powers, 2020).

$$F1 - Score = 2 \times \frac{Precision_i\ \times\ Recall_i}{Precision_i +\ Recall_i}, \qquad \forall\ i\ is\ the\ numer\ of\ classes$$

Developing and implementing a CNN model for detecting and classifying cassava leaf diseases aligns with the third research objective outlined in section 1.2. Specifically, objective three (RO3) uses image data to develop and implement a CNN model for disease detection and classification (see Section 1.2.1).

4. Results

This study aimed to compile a CNN which gives the highest possible accuracy based on multiple parameters. An iterative process of optimizing the model and fine-tuning it was used. The best model achieved a training accuracy of 99% and a test accuracy of over 90%. An average testing accuracy indicates the overfitting of the model due to an imbalanced class of the dataset, although data augmentation see section 3.5.4 was implemented. All the images were trained and tested for shape (227x227) pixels. However, a balanced approach was able to achieve 90% accuracy.

4.1 Overall result of the experiments

Run Name	Activation Function (CCE=Categorical Cross Entropy)	Optimizer	Epochs	Trainable parameters	Train Duration (mins)	Dropout Ratio	Accuracy	Precision	Recall	F1 Score
bedecked-lynx-31	CCE,ReLU	Adam	50	1,94,36,101	22	0.5	45%	NA	NA	NA
bemused-penguin-717	CCE,ReLU	Adam	100	1,32,76,171	60	0.1	68%	0.46	0.56	0.5
carefree-mole-118	FocalLoss,ReLU	RMSProp	10	1,32,76,171	5.6	0.1	61%	0.37	0.61	0.46
efficient-koi-221	CCE, ReLU	RMSProp	50	36,41,413	19	0.2	65%	0.42	0.45	0.43
overjoyed-horse-209	CCE, ReLU	Adam	100	1,94,36,101	43	0.2	63%	0.35	0.56	0.43
placid-snake-567	CCE,ReLU	Adam	100	68,21,585	49	0.2	93%	0.93	0.94	0.94
victorious-gnat-76	FocalLoss,ReLU	RMSProp	36	1,32,76,171	18	0.1	61%	0.37	0.61	0.46

Figure 28: Evaluation chart of different CNN model variants

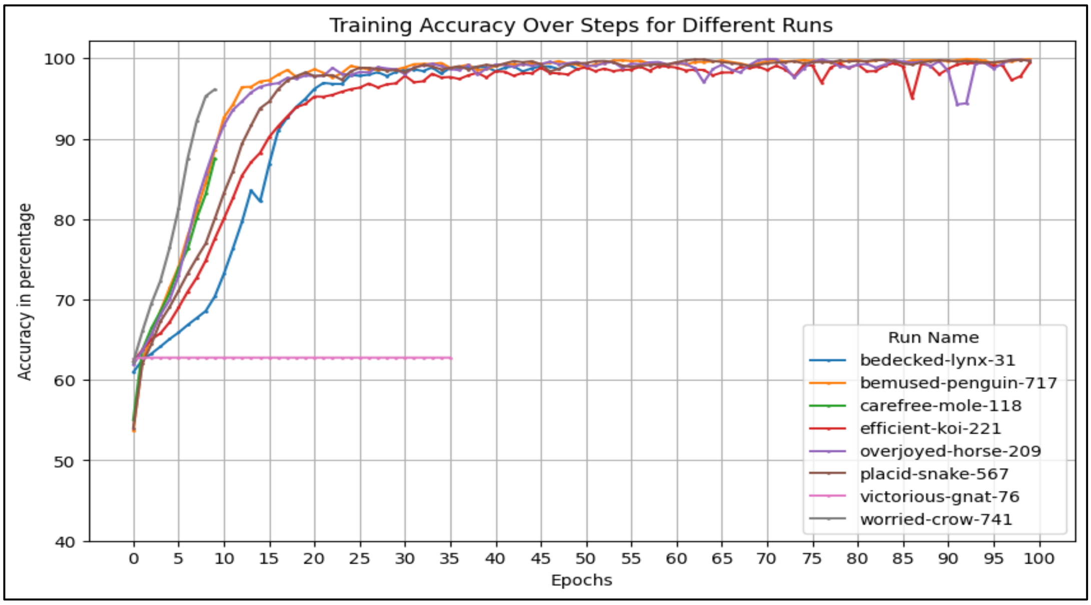
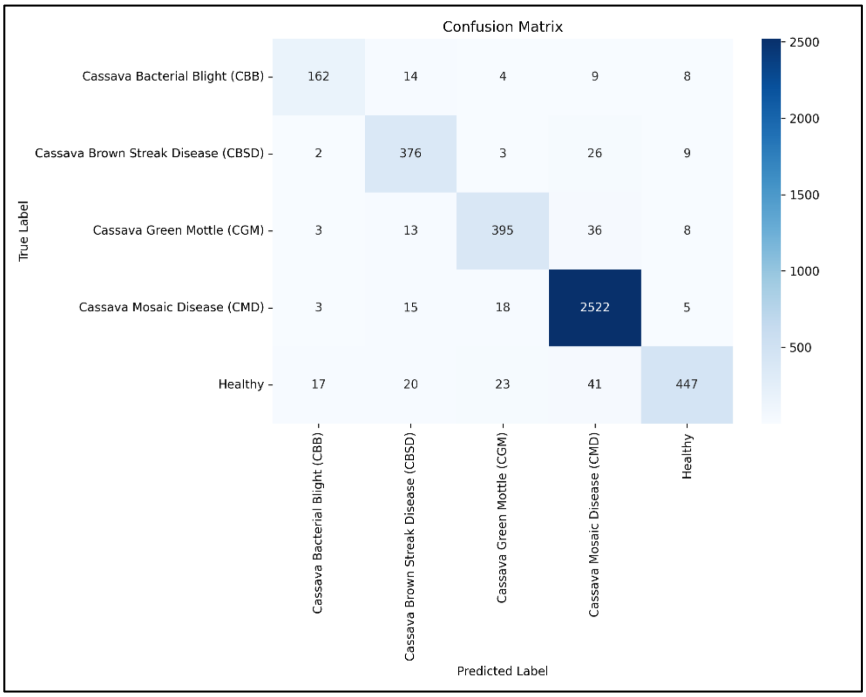


Figure 29: Graph of training accuracy over epochs

Figure 29 illustrates the training accuracy of experiments on different neural network configurations for cassava leaf disease detection, highlighting the importance of balancing model complexity, regularisation, and hyperparameter settings. The first experiment, placid-snake-567, used the Adam optimizer with a ReLU activation function, 100 epochs, and a moderate dropout rate of 0.2, achieving the highest accuracy of 93%. This suggests that a balanced approach with sufficient epochs and proper regularization can help the model generalize well. In contrast, the second experiment, bemused-penguin-717, with a lower dropout rate of 0.1 and fewer trainable parameters, achieved an accuracy of 68%, indicating potential overfitting due to insufficient regularization. The third experiment, bedecked-lynx-31, implemented a higher dropout rate of 0.5 but only trained for 50 epochs, resulting in an even lower accuracy of 64%, suggesting underfitting where the model could not learn effectively due to excessive. The rest of the runs have similar results of above 60% accuracy in test data. How the loss function decreased with each iteration is explained below.



	Cassava Bacterial Blight (CBB)	Cassava Brown Streak Disease (CBSD)	Cassava Green Mottle (CGM)	Cassava Mosaic Disease (CMD)	Healthy
Cassava Bacterial Blight (CBB)	162	14	4	9	8
Cassava Brown Streak Disease (CBSD)	2	376	3	26	9
Cassava Green Mottle (CGM)	3	13	395	36	8
Cassava Mosaic Disease (CMD)	3	15	18	2522	5
Healthy	17	20	23	41	447

Figure 30: Confusion matrix for best model placid-snake-567

The diagonal entries represent the number of correct predictions, indicating that the model performed best at identifying CMD, with 2522 correct predictions. Other diseases like CBB, CBSD, and CGM showed fewer correct predictions, with some misclassifications. For instance, CBB was sometimes confused with other categories, as indicated by the off-diagonal values, such as 14 instances where CBB was misclassified as CBSD. The model also correctly identified 447 healthy samples but made some errors, as shown by misclassifying healthy samples as other diseases. While the model is highly effective in detecting CMD, it struggles to distinguish between other diseases, suggesting a need for further refinement or balanced training data to improve accuracy across all categories.

