Ethiopian Soybeans Seed Type Classification Using Deep Learning

Haimanot Kiber Temesgen

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfilment of the Requirement for the Degree of Masters' in Computer Science and Engineering (Specialization in Data Science)

Office of Graduate Studies
Adama Science and Technology University

June, 2024

Adama, Ethiopia

Ethiopian Soybeans Seed Type Classification Using Deep Learning

By

Haimanot Kiber Temesgen

Advisor: Dr. Getinet Yilma

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfilment of the Requirement for the Degree of Masters' in Computer Science and Engineering (Specialization in Data Science)

Office of Graduate Studies

Adama Science and Technology University

June, 2024

Adama, Ethiopia

DECLARATION

I hereby declare that this Master thesis entitled *“Ethiopian Soybeans Seed Type Classification Using Deep Learning”* is my original work. That is, it has not been submitted for the award of any academic degree, diploma or certificate in any other university. All the sources of materials that are used for this thesis have been duly acknowledged through citation.

Haimanot Kiber Temesgen	________________	________________
Name of Student	Signature	Date

RECOMMENDATION

I, the major advisor of this thesis, hereby certify that I have read the revised version of thesis *"Ethiopian Soybeans Seed Type Classification Using Deep Learning"* prepared under our guidance by Haimanot Kiber Temesgen submitted in partial fulfilment of the requirements for the degree of Master's of Science in Computer Science and Engineering. Therefore, I recommend the submission of revised version of the thesis to the department following the applicable procedures.

__________________________	_______________________	________________
Major Advisor	Signature	Date

APPROVAL PAGE

I the advisor of the thesis entitled "*Ethiopian Soybeans Seed Type classification Using Deep Learning*" and developed by Haimanot Kiber Temesgen, hereby certify that the recommendation and suggestion made by the board of examiners are appropriately incorporated into the final version of the thesis.

Major Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Haimanot Kiber Temesgen have read and evaluated the thesis entitled *"Ethiopian Soybeans Seed Type Classification Using Deep Learning"* and examined the candidate during open defence. This is, therefore, to certify that the thesis is accepted for partial fulfilment of the requirement of the degree of Masters of Science in Computer science and Engineering.

Chairperson	Signature	Date
Internal Examiner	Signature	Date
External Examiner	Signature	Date

Final approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

Department Head	Signature	Date
School Dean	Signature	Date
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGMENT

It is my pleasure to express deepest gratitude to Dr. Getinet Yilma, who is very kind and generously, devoted much of his time and experienced in helping, guiding and advising me throughout this research work. Indeed, this work is the outcome of his great continuous efforts and wide experience in the field of Computer Engineering. He has been a constant presence throughout my study and was able to keep me motivated, challenged and productive at all time. I would also like to extend my heartfelt acknowledgments to my fellow graduate students who helped me during this research work. Lastly, I would like to thank my family and friends for their love and moral support through their prayers. I am deeply grateful to all of them. I am pent-up to the highest degree indebted to the Almighty God who gave me health and era to reach here with happiness and glory.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ARI	Agricultural Research Institute
CNN	Convolutional Neural Network
DL	Deep Learning
ECX	Ethiopian Commodity Exchange
EIAR	Ethiopian Institute of Agricultural Research
BARC	Bako Agricultural Research Center
KNN	K-means Nearest Neighbour
ML	Machine Learning
PNN	Probabilistic Neural Network
SVM	Support Vector Machine
TCNN	Temporal Convolutional Neural Network
VGG	Visual Geometry Group

ABSTRACT

Soybean, a crucial oil crop worldwide, is mainly grown for its protein-rich seeds containing high levels of both protein and oil. It ranks second in global oil consumption after palm oil, holding significant economic value as both a food commodity and a source of foreign exchange. In Ethiopia, the current method of identifying soybean seed varieties relies on manual classification by domain experts, a process that is time-consuming, labour-intensive, and prone to errors. Thus, there is a need for an efficient classification model to streamline the selection of soybean seeds.

This study aims to experimentally evaluate a proposed system for classifying soybean seed varieties using deep learning models. The dataset comprises 7,200 images representing eight different soybean varieties: Cheri, Jalale, Dhidhessa, Korme, Boshe, Katta, Ethio-Yugoslavia, and Gute. The images were divided into 80% for training and 20% for testing. Pre-processing techniques such as Median and Gaussian filtering were applied to remove noise introduced during image acquisition, and image augmentation was used to expand the dataset. Three deep learning models, Custom CNN, InceptionV3, and VGG16, were trained on both original and pre-processed images. The performance of each model was evaluated using metrics such as accuracy, recall, f1-score, precision, and confusion matrix, under various conditions. Custom CNN, InceptionV3 and VGG16 achieved accuracies rates of 100%, 98.25%and 99.39% respectively, across the different image sets. The overall better performance was achieved by the Custom CNN model with Median filtering, attaining an accuracy of 100%. This model is poised to significantly aid agricultural research institutes and the Ethiopian Commodity Exchange in swiftly and accurately identifying soybean varieties.

Keywords*: soybean seed varieties, deep learning, image classification, Gaussian filtering, Median filtering, Custom CNN, InceptionV3, VGG16*

CHAPATER ONE

INTRODUCTION

1.1 Background of the Study

Soybean is indeed one of the most important oil crops in the world, primarily cultivated for its oil and protein-rich seeds. It has gained significant popularity due to its versatile applications and nutritional value. While soybean production and consumption have experienced substantial growth in the western hemisphere, particularly in countries like the United States, Brazil, and Argentina, the output in Africa has generally lagged behind. However, there have been efforts to promote soybean cultivation and improve productivity in various African countries to meet the rising demand for oil and protein-rich food sources (Rehima Musema, Samuel Diro, Beza Erko, Desalegn Teshale, Regasa Dibaba, Abush Tesfaye 2022).

Soybeans are considered a multipurpose crop, making them suitable for both small-scale and large-scale farming. The increasing demand for soybean oil for cooking and food purposes, as well as its use in various industrial applications, contributes to its economic significance. Additionally, soybeans are an excellent source of protein, containing all nine essential amino acids, which makes them a valuable dietary component for both humans and animals.

The majority of soybean production, approximately 85%, is used for animal feed due to its high protein content. Livestock, such as poultry, pigs, and cattle, greatly benefit from soybean-based feed as it provides them with essential nutrients. The remaining portion of soybean production is utilized for direct human consumption through various forms like tofu, soy milk and other soy-based food products. The nutritional benefits, versatility, and economic value of soybeans have contributed to their prominence as an important crop worldwide, serving as a vital source of oil and protein for both human and animal consumption (Markett, 2022).

Soybeans are important due to their high protein and oil content. The protein content, exceeding 40%, makes them a valuable source of plant-based protein. Soybean oil, which contains no cholesterol, is widely used in food production and is the second most consumed oil globally after palm oil. However, the soybean industry faces challenges such as low yield, with diseases being a significant factor. Despite these challenges, soybeans continue to be a significant food crop globally, serving as a source of edible oil and livestock feed (Delele et al., 2022). To address the challenge of increasing food demand with limited agricultural land, strategies include developing improved seeds, adopting precision agriculture, promoting crop rotation and diversification, exploring vertical farming and urban agriculture, investing in research, enhancing infrastructure, and providing education and training for farmers. These measures aim to boost agricultural productivity sustainably, ensuring a balance between production and consumption in the face of a growing global population (Tesfay, 2023).

Seed variety classification is vital for seed producers and farmers to ensure the quality of seeds and optimize crop production. While visual inspection has been the traditional method, it has drawbacks including high error rates, imprecise results, and time-consuming procedures, particularly for similar species. To address these limitations, alternative techniques like machine learning, computer vision, and spectroscopy are being explored. These approaches offer benefits such as increased accuracy, faster processing times, and reduced dependence on subjective human judgment. By improving the classification process, these techniques can support seed producers and farmers in delivering high-quality seeds and meeting market demands (Nuthalapati et al., 2024).

Deep learning, a powerful technique for data categorization and prediction, can be effectively utilized to address various challenges (Dönmez, 2024; Sonmez et al., 2024; Yu et al., 2024). In the context of Ethiopian soybeans, deep learning can be employed to classify different properties of soybeans, enabling precise identification and optimization of agricultural practices for improved crop yields. Deep learning leverages computational models that consist of multiple layers to learn representations of data at different levels of abstraction. In agriculture, deep learning models can be applied in diverse ways to explore the agricultural characteristics of soybeans (Teodoro et al., 2024). The thesis aimed to classify eight varieties of Ethiopian soybean seeds using a deep learning approach. This method enables rapid, non-destructive, low-cost, and accurate classification based on external appearances. The use of CNNs allows the model to automatically learn relevant features from seed images, streamlining the classification process and potentially benefiting agricultural quality assessment and breeding programs (Tempola et al., 2024).

1.2 Motivation of the Study

Numerous studies have delved into the realms of soybean disease analysis, seed identification, and classification utilizing machine learning (ML) and deep learning techniques. This indicates the considerable importance of this field, as evidenced by its continuous research attention and ongoing status as a significant research problem. Despite this, there remains a noticeable gap in research pertaining to the classification of Ethiopian soybean seed varieties using deep learning approaches, which distinguishes our work from existing studies. Recognizing this gap, we aim to contribute by developing a model capable of accurately identifying common soybean seed varieties that pose challenges for experts at the Ethiopian Institute of Agricultural Research, traders at the Ethiopian Commodity Exchange (ECX), and local farmers.

The motivation for developing an Ethiopian soybean seed type classification model using deep learning is rooted in the critical need to enhance agricultural productivity and food security in the country. Soybean is a vital crop in Ethiopia, essential for nutrition and income generation for farmers. Accurate classification of soybean seed types is crucial for selecting superior varieties, improving yields, and ensuring high-quality seeds. Traditional classification methods are labor-intensive and error-prone, while deep learning models offer increased accuracy and efficiency. Integrating advanced technologies like deep learning into agriculture modernizes practices, supports breeding programs by identifying desirable traits, and provides economic benefits by boosting marketable yields. Furthermore, this research contributes to the broader field of artificial intelligence in agriculture, fostering innovation and technological advancement. Ultimately, the study aims to revolutionize soybean seed classification, supporting sustainable farming practices and contributing to Ethiopia's food security and economic development.

1.3 Statement of Problem

Soybean seeds play a crucial role in agriculture, serving as a primary source of protein and oil for both humans and animals. In Ethiopia, soybean cultivation has gained significant importance due to its nutritional value, economic potential, and contribution to food security. However, accurately classifying different soybean seed types is a challenging task, primarily due to the inherent similarities and subtle differences in their visual appearance (Ayele Nigus Asres, 2020). Currently, seed classification in Ethiopia is mainly conducted manually by experts, relying on visual inspection and morphological characteristics (Gudina et al., 2023) (Butuner et al., 2023). This process is time-consuming, subjective, and prone to errors, leading to inconsistencies in seed type identification. As a result, farmers, researchers, and seed producers face difficulties in obtaining reliable and consistent information about seed varieties, hindering effective decision-making processes in soybean cultivation, breeding, and seed production. To address this problem, there is a need to develop an automated and accurate seed type classification system for Ethiopian soybeans (Abuhayi & Mossa, 2023). Deep learning, a subfield of machine learning, has shown remarkable success in image classification tasks, making it a promising approach for seed type classification. By leveraging the power of deep neural networks, it is possible to train a model that can effectively distinguish and classify different soybean seed types based on their visual characteristics. Therefore, this research aims to develop and evaluate a deep learning-based classification model for Ethiopian soybean seeds. The model will be trained on a carefully curate dataset of soybean seed images, encompassing various seed types commonly found in Ethiopia. By utilizing deep learning techniques, the study seeks to achieve high accuracy and robustness in automatically classifying soybean seed types, providing a reliable and efficient solution to the existing challenges faced in manual classification.

The outcomes of this research will not only contribute to the field of agricultural technology but also have practical implications for farmers, researchers, and seed producers in Ethiopia. By automating the seed type classification process, this study aims to enhance the accuracy, speed, and consistency of seed identification, facilitating informed decision-making and promoting the development of improved soybean cultivars. Ultimately, this research aims to support the growth and productivity of soybean farming in Ethiopia, contributing to food security and economic development in the region.

1.4 Research questions

1. How can a diverse and representative dataset of Ethiopian soybean seed images be effectively collected and prepared for deep learning-based classification?
2. Which deep learning architectures are better suited to classifying Ethiopian soybean seed types based on phenotype features like colour , size, shape etc.?
3. What strategies can be applied to measure the performance of deep learning models on the collected dataset?

1.5 Objectives

1.5.1 General Objective

The general objective of this thesis is to develop a deep learning-based classification model for distinguishing Ethiopian soybean seed varieties.

1.5.2 Specific Objectives

To achieve this, the study pursued the following specific objectives:

- To develop dataset of labelled soybean seed images representing diverse varieties and types.
- To adopt and employ deep learning models tailored to the task of soybean seed type classification
- To train the models on the prepared dataset using appropriate training strategies.
- To evaluate the performance of the proposed classification models using appropriate metrics.

1.6 Scope and Limitation of the Study

1.6.1 Scope

The study's scope centered on employing deep learning techniques to develop a classification model specifically geared towards eight commonly grown varieties of soybean seeds in Ethiopia. These varieties include **Cheri, Jalale, Dhidhessa, Korme, Boshe, Katta, Ethio-Yugoslavia and Gute.** The primary goal was to precisely classify and differentiate among these particular soybean seed types. By providing unique insights and solutions tailored to the classification of these varieties, the study aimed to contribute to the enhancement of Ethiopian soybean seed production and quality.

1.6.2 Limitation

One constraint of this research paper concerns the availability and range of datasets utilized. Due to factors such as limited data access and financial constraints, the study focused solely on seven particular soybean varieties. Consequently, it did not include grades for these varieties of soybean seeds. Therefore, the study's findings may not be universally applicable to all existing varieties and grades of soybean seeds.

1.7 Significance of the Study

Soybean seeds, rich in protein, hold significant nutritional value for people in both developed and developing nations, aiding in meeting their dietary requirements. These versatile seeds find various applications, including being used in home cooking for diverse dishes, the production of soybean milk, and as livestock feed in regions like Ethiopia. Deep learning can be applied to identify Ethiopian soybean seed varieties accurately, aiding in selecting the better seeds for specific agricultural needs. It enables farmers, researchers, and stakeholders like the Ethiopian Commodity Exchange (ECX) to make informed decisions by detecting subtle differences among seed types and recommending the most suitable ones for particular environments, thereby enhancing agricultural efficiency and effectiveness. Therefore, the implementation of a classification model can assist Ethiopian farmers and experts in optimizing crop potential, minimizing crop failure, and ultimately achieving higher yields and promoting sustainable agricultural practices. Here are several key points that highlight its significance:

Improved Crop Management: Precision Agriculture: Accurate classification of soybean seed varieties allows for more precise agricultural practices. Different varieties may require specific treatments in terms of irrigation, fertilization, and pest control. Deep learning models can help in identifying the right variety, leading to optimized resource use and better crop management.