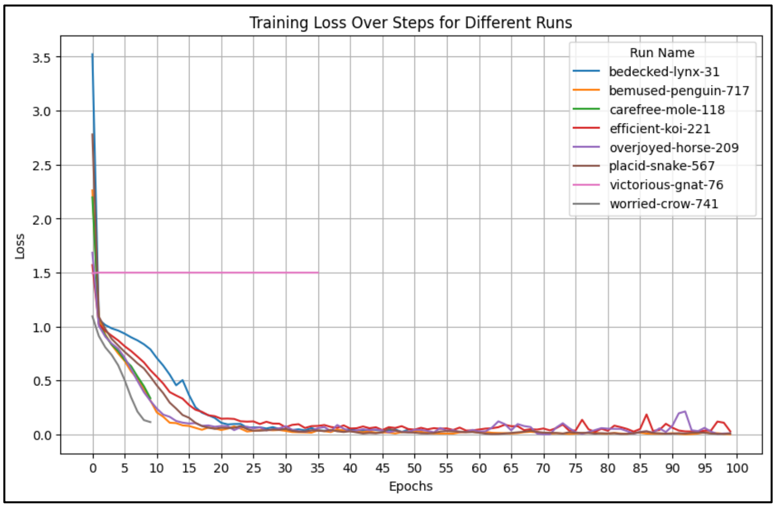


Figure 31: Graph of loss over epochs

Figure 31 illustrates the loss over epochs while training; on the x-axis, epochs represent the number of iterations the model has undergone, while the y-axis denotes the loss value, which measures the error between the predicted outputs and the actual values. From experiments, it is evident that most runs achieved a loss of less than 0.5 in a mere 15-20 epochs, indicative of the prowess of the Adam optimizer see section 3.6.7.2. Several runs, such as bedecked-lynx-31 and bemused-penguin-717, start with a high loss but experience a rapid decline within the first few epochs, reflecting the early phase of training where the model quickly adjusts to the data. In contrast, runs such as victorias-gnat-76 suffered from a vanishing gradient problem, as it remained stuck at a loss value of 1.5 throughout the training period, where Focal Loss was used.

4.2 Key findings

- While focal Loss is known to better tackle imbalanced datasets, contrary to Lin et al. (2017), it did not outperform Categorical Cross-Entropy loss in this instance. This result underscores the need for further research and experimentation to enhance the understanding and application of loss functions in deep Learning.
- While increasing the dropout layer can help mitigate overfitting (Nitish Srivastava et al., 2014), it is essential to note that it also comes with a trade-off. In this case, the trade-off was a significant decrease in the model's overall accuracy run bedecked-lynx-31 illustrates the same.
- Increasing the layers of CNN ramped up the model's complexity and total trainable parameters. This, in turn, led to a longer training time and a reduction in the model's generalisation ability, highlighting the need for careful consideration in model design.
- Deep Learning using CNN has significantly improved using state-of-the-art optimisers such as Adam, which achieved over 90% accuracy in training in a mere 20 epochs (run name: efficient-koi-221).
- Adam optimizer performed faster convergence and better accuracy than RMSprop optimizer.

5. Discussions

This section discusses the significance of the literature review based on different aspects of the research and how it aligns with or differs from other findings.

5.1 Overfitting and class imbalance

Despite using dropout layers and data augmentation, the finding that there is overfitting to the CMD class aligns with existing literature highlighting the challenges of class imbalance in deep learning models (Kamilaris & Prenafeta-Boldú, 2018). In plant disease detection, the class imbalance can lead to models becoming biased towards more prevalent classes, as seen with the CMD class in this study, which has over 10,000 images. This is consistent with previous findings, where CNNs perform better in dominant classes due to their higher representation in the training data (Abayomi-Alli et al., 2021; Lu et al., 2021).

5.2 Impact of dropout layers on model accuracy

The observation that increasing the dropout rate reduces the overall accuracy supports the findings of Srivastava et al. (2014), who noted that while dropout helps in the prevention of overfitting by randomly dropping neurons, excessive dropout can lead to underfitting, where the model fails to learn adequately from the training data. From Figure 28 for the run bedecked-lynx-31, the dropout layer used is 0.5, which means fifty per cent of neurons will deactivate while training, which led to underfitting, rendering insignificant results.

5.3 Model complexity and generalisation

Increasing the complexity of the CNN model by adding more layers resulted in a longer training time and reduced generalisation, which aligns with findings from (He et al., 2016; Lu, 2023). These studies indicate that deeper networks can learn more complex features but are also more prone to overfitting, mainly when training data is limited or imbalanced. The results align with the broader understanding that while deeper architectures like ResNet can enhance performance, they require careful management of training resources and overfitting risks (He et al., 2016; Lu, 2023).

5.4 Effectiveness of optimizers: Adam vs. RMSprop

The superior performance of the Adam optimizer used by runs placid-snake-567, bemused-penguin-717, bedecked-lynx-31, achieving over 90% accuracy within 20 epochs see Figure 28, supports the findings of Kingma and Ba (2015), who demonstrated Adam's effectiveness in adapting learning rates based on gradient moments, leading to faster convergence and better performance than other optimizers like RMSprop. This study confirms that Adam's adaptive learning rate capabilities make it suitable for complex classification tasks, such as cassava disease detection, where rapid and efficient learning is crucial (Kingma & Ba, 2015).

5.5 Focal Loss vs. Categorical Cross Entropy

Despite the theoretical benefits of focal loss in handling class imbalance (Lin et al., 2017), its performance in this study was not superior to categorical cross-entropy. For the run (victorious-gnat-76), where the focal loss was used with the RMSprop optimizer, the training accuracy was only 63%, with the loss plateauing around 1.5, indicating a possible vanishing gradient problem. This result suggests that while focal loss can benefit specific contexts, its effectiveness may be limited when used with specific optimizers or network architectures. These findings suggest that the choice of loss function and optimizer must be carefully considered based on the specific characteristics of the dataset and the model architecture (Lin et al., 2017).

5.6 Accuracy in context of model variations

Accuracy remains a critical metric in evaluating the performance of different CNN configurations. As seen in this study, placid-snake-567 achieved the highest accuracy of 93%, which significantly surpasses the performance of another study (Ding, 2022), which achieved an accuracy of 75%. This can be attributed to Categorical Cross Entropy (CCE) loss with the Adam optimizer, suggesting that this combination is particularly effective for cassava leaf disease classification (Kingma & Ba, 2015). In contrast, runs using Focal Loss, such as victorious-gnat-76 and carefree-mole-118, achieved lower accuracy levels, indicating that while Focal Loss is designed to handle class imbalance, it may not be as effective as CCE in this specific application. The lower accuracy observed in these runs aligns with findings from Lu et al. (2023) that suggest focal loss may require more careful tuning and may not universally outperform traditional loss functions like CCE, especially in multiclass scenarios. Additionally, models with a higher number of trainable parameters, such as bemused-penguin-717, did not achieve the highest accuracy, suggesting that increasing model complexity does not necessarily translate to better performance, as He et al. (2016) noted.

6. Conclusions

This study, which focused on using Convolutional Neural Networks (CNNs) to detect and classify diseases in cassava leaves, evaluated different architectures, optimisations, and loss functions. The findings are clear: the CNN model using Categorical Cross Entropy with the Adam optimiser achieved the highest accuracy of over 90%, reinforcing the credibility of the research.

This work's implications include the potential for deploying these models in real-world agricultural settings. This would enable farmers to detect diseases early and manage crops more effectively, thereby significantly reducing losses and improving yield. The study also underscores the importance of addressing class imbalance and optimising model parameters, which are critical for improving model performance in practical applications.

This research demonstrates the effectiveness of deep learning techniques, particularly CNNs, in agricultural disease detection. The comparative analysis of different optimisers and loss functions contributes to the existing knowledge by identifying optimal configurations for cassava disease classification, providing a valuable reference for future studies.

6.1 Revisiting research questions

RQ1: How can machine learning models be effectively utilized to detect and classify diseases in cassava leaves based on image data?

This study comprehensively explores using Convolutional Neural Networks for disease detection in cassava leaves. By employing various CNN architectures and optimisers, the research demonstrates that machine learning models can be effectively used to analyse image data to classify cassava diseases. The successful classification of diseases such as CMD, CBB, and CGM, as evidenced by the results from different runs (e.g., placid-snake-567 achieving 93% accuracy), shows that CNNs are capable of detecting subtle features in images that are indicative of specific diseases. This aligns with previous research highlighting the power of CNNs in image-based classification tasks (Hoeser et al., 2020; Lu et al., 2023).

RQ2: What impact do the identified diseases have on cassava crop health and yield, and how can early detection improve agricultural outcomes?

The study indirectly addresses the impact of identified diseases on cassava crop health and yield by focusing on CNNs' early detection capabilities. Accurate early detection of diseases such as CMD, which is overrepresented in the dataset, implies that actionable insights can be provided to farmers and agricultural stakeholders, leading to timely interventions. The research demonstrates that machine learning models can effectively analyse image data to classify different cassava diseases, achieving 93% accuracy in one instance. This study aligns with previous research emphasising the importance of early disease detection for maintaining crop health and maximising yield, such as the work of Hoeser et al. (2020) and Lu et al. (2023).

RQ3: Which machine learning model of CNN demonstrates the highest accuracy and effectiveness in detecting and classifying diseases in cassava leaves?

The study evaluates different CNN architectures, optimisers, and loss functions to identify the most effective model for disease detection in cassava leaves. The results indicate that the CNN model using Categorical Cross Entropy with the Adam optimiser (placid-snake-567 run) demonstrates the highest accuracy and effectiveness, achieving a 93% accuracy rate. This finding is consistent with the literature, which often cites Adam as a highly effective optimiser for training deep learning models due to its adaptive learning rate capabilities (Kingma & Ba, 2015). In comparison, models using Focal Loss with RMSprop, like victorious-gnat-76, achieved lower accuracy, highlighting the superior performance of the Adam optimiser in this context.

6.2 Limitations

- **Basic CNN model used:** This study used a basic CNN model due to computation limitations. A hybrid approach such as Transfer Learning requires more computation power.
- **Insufficient Layers of CNN:** CNN works on the principle of extracting features using kernels (section 3.6.4); however, in this study, a low number of layers is used due to limited computing power.
- **Data Imbalance:** The data collected from Kaggle is exceptionally imbalanced, with CMD dominating features, which led to incorrect classification on several models (see Appendix 8.4).
- **No use of Bagging:** Bagging refers to combining multiple machine learning models to train separately and make predictions using all three to obtain more accurate predictions.
- **Inadequate Data:** Deep Learning algorithms require a substantial amount of data to perform effectively, and the data present in Kaggle is extremely limited, with approximately 20,000 images. This data can only contribute to a small number of features from overall images, leading to underfitting.
- **Detection Only:** This study only focuses on the type of disease the plant has, which is limited to four; it does not discuss how to tackle them.

6.3 Assumptions

- **Independent Runs:** Each run is assumed to be independent of the others, meaning that the results of one run do not influence another. This independence is crucial for drawing valid conclusions from comparisons.
- **No External Interference During Training:** It is assumed that the training process was not interrupted or influenced by external factors, such as hardware failures or significant changes in computational resources, which could affect the training dynamics and the resulting loss values.
- **No Data Leakage:** Data leakage is not just a technical term but a potential threat to model generalisation. It occurs when training data contains more information than expected, such as information from test data, leading to poor generalisation of real-world unseen data.

6.4 Future work and implications

- The classification of disease in cassava can be further improved using a more hybrid approach to developing a machine learning model, such as XGBoost, to train the model and CNN for feature extraction.
- The use of a functional (non-sequential, not used in this study) model allows for the creation of more sophisticated deep learning models that go beyond the linear stack of layers. This can further help create models with non-linear shared layers, multiple inputs beyond images, and extra information that can be used for training. This approach will significantly increase the effectiveness of disease management, providing reassurance and confidence in the system's capabilities.
- A more diverse dataset, which contains huge amounts of image data of cassava leaves, would help the algorithm, particularly deep learning, extract more features and generalise the model more for better performance on unseen data.
- One promising direction is to develop a system that identifies the presence of a disease and assesses its severity. The model could simultaneously predict the disease type and severity level

by incorporating multi-task learning approaches. This information could be invaluable for farmers, allowing them to prioritise interventions and allocate resources more effectively (Kamilaris & Prenafeta-Boldú, 2018).

- Additionally, the proactive nature of real-time monitoring and early warning systems could be a game-changer in disease management. By using temporal data, such as satellite imagery or sensor data, to monitor changes in plant health over time, farmers could predict the potential spread of diseases and assess the risk to crops. This proactive approach holds the potential to significantly reduce the overall impact of diseases on cassava production, instilling hope and optimism for a healthier crop yield (Picon et al., 2019).

If more time had been available, the study could have focused on several improvements. A larger and more balanced dataset could have been developed to address class imbalance through additional data collection or advanced augmentation techniques. Exploring more sophisticated algorithms, such as XGBoost, RandomForest, and advanced CNNs with different image sizes, could have provided more profound insights. Additionally, experimenting with new pooling methods like Local Average Pooling and MinPooling might have enhanced feature extraction. Integrating advanced models like LSTMs and GANs could also have improved the model's ability to detect and classify diseases by capturing complex patterns and generating synthetic training data.

7. References

Abayomi-Alli, O. O., Damaševičius, R., Misra, S., & Maskeliūnas, R. (2021). Cassava disease recognition from low-quality images using enhanced data augmentation model and deep learning. *Expert Systems*. https://doi.org/10.1111/exsy.12746

Acharya, S., Kar, T., Umesh Chandra Samal, & Prasant Kumar Patra. (2023). Performance Comparison between SVM and LS-SVM for Rice Leaf Disease detection. *ICST Transactions on Scalable Information Systems*. https://doi.org/10.4108/eetsis.3940

Adamopoulou M. (2023). A Comprehensive Overview of Deep Learning Techniques for Retinal Vessel Segmentation. *American Journal of Biomedical Science & Research*, *19*(5), 622–645. https://doi.org/10.34297/ajbsr.2023.19.002631

Ahishakiye, E., Mwangi, W., Petronilla Muthoni, Fredrick, K., Godliver Owomugisha, Danison, T., & Nkalubo, L. (2022). Deep Gaussian Convolutional Neural Network Model in Classification of Cassava Diseases using Spectral Data. *Research Square (Research Square)*. https://doi.org/10.21203/rs.3.rs-1750871/v1

Ahmed, I., & Yadav, P. K. (2022). Plant disease detection using machine learning approaches. *Expert Systems*. https://doi.org/10.1111/exsy.13136

Akano, A., Dixon, A., Mba, C., Barrera, E., & Fregene, M. (2002). Genetic mapping of a dominant gene conferring resistance to cassava mosaic disease. *Theoretical and Applied Genetics*, *105*(4), 521–525. https://doi.org/10.1007/s00122-002-0891-7

Alford, J., & Tuba, E. (2024). *Cassava Plant Disease Detection Using Transfer Learning with Convolutional Neural Networks*. https://doi.org/10.1109/isdfs60797.2024.10527320

Ali, G. A., & ahmed hussein ali. (2023). Convolutional Neural Networks using FPGA-based Pipelining. *Iraqi Journal for Computer Science and Mathematics*, 215–223.

https://doi.org/10.52866/ijcsm.2023.02.02.019

Alkurdy, N. H., Aljobouri, H. K., & Wadi, Z. K. (2023). Ultrasound renal stone diagnosis based on convolutional neural network and VGG16 features. *International Journal of Electrical and Computer Engineering (IJECE)*, *13*(3), 3440. https://doi.org/10.11591/ijece.v13i3.pp3440-3448

allglenn. (2023, December 19). *Understanding the Importance of Loss Functions in Deep Learning*. https://blog.stackademic.com/understanding-the-importance-of-loss-functions-in-deep-learning-dd5ff33551d5

Alonso Chavez, V., Milne, A. E., van den Bosch, F., Pita, J., & McQuaid, C. F. (2021). Modelling cassava production and pest management under biotic and abiotic constraints. *Plant Molecular Biology*. https://doi.org/10.1007/s11103-021-01170-8

Anitha, J., & Saranya, N. (2022). Cassava Leaf Disease Identification and Detection Using Deep Learning Approach. *INTERNATIONAL JOURNAL of COMPUTERS COMMUNICATIONS & CONTROL*, *17*(2). https://doi.org/10.15837/ijccc.2022.2.4356

Annet Namuddu, Seal, S., Sharon van Brunschot, Malka, O., Kabaalu, R., Morin, S., Omongo, C., & Colvin, J. (2023). Distribution of Bemisia tabaci in different agro-ecological regions in Uganda and the threat of vector-borne pandemics into new cassava growing areas. *Frontiers in Sustainable Food Systems*, *7*. https://doi.org/10.3389/fsufs.2023.1068109

Arabameri, A., Chandra Pal, S., Costache, R., Saha, A., Rezaie, F., Seyed Danesh, A., Pradhan, B., Lee, S., & Hoang, N.-D. (2021). Prediction of gully erosion susceptibility mapping using novel ensemble machine learning algorithms. *Geomatics, Natural Hazards and Risk*, *12*(1), 469–498. https://doi.org/10.1080/19475705.2021.1880977

Arepalli Gopi, R, S. L., & Joseph. (2024). *Disclosing the Potential of Deep Learning in Cassava Leaf Disease Analysis by using CNN and Neural Networks Approach*.

https://doi.org/10.1109/icict60155.2024.10544880

Arfat Ahmad Khan, Muhammad Asif Nauman, Shoaib, M., Jahangir, R., Roobaea Alroobaea, Majed Alsafyani, Binmahfoudh, A., & Chitapong Wechtaisong. (2022). Crowd Anomaly Detection in Video Frames Using Fine-Tuned AlexNet Model. *Electronics*, *11*(19), 3105–3105. https://doi.org/10.3390/electronics11193105

Arief Purnama Muharram, & Sajid, F. (2024). *Supervised Machine Learning Approach for Predicting Cardiovascular Complications Risk in Patients with Diabetes Mellitus*. https://doi.org/10.36227/techrxiv.24243631.v5

Ashwin Abraham John. (2022). Identification of Diseases in Cassava Leaves using Convolutional Neural Network. *2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT)*. https://doi.org/10.1109/ccict56684.2022.00013

Basile Njei, Ulrick Sidney Kanmounye, Seto, N., McCarty, T. R., Mohan, B. P., Fozo, L., & Udayakumar Navaneethan. (2023). Artificial intelligence in medical imaging for cholangiocarcinoma diagnosis: A systematic review with scientometric analysis. *Journal of Gastroenterology and Hepatology*, *38*(6), 874–882. https://doi.org/10.1111/jgh.16180

Bhagwat, R., & Dandawate, Y. H. (2021). *A Review on Advances in Automated Plant Disease Detection*. *11*(4), 251–264. https://doi.org/10.46604/ijeti.2021.8244

Burak Gülmez. (2024). Advancements in rice disease detection through convolutional neural networks: A comprehensive review. *Heliyon*, *10*(12), e33328–e33328. https://doi.org/10.1016/j.heliyon.2024.e33328

Byra, M., Styczynski, G., Cezary Szmigielski, Kalinowski, P., Michalowski, L., Rafal Paluszkiewicz, Bogna Ziarkiewicz-Wroblewska, Krzysztof Zieniewicz, & Nowicki, A. (2020). Adversarial attacks on deep learning models for fatty liver disease classification by modification of ultrasound image reconstruction method. *ArXiv (Cornell University)*.