Enhanced Yield and Quality: Selective Breeding: Identifying high-yield and disease-resistant varieties can lead to selective breeding programs that enhance overall crop productivity and quality. Deep learning models can assist in recognizing these superior varieties quickly and accurately.

Economic Benefits: Market Segmentation: Farmers can better understand which varieties are in demand and can segregate their harvests accordingly. This can lead to higher market prices and better economic returns.

Reduction of Losses: Accurate classification reduces the risk of mixing varieties, which can lead to market rejection or lower prices. This can also minimize post-harvest losses.

Research and Development: Genetic Research: Accurate classification aids in genetic research by providing precise data on different varieties. This information is crucial for developing new and improved soybean varieties. Data Collection: Large datasets generated from classification tasks can be used for further research in plant biology and genetics.

Quality Control: Seed Purity: Ensuring that seed batches are pure and not mixed with other varieties is essential for maintaining quality. Deep learning models can help in verifying the purity of seed lots. Certification and Standards: Meeting the standards set by agricultural authorities and seed certification bodies becomes easier with accurate classification.

Automation and Efficiency: Labor Savings: Automated classification using deep learning reduces the need for manual inspection, which is labor-intensive and time-consuming. This increases overall efficiency and reduces labor costs. Scalability: Deep learning models can handle large volumes of data and can be scaled to process information from extensive farming operations, making it suitable for both small and large-scale farming (Kamilaris, A., & Prenafeta-Boldú, F. X. 2018).

Disease and Pest Management: Early Detection: Some soybean varieties might be more susceptible to certain diseases or pests. Classifying varieties accurately can help in early detection and targeted intervention, reducing the spread of diseases and pest infestations (Ferentinos, K. P. 2018).

Environmental Impact: Sustainable Practices: By knowing the specific requirements and characteristics of each soybean variety, farmers can adopt more sustainable farming practices that are better suited to their local environment and reduce negative environmental impacts (Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. 2018).

1.8 Organization of the Paper

This research paper is organized into five chapters, each serving a specific purpose,

Chapter One: Introduction

Provides an overview, including background, problem statement, research questions, objectives, scope, limitations, motivation, and significance of the study.

Chapter Two: Review of Related Literature

Discusses various scholarly works related to the topic and highlights previous studies.

Chapter Three: Methodology of the Study

Describes the research methods, approaches, techniques, and tools used for data collection and analysis.

Chapter Four: Results and Discussions (Experimentations)

Presents and analyzes the results of the experiments conducted.

Chapter Five: Conclusion and Recommendations

Summarizes findings, addresses research objectives, and offers recommendations for future research or actions.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

In this chapter, the relevant literature pertaining to the foundational concepts underpinning this research paper was examined. Initially, an overview is provided on the types of Ethiopian soybean seeds and methodologies employed by the Ethiopian Commodity Exchange (ECX) and the Ethiopian Institute of Agricultural Research (EIAR). This serves as an introduction to the existing literature on a subject, facilitating the synthesis of previous research discoveries.

The literature review delves into a comprehensive examination of an alternative classification model for soybean seeds employing deep learning methodologies. Additionally, digital image processing techniques, encompassing image acquisition, pre-processing, various segmentation methods, feature extraction, and evaluation techniques, are thoroughly discussed. This chapter concentrates specifically on soybean seeds, elucidating the classification models developed through the utilization of deep learning techniques.

2.2 Ethiopian Soybeans

Soybean cultivation has endured in Africa since the late 1800s, as evidenced by records dating back to the early 1900s. In Ethiopia, soybean was introduced in the 1950s, with formal agricultural research initiatives commencing in the 1970s. Research efforts on soybeans began in Ethiopia in 1970, initially in Hawassa, Jimma, Bako, and Pawe, subsequent to the crop's introduction to the nation in 1950 (Hunde Desissa, 2019). The first five soybean varieties were officially registered during the 1973/74 period. Soybean holds significant economic importance in the region, offering numerous agronomic and commercial applications. It serves as a vital source of foreign exchange and provides essential protein for both human consumption and livestock feed. Although the majority of soybean production in Ethiopia is concentrated in the regions of Amhara, Oromia, and Benishangul Gumuz, its demand, both domestically and internationally, has been steadily increasing in recent years. However, challenges such as inadequate post-harvest management, limited access to technology, and marketing constraints have hindered the crop's ability to fully satisfy this growing demand (Rehima Musema, Samuel Diro. Beza Erko, Desalegn Teshale, Regasa Dibaba, Abush Tesfaye, 2022), (Beyene & JPAA, 2022), (Tesfaye, 2018).

Soybean is a highly productive crop, yielding significant amounts of protein and oil per unit of land. It is widely cultivated worldwide for various food products and has seen increased adoption by large-scale commercial farmers in Ethiopia, leading to doubled production levels over the past decade. Despite occupying a small portion of land, soybean contributes substantially to oilseed production in the country, with around 209,000 small-scale farmers primarily cultivating it in western regions like Oromia, Benishangul-Gumuz, and Amhara (USDA, 2020), (K. Achamyelh, Z. Shumeta, A. Tesfaye, and M. Hailemariam, 2020).

Ethiopian soybean seed types differ significantly from those of other countries due to the unique agro-ecological conditions, traditional farming practices, and genetic variations present in the region. These differences can impact traits such as seed size, color, oil content, and resistance to local pests and diseases. Ethiopian soybean varieties are often adapted to specific local conditions, which may not be suitable for varieties developed in other parts of the world. Developing a classification model specifically for Ethiopian soybean seed types is advantageous as it ensures the identification and promotion of locally adapted, high-performing varieties. This targeted approach can enhance crop yields, improve disease resistance, and increase the nutritional value of soybeans produced in Ethiopia. Moreover, a tailored classification model can support breeding programs, streamline seed selection processes, and ultimately contribute to sustainable agricultural practices and food security in the region . By leveraging deep learning technologies, this model can achieve high accuracy and efficiency in seed classification, reducing reliance on manual methods and minimizing errors, thereby benefiting both farmers and researchers .

2.3 Ethiopian Soybean Varieties

In this section, the different types of soybean seeds typically cultivated in Ethiopia are discussed. Therefore, Ethiopia boasts a diverse selection of soybean seed varieties, including Cheri, Jalale, Dhidhessa, Korme, Boshe, Katta, and And Ethio-Yugoslavia. These distinct variations are carefully grown and assessed to determine their effectiveness in relation to a range of agronomic factors within varying climatic and soil conditions. The plant is classified as an oilseed and pulse. In Ethiopia, under the Federal Democratic Republic of Ethiopia Ministry of Agriculture and Natural Resources, the Plant Variety Release Protection and Seed Quality Control Directorate register different varieties of soybean (K. Achamyelh, Z. Shumeta, A. Tesfaye, and M. Hailemariam, 2020).

Figure 2.1: Sample images of eight Soybean Seed Varieties

2.4 Soybean Seed Classification Technique

Soybean seed classification identifies seed types based on physical traits, aiding in selecting the right seeds for optimal growth conditions. Deep learning, advancement from traditional neural networks, outperforms previous methods (Alzubaidi et al., 2021). Deep learning (DL) utilizes artificial neural networks to tackle intricate problems across diverse fields like image classification, speech recognition, and natural language processing. In soybean seed classification, DL can discern seed types by analyzing their physical and structural traits. Convolutional neural networks (CNNs) are commonly employed for this task due to their proficiency in image classification and demonstrated accuracy across domains (Cong & Zhou, 2023).

2.5 Digital Image Processing

Digital image processing involves the manipulation of digital images using algorithms and techniques to enhance quality, extract information, or perform tasks like recognition or compression. It includes acquisition, enhancement, restoration, segmentation, feature extraction, compression, object recognition, and image analysis (Archana & Jeevaraj, 2024).

In agriculture, image processing holds the potential to automate crucial tasks such as weed detection, food grading, harvest control, and fruit picking with exceptional precision. Digital images play a pivotal role across all agricultural processes, constituting a significant aspect of our digital landscape. Digital image processing revolves around employing methods to enhance images or extract valuable data from them, enabling the derivation of insightful results. The evolution of digital image processing techniques is progressing rapidly (Pareek et al., 2023).

2.5.1 Image Acquisition

Image acquisition refers to the process of capturing digital images using various devices such as cameras, scanners, or sensors. It is the initial step in the digital image processing pipeline and involves converting real-world scenes into digital representations. Proper image acquisition is crucial as it directly impacts the quality and usability of the resulting images. Factors such as lighting conditions, camera settings, and sensor quality can affect the quality and fidelity of the acquired images. Various techniques and technologies are employed for image acquisition depending on the specific application requirements, such as visible light imaging, infrared imaging, or multispectral imaging (Luo et al., 2023).

2.5.2 Image Pre-processing

Image pre-processing involves applying various techniques to images before further analysis. This includes resizing, cropping, rotation, denoising, contrast adjustment, normalization, colour correction, and feature extraction. These steps enhance image quality, reduce noise, correct distortions, and extract relevant features for better performance in subsequent processing tasks.

Resizing: Images may need to be resized to a specific resolution or aspect ratio to fit the requirements of the subsequent tasks. Resizing can be done using various interpolation methods such as nearest neighbour, bilinear, or bicubic (Pareek et al., 2023).

Cropping: Cropping involves removing unwanted parts of an image to focus on the region of interest. It can help eliminate irrelevant background or isolate specific objects within an image.

Rotation and flipping: Images may need to be rotated or flipped to align them properly or to augment the dataset. Rotation can be performed at different angles, such as 90, 180, or 270 degrees, while flipping can be horizontal or vertical.

Gray scaling: Converting a colour image to gray-scale reduces the image's dimensionality and simplifies subsequent processing steps, especially when colour information is not necessary.

Denoising: Noise in images can degrade their quality and affect the performance of subsequent algorithms. Denoising techniques like Gaussian blur, median filtering, or bilateral filtering can be used to reduce noise while preserving important image details.

Contrast adjustment: Adjusting the contrast of an image can improve its visual quality and enhance the visibility of important features. Techniques like histogram equalization or adaptive contrast enhancement can be applied to enhance the dynamic range of pixel intensities.

Normalization: Normalizing the pixel values of an image ensures that they fall within a specific range, such as [0, 1] or [-1, 1]. It can help in standardizing the data and making it more suitable for certain algorithms or models.

Image enhancement: Various enhancement techniques like sharpening, smoothing, or edge detection can be applied to highlight important features or suppress unwanted details in an image.

Colour correction: Colour balance, white balance, and colour mapping techniques are used to correct colour inconsistencies or adjust the colour distribution of an image.

Feature extraction: Pre-processing steps like edge detection, image gradients, or texture analysis can be performed to extract specific features from an image. These features can be used for tasks such as object recognition, segmentation, or classification.

2.5.3 Image Segmentation

Image segmentation involves dividing a digital image into segments to simplify it or represent it in a more meaningful way. This process can utilize various techniques like edge detection, region detection, thresholding, and statistical classification (Kumar & Sachar, 2024). The aim is to group pixels into coherent regions within the image, such as objects or parts of objects. Image segmentation methods fall into two main categories: region-based and edge-based, each with its own set of techniques and algorithms to refine the segmentation process

(Amiriebrahimabadi et al., 2024). Image segmentation methods encompass two main categories (Kumar & Sachar, 2024) region-based and edge-based. Region-based methods involve partitioning an image into coherent regions based on similarity criteria, such as region growing and watershed transform. Edge-based methods focus on detecting boundaries or edges between regions, using techniques like edge detection filters and active contour models. Both approaches have their advantages and are chosen based on the characteristics of the image and the requirements of the application.

2.5.4 Feature Extraction

Feature extraction is a crucial step in data pre-processing, particularly in image analysis tasks. Its primary objective is to generate new features from the existing ones in a dataset, often with the aim of enhancing the performance of subsequent machine learning or computer vision algorithms. This process involves transforming raw data, such as pixels in an image, into a more compact and meaningful representation that captures relevant information for the given task. In the context of image analysis, once an image has been segmented into regions or objects using various segmentation techniques, the resulting collections of segmented pixels need to be further processed to extract features that are suitable for subsequent analysis. This might involve extracting information related to colour, texture, shape, or other visual characteristics of the objects in the image (Soares et al., 2023).

For example, in the study mentioned, which addresses the categorization of soybean seeds, the researchers aimed to improve the effectiveness of classification by incorporating features related to both colour and texture. Colour features might capture information about the distribution of colours within different regions of the seed, while texture features could describe the spatial arrangement of pixel intensities, providing insights into surface properties such as smoothness or roughness. By incorporating colour and texture features, the researchers likely aimed to capture a more comprehensive representation of the soybean seeds, which could help mitigate the issue of form variance within each class. This approach can potentially improve the accuracy and robustness of the classification model, enabling more reliable categorization of soybean seeds based on their visual characteristics (Yang et al., 2021).

2.5.5 Image Classification

Image classification, a foundational method in computer vision, aims to understand an image by recognizing and describing its visual components in relation to real-world objects. This involves identifying features within the image and linking them to specific objects or categories, typically represented by unique grey levels or colours (de Medeiros et al., 2020). Within digital image processing, image classification is immensely significant as it enables the extraction of meaningful insights from visual data. The process of image classification entails categorizing an image into predefined classes or groups, requiring advanced algorithms and techniques to effectively analyze and interpret the visual content, enabling machines to accurately identify and classify objects depicted in them (Lin et al., 2023).

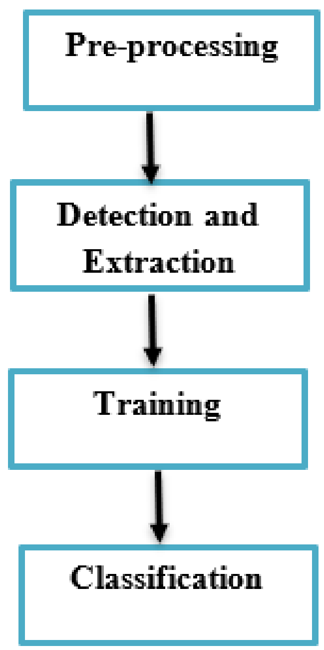


Figure 2.2: Steps for Image Processing

2.6 Image Classification Methods

Image classification, a fundamental task in computer vision, entails assigning labels or categories to images according to their visual characteristics. Various methods, from conventional machine learning techniques to deep learning methodologies, are employed for this purpose.

2.6.1 K-Nearest Neighbour (KNN)

K-Nearest Neighbour (KNN) is a simple yet effective algorithm used for classification and regression tasks. It determines the class of a data point based on the majority vote of its nearest neighbours in the feature space. KNN excels in accurately classifying data points near the decision boundary but can be time-consuming to train with large datasets. It performs well with lower-dimensional data. In KNN, the distance between neighbours is calculated, and the value of k determines the number of neighbours considered when classifying a sample data point. This method identifies a cluster of k nearest training items to the test object, assigns a label based on the prevalent class in that cluster, and then classifies the cluster (Kaesmetan & Overbeek, 2022). In the context of image classification, KNN works by comparing an unlabeled image to a set of labelled images in a training dataset. The algorithm assigns a label to the unlabeled image based on the majority label of its K nearest neighbours in the feature space (L. Wang et al., 2022).

2.6.2 Naïve Bayesian Classification

Naive Bayes classifiers are indeed straightforward probabilistic classifiers widely used in machine learning. They are based on applying Bayes' theorem with the strong assumption of independence between each pair of features. Despite this simplification, Naive Bayes classifiers often perform surprisingly well in practice, especially in text classification tasks like spam filtering and sentiment analysis. The Bayes theorem, upon which Naive Bayes classifiers are built, allows us to calculate the probability of a hypothesis given our prior knowledge. In the context of classification, it helps us determine the probability that a given data instance belongs to a particular class given its features.

Training a Naive Bayes classifier involves estimating the probabilities required by the Bayes theorem from the training data. This can be done effectively in a supervised learning environment, where the correct class labels are provided for the training instances. The accuracy of the probability model depends on the quality and representativeness of the training data. Bayesian networks, on the other hand, are graphical models that represent the dependencies between random variables using nodes and directed connections. These models are also known as probabilistic directed acyclic graphs (DAGs). Each node in the graph represents a random variable, and the directed edges indicate the probabilistic dependencies between variables. Bayesian networks are powerful tools for representing and reasoning about uncertain domains, and they find applications in various fields including machine learning, decision making, and probabilistic inference (L. Wang et al., 2022).

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \tag{2.1}$$

2.6.3 Support Vector Machine (SVM)

Support Vector Machines (SVM) are powerful supervised learning algorithms used for both regression and classification tasks. They differ from other methods by aiming to maximize the margin between classes. SVM networks require labelled data for training and analyze data for regression and classification. They are versatile and effective in various domains but may have limitations with large datasets due to computational complexity (Macuácua et al., 2023).

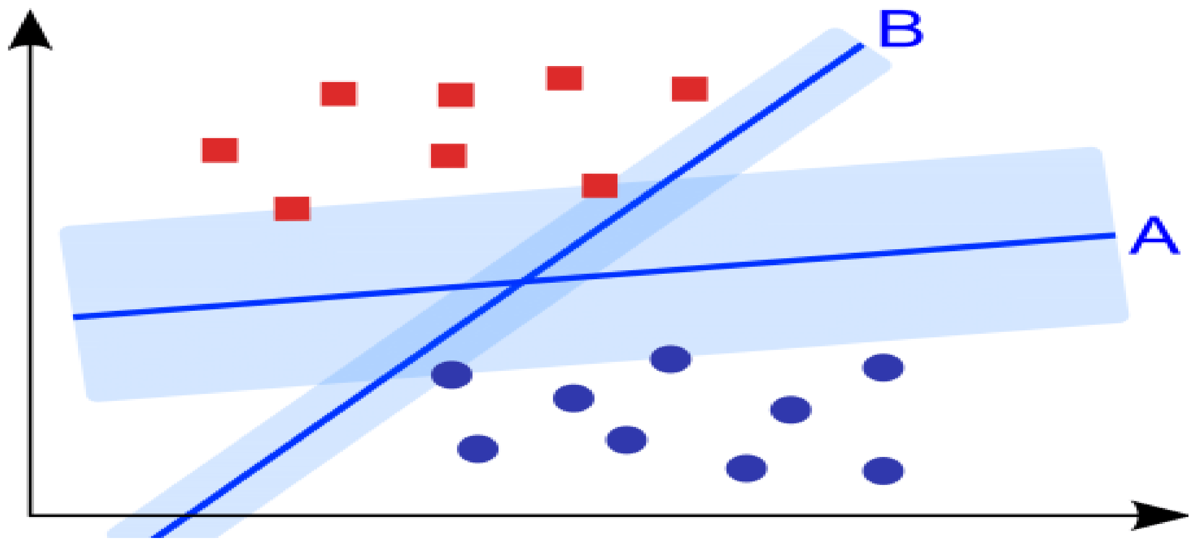


Figure 2.3: Visualization of A SVM splitting a dataset into two classes (Zhang et al., 2023)

2.6.4 Artificial Neural Network (ANN)

Neural networks, both biological and artificial, consist of interconnected neurons that process and transmit signals. In the context of artificial neural networks (ANNs), nodes serve as processing components akin to biological neurons. ANNs are increasingly utilized due to their ability to perform tasks comparable to the human brain, such as pattern recognition, autonomous vision, and controlling mechanical devices. They enable the development of systems capable of autonomous learning (Abraham et al., 2020). Typically, neural network architecture comprises three layers:

Input Layer: This layer accepts the input feature vector, which serves as the initial data for the network.

Output Layer: The output layer produces the response or prediction generated by the neural network based on the input data.

Intermediate Layer(s): Intermediate layers, also known as hidden layers, contain neurons that facilitate connections between the input and output layers. These layers play a crucial role in processing and transforming the input data to produce the desired output.

This architectural structure enables neural networks to learn from data and make predictions or classifications based on learned patterns. The flexibility and adaptability of neural networks make them valuable tools in various fields, including machine learning, computer vision, robotics, and more.

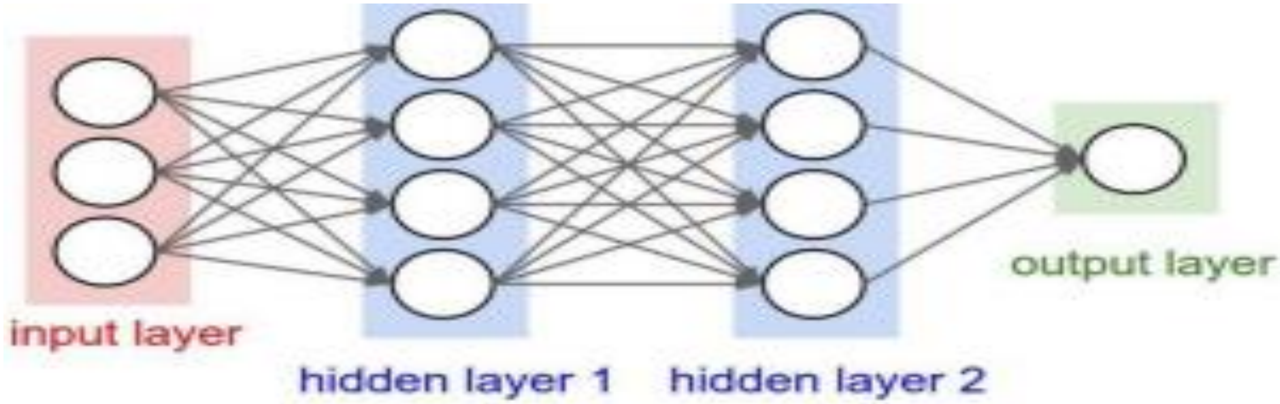


Figure 2.4: Basic Deep Neural Network (Z. J. Wang et al., 2021)

2.6.5 Deep Learning (DL)

Deep learning, a subset of machine learning, involves training computational models with multiple layers of processing. It can be implemented through various architectures such as Unsupervised Pre-trained Networks, CNNs, RNNs, and Recursive Neural Networks (Y. chen Wu & Feng, 2018). There are several implementations of deep learning, each suited for different tasks:

Unsupervised Pre-trained Networks: These networks leverage unlabeled data for training. They are initialized with weights learned from an unsupervised learning task, such as auto-encoders or generative adversarial networks (GANs). These pre-trained models can then be fine-tuned on labelled data for specific tasks.

Convolutional Neural Networks (CNNs): CNNs are particularly effective for tasks involving image recognition and classification. They consist of layers designed to automatically and adaptively learn spatial hierarchies of features from input images.

Recurrent Neural Networks (RNNs): RNNs are suitable for sequential data processing tasks, such as natural language processing and time-series analysis. They are capable of capturing temporal dependencies by allowing information to persist over time through loops within the network architecture.

Recursive Neural Networks: These networks are similar to RNNs but are better at capturing hierarchical structures within sequential data. They excel in tasks where the input data exhibits recursive or nested relationships, such as parsing syntactic structures in language processing. One of the key advantages of deep learning is its ability to automatically learn features from raw data, eliminating the need for manual feature engineering. This process, known as feature learning, enables the extraction of meaningful representations from the data, which can then be utilized as inputs for downstream regression or classification tasks (Dastres et al., 2021).

2.6.6 Convolutional Neural Network (CNN or ConvNet)

Convolutional Neural Networks (CNNs) are widely employed in the realm of deep learning for visual image analysis. They streamline the need for pre-processing by autonomously learning features through filters, bypassing manual feature design. The CNN architecture comprises input, output, and numerous hidden layers. Specifically designed to process grid-structured inputs, CNNs excel at handling data with high spatial correlations in local grid regions, with the quintessential example being two-dimensional images. CNNs consist of multiple layers, each serving a distinct function on the input data, including convolutional, down-sampling, and activation layers. Initial convolutional layers within the network identify basic features like edges, whereas subsequent layers capture more intricate semantic features. These convolutional layers play a pivotal role in extracting features essential for image classification (Y. chen Wu & Feng, 2018). The convolutional layer serves as the initial stage in feature extraction within Convolutional Neural Networks (CNNs), facilitating the preservation of spatial relationships in images by connecting pixels with their nearby counterparts. CNNs with varying depths employ multiple convolutional layers to discern different aspects of the input. Typically, the first convolutional layer identifies fundamental features like lines, edges, and texture, whereas subsequent layers delve into more intricate and abstract characteristics (Nawrocka et al., 2023).

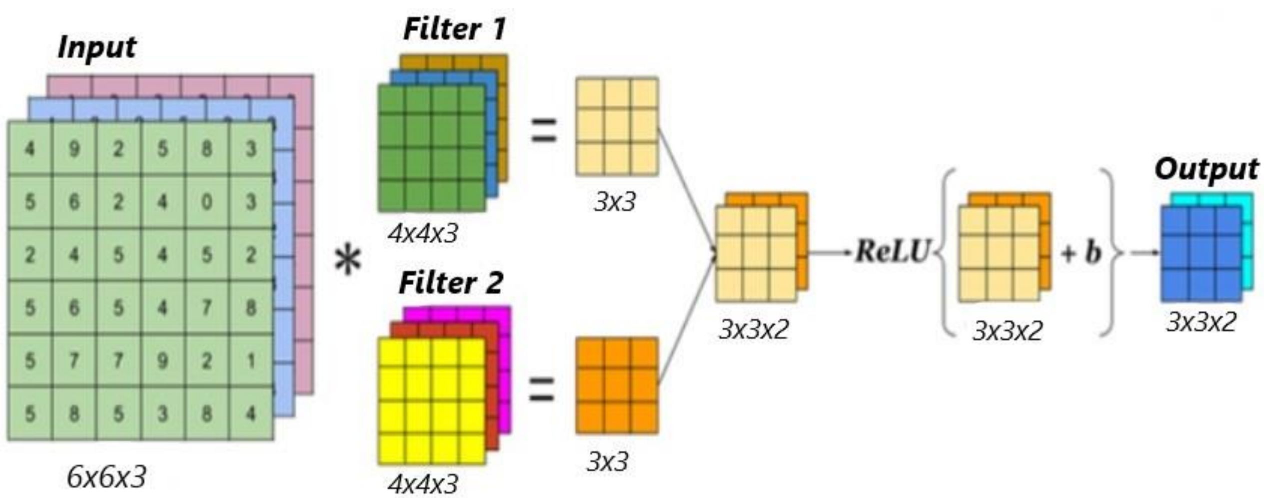


Figure 2.5: Convolution Layer (Butuner et al., 2023)

The Pooling Layer in a CNN follows the Convolutional Layer, aiming to reduce feature size and computational load. It operates independently on each feature map, summarizing features from the previous layer using methods like max, min, or average pooling. This layer helps prevent over-fitting by extracting key features and reducing complexity.

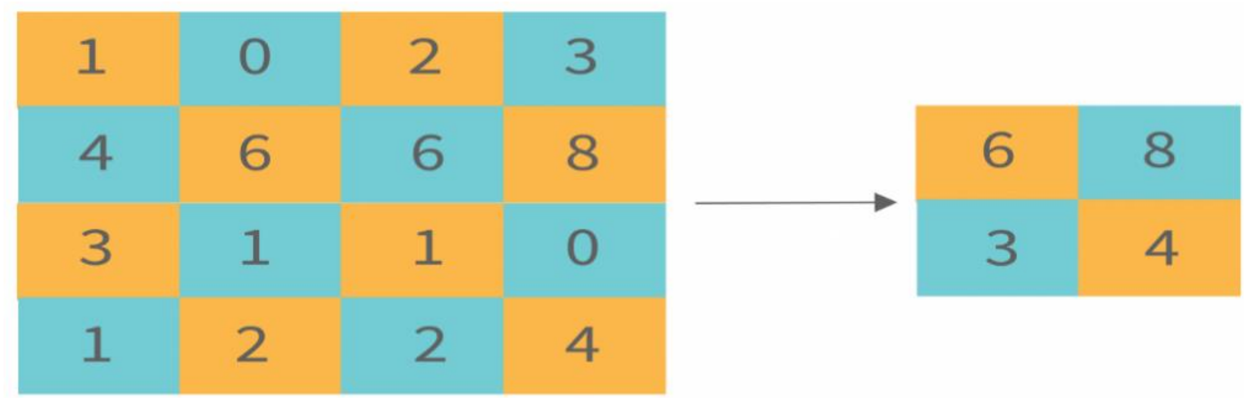


Figure 2.6: Pooling layer (Yang et al., 2021)

The Fully Connected (FC) Layer comprises neurons, weights, and biases, facilitating connections between neurons from different layers in a CNN. Typically positioned before the output layer, FC layers constitute the final stages of CNN architecture. These layers establish relationships between parameters to determine their impact on labels, forming a fully connected network. By leveraging convolutional and pooling layers, we can notably reduce time and space complexity. Ultimately, a fully connected network can be constructed to classify images (Nawrocka et al., 2023).