https://doi.org/10.1109/ius46767.2020.9251568

Carvajal-Yepes, M., Olaya, C., Lozano, I., Cuervo, M., Castaño, M., & Cuellar, W. J. (2014). Unraveling complex viral infections in cassava (Manihot esculenta Crantz) from Colombia. *Virus Research*, *186*, 76–86. https://doi.org/10.1016/j.virusres.2013.12.011

Ch, E., K. Revanth Kumar, Sai, A., Munagala Ruchita, & Manob Jyoti Saikia. (2024). Signal to Image Conversion and Convolutional Neural Networks for Physiological Signal Processing: A Review. *IEEE Access*, *12*, 66726–66764. https://doi.org/10.1109/access.2024.3399114

Chang, H.-H., Chiang, J.-H., Tsai, C.-C., & Chiu, P.-F. (2023). Predicting hyperkalemia in patients with advanced chronic kidney disease using the XGBoost model. *BMC Nephrology*, *24*(1). https://doi.org/10.1186/s12882-023-03227-w

Chatterjee, D. (2019). The Rise of Deep Learning in Radiology: An Overview of Recent Research. *International Journal for Research in Applied Science and Engineering Technology*, *7*(6), 2353–2361. https://doi.org/10.22214/ijraset.2019.6397

Chavarriaga-Aguirre, P., Brand, A., Medina, A., Prías, M., Escobar, R., Martinez, J., Díaz, P., López, C., Roca, W. M., & Tohme, J. (2016). The potential of using biotechnology to improve cassava: a review. *In Vitro Cellular & Developmental Biology - Plant*, *52*(5), 461–478. https://doi.org/10.1007/s11627-016-9776-3

Colvin, J., Omongo, C. A., Maruthi, M. N., Otim-Nape, G. W., & Thresh, J. M. (2004). Dual begomovirus infections and high Bemisia tabaci populations: two factors driving the spread of a cassava mosaic disease pandemic. *Plant Pathology*, *53*(5), 577–584. https://doi.org/10.1111/j.0032-0862.2004.01062.x

Dara, R., Hazrati Fard, S. M., & Kaur, J. (2022). Recommendations for ethical and responsible use of artificial intelligence in digital agriculture. *Frontiers in Artificial Intelligence*, *5*. https://doi.org/10.3389/frai.2022.884192

Das, S., Kaur, M., Chhabra, V., Nandi, T., Mishra, P., & Ghosh, S. (2024). A Systematic Review of Artificial Intelligence: A Future Guide to Sustainable Agriculture. *International Journal of Environment and Climate Change*, *14*(4), 562–573. https://doi.org/10.9734/ijecc/2024/v14i44139

Dharani, M. K., R. Thamilselvan, Gudadhe, S. P., Manasi Arvindrao Joshi, & Yadav, V. (2022). Leaf Disease Detection using Deep Learning Models. *2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)*. https://doi.org/10.1109/ictacs56270.2022.9988660

Dheiver Francisco Santos. (2023). XGBoosting Early Detection: Advancing Parkinson’s Disease Diagnosis through Machine Learning. *MedRxiv (Cold Spring Harbor Laboratory)*. https://doi.org/10.1101/2023.10.23.23297368

Ding, B. (2022). Cassava Leaf Disease Classification Based On CNN. *Highlights in Science, Engineering and Technology*, *16*, 63–69. https://doi.org/10.54097/hset.v16i.2363

Dr Subhadra P.S, Dr. A. Kalaivani, Dr. Rohit Markan, & Ramesh. (2024). Rise of Artificial Intelligence in Business and Industry. *Journal of Informatics Education and Research*, *4*(2). https://doi.org/10.52783/jier.v4i2.850

Eftekhari, N., Pourreza, H.-R., Masoudi, M., Ghiasi-Shirazi, K., & Saeedi, E. (2019). Microaneurysm detection in fundus images using a two-step convolutional neural network. *BioMedical Engineering OnLine*, *18*(1). https://doi.org/10.1186/s12938-019-0675-9

Elliott, K., Berry, J. C., Kim, H., & Bart, R. (2022). *A comparison of ImageJ and machine learning based image analysis methods to measure cassava bacterial blight disease severity*. *18*(1). https://doi.org/10.1186/s13007-022-00906-x

Esperance Munganyinka, Paolo Margaria, Samar Sheat, Ateka, E. M., Tairo, F., Ndunguru, J., & Winter, S. (2018). Localization of cassava brown streak virus in Nicotiana rustica and cassava Manihot esculenta (Crantz) using RNAscope® in situ hybridization. *Virology Journal*, *15*(1). https://doi.org/10.1186/s12985-018-1038-z

Evaggelia Arapostathi, Panopoulou, C., Antonopoulos, A., Anastasios Katsileros, Konstantinos Karellas, Dimopoulos, C., & Antonios Tsagkarakis. (2023). Early Detection of Potential Infestation by Capnodis tenebrionis (L.) (Coleoptera: Buprestidae), in Stone and Pome Fruit Orchards, Using Multispectral Data from a UAV. *Agronomy*, *14*(1), 20–20. https://doi.org/10.3390/agronomy14010020

G V S Sree Santosh, G Chaitanya Kumar, G Sandeep, & Satish, E. (2022). German Traffic Sign Recognition Using Convolutional Neural Network. *2022 6th International Conference on Electronics, Communication and Aerospace Technology*. https://doi.org/10.1109/iceca55336.2022.10009588

G. Arulkumaran, B. Gopi, Khan, V., & R. Parameshwaran. (2022). Deep convolutional neural network-based cassava plant leaf disease classification. *AIP Conference Proceedings*. https://doi.org/10.1063/5.0110659

Garifulla, M., Shin, J., Kim, C., Kim, W. H., Kim, H. J., Kim, J., & Hong, S. (2021). A Case Study of Quantizing Convolutional Neural Networks for Fast Disease Diagnosis on Portable Medical Devices. *Sensors*, *22*(1), 219. https://doi.org/10.3390/s22010219

Glorot, X., Bordes, A., & Yoshua Bengio. (2011). Deep Sparse Rectifier Neural Networks. *PMLR*, 315–323. https://proceedings.mlr.press/v15/glorot11a.html

Godliver Owomugisha, Ephraim Nuwamanya, Quinn, J. A., Biehl, M., & Mwebaze, E. (2020). Early detection of plant diseases using spectral data. *University of Groningen Research Database (University of Groningen / Centre for Information Technology)*. https://doi.org/10.1145/3378184.3378222

Goh, K.-S., Edward Yi Chang, & Cheng, K.-T. (2001). *Support vector machine pairwise classifiers* *with error reduction for image classification*. https://doi.org/10.1145/500933.500944

Gomez, R. (2018, May 23). *Understanding Categorical Cross-Entropy Loss, Binary Cross-Entropy Loss, Softmax Loss, Logistic Loss, Focal Loss and all those confusing names*. https://gombru.github.io/2018/05/23/cross_entropy_loss/

Gou, W., Zhang, H., & Zhang, R. (2023). Multi-Classification and Tree-Based Ensemble Network for the Intrusion Detection System in the Internet of Vehicles. *Sensors*, *23*(21), 8788–8788. https://doi.org/10.3390/s23218788

Guo, S. (2023). Leaf Disease Detection by Convolutional Neural Network (CNN). *Highlights in Science Engineering and Technology*, *72*, 1141–1146. https://doi.org/10.54097/aex9r523

Gururaj Harinahalli Lokesh, Soundarya Bidare Chandregowda, Vishwanath, J., Ravi, V., Ravi, P., & Alanoud Al Mazroa. (2024). Intelligent Plant Leaf Disease Detection Using Generative Adversarial Networks: a Case-study of Cassava Leaves. *The Open Agriculture Journal*, *18*(1). https://doi.org/10.2174/0118743315288623240223072349

Habibu Mugerwa, Colvin, J., Alicai, T., Omongo, C. A., Kabaalu, R., Visendi, P., Sseruwagi, P., & Seal, S. (2021). Genetic diversity of whitefly (Bemisia spp.) on crop and uncultivated plants in Uganda: implications for the control of this devastating pest species complex in Africa. *Journal of Pest Science*, *94*(4), 1307–1330. https://doi.org/10.1007/s10340-021-01355-6

Hasan, T., Marjuk Ahmed Siddiki, & Md Naim Hossain. (2023). Detection of Bangladeshi-Produced Plant Disease Using a Transfer Learning Based on Deep Neural Model. *Journal of Computer Science and Technology Studies*, *5*(3), 55–69. https://doi.org/10.32996/jcsts.2023.5.3.6

Hauptman, A., Balasubramaniam, G. M., & Arnon, S. (2023). Machine Learning Diffuse Optical Tomography Using Extreme Gradient Boosting and Genetic Programming. *Bioengineering*, *10*(3), 382. https://doi.org/10.3390/bioengineering10030382

Hoeser, T., Bachofer, F., & Kuenzer, C. (2020). Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review—Part II: Applications. *Remote Sensing*, *12*(18), 3053. https://doi.org/10.3390/rs12183053

Hoeser, T., & Kuenzer, C. (2020). Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review-Part I: Evolution and Recent Trends. *Remote Sensing*, *12*(10), 1667. https://doi.org/10.3390/rs12101667