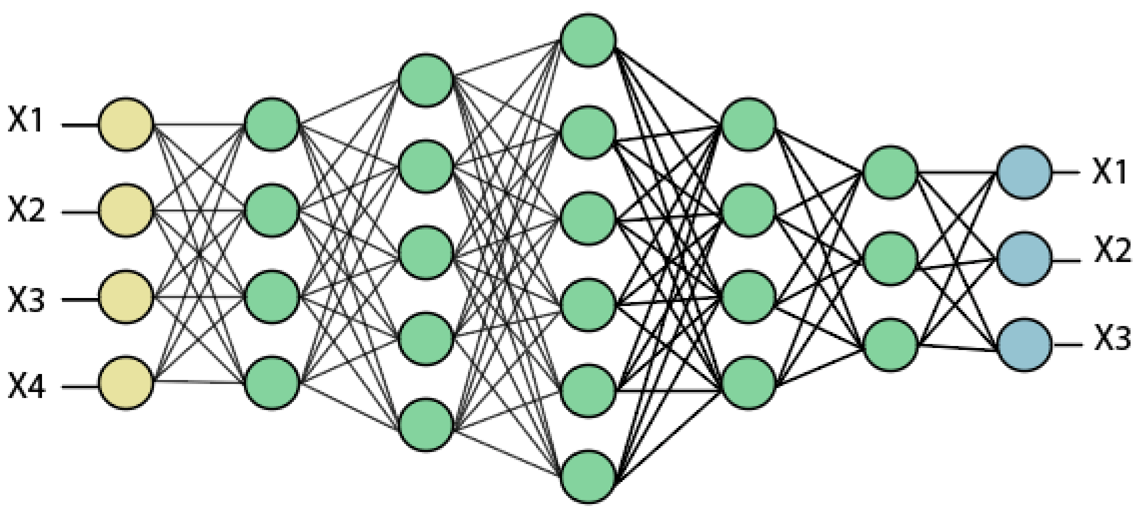


Figure 2.7: Fully Connected layer (Dönmez, 2024)

The core components of a convolutional neural network (CNN) include the convolutional layer, pooling layer, nonlinear activation layer, and fully connected layer. Typically, images undergo pre-processing before being inputted into the neural network via the input layer. They then traverse through alternating convolutional and pooling layers before being classified by the fully connected layer (Han et al., 2023). In the design process outlined in the thesis, these steps were followed: first, a basic CNN structure was developed. Then, three more sophisticated CNN models utilizing VGG-16, ResNet-50, and Inception-v3 architectures were constructed. Finally, data augmentation and ensemble techniques were integrated into the complex CNN models. The CNN architecture is formed by integrating different layers (convolutional, pooling, and fully connected) in a specific arrangement. Following the development of effective CNN architectures, the applications of image processing and image classification are showcased, demonstrating the network's capability in handling visual data tasks (Qu et al., 2023).

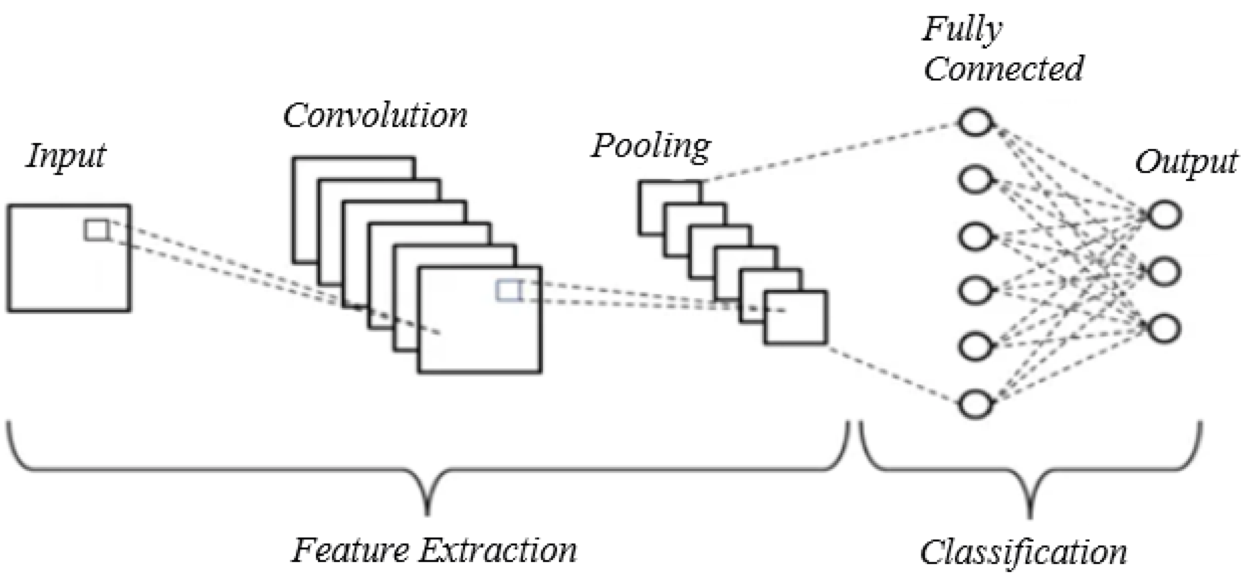


Figure 2.8: Basic CNN Architecture (Phung & Rhee, 2018)

AlexNet: AlexNet's victory in the 2012 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) sparked a rapid acceleration in deep learning research. Its interconnected design played a pivotal role in solving numerous challenges, including the classification of obscured objects, a vital aspect of artificial neural intelligence (Q. Wu et al., 2019).

Visual Geometry Group (VGG): VGG replaced layers of 5x5 and 11x11 filters with stacks of 3x3 filters, demonstrating experimentally that this parallel arrangement can have a comparable impact. Renowned for its computational efficiency, the VGG CNN model serves as a strong baseline in computer vision applications. Its versatility enables it to handle diverse tasks, including object detection, making it widely favoured in the field.

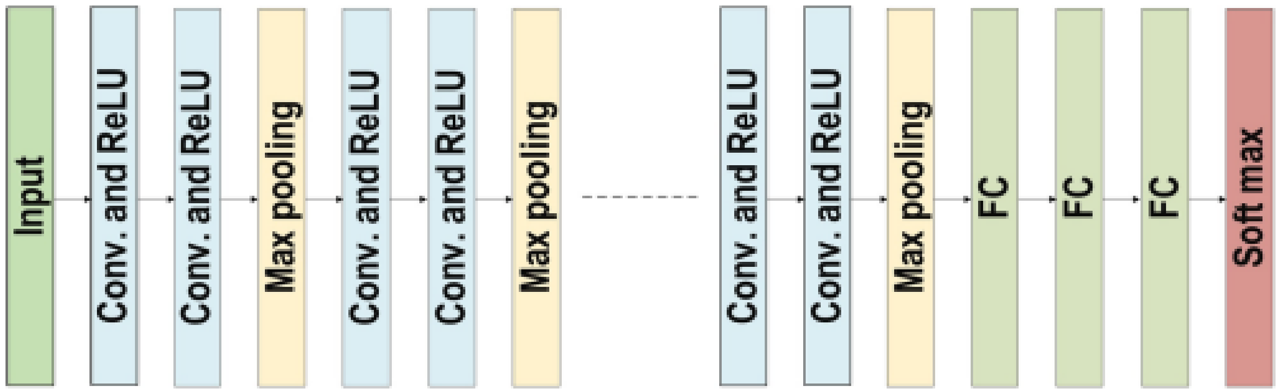


Figure 2.9: Visual Geometry Group (H. C. Li et al., 2023)

GoogLeNet (InceptionV3): Developed by Google, Inception introduces deeper CNNs, notably Inception-v3 with 48 layers. It achieves strong performance with fewer parameters compared to other architectures. Pre-trained on a vast ImageNet dataset, it can classify images into 1000 categories, with an input size of 299x299 pixels.

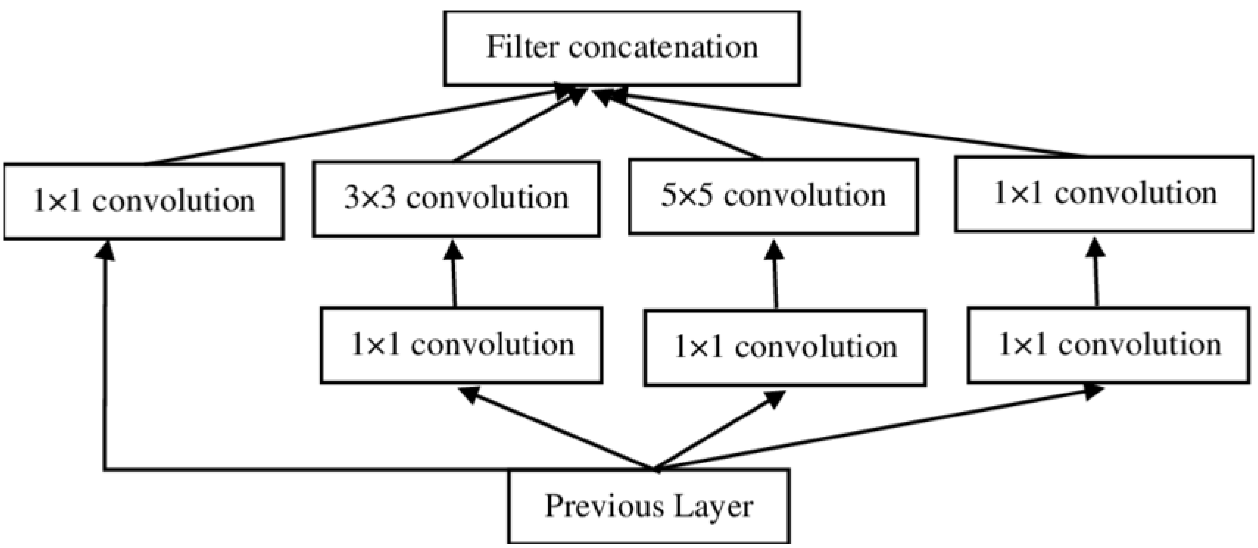


Figure 2.10: GoogleNet Block (Véstias, 2019)

2.7 Related Work

This section explores various studies that have utilized deep learning for the classification of soybean seed types. Table 2.1 highlights several issues identified in related works, categorized according to classification type, dataset selection, computational resource requirements, training complexity and inadequate results on minority classes. These problems are discussed to shed light on the shortcomings observed in the reviewed literature.

Table 2.1: Summary of the Related Work

Authors	Title	Method/Algorithm	Results	Paper gab
(Çetin, 2022)	Machine Learning for Varietal Binary Classification of Soybean (Glycine max (L.) Merrill) Seeds Based on Shape and Size Attributes	Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), and Multilayer Perceptron (MLP) are used machine learning algorithms for classification tasks.	Ceyhan and Traksoy soybean seeds had the highest accuracy (90% RF, 89% MLP), Çevik and İlksoy (88% MLP) and Çevik and Traksoy (87.50% RF).	Soybean variety variations fine-tune models for better performance.
(Wei et al., 2020)	Non-destructive Classification of Soybean Seed Varieties by Hyperspectral Imaging and Ensemble Machine Learning Algorithms	Linear discriminant (RSLD) algorithm is used to classify soybean seeds.	RSLD method achieves 99.2% accuracy, outperforming LD (98.6%) and LSVM (69.7%) methods.	The RSLD algorithm achieves high accuracy, but struggles with fewer features. Spectral indices and morphological traits could improve accuracy.
(Yang et al., 2021)	High-throughput soybean seeds phenotyping with convolutional neural networks	Synthetic image generation, transfer learning, and instance segmentation model training, all within	Automated labelled dataset creation reduces manual annotation costs.	Lack of depth information in 2D images restricts comprehensive

	and transfer learning	the context of deep learning and computer vision technology were employed.	Transfer learning lowers computing costs by fine-tuning pre-trained model weights. Scalable pipeline extends to high-throughput object segmentation beyond soybean seeds.	seed analysis. High computing costs in instance segmentation model training need improvement. Synthetic image generation limited to single-class objects; multi-class extension required.
(Zhang et al., 2023)	A Soybean Classification Method Based on Data Balance and Deep Learning	Convolutional neural networks (CNNs) such as DenseNet, VGG, AlexNet, and ResNet were employed.	With DenseNet achieving 98.48% accuracy	Data imbalance
(H. Li et al., 2021)	Identification of soybean varieties based on hyperspectral imaging technology and one-dimensional convolutional neural network	1D Convolutional Neural Network (CNN) algorithm with Near Infrared Hyperspectral Imaging (NIR HSI) was used.	The CNN outperforms methods like SVM, KNN, and PLS-DA, showing stability and high accuracy even with more seed varieties.	Lack of extensive interpretability analysis, computational efficiency.

2.8 Gap Analysis

Under this chapter, numerous research papers were reviewed and gaps were identified that could be addressed in future work. Based on this analysis, the proposed model aims to address these gaps by incorporating common soybean seed varieties that pose challenges for identification by Agricultural Research Institute/Centers experts, ECX traders, and farmers. Additionally, related works were examined, focusing on specific papers closely related to this thesis and the methods or techniques utilized. Consequently, the proposed system seeks to elucidate the performance and limitations of these papers and work towards filling in these gaps. To achieve this, the customized CNN model, specifically designed to address the deficiencies observed in earlier or related works, was introduced and applied to classify soybean seed varieties.

CHAPTER THREE

DESIGN AND METHODOLOGY

3.1 Introduction

This chapter offers a thorough overview of the proposed system architecture for the classification model intended to categorize soybean seed varieties. The system comprises several phases, beginning with image acquisition, proceeding to image preprocessing, and culminating in the classification of soybean seed varieties. Both the training and testing phases of the classification model are incorporated to guarantee precise classification outcomes.

3.2 Research Design

This research aimed to develop a deep learning-based model for classifying different soybean seed varieties. To achieve this, a diverse dataset of soybean seed images was collected and prepared using preprocessing techniques, including noise removal. Various deep learning architectures, such as the Customized CNN model, VGG16, and InceptionV3, were considered, and the model was trained using the prepared dataset. The performance of the model was then evaluated, compared to other pre-trained models, and tested. The results and findings were thoroughly discussed, and potential avenues for future improvement were identified. The research adopts an experimental approach, involving dataset preparation, algorithm implementation, and performance evaluation, with the aim of addressing real-world challenges related to soybean seed varieties classification.

3.3 Research Framework

To achieve the research objective outlined in this paper, a structured research process was followed, as illustrated in Figure 3-1. The study comprises three main phases, outlined below:

Phase One: The research begins with an extensive literature review aimed at identifying and comprehensively understanding the problem. Various sources are examined to gain insights into the specific issue under investigation.

Phase Two: This phase focuses on data preparation and collection. A total of 5200 soybean images are captured, labelled, and assigned to their respective seed varieties. Additionally, data augmentation techniques are applied using Python code to enhance the model training, including rotation range, width range, and height range.

Phase Three: In this phase, pre-processing and deep learning techniques are implemented for the classification of soybean seed varieties. The model's performance is evaluated by comparing accuracies, and the model is tested using hidden soybean images from the datasets.