Hong, H., Du, R., Wang, Z., Li, X., & Yuan, G. (2023). A Malicious Code Detection Method Based on Stacked Depthwise Separable Convolutions and Attention Mechanism. *Sensors*, *23*(16), 7084–7084. https://doi.org/10.3390/s23167084

Horowitz, A. R., Antignus, Y., & Gerling, D. (2011). Management of Bemisia tabaci Whiteflies. *The Whitefly, Bemisia Tabaci (Homoptera: Aleyrodidae) Interaction with Geminivirus-Infected Host Plants*, 293–322. https://doi.org/10.1007/978-94-007-1524-0_11

Hou, N., Li, M., He, L., Xie, B., Wang, L., Zhang, R., Yu, Y., Sun, X., Pan, Z., & Wang, K. (2020). Predicting 30-days mortality for MIMIC-III patients with sepsis-3: a machine learning approach using XGboost. *Journal of Translational Medicine*, *18*(1). https://doi.org/10.1186/s12967-020-02620-5

Houngue, J. A., Pita, J. S., Ngalle, H. B., Zandjanakou-Tachin, M., Kuate, A. F., Cacaï, G. H. T., Bell, J. M., & Ahanhanzo, C. (2019). Response of cassava cultivars to African cassava mosaic virus infection across a range of inoculum doses and plant ages. *PLOS ONE*, *14*(12), e0226783. https://doi.org/10.1371/journal.pone.0226783

Huang, Y., Guo, J., Chen, W.-H., Lin, H.-Y., Tang, H., Wang, F., Xu, H., & Bian, J. (2024). A scoping review of fair machine learning techniques when using real-world data. *MedRxiv (Cold Spring Harbor Laboratory)*. https://doi.org/10.1101/2024.03.03.24303669

Hussein Mohammed Essa, & Murshid, A. M. (2023). Optimizing Image Processing with CNNs through Transfer Learning: Survey. *Mağallaẗ Al-Kitāb Li-l-ʿulūm Al-Şirfaẗ*, *7*(1), 57–68. https://doi.org/10.32441/kjps.07.01.p6

Hyperparameter Optimization for Transfer Learning-based Disease Detection in Cassava Plants. (2023). *Journal of Scientific & Industrial Research*, *82*(05). https://doi.org/10.56042/jsir.v82i05.1089

Innamorati, C., Ritschel, T., Weyrich, T., & Mitra, N. J. (2019). Learning on the Edge: Investigating Boundary Filters in CNNs. *International Journal of Computer Vision*, *128*(4), 773–782. https://doi.org/10.1007/s11263-019-01223-y

Intan Nurma Yulita, Firman Ardiansyah, Anton Satria Prabuwono, Muhammad Rasyid Ramdhani, Mokhamad Arfan Wicaksono, Agus Trisanto, & Asep Sholahuddin. (2023). Recyclable Waste Classification using SquezeeNet and XGBoost. *International Journal of Advanced Computer Science and Applications*, *14*(10). https://doi.org/10.14569/ijacsa.2023.0141037

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *JMLR.org*. https://doi.org/10.48550/arXiv.1502.03167

Jain, U. (2023). Detecting Parkinson's Disease Using XGBoost. *International Journal of Science and Research (IJSR)*, *12*(4), 1339–1343. https://doi.org/10.21275/sr23417234157

James Daniel Omaye, Emeka Ogbuju, Ataguba, G., Oluwayemisi Jaiyeoba, Aneke, J., & Oladipo, F. (2024). Cross-comparative review of Machine learning for plant disease detection: apple, cassava, cotton and potato plants. *Artificial Intelligence in Agriculture*. https://doi.org/10.1016/j.aiia.2024.04.002

Jiao, W., Hao, X., & Qin, C. (2021). The Image Classification Method with CNN-XGBoost Model Based on Adaptive Particle Swarm Optimization. *Information*, *12*(4), 156. https://doi.org/10.3390/info12040156

Jing, H., Lin, G., Zhang, H., & Chen, T. (2022). A Face Recognition Algorithm Based on Improved Resnet. *Frontiers in Computing and Intelligent Systems*, *1*(1), 22–25. https://doi.org/10.54097/fcis.v1i1.1100

Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0192-5

Jwo, D.-J., & Chiu, S.-F. (2022). Deep Learning Based Automated Detection of Diseases from Apple Leaf Images. *Computers, Materials & Continua*, *71*(1), 1849–1866. https://doi.org/10.32604/cmc.2022.021875

K E, S. (n.d.). *Convolutional Neural Network | Deep Learning*. https://developersbreach.com/convolution-neural-network-deep-learning/

K. Thaiyalnayaki, S. Raghul, U. Karan kumar, & Ramachandran, S. (2022). *Automatic classification of cassava using data augmentation and CNN*. https://doi.org/10.1063/5.0072729

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, *147*, 70–90. https://doi.org/10.1016/j.compag.2018.02.016

Kathuria, A. (2018, June 13). *Intro to optimization in deep learning: Momentum, RMSProp and Adam*. Paperspace Blog; Paperspace Blog. https://blog.paperspace.com/intro-to-optimization-momentum-rmsprop-adam/

Kawano, K., Yoichiro Otaki, Suzuki, N., Fujimoto, S., Kunitoshi Iseki, Moriyama, T., Yamagata, K., Kazuhiko Tsuruya, Narita, I., Kondo, M., Yugo Shibagaki, Kasahara, M., Koichi Asahi, Watanabe, T., & Tsuneo Konta. (2022). Prediction of mortality risk of health checkup participants using machine learning-based models: the J-SHC study. *Scientific Reports*, *12*(1). https://doi.org/10.1038/s41598-022-18276-8

Kennedy Okokpujie, Okokpujie, I. P., Ayomikun, O. I., Abidemi Orimogunje, & Ogundipe, A. T.

(2023). Development of a Web and Mobile Applications-Based Cassava Disease Classification Interface Using Convolutional Neural Network. *Mathematical Modelling of Engineering Problems*, *10*(1), 119–128. https://doi.org/10.18280/mmep.100113

Kingma, D. P., & Ba, J. (2014, December 22). *Adam: A Method for Stochastic Optimization*. ArXiv.org. https://arxiv.org/abs/1412.6980

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, *60*(6), 84–90.

Kumar Laxmikant, R Bhuvaneswari, & Natarajan, B. (2023). *An Efficient Approach to Detect Diabetes using XGBoost Classifier*. https://doi.org/10.1109/wisscon56857.2023.10133854

Kumar, J., Chawla, R., Drishty Katiyar, Arjun Chouriya, Nath, D., Sahoo, S., Ali, A. H., & Singh, B. (2023). Optimizing Irrigation and Nutrient Management in Agriculture through Artificial Intelligence Implementation. *International Journal of Enviornment and Climate Change*, *13*(10), 4016–4022. https://doi.org/10.9734/ijecc/2023/v13i103077

Kusumo, B. S., Heryana, A., Krisnandi, D., Yuwana, S., Zilvan, V., & Pardede, H. F. (2020). Deep Convolutional Neural Networks-Based Plants Diseases Detection Using Hybrid Features. *Computer Engineering and Applications Journal*, *9*(3), 215–225. https://doi.org/10.18495/comengapp.v9i3.346

LeCun, Y., Bottou, L., Orr, G. B., & Müller, K.-R. (1998). Efficient BackProp. *Lecture Notes in Computer Science*, 9–50. https://doi.org/10.1007/3-540-49430-8_2

Li, B., Ai, D., & Liu, X. (2022). CNN-XG: A Hybrid Framework for sgRNA On-Target Prediction. *Biomolecules*, *12*(3), 409. https://doi.org/10.3390/biom12030409

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). Focal Loss for Dense Object Detection. *2017 IEEE International Conference on Computer Vision (ICCV)*.

https://doi.org/10.1109/iccv.2017.324

Lodhi, S., None Sakshi, & Kukreja, V. (2022). Deep Learning Based Method To Detect Diseases In Leaves Of Cassava Plant. *2022 International Conference on Data Analytics for Business and Industry (ICDABI)*. https://doi.org/10.1109/icdabi56818.2022.10041484

Maeta, K., Nishiyama, Y., Fujibayashi, K., Gunji, T., Sasabe, N., Iijima, K., & Naito, T. (2018). Prediction of Glucose Metabolism Disorder Risk Using a Machine Learning Algorithm: Pilot Study. *JMIR Diabetes*, *3*(4), e10212. https://doi.org/10.2196/10212

Mandar Sapre, Jatti, V. S., Tiwari, P., Niranjan Kodachakki, & Atharva Undale. (2023). *Disease Classification in Cassava Plant by Artificial Neural Network*. 75–84. https://doi.org/10.1007/978-981-99-7456-6_6

Maruthi, M. N., Whitfield, E. C., Otti, G., Tumwegamire, S., Kanju, E., Legg, J. P., Mkamilo, G., Kawuki, R., Benesi, I., Zacarias, A., Munga, T., Mwatuni, F., & Mbugua, E. (2019). A method for generating virus-free cassava plants to combat viral disease epidemics in Africa. *Physiological and Molecular Plant Pathology*, *105*, 77–87. https://doi.org/10.1016/j.pmpp.2018.09.002

McKinney, W. (2010). Data Structures for Statistical Computing in Python. *Proceedings of the 9th Python in Science Conference*, *445*. https://doi.org/10.25080/majora-92bf1922-00a

Mehta, S., Kukreja, V., & Gupta, R. (2023). *Decentralized Detection of Cassava Leaf Diseases: A Federated Convolutional Neural Network Solution*. https://doi.org/10.1109/iccpct58313.2023.10245357

Michael Olumuyiwa Adio, Ogunmakinde Jimoh Ogunwuyi, Mayowa Oyedepo Oyediran, Adebimpe Omolayo Esan, & Olufikayo Adepoju Adedapo. (2024). Performance Evaluation of Some Selected Classification Algorithms in a Facial Recognition System. *ABUAD Journal of Engineering Research and Development (AJERD)*, *7*(1), 169–177.

https://doi.org/10.53982/ajerd.2024.0701.17-j

Miftahus Sholihin, Mohd, & Mohd Norasri Ismail. (2023). AlexNet-Based Feature Extraction for Cassava Classification: A Machine Learning Approach. *Baghdad Science Journal*, *20*(6(Suppl.)), 2624–2624. https://doi.org/10.21123/bsj.2023.9120

Momeni, A., Thibault, M., & Olivier Gevaert. (2018). Dropout-Enabled Ensemble Learning for Multi-Scale Biomedical Data. *BioRxiv (Cold Spring Harbor Laboratory)*. https://doi.org/10.1101/440362

Monger, W. A., Seal, S., Cotton, S., & Foster, G. D. (2001). Identification of different isolates of Cassava brown streak virus and development of a diagnostic test. *Plant Pathology*, *50*(6), 768–775. https://doi.org/10.1046/j.1365-3059.2001.00647.x

Mrisho, L. M., Mbilinyi, N. A., Ndalahwa, M., Ramcharan, A. M., Kehs, A. K., McCloskey, P. C., Murithi, H., Hughes, D. P., & Legg, J. P. (2020). Accuracy of a Smartphone-Based Object Detection Model, PlantVillage Nuru, in Identifying the Foliar Symptoms of the Viral Diseases of Cassava-CMD and CBSD. *Frontiers in Plant Science*, *11*, 590889. https://doi.org/10.3389/fpls.2020.590889

Muhammad riki Atsauri, Mawengkang, H., & Syahril Efendi. (2023). Enhancing Unbalanced Data Classification with Cross-Validation and Extreme Gradient Boosting: A Comprehensive Analysis. *JOURNAL of INFORMATICS and TELECOMMUNICATION ENGINEERING*, *7*(1), 30–42. https://doi.org/10.31289/jite.v7i1.8690

N.R., L., S., N., C.H., D., M., H., & KS KS, C. (2019). A Review on Weed Detection Using Image Processing. *Indian Journal of Science and Technology*, *12*(48), 1–3. https://doi.org/10.17485/ijst/2019/v12i48/149410

Naufal, M. F., Kusuma, S. F., Prayuska, Z. A., Yoshua, A. A., Lauwoto, Y. A., Dinata, N. S., & Sugiarto,

D. (2021). Comparative Analysis of Image Classification Algorithms for Face Mask Detection. *Journal of Information Systems Engineering and Business Intelligence*, *7*(1), 56–66. https://doi.org/10.20473/jisebi.7.1.56-66

Nerkar, B., & Talbar, S. (2020). Fusing Convolutional Neural Networks to Improve the Accuracy of Plant Leaf Disease Classification. *Current Journal of Applied Science and Technology*, 9–19. https://doi.org/10.9734/cjast/2020/v39i3931099

Neutelings, I. (n.d.). *Neural networks*. TikZ.net. https://tikz.net/neural_networks/

Nguyen, K., Nguyen, M., Dang, K., Pham, B., Huynh, V., Vo, T., Ngo, L., & Ha, H. (2023). Early Alzheimer's disease diagnosis using an XG-Boost model applied to MRI images. *Biomedical Research and Therapy*, *10*(9), 5896–5911. https://doi.org/10.15419/bmrat.v10i9.832

Nissar, I., Rizvi, D., Masood, S., & Mir, A. (2019). Voice-Based Detection of Parkinson's Disease through Ensemble Machine Learning Approach: A Performance Study. *EAI Endorsed Transactions on Pervasive Health and Technology*, *5*(19), 162806. https://doi.org/10.4108/eai.13-7-2018.162806

Nitish Srivastava, Hinton, G. E., Krizhevsky, A., & Ruslan Salakhutdinov. (2014, June). *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*. ResearchGate; unknown. https://www.researchgate.net/publication/286794765_Dropout_A_Simple_Way_to_Prevent_Neural_Networks_from_Overfitting

Odipio, J., Ogwok, E., Taylor, N. J., Halsey, M., Bua, A., Fauquet, C. M., & Alicai, T. (2013). RNAi-derived field resistance to Cassava brown streak disease persists across the vegetative cropping cycle. *GM Crops & Food*, *5*(1), 16–19. https://doi.org/10.4161/gmcr.26408

Ogbonna, A. C., Ramu, P., Esuma, W., Nandudu, L., Morales, N., Powell, A., Kawuki, R., Bauchet, G., Jannink, J.-L., & Mueller, L. A. (2021). A population based expression atlas provides insights into disease resistance and other physiological traits in cassava (Manihot esculenta Crantz). *Scientific Reports*, *11*(1). https://doi.org/10.1038/s41598-021-02794-y

Onah, I. S., Aniaku, S. E., & Ezugorie, O. M. (2022). Analysis and optimal control measures of diseases in cassava population. *Optimal Control Applications and Methods*, *43*(5), 1450–1478. https://doi.org/10.1002/oca.2901

Oyewola, D. O., Dada, E. G., Misra, S., & Damaševičius, R. (2021). Detecting cassava mosaic disease using a deep residual convolutional neural network with distinct block processing. *PeerJ Computer Science*, *7*, e352. https://doi.org/10.7717/peerj-cs.352

Pai, A., Atharv Raotole, Sailee Shirodkar, Bose, S., & Kolekar, M. H. (2023). *From Pixels to Prognosis: Exploring Convolutional Neural Networks for Cassava Leaf Disease Diagnosis*. https://doi.org/10.1109/ocit59427.2023.10430882

Panigrahi, K. P., Sahoo, A. K., & Das, H. (2020, June 1). *A CNN Approach for Corn Leaves Disease Detection to support Digital Agricultural System*. IEEE Xplore. https://doi.org/10.1109/ICOEI48184.2020.9142871

Pardo, J. M., Alvarez, E., Becerra Lopez-Lavalle, L. A., Olaya, C., Leiva, A. M., & Cuellar, W. J. (2022). Cassava Frogskin Disease: Current Knowledge on a Re-Emerging Disease in the Americas. *Plants*, *11*(14), 1841. https://doi.org/10.3390/plants11141841

Pardo, J. M., Chittarath, K., Vongphachanh, P., Hang, L. T., Oeurn, S., Arinaitwe, W., Rodriguez, R., Sophearith, S., Malik, A. I., & Cuellar, W. J. (2023). Cassava Witches' Broom Disease in Southeast Asia: A Review of Its Distribution and Associated Symptoms. *Plants*, *12*(11), 2217. https://doi.org/10.3390/plants12112217

Park, J. E., Kickingereder, P., & Kim, H. S. (2020). Radiomics and Deep Learning from Research to Clinical Workflow: Neuro-Oncologic Imaging. *Korean Journal of Radiology*, *21*(10), 1126. https://doi.org/10.3348/kjr.2019.0847

Pettit, R. W., Marlatt, B. B., Miles, T. J., Uzgoren, S., Corr, S. J., Shetty, A., Havelka, J., & Rana, A. (2023). The utility of machine learning for predicting donor discard in abdominal transplantation. *Clinical Transplantation*, *37*(5), e14951. https://doi.org/10.1111/ctr.14951

Pham, C., & Hong Nhung Tran. (2021). *Cunninghamella elegans causing cassava root rot in Vietnam*. *16*(1). https://doi.org/10.1007/s13314-021-00427-x

Plata, I. T., Panganiban, E. B., Alado, D. B., Taracatac, A. C., Bartolome, B. B., & Labuanan, F. R. E. (2022a). Comparative Performance Analysis of Real-Time Methods for Cassava Phytoplasma Disease (CPD) Detection based on Deep Learning Neural Networks. *International Journal of Emerging Technology and Advanced Engineering*, *12*(2), 55–64. https://doi.org/10.46338/ijetae0222_07

Plata, I. T., Panganiban, E. B., Alado, D. B., Taracatac, A. C., Bartolome, B. B., & Labuanan, F. R. E. (2022b). Drone-based Geographical Information System (GIS) Mapping of Cassava Pythoplasma Disease (CPD) for Precision Agriculture. *International Journal of Emerging Technology and Advanced Engineering*, *12*(2), 1–9. https://doi.org/10.46338/ijetae0222_01

Polat, H., & Danaei Mehr, H. (2019). Classification of Pulmonary CT Images by Using Hybrid 3D-Deep Convolutional Neural Network Architecture. *Applied Sciences*, *9*(5), 940. https://doi.org/10.3390/app9050940

Powers, D. M. W. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *ArXiv:2010.16061 [Cs, Stat]*. https://arxiv.org/abs/2010.16061

Pranav Ranaware, & Wagh, M. (2023). A HYBRID ENSEMBLE APPROACH FOR PLANT DISEASE DETECTION AND CLASSIFICATION. *International Research Journal of Modernization in*