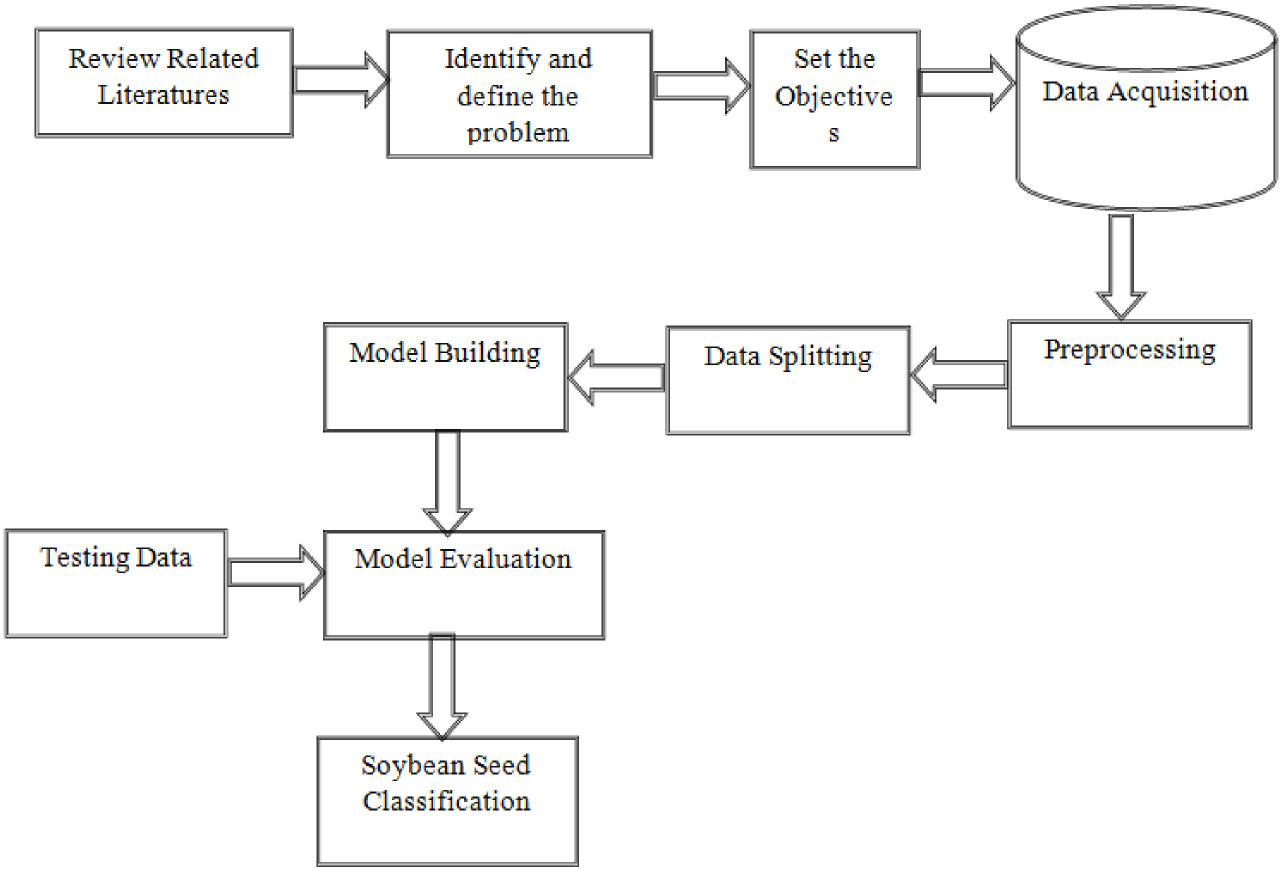


Figure 3.1: Research Process framework

3.4 Data Collection

The images of soybean seeds were gathered from the Bako Agricultural Research Center (BARC), with 80% of the dataset utilized for training and 20% for testing/validation of the model. The initial step encompassed assembling a diverse and extensive dataset containing various soybean seed images, crucial for effective model training. Subsequently, the collected data underwent pre-processing and cleaning procedures to ensure its suitability for training the model. The model was then constructed and trained to precisely discern and categorize different patterns of soybean seed varieties. Ultimately, the model underwent rigorous testing and evaluation to validate its accuracy and dependability.

3.5 The Proposed Soybean Seed Identification Model

The model proposed relies on Convolutional Neural Networks (CNNs), known for their efficacy in seed identification and classification. In order to construct a classification model for Ethiopian soybean seed varieties using deep learning techniques, it's advantageous to harness state-of-the-art deep learning architectures, including CNNs and pre-trained models. The envisioned architecture for classifying Ethiopian soybean seed varieties comprises several stages, including image acquisition, pre-processing, feature extraction, and classification (Gulzar et al., 2020). Initially, images of soybean seeds with class labels were collected from the Bako Agricultural Research Center (BARC). These images were then utilized to train deep learning models like customized CNN, VGG16, and InceptionV3. During training, the models learned to minimize classification error by adjusting their parameters. Subsequently, the trained models were evaluated on a separate validation dataset to gauge their performance and accuracy.

In the pre-processing step, images were resized and filtered using Gaussian and median techniques. Image augmentation was then applied to prevent over-fitting, expanding the dataset for training and testing. Features were extracted using a customized CNN model, VGG16, and InceptionV3. The CNN model consisted of convolutional, pooling, and fully connected layers for feature extraction and classification. Finally, a Softmax classifier was trained on the extracted features from the training dataset and evaluated on the test dataset to classify different soybean seed varieties accurately.

Images were obtained using a suitable camera. Specifically, a Samsung Galaxy Tab 2016 and Samsung Galaxy A13 Smartphone, both digital cameras, were employed to capture digital pictures of various soybean seed types. The resolution of the camera was adjusted to 8 megapixels (2448 × 3264) and 50 megapixels (3060 x 4080) for the images. In order to capture the images, a white paper was placed on a rectangular table, and the Smartphone was positioned at a distance of 10.5 cm from the seeds. Approximately 650 images were taken for each type of seed to ensure an even distribution for classification purposes. Afterwards, the images of the soybean seeds were transferred to a computer for additional analysis. Specifically, the images were saved in the JPG (Joint Photographers Group) file format.

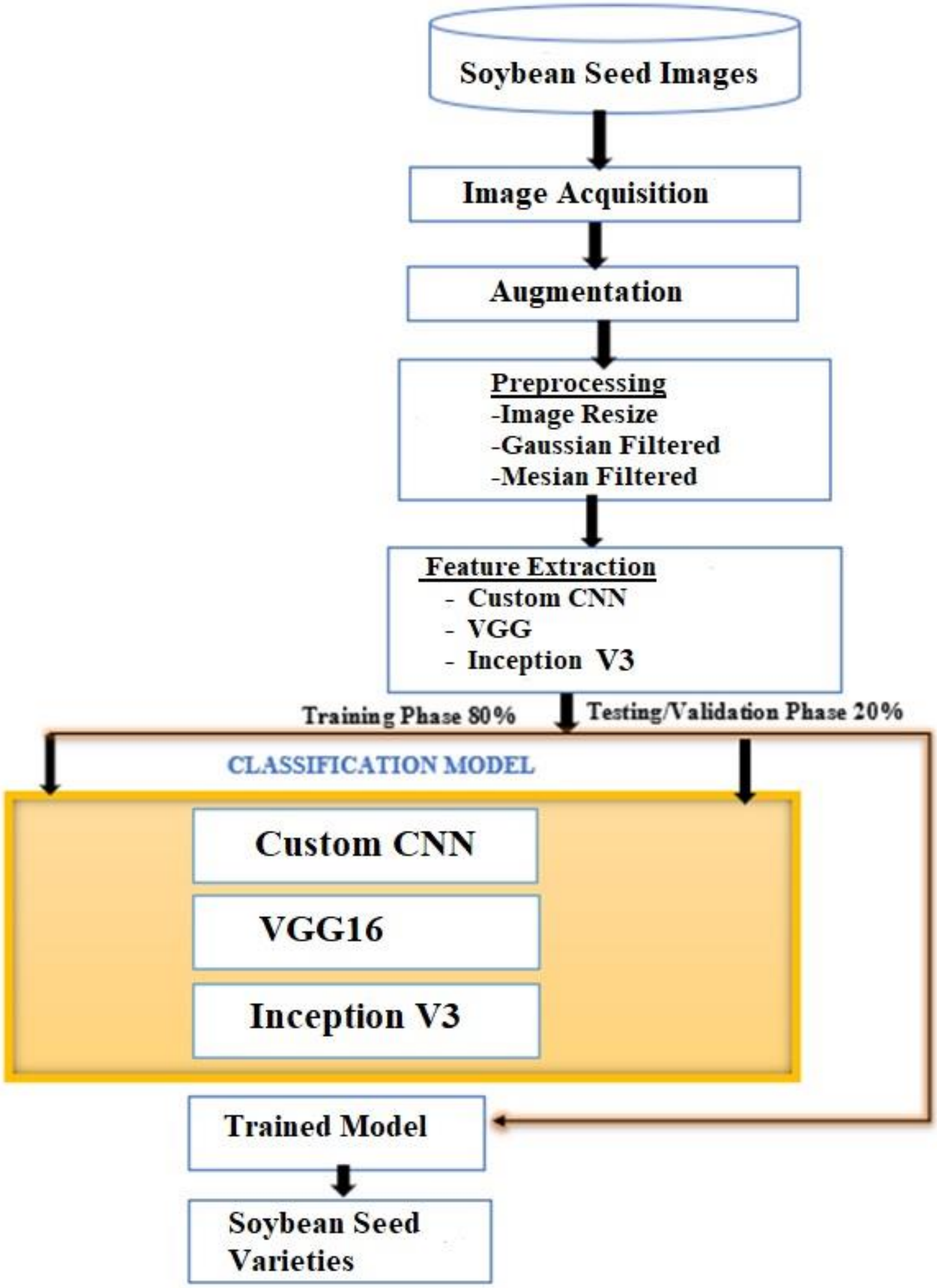


Figure 3.2: Proposed Deep Learning Model Architecture

The research aimed to classify eight different soybean seed varieties, namely Cheri, Jalale, Dhidhessa, Korme, Boshe, Katta, Ethio-Yugoslavia and Gute. These specific seed varieties were sourced from the Bako Agricultural Research Center (BARC).

To ensure accuracy, domain experts involved in the study verified the varieties of the sampled soybean seeds during the stage of image acquisition. The total count of images captured for each seed variety is displayed in Table 3.1.

Table 3.1: Number of samples Soybean images taken from each variety

No.	Varieties	Gathered (Captured)	Augmented	Total number of images for each class
1.	Boshe	650	250	900
2.	Cheri	650	250	900
3.	Dhidhessa	650	250	900
4.	Ethio-Yugo	650	250	900
5.	Gute	650	250	900
6.	Jalale	650	250	900
7.	Katta	650	250	900
8.	Korme	650	250	900
Total		**5200**	**2000**	**7200**

3.6 Data Pre-processing

During the pre-processing stage, the images were resized to a specific dimension of 227 x 227 and transformed into NumPy arrays to ensure compatibility with Keras, a popular deep learning framework. This selection of image size corresponds to the dimensions commonly used in state-of-the-art models that process images. The resizing and conversion of the images were carried out using OpenCV, a computer vision library, which facilitated efficient handling of the images. To ensure proper down sampling of the spatial input in our model, the input layer is structured to be divisible by two multiple times following each convolutional layer. This design choice enables effective pooling operations and contributes to the overall suitability and efficiency of our network. By maintaining this divisibility, we can optimize the down sampled representations and enhance the performance of our model. To enhance the efficiency of the proposed model, a pre-processing technique was integrated to ensure precise classification of images captured from various angles and perspectives (Umamaheswari & Karthikeyan, 2019).

3.6.1 Cleaning and Filtering

Noise Removal: Images can suffer from fluctuations in intensity values caused by environmental factors or camera imperfections, leading to distortions.

Median Filter: Median filters are highly favoured for their ability to effectively reduce specific types of random noise, offering superior noise reduction with minimal blurring compared to similar linear smoothing filters. This makes them popular for applications where preserving image details is crucial. Median filters are commonly employed to remove noise from images for specific purposes while preserving boundaries, making them a predictable technique in image processing, particularly effective in reducing speckle noise and salt and pepper noise (Zhu & Huang, 2012).

Median Filter Algorithm

Input: RGB image

Output: Filtered image

Procedures:

1 *Begin*
2 *Read the RGB image (i)*
3 *Convert the RGB color image*
4 *Select a kernel size (e.g, 5x5 pixels)*
5 *Iterate over each pixel in the converted image*
6 *Calculate the median value of the pixel values within the neighbourhood*
7 *Replace the pixel value with the calculated median value*
8 *Repeat the step 5-8 for all pixels in the entire image*
9 *Save the filtered image as a denoised median image*
10 *End*

Gaussian filters: Gaussian blur, commonly used in graphics software, reduces image noise by applying a blurring effect using a Gaussian function. In image processing, Gaussian smoothing involves applying this technique to blur an image, effectively reducing noise and decreasing image detail (K et al., 2020).

Gaussian Filter algorithm

Input: RGB image

Output: Denoised Filtered image

Procedures:

1 *Begin*
2 *Read the RGB image (i)*

3 *Convert the RGB color image to gray scale*

4 *Select a kernel size*

5 *Apply a Gaussian blur to the Grayscale image using the chosen kernel size*

6 *Repeat steps 2-5 for all pixels in the entire image*

7 *Save the filtered image as a denoised filtered image*

8 *End*

Image Resizing: Resizing images serves to standardize their dimensions for further processing or to meet specific requirements. In this scenario, the resized images are saved in .BMP format with a JPEG quality of 200. OpenCV library in Python is utilized to resize the input image from a specified directory to desired dimensions (227x227 pixels in this instance) and save the resized image to a designated output directory.

Algorithm of Image Resizing

Input: RGB image

Output: Resized image (227x227)

Procedures:

1 *Begin*

2 *Read the RGB image (i)*

3 *Determine the desired dimensions for the resized image (new_width, new_height)*

4 *Calculate the scaling factors for width and height:*

Width_scale = new_width/current_width

Height_scale = new_height/current_height

5 *Assign the pixel value from the original image (227x227)*

6 *Repeat steps for all pixels in the new image*

7 *Save the resized image (227x227)*

8 *End*

3.6.2 Data Augmentation

Data augmentation was used to expand the training dataset, improving model generalization. It prevents over-fitting by creating more diverse data. Techniques like horizontal flipping were applied to increase dataset size and enhance model performance (Xu et al., 2023).

Data Augmentation Algorithm

Input: RGB image

Output: Augmented images

Procedures:

1 *Begin*

2 *Read the original dataset (D)*

3 *Initialize an empty augmented dataset (A)*

4 *Set the desired number of augmented samples to generate (N)*

5 *Select a sample from the original dataset (S)*

Apply a set of transformation operations to the selected sample

Randomly rotate the sample within a specific range

Randomly flip the sample horizontally

6 *Repeat steps 5-8 until N augmented samples are generated (250 images)*

7 *Save the augmented dataset (A)*

8 *End*

3.7 Image Classification Model

Classification is a deep learning method where input samples are labelled into predefined categories based on extracted features. In soybean seed classification, features like size, shape, colour, and texture are extracted from seed images and fed into CNN models like VGG16 and InceptionV3. These models independently learn to accurately classify seed images by extracting relevant features.

3.7.1 Soybean Seed Identification Network

The custom CNN, designed for soybean seed identification, comprises an input layer, an output layer, six convolution layers, pooling layers, and fully connected layers. Each layer plays a specific role in capturing features from the image dataset. In the convolution layers, essential parameters include input shape, filter count, kernel size, stride, padding, and activation function. The input shape standardizes RGB images to 224x224x3. The initial layer learns basic features like color and edges with 32 filters and an input size of 224x224x3. Subsequent layers have increasing filter counts (64, 96, 128, 160, and 192) and decreasing input sizes (112x112, 56x56, 28x28, 14x14, and 7x7). A 3x3 kernel size is used for feature extraction, with a stride of 1 and "same" padding. The ReLU activation function sets negative values to zero. Pooling layers follow convolution to reduce feature map dimensionality and enhance reliability. Average pooling with a 2x2 filter size and stride of 2 is employed. Fully connected layers follow, including two layers: the first with 128 neurons and 25% dropout, and the second with 8 neurons and a SoftMax classifier. Dropout regularization of 25% is applied to prevent over-fitting. The SoftMax classifier transforms scores into probabilities between zero and one. Figure 3.3 illustrates the architecture of the Soybean Seed Identification Network.

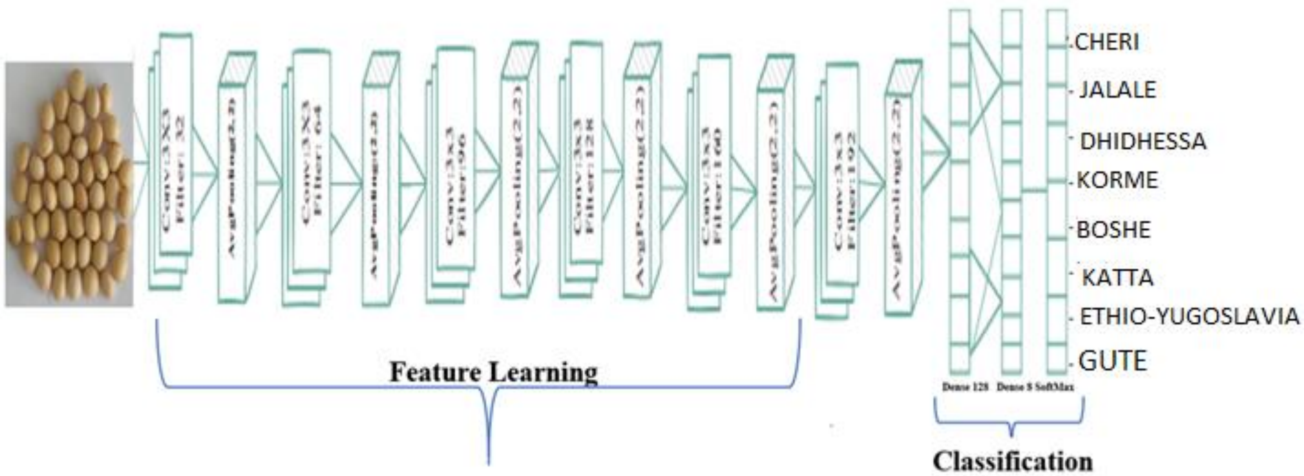


Figure 3.3: Architecture of Custom CNN

Algorithm for the Custom CNN learning phase

Input: *Images of size 224x224x3(RGB)*
Output: *A probability distribution over the 8 classes.*
Procedures:
1. *Initialize the model architecture using the sequential class from tensorflow. keras.* 2. *Add a conv2d layer with 32 filters, a kernel size of 3x3, 'same' padding, and 'relu' activation, taking the input shape of (224, 224, 3).* 3. *Add a dropout layer with a rate of 0.25 to prevent over-fitting.* 4. *Add an averagepooling2d layer with a pool size of 2x2 to reduce spatial dimensions.* 5. *Repeat steps 2-4 for additional conv2d, dropout, and averagepooling2d layers with increasing filter sizes (64, 96, 128, 160, and 192).* 6. *Flatten the output from the previous layer to a 1d vector.* 7. *Add a dense layer with 128 units and 'relu' activation.* 8. *Add a dropout layer with a rate of 0.25.* **9.** *Add a dense layer with 8 units (corresponding to the number of classes) and 'softmax' activation for classification.*

Algorithm for the *Custom CNN* learning classification phase

Input: *test images of size 224x224x3 (RGB) are provided for classification*
Output*: the predicted class labels for the test images based on the trained model.*
Procedures:
1. *Load the SBSINET model weights obtained during the learning phase* 2. *Pre-process the test images (resizing, Augmentation, filtering)to match the input requirements of the model* 3. *Pass the pre-processed test images through the model* 4. *Obtain the predicted probabilities for each class using the softmax activation function.* 5. *Select the class with the highest probability as the predicted class for each test images.*

Table 3.2: Output Shape and Number of Parameters for Custom CNN

Layer (type)	Output Shape	Number of Parameters
Input Layer		
Conv2d (Conv2d)	224, 224, 32	896
Dropout (Dropout)	224, 224, 32	0
Average_Pooling2d (Average pooling2d)	112, 112, 32	0
Conv2d_1 (Conv2D)	112, 112, 64	18496
Dropout_1 (Dropout)	112, 112, 64	0
Average_Pooling2d_1 (Average pooling2d)	56, 56, 64	0
Conv2d_2 (Conv2D)	56, 56, 96	55392
Dropout_2 (Dropout)	56, 56, 96	0
Average_Pooling2d_2 (Average pooling2d)	28, 28, 96	0
Conv2d_3 (Conv2D)	28, 28, 128	110720
Dropout_3 (Dropout)	28, 28, 128	0
Average_Pooling2d_3 (Average pooling2d)	14, 14, 128	0
Conv2d_4 (Conv2D)	14, 14, 160	184480
Dropout_4 (Dropout)	14, 14, 160	0
Average_Pooling2d_4 (Average pooling2d)	7, 7, 160	0
Conv2d_5 (Conv2D)	7, 7, 192)	276672
Dropout_5 (Dropout)	7, 7, 192)	0
Average_Pooling2d_5 (Average pooling2d)	3, 3, 192	0
Flatten (Flatten)	1728	0
Dense (Dense)	128	221312
Dropout_6 (Dropout)	128	0
Dense_1 (Dense)	(8)	1032
Total params:		869,000
Trainable params:		869,000
Non-trainable params:		0

3.7.2 InceptionV3 Model

InceptionV3, a model optimized for ImageNet by Google, is designed to efficiently scale deep networks by utilizing computational resources more effectively. Its architecture is capable of processing images sized at 299 by 299 pixels (Kim, 2022). Figure 3.4 depicts the architecture of InceptionV3. Notably, the model reduces the number of connections between nodes, leading to enhanced efficiency and computational performance. To efficiently extract data features, the convolution operation in InceptionV3 employs connected matrices of various sizes, such as 1x1, 3x3, and 5x5, enabling dense operations on input data (Aung et al., 2019). Moreover, max-pooling is incorporated to improve the model's capability to capture relevant information from the input.

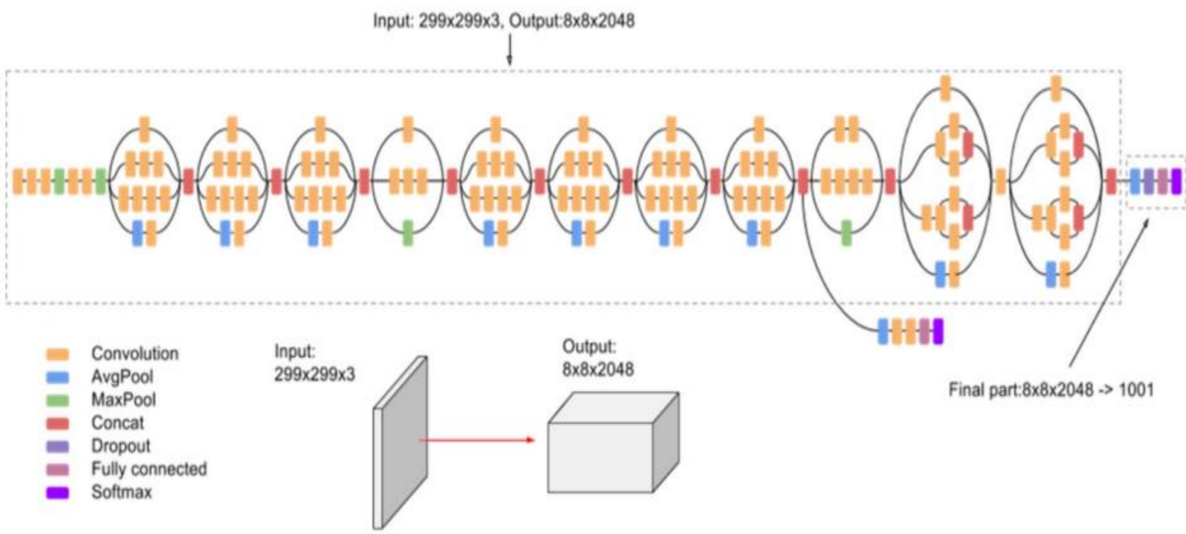


Figure 3.4: Structure of the InceptionV3 Model (Aung et al., 2019)

Table 3.3: Output Shape and Number of Parameters for InceptionV3 Model

Layers (type)	Output Shape	Number of Parameters
inception_v3 (Functional)	(None, 5, 5, 2048)	21802784
global_average_pooling2d(GlobalAveragePooling2D)	(None, 2048)	0
dense (Dense)	(None, 128)	262272
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 8)	1032
Total params:		22,066,088
Trainable params:		263,304
Non-trainable params:		21,802,784

GlobalAveragePooling2D: Averages each feature map to a single value, creating a 1D tensor with 2048 elements.

Dense Layers: A fully connected layer reduces dimensionality to 128, followed by dropout to prevent overfitting, and a final dense layer with 8 units and softmax activation for classification.

Freezing Layers: Freezes InceptionV3 layers to retain pre-trained features, training only the new classification layers.

3.7.3 Visual Geometry Group (VGG16) Network

The Visual Geometry Group at the University of Oxford developed VGG specifically for the ImageNet dataset, characterized by its utilization of 16 layers and 3x3 filters in the convolutional layers. It is designed to process images of size 224 x 224. VGG16 is a well-known deep learning model renowned for its use of 3x3 filter convolution layers with a stride of 1 and consistent same padding (Aung et al., 2019). It also incorporates 2x2 max pooling layers with a stride of 2. The model consists of 13 convolutional layers, 5 pooling layers, and 3 dense layers, evenly distributed throughout the architecture. The "16" in VGG16 refers to the 16 layers with varying weights, totaling over 138 million parameters, making it a large network. The initial conv2d layer in VGG16 comprises 64 filters and accepts input images of dimensions 224x224x3. This output is then passed through subsequent convolutional layers with 64 filters each, followed by max pooling. The following convolutional layers have 128, 256, 512, and 512 filters, with input sizes decreasing to 112x112, 56x56, 28x28, and 14x14, respectively. The final fully connected layer contains 4096 neurons, followed by two additional fully connected layers, each with 4096 neurons. The final output is generated using SoftMax activation with eight classes.

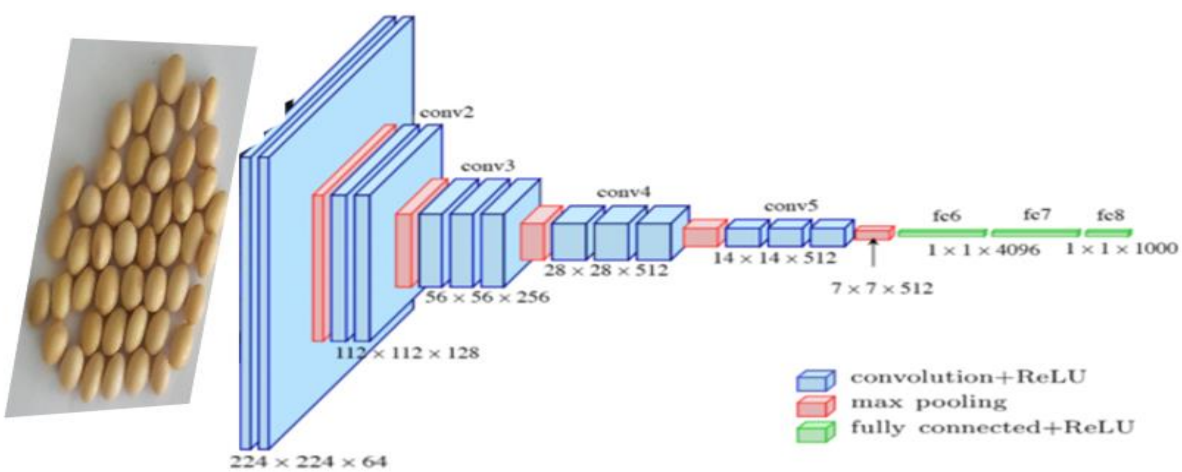


Figure 3.5: Architecture of VGG

Table 3.4: Output Shape and Number of Parameters for VGG16 Model

Layer (type)	Output Shape	Number of Parameters
block1_conv1 (Conv2D)	224, 224, 64	1792
block1_conv2 (Conv2D)	224, 224, 64	36928
block1_pool (MaxPooling2D)	112, 112, 64	0
block2_conv1 (Conv2D)	112, 112, 128	73856
block2_conv2 (Conv2D)	112, 112, 128	147584
block2_pool (MaxPooling2D)	56, 56, 128)	0
block3_conv1 (Conv2D)	56, 56, 256	295168
block3_conv2 (Conv2D)	56, 56, 256	590080
block3_conv3 (Conv2D)	56, 56, 256	590080
block3_pool (MaxPooling2D)	28, 28, 256	0
block4_conv1 (Conv2D)	28, 28, 512	1180160
block4_conv2 (Conv2D)	28, 28, 512	2359808
block4_conv3 (Conv2D)	28, 28, 512	2359808
block4_pool (MaxPooling2D)	14, 14, 512	0
block5_conv1 (Conv2D)	14, 14, 512	2359808
block5_conv2 (Conv2D)	14, 14, 512	2359808
block5_conv3 (Conv2D)	14, 14, 512	2359808
block5_pool (MaxPooling2D)	7, 7, 512	0
flatten (Flatten)	25088	0
dense_3 (Dense)	128	3211392
dropout_2 (Dropout)	128	0
dense_4 (Dense)	128	16512
dropout_3 (Dropout)	128	0
dense_5 (Dense)	8	1032
Total params:	**17,943,624**	

3.8 Performance Evaluation Metrics

Assessing the performance and generalization capability of a model on a given task is crucial in deep learning approaches. Evaluation metrics embedded within these tasks are fundamental for achieving an optimized classifier (Gulzar et al., 2020). Throughout the training process, the model's performance is evaluated by analyzing its output at each epoch. Evaluation in deep learning involves determining true positives/negatives, where the model accurately classifies actual positives/negatives, and false positives, where it misclassifies. Performance metrics like accuracy, precision, recall, and F1 score measure effectiveness. Various evaluation techniques, from traditional ones like accuracy to advanced ones like precision and recall, are utilized (Alzubaidi et al., 2021).