Engineering Technology and Science. https://doi.org/10.56726/irjmets41756

Qiu, M. (2024). Art Image Attribute Classification Based on Support Vector Machine. *Advances in Transdisciplinary Engineering*. https://doi.org/10.3233/atde231231

Quy Thanh Lu. (2023). An Approach for Classification of Diseases on Leaves. *International Journal of Advanced Computer Science and Applications*, *14*(10). https://doi.org/10.14569/ijacsa.2023.01410112

R. Aruna, M Prabu, S. Ananthi, V.C. Bharathi, R. Sathya, & B. Suchithra. (2023). *Vision based Cassava Plant Leaf Disease Classification using Machine Learning Techniques*. https://doi.org/10.1109/icacrs58579.2023.10404468

Rajasree R, Christalin, B., Paul, S., Appu M, & Aswathy N. (2023). *An optimized Faster R-CNN model for Cassava Brown Streak Disease Classification*. https://doi.org/10.1109/access57397.2023.10200536

Raju, L., Sai, V., Korada Harika, A. Dharani, Satish, K., M. Lavanya, & G. Tharun. (2024). Parkinson's Disease Detection Using Machine Learning. *International Journal of Research Publication and Reviews*, *5*(4), 1814–1823. https://doi.org/10.55248/gengpi.5.0424.0946

Ramaneswaran, S., Srinivasan, K., Vincent, P. M. D. R., & Chang, C.-Y. (2021). Hybrid Inception v3 XGBoost Model for Acute Lymphoblastic Leukemia Classification. *Computational and Mathematical Methods in Medicine*, *2021*, 1–10. https://doi.org/10.1155/2021/2577375

Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J., & Hughes, D. P. (2017). Deep Learning for Image-Based Cassava Disease Detection. *Frontiers in Plant Science*, *8*. https://doi.org/10.3389/fpls.2017.01852

Ramcharan, A., McCloskey, P., Baranowski, K., Mbilinyi, N., Mrisho, L., Ndalahwa, M., Legg, J., & Hughes, D. P. (2019). A Mobile-Based Deep Learning Model for Cassava Disease Diagnosis. *Frontiers in Plant Science*, *10*. https://doi.org/10.3389/fpls.2019.00272

Ravi, V., Acharya, V., & Pham, T. D. (2021). Attention deep learning-based large-scale learning classifier for Cassava leaf disease classification. *Expert Systems*. https://doi.org/10.1111/exsy.12862

Sahoo, L., Mohapatra, D., Raghuvanshi, H. R., Kumar, S., Kaur, R., Anshika, Sapna, Chawla, R., & Afreen, N. (2024). Transforming Agriculture through Artificial Intelligence: Advancements in Plant Disease Detection, Applications, and Challenges. *Journal of Advances in Biology & Biotechnology*, *27*(5), 381–388. https://doi.org/10.9734/jabb/2024/v27i5796

Sai Swaroop Reddy, Khan, I., & Kanchana M. (2024). Plant Disease Detection Using Deep-Learning. *Research Square (Research Square)*. https://doi.org/10.21203/rs.3.rs-4374075/v1

Sat Byul Seo, & Cho, H. (2022). Deep learning classification of early normal-tension glaucoma and glaucoma suspect eyes using Bruch's membrane opening-based disc photography. *Frontiers in Medicine*, *9*. https://doi.org/10.3389/fmed.2022.1037647

Shamshad, F., Khan, S., Zamir, S. W., Khan, M. H., Hayat, M., Khan, F. S., & Fu, H. (2022, January 24). *Transformers in Medical Imaging: A Survey*. ArXiv.org. https://doi.org/10.48550/arXiv.2201.09873

Shao, S., Fu, D., Yang, T., Mu, H., Gao, Q., & Zhang, Y. (2023). Analysis of Machine Learning Models for Wastewater Treatment Plant Sludge Output Prediction. *Sustainability*, *15*(18), 13380–13380. https://doi.org/10.3390/su151813380

Shoaib, M., Shah, B., Hussain, T., Ali, A., Ullah, A., Alenezi, F., Gechev, T., Ali, F., & Syed, I. (2022). A deep learning-based model for plant lesion segmentation, subtype identification, and survival probability estimation. *Frontiers in Plant Science*, *13*. https://doi.org/10.3389/fpls.2022.1095547

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep

Learning. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0197-0

Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *Computer Science*. https://doi.org/10.48550/arXiv.1409.1556

Singh, R., Sharma, A., Sharma, N., Sharma, K., & Gupta, R. (2023). A Deep Learning-Based InceptionResNet V2 Model for Cassava Leaf Disease Detection. *Lecture Notes in Networks and Systems*, 423–432. https://doi.org/10.1007/978-981-99-1946-8_38

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Song, X., Li, H., Chen, Q., Zhang, T., Huang, G., Zou, L., & Du, D. (2023). Predicting pneumonia during hospitalization in flail chest patients using machine learning approaches. *Frontiers in Surgery*, *9*. https://doi.org/10.3389/fsurg.2022.1060691

Sriram G, Vignesh M, D, S., & Ayyasamy S. (2023). Flourishing Fields: Revolutionizing Agriculture with Soft Computing-based Plant Disease Detection. *Research Square (Research Square)*. https://doi.org/10.21203/rs.3.rs-2859449/v1

Stubblefield, J., Hervert, M., Causey, J., Qualls, J., Dong, W., Cai, L., Fowler, J., Bellis, E. S., Walker, K., Moore, J. H., Nehring, S. M., & Huang, X. (2020). *Cardiac or Infectious? Transfer Learning with Chest X-Rays for ER Patient Classification*. https://doi.org/10.1101/2020.04.11.20062091

Syed Mursleen Riaz, Ahsan, M., & Muhammad Usman Akram. (2022). *Diagnosis Of Cassava Leaf Diseases and Classification Using Deep Learning Techniques*. https://doi.org/10.1109/icosst57195.2022.10016854

Tomlinson, K. R., Bailey, A. M., Alicai, T., Seal, S., & Foster, G. D. (2017). Cassava brown streak disease: historical timeline, current knowledge and future prospects. *Molecular Plant*

Pathology, *19*(5), 1282–1294. https://doi.org/10.1111/mpp.12613

Torkpo, S. K., Amponsah, E., Arhin, C. D., & Offei, S. K. (2021). Occurrence of cassava mosaic begomovirus-associated satellites on cassava in Ghana. *Cogent Food & Agriculture*, *7*(1), 1963929. https://doi.org/10.1080/23311932.2021.1963929

Ummadi Sreeja. (2024). AUTOMATED CLASSIFICATION OF EMPHYSEMA USING CONVOLUTIONAL NEURAL NETWORKS. *INTERANTIONAL JOURNAL of SCIENTIFIC RESEARCH in ENGINEERING and MANAGEMENT*, *08*(06), 1–5. https://doi.org/10.55041/ijsrem35769

Varshney, D., Babukhanwala, B., Khan, J., Saxena, D., & Singh, A. kumar. (2021, June 1). *Machine Learning Techniques for Plant Disease Detection*. IEEE Xplore. https://doi.org/10.1109/ICOEI51242.2021.9453053

Vasileios Balafas, Emmanouil Karantoumanis, Malamati Louta, & Nikolaos Ploskas. (2023). Machine Learning and Deep Learning for Plant Disease Classification and Detection. *IEEE Access*, *11*, 114352–114377. https://doi.org/10.1109/access.2023.3324722

Venkatasubbu, S., & Krishnamoorthy, G. (2022). Ethical Considerations in AI Addressing Bias and Fairness in Machine Learning Models. *Journal of Knowledge Learning and Science Technology ISSN: 2959-6386 (Online)*, *1*(1), 130–138. https://doi.org/10.60087/jklst.vol1.n1.p138

Vetal, S., & R.S., K. (2017). Tomato Plant Disease Detection using Image Processing. *IJARCCE*, *6*(6), 293–297. https://doi.org/10.17148/ijarcce.2017.6651

Vinod Nair, & Hinton, G. E. (2010). *Rectified Linear Units Improve Restricted Boltzmann Machines*. Undefined. https://www.semanticscholar.org/paper/Rectified-Linear-Units-Improve-Restricted-Boltzmann-Nair-Hinton/a538b05ebb01a40323997629e171c91aa28b8e2f

Wagner, M. W., Namdar, K., Biswas, A., Monah, S., Khalvati, F., & Ertl-Wagner, B. B. (2021). Radiomics, machine learning, and artificial intelligence—what the neuroradiologist needs to know. *Neuroradiology*. https://doi.org/10.1007/s00234-021-02813-9

Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., & Xu, W. (2016, June 1). *CNN-RNN: A Unified Framework for Multi-label Image Classification*. IEEE Xplore. https://doi.org/10.1109/CVPR.2016.251

Wang, K., Tian, Y., Liu, S., Zhang, Z., Shen, L., Meng, D., & Li, J. (2022). Risk Factors and Predictive Model for Dermatomyositis Associated with Rapidly Progressive Interstitial Lung Disease. *Pharmacogenomics and Personalized Medicine*, *Volume 15*, 775–783. https://doi.org/10.2147/pgpm.s369556

Wang, X., Chen, X., & Li, M. (2011). Classification of high spatial resolution remote sensing image using SVM and local spatial statistics Getis-Ord Gi. *Proceedings of SPIE, the International Society for Optical Engineering/Proceedings of SPIE*. https://doi.org/10.1117/12.901810