Accuracy: Accuracy assesses the proportion of correctly identified cases, encompassing true positives and true negatives, relative to the entire population. It is computed by dividing the number of correctly predicted positive and negative samples by the total sample size. Essentially, accuracy measures the ratio of accurately predicted classes to the total number of samples evaluated.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}\text{x}100 \tag{3.1}$$

Precision: Precision represents the ratio of correctly predicted positive samples to all samples that were predicted as positive. Mathematically, it is defined as:

$$\text{Precision} = \frac{TP}{TP+FP}\text{x}100 \tag{3.2}$$

Recall: Recall, also known as sensitivity or true positive rate, quantifies the proportion of all correctly identified positive samples to all samples that are actually positive. It measures the ability of the model to correctly identify positive patterns among all positive instances.

$$\text{Recall} = \frac{TP}{TP+FN}\text{x}100 \tag{3.3}$$

F1-Score: The F1-score represents the harmonic mean of precision and recall. It combines precision and recall equally, calculating the harmonic average of their rates to provide a balanced assessment of the model's performance.

$$F1-Score = \frac{2TP}{2TP+FP+fN}\text{x}100 \text{ Or } F1-Score = 2\,\frac{Precision* \; Recall}{Precision* \; Recall}\text{x}100 \tag{3.4}$$

CHAPTER FOUR

RESULTS AND DISCUSSIONS (EXPERIMENTATIONS)

4.1 Introduction

The goal of this thesis was to assess the efficacy of our proposed system design, a crucial component of the overall framework, to gain insights into its performance, merits, and limitations. A detailed experimental evaluation of the proposed architecture for classifying soybean seed varieties is provided, covering the implementation process, dataset preparation, and training/testing methodologies. The classification results are compared with those obtained from a manual system, enabling a comprehensive assessment. Additionally, the experimental outcomes of our proposed model are compared with those of other models, offering further analysis and discussion.

4.2 Dataset Preparation

This study utilized the Samsung Galaxy Tab 2016 SM-T580 and Samsung Galaxy A13 smart-phones to capture an image dataset of soybean seed varieties. The images were captured at resolutions of 8 megapixels (2448 × 3264) and (3060 x 4080) in JPG format. A total of 7200 images were collected, representing eight soybean varieties: Cheri, Jalale, Dhidhessa, Korme, Boshe, Katta, Ethio-Yugoslavia, and Gute. These images were divided into two sets, with 80% used for training and the remaining 20% for testing/validation. Soybean seed samples were sourced from the Bako Agricultural Research Center (GARC) and authenticated by domain experts, particularly soybean breeder professionals at the center. he Bako Research Center, located in the Oromia Region of Ethiopia, is a prominent agricultural research institution established to address the challenges of food security and agricultural productivity in the region. Founded in 1968, the center is part of the Ethiopian Institute of Agricultural Research (EIAR) network and plays a crucial role in developing and disseminating innovative agricultural technologies and practices tailored to the diverse agro-ecological zones of Ethiopia. The center's research focuses on a variety of crops, including maize, teff, sorghum, and other cereals, as well as legumes and oilseeds. In addition to crop improvement, Bako Research Center also emphasizes soil fertility management, pest and disease control, and sustainable farming practices. Its multidisciplinary approach integrates agronomy, soil science, plant pathology, entomology, and socio-economics to ensure holistic and practical solutions for farmers.

Bako Research Center collaborates with national and international partners, including universities, non-governmental organizations, and international research institutions, to enhance its research capacity and extend its impact. Through training programs, field demonstrations, and extension services, the center actively disseminates its findings to farmers, helping to improve agricultural productivity and livelihoods in the region.

Overall, the Bako Research Center is a pivotal institution in Ethiopia's agricultural research landscape, contributing significantly to the advancement of sustainable agriculture and food security in the country.

The following section provides a detailed summary of the soybean seed varieties and quantities utilized in our experiments.

Table 4.1: Dataset Description

No.	Type of Soybean Seed Varieties	Spatial Resolution (Resized Resolution)	Image Format	Quantity	Sources
5	Boshe	227x227	BMP	900	BARC
1	Cheri	227x227	BMP	900	BARC
3	Dhidhessa	227x227	BMP	900	BARC
7	Ethio-Yugoslavia	227x227	BMP	900	BARC
8	Gute	227x227	BMP	900	BARC
2	Jalale	227x227	BMP	900	BARC
6	Katta	227x227	BMP	900	BARC
4	Korme	227x227	BMP	900	BARC
Total				7200	

4.3 Implementation

The aim of this thesis was to classify different varieties of soybean seeds using a deep learning model. To achieve this, three distinct customized CNN models, VGG16, and InceptionV3, were trained. In addition to the original images, two noise removal techniques, Gaussian and Median filtering, were applied as pre-processing steps to improve the accuracy of the classification results. The implementation of the models was carried out using the Python programming language on a system with the following specifications: an 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz processor, 16GB RAM, and 1TB hard disk space. To train each of the models, a 12GB NVIDIA Tesla K80 Graphics Processing Unit (GPU) provided by Google Colab was utilized. The system was developed using Keras with TensorFlow as the backend. For training the models, they were trained for 30 epochs with a batch size of 32 and an initial learning rate of 0.001 (1e-3). The dataset was divided into training and testing sets, with 80% of the data allocated for training the model and the remaining 20% for testing its performance.

Generating Images for Test and Train Datasets

```
In [2]: datagen = ImageDataGenerator(
            rescale = 1./255,
            rotation_range = 360)

In [3]: train_generator = datagen.flow_from_directory(r'C:\Users\user\Desktop\Soybean_Seads\train',  target_size = (227, 227), batch_si

        Found 5760 images belonging to 8 classes.

In [4]: test_generator = datagen.flow_from_directory(
            r'C:\Users\user\Desktop\Soybean_Seads\test',
            target_size = (227, 227),
            batch_size = 30,
            class_mode = 'categorical',
            shuffle = False)

        Found 1440 images belonging to 8 classes.
```

Figure 4.1: Generated images for test and train datasets

4.4 Training the model

The process of developing the models involved the utilization of different algorithms, including a customized CNN model, as well as the VGG16 and InceptionV3 models. These algorithms were applied to two versions of the dataset: the original image dataset and the datasets that underwent noise removal techniques like Gaussian filtering and Median filtering. The training of the models was carried out for a total of 30 epochs.

4.4.1 Training on Custom CNN model

Experiment-I focused on training the customized CNN model using the original dataset images before applying any pre-processing techniques.

The training process of the CNN model using the original images (prior to noise removal) for duration of 30 epochs is clearly depicted in figure 4.2.

```
Epoch 1/30
130/130 [==============================] - 653s 5s/step - loss: 1.0969 - accuracy: 0.5272 - val_loss: 0.8574 - val_accuracy: 0.
6538
Epoch 2/30
130/130 [==============================] - 649s 5s/step - loss: 0.9172 - accuracy: 0.6255 - val_loss: 0.7870 - val_accuracy: 0.
6721
Epoch 3/30
130/130 [==============================] - 654s 5s/step - loss: 0.7659 - accuracy: 0.6974 - val_loss: 0.7620 - val_accuracy: 0.
7077
Epoch 4/30
130/130 [==============================] - 637s 5s/step - loss: 0.7025 - accuracy: 0.7296 - val_loss: 0.6444 - val_accuracy: 0.
7385
Epoch 5/30
130/130 [==============================] - 637s 5s/step - loss: 0.6235 - accuracy: 0.7726 - val_loss: 0.5856 - val_accuracy: 0.
7865
```

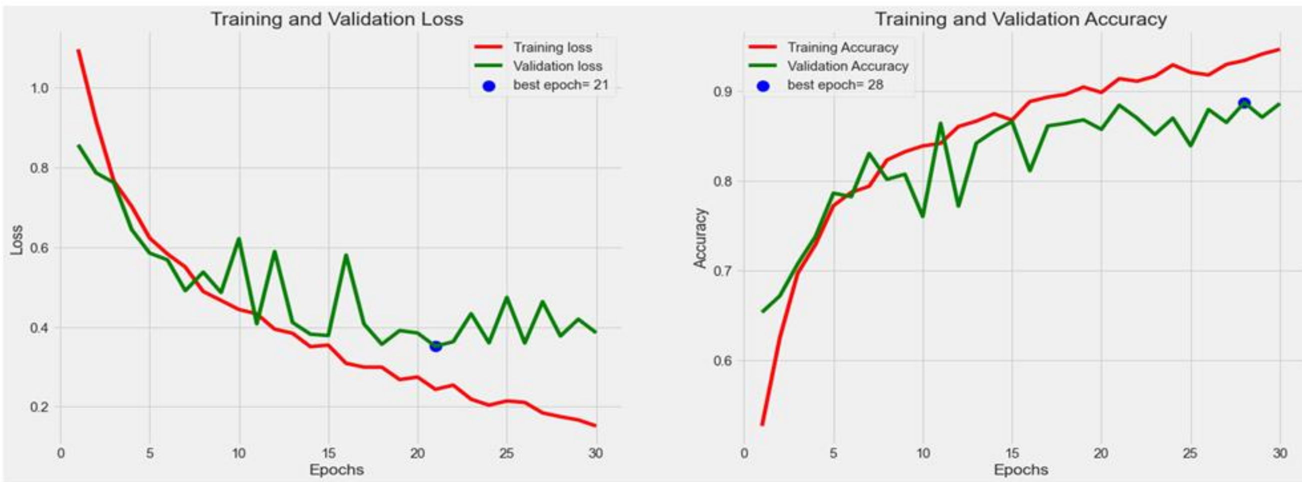


Figure 4.2: Training of customized CNN model on Original Datasets

Based on the figure above, it is evident that the training and validation accuracy of the model display inconsistent improvements, whereas the training and test loss demonstrate a decreasing trend with high variability. This suggests a potential problem of over-fitting and relatively lower accuracy performance. The overall accuracy of the model is reported as 88.6%, indicating room for improvement. To address the issue of over-fitting and enhance the model's accuracy, it is recommended to explore strategies such as regularization techniques (L1 or L2 regularization) to prevent over-reliance on specific features and promote generalization. Additionally, data augmentation methods can be employed to increase the diversity and size of the training dataset, improving the model's ability to generalize to unseen data. Experimenting with different hyper-parameters, such as learning rate, batch size, or network architecture modifications, may also prove beneficial in optimizing the model's performance.

Experiment-II involved training the CNN model after applying a Gaussian filtering technique to the dataset.

The training process of the CNN model using Gaussian filtered images for duration of 30 epochs is clearly depicted in figure 4.3.

```
Epoch 1/30
180/180 [==============================] - 31s 156ms/step - loss: 2.0243 - accuracy: 0.2373 - val_loss: 1.5224 - val_accuracy:
0.2389
Epoch 2/30
180/180 [==============================] - 23s 126ms/step - loss: 1.4566 - accuracy: 0.3198 - val_loss: 1.3528 - val_accuracy:
0.4417
Epoch 3/30
180/180 [==============================] - 23s 129ms/step - loss: 1.3764 - accuracy: 0.3766 - val_loss: 1.2647 - val_accuracy:
0.4500
Epoch 4/30
180/180 [==============================] - 24s 133ms/step - loss: 1.2844 - accuracy: 0.4418 - val_loss: 1.1447 - val_accuracy:
0.4646
Epoch 5/30
180/180 [==============================] - 23s 130ms/step - loss: 1.2193 - accuracy: 0.4741 - val_loss: 1.0173 - val_accuracy:
0.5778
Epoch 6/30
180/180 [==============================] - 25s 139ms/step - loss: 1.1180 - accuracy: 0.5323 - val_loss: 0.9970 - val_accuracy:
0.5653
```

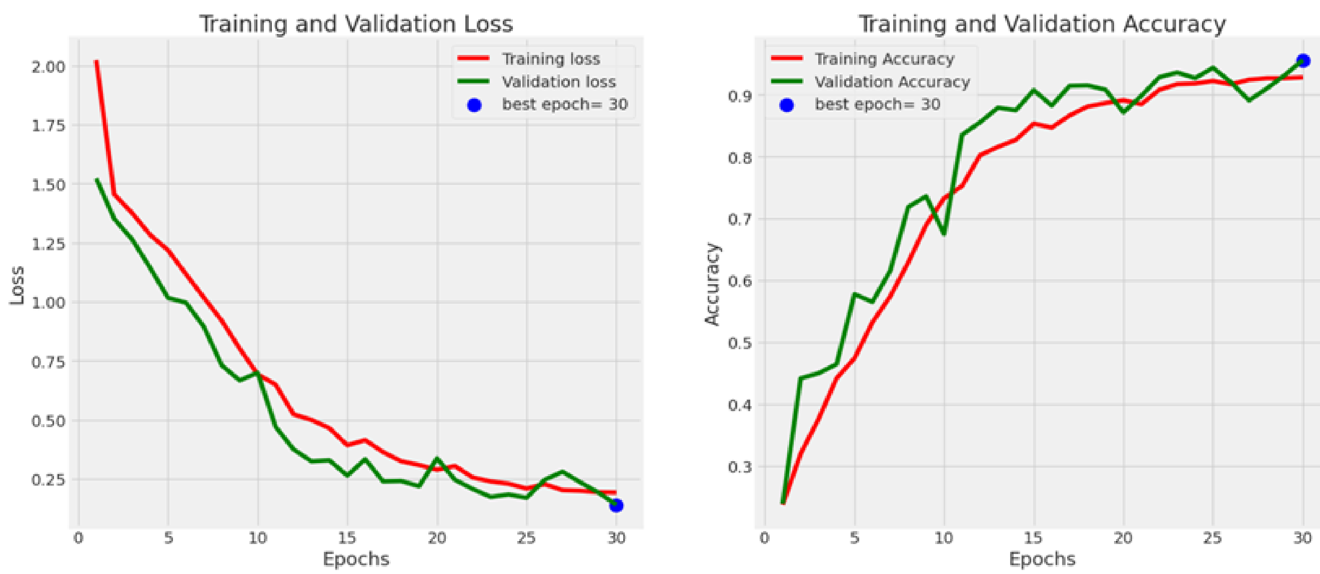


Figure 4.3: Training on CNN after Applying Gaussian Filtered

Based on figure 4.3, the accuracy in both training and validation displays inconsistent progress over time, whereas the loss in training and testing shows a downward trend with considerable fluctuation. Moreover, the diagram suggests a moderate level of over-fitting in the model. Despite these findings, the model achieves an overall accuracy of 98.63%, surpassing CNN on the initial images. Hence, it's advisable to investigate alternative approaches to enhance the model's performance further.