Weldegebriel, H. T., Liu, H., Haq, A. U., Bugingo, E., & Zhang, D. (2020). A New Hybrid Convolutional Neural Network and eXtreme Gradient Boosting Classifier for Recognizing Handwritten Ethiopian Characters. *IEEE Access*, *8*, 17804–17818. https://doi.org/10.1109/access.2019.2960161

Wicht, B. (2018, January). *(PDF) Deep Learning feature Extraction for Image Processing*. https://www.researchgate.net/publication/322505397_Deep_Learning_feature_Extraction_for_Image_Processing

Wu, X., Sahoo, D., & Hoi, S. C. H. (2020). Recent advances in deep learning for object detection. *Neurocomputing*. https://doi.org/10.1016/j.neucom.2020.01.085

Xu, Z., Zhang, Q., Xiang, S., Li, Y., Huang, X., Zhang, Y., Zhou, X., Li, Z., Yao, X., Li, Q., & Guo, X. (2022). Monitoring the Severity of Pantana phyllostachysae Chao Infestation in Moso Bamboo Forests Based on UAV Multi-Spectral Remote Sensing Feature Selection. *Forests*, *13*(3), 418. https://doi.org/10.3390/f13030418

Y Vijayalata, Nikhil Billakanti, Karthik Veeravalli, Ashlin, & Kota, L. (2022). Early Detection of Casava Plant Leaf Diseases using EfficientNet-B0. *2022 IEEE Delhi Section Conference (DELCON)*. https://doi.org/10.1109/delcon54057.2022.9753210

YADAV, J. S., OGWOK, E., WAGABA, H., PATIL, B. L., BAGEWADI, B., ALICAI, T., GAITAN-SOLIS, E., TAYLOR, N. J., & FAUQUET, C. M. (2011). RNAi-mediated resistance to Cassava brown streak Uganda virus in transgenic cassava. *Molecular Plant Pathology*, *12*(7), 677–687. https://doi.org/10.1111/j.1364-3703.2010.00700.x

Yadav, R., Pandey, M., & Santosh Kumar Sahu. (2022). Cassava plant disease detection with imbalanced dataset using transfer learning. *2022 IEEE World Conference on Applied Intelligence and Computing (AIC)*. https://doi.org/10.1109/aic55036.2022.9848882

Yadi Sitinjak, Muhaymin, & Marlince Nababan. (2023). LIVER DISEASE CLASSIFICATION ANALYSIS USING THE XGBOOST METHOD. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima(JUSIKOM PRIMA)*, *7*(1), 132–141. https://doi.org/10.34012/jurnalsisteminformasidanilmukomputer.v7i1.4130

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional Neural networks: an Overview and Application in Radiology. *Insights into Imaging*, *9*(4), 611–629. https://doi.org/10.1007/s13244-018-0639-9

Yan, Z., & Yang, Y. (2014). Application of ECOC SVMS in Remote Sensing Image Classification. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XL-2*, 191–196. https://doi.org/10.5194/isprsarchives-xl-2-191-2014

Yang, S., Cao, L., Zhou, Y., & Hu, C. (2023). A Retrospective Cohort Study: Predicting 90-Day Mortality for ICU Trauma Patients with a Machine Learning Algorithm Using XGBoost Using MIMIC-III Database. *Journal of Multidisciplinary Healthcare*, *Volume 16*, 2625–

2640. https://doi.org/10.2147/jmdh.s416943

Yogesh, al. (2023). Machine and Deep Learning Approaches for Plant Disease Detection: A Comprehensive Review. *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(11s), 745–757. https://doi.org/10.17762/ijritcc.v11i11s.10094

Yoo, G., Kim, H., & Hong, S. (2023). Prediction of Cognitive Load from Electroencephalography Signals Using Long Short-Term Memory Network. *Bioengineering*, *10*(3), 361. https://doi.org/10.3390/bioengineering10030361

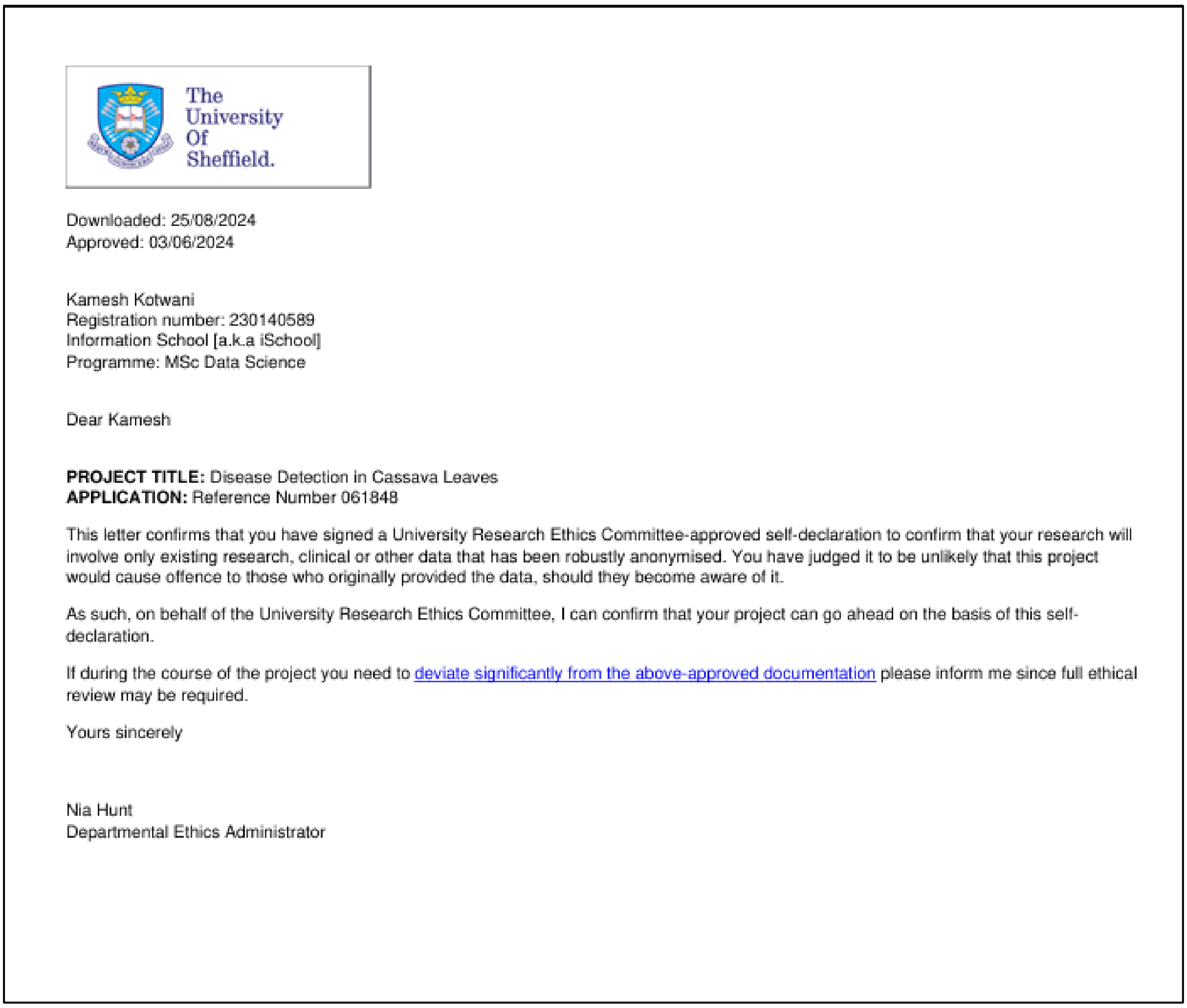
Zafar, A., Aamir, M., Mohd Nawi, N., Arshad, A., Riaz, S., Alruban, A., Dutta, A. K., & Almotairi, S. (2022). A Comparison of Pooling Methods for Convolutional Neural Networks. *Applied Sciences*, *12*(17), 8643. https://doi.org/10.3390/app12178643

Zhang, L., & Weng, Y. (2010). *Optical Flow Analysis Based on Cellular Neural Networks*. https://doi.org/10.1109/iptc.2010.78

Zhou, M., & Lai, W. (2023). Coal gangue recognition based on spectral imaging combined with XGBoost. *PLOS ONE*, *18*(1), e0279955. https://doi.org/10.1371/journal.pone.0279955

8. Appendix

8.1 Ethics Approval Form:



The
University
Of
Sheffield.

Downloaded: 25/08/2024
Approved: 03/06/2024

Kamesh Kotwani
Registration number: 230140589
Information School [a.k.a iSchool]
Programme: MSc Data Science

Dear Kamesh

PROJECT TITLE: Disease Detection in Cassava Leaves
APPLICATION: Reference Number 061848

This letter confirms that you have signed a University Research Ethics Committee-approved self-declaration to confirm that your research will involve only existing research, clinical or other data that has been robustly anonymised. You have judged it to be unlikely that this project would cause offence to those who originally provided the data, should they become aware of it.

As such, on behalf of the University Research Ethics Committee, I can confirm that your project can go ahead on the basis of this self-declaration.

If during the course of the project you need to deviate significantly from the above-approved documentation please inform me since full ethical review may be required.

Yours sincerely

Nia Hunt
Departmental Ethics Administrator

8.2 Dataset Link: Cassava Leaf Disease Classification | Kaggle