Experiment-III involves training on CNN after applying a median filter.

Figure 4.4 illustrates the training processes of the CNN model on the Median Filtered images, utilizing 30 epochs.

```
Epoch 1/30
180/180 [==============================] - 1193s 6s/step - loss: 1.4484 - accuracy: 0.3998 - val_loss: 1.0143 - val_accuracy:
0.5111
Epoch 2/30
180/180 [==============================] - 941s 5s/step - loss: 0.9730 - accuracy: 0.5314 - val_loss: 0.9362 - val_accuracy: 0.
5208
Epoch 3/30
180/180 [==============================] - 1051s 6s/step - loss: 0.8452 - accuracy: 0.6064 - val_loss: 0.7829 - val_accuracy:
0.6403
Epoch 4/30
180/180 [==============================] - 1071s 6s/step - loss: 0.7233 - accuracy: 0.6785 - val_loss: 0.6434 - val_accuracy:
0.7069
Epoch 5/30
180/180 [==============================] - 1023s 6s/step - loss: 0.5867 - accuracy: 0.7502 - val_loss: 0.6396 - val_accuracy:
0.7340
```

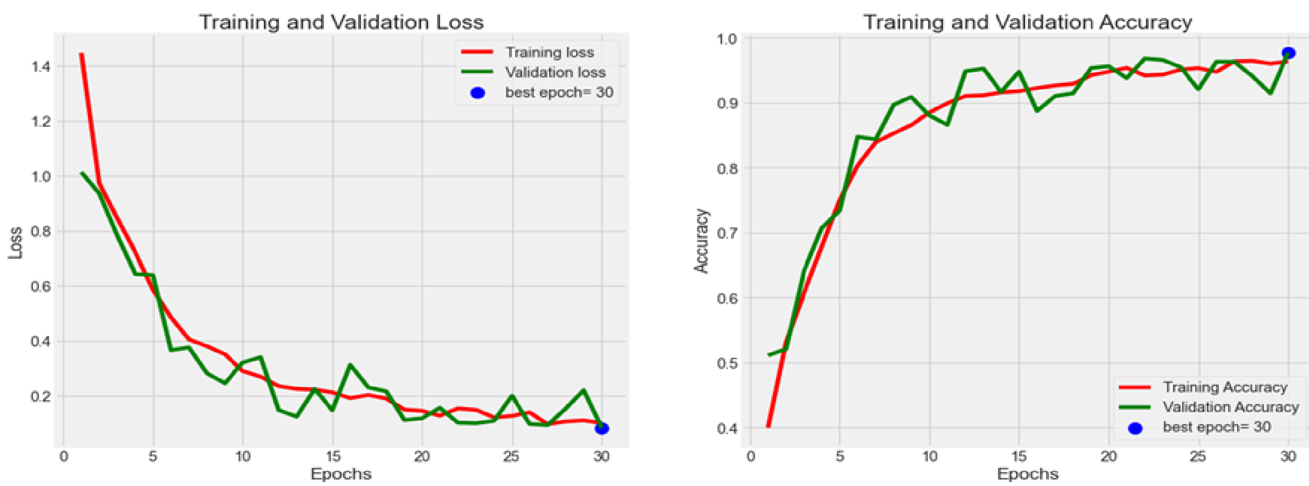


Figure 4.4: Training on Custom CNN after Applying Median Filtered

The graphical representation suggests that there is no indication of over-fitting when compared to previous outcomes. The attained accuracy of 100% and the minimal loss value of 1.0099e-04 provide further validation to these observations. Consequently, the performance of the customized CNN model, utilizing the Median filter, surpasses that of the original image and the image treated with the Gaussian filter in terms of accuracy.

4.4.2 Training on InceptionV3 Model

Experiment I: Training on InceptionV3 Original Images

Figure 4.5 likely shows the training progress of the InceptionV3 model over 30 epochs using noised images. The x-axis represents epochs (1-30), while the y-axis represents performance metrics like loss and accuracy.

Training Loss/Accuracy Curves: These curves illustrate how the model's performance on the training dataset changes. Generally, you expect the training loss to decrease and accuracy to increase over time.

Validation Loss/Accuracy Curves: These curves show performance on the validation dataset. Ideally, validation loss should decrease and validation accuracy should increase, indicating good generalization.

Early Epochs (1-10): Rapid improvement in both metrics.

Middle Epochs (11-20): Slower improvements; potential start of overfitting if validation loss increases. Late Epochs (21-30): Metrics plateau; overfitting more apparent if validation and training metrics diverge significantly. This figure helps in understanding model behaviour, diagnosing overfitting, and guiding hyperparameter tuning.

```
Epoch 1/30
130/130 [==============================] - 37s 148ms/step - loss: 1.7321 - accuracy: 0.3401 - val_loss: 1.3289 - val_accuracy:
0.4019
Epoch 2/30
130/130 [==============================] - 20s 155ms/step - loss: 1.1192 - accuracy: 0.5111 - val_loss: 1.1072 - val_accuracy:
0.5298
Epoch 3/30
130/130 [==============================] - 17s 133ms/step - loss: 1.0657 - accuracy: 0.5450 - val_loss: 0.9230 - val_accuracy:
0.5846
Epoch 4/30
130/130 [==============================] - 19s 149ms/step - loss: 0.9387 - accuracy: 0.6175 - val_loss: 0.8904 - val_accuracy:
0.6481
Epoch 5/30
130/130 [==============================] - 18s 141ms/step - loss: 0.9413 - accuracy: 0.6082 - val_loss: 0.8696 - val_accuracy:
0.6337
```

Figure 4.5: Training on InceptionV3 Original Images

Figure 4.5 shows the InceptionV3 model's inconsistent training and validation accuracy, decreasing loss with high variance, and signs of overfitting, achieving 78.85% accuracy.

Recommendations to Enhance Performance:

- Data Augmentation: Increase training data variety.
- Regularization: Use dropout, weight decay, or L2 regularization.
- Learning Rate Adjustments: Try different learning rates or schedules.
- Noise Reduction: Pre-process to reduce image noise.
- Early Stopping: Halt training when validation performance degrades.
- Model Tweaks: Modify InceptionV3 or use transfer learning.
- Hyper-parameter Tuning: Optimize batch size, epochs, and learning rates.

Experiment II: Training on InceptionV3 after Applying Gaussian Filtered

The diagram shows the training progress of the pre-trained InceptionV3 model using Gaussian filtered images, highlighting:

Training and Validation Accuracy: Metrics over 30 epochs, showing improved stability.

Training and Validation Loss: Smoother curves, indicating reduced variance and overfitting.

Smoother Curves: Due to Gaussian filtering.

Better Generalization: Reduced gap between training and validation metrics.

Improved Performance: Higher accuracy and lower loss compared to unfiltered images.

Gaussian filtering enhances model stability and performance.

```
Epoch 1/30
180/180 [==============================] - 600s 3s/step - loss: 4.5519 - accuracy: 0.4627 - val_loss: 0.8402 - val_accuracy: 0.
6278
Epoch 2/30
180/180 [==============================] - 604s 3s/step - loss: 0.8637 - accuracy: 0.6056 - val_loss: 0.7454 - val_accuracy: 0.
6681
Epoch 3/30
180/180 [==============================] - 602s 3s/step - loss: 0.7652 - accuracy: 0.6465 - val_loss: 0.6474 - val_accuracy: 0.
7021
Epoch 4/30
180/180 [==============================] - 582s 3s/step - loss: 0.7337 - accuracy: 0.6479 - val_loss: 0.5311 - val_accuracy: 0.
7576
Epoch 5/30
180/180 [==============================] - 579s 3s/step - loss: 0.6739 - accuracy: 0.6863 - val_loss: 0.4997 - val_accuracy: 0.
7764
```

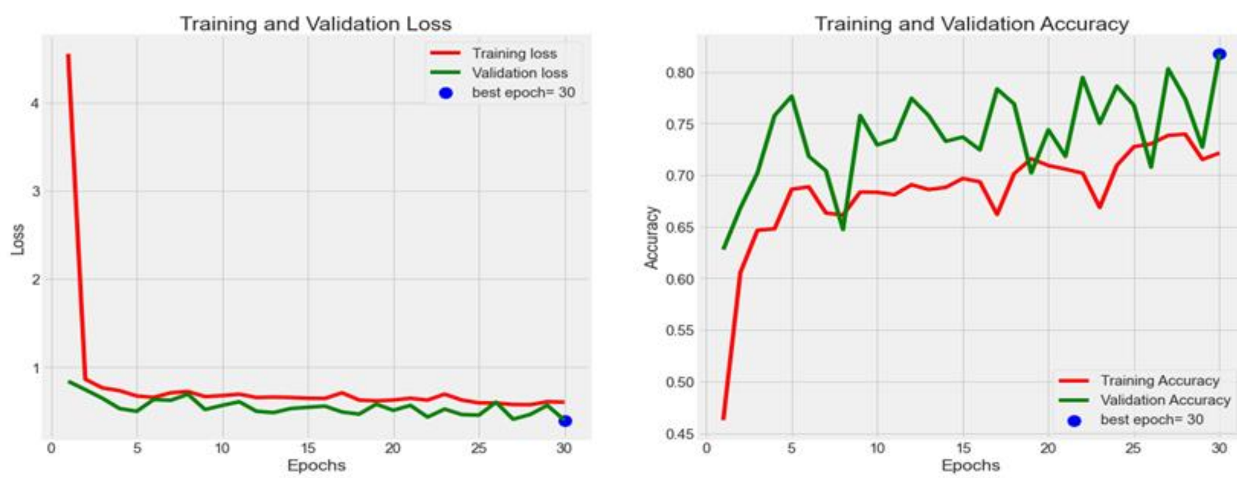
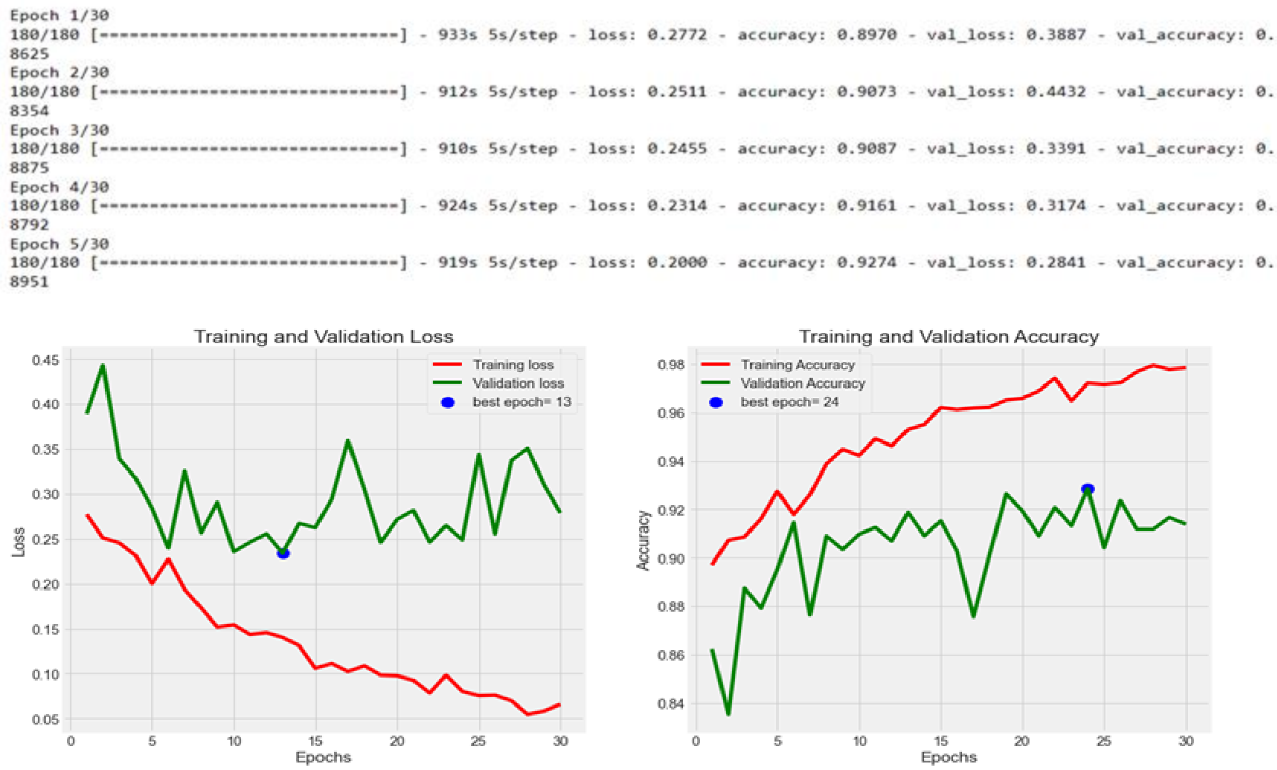


Figure 4.6: Training on InceptionV3 after Applying Gaussian Filtered Images

According to the figure 4.6, the training and validation accuracy of the InceptionV3 model continuously increases, while training and test loss decreases with high variance over 30 epochs. The model achieves an overall accuracy of 81.74%, which is better than using the original noised images. However, the figure also indicates signs of overfitting and shows that the model performs less effectively compared to the Custom CNN model.

Experiment III: Train of InceptionV3 after Applying Median Filtered

As clearly demonstrated in Figure 4.7, the training and validation accuracy of the InceptionV3 model continuously increases, while the training and test loss decrease with high variance over 30 epochs.



```
Epoch 1/30
180/180 [==============================] - 933s 5s/step - loss: 0.2772 - accuracy: 0.8970 - val_loss: 0.3887 - val_accuracy: 0.
8625
Epoch 2/30
180/180 [==============================] - 912s 5s/step - loss: 0.2511 - accuracy: 0.9073 - val_loss: 0.4432 - val_accuracy: 0.
8354
Epoch 3/30
180/180 [==============================] - 910s 5s/step - loss: 0.2455 - accuracy: 0.9087 - val_loss: 0.3391 - val_accuracy: 0.
8875
Epoch 4/30
180/180 [==============================] - 924s 5s/step - loss: 0.2314 - accuracy: 0.9161 - val_loss: 0.3174 - val_accuracy: 0.
8792
Epoch 5/30
180/180 [==============================] - 919s 5s/step - loss: 0.2000 - accuracy: 0.9274 - val_loss: 0.2841 - val_accuracy: 0.
8951
```


Figure 4.7: Training on Inceptionv3 after Applying Median Filtered

According to the visual representation in figure 4.7, depicting the training processes of the pre-trained InceptionV3 model applied to median filtered images,

Training and Validation Accuracy: Both metrics increase continuously up to 30 epochs.

Training and Test Loss: Both decrease with less variance compared to previous methods.

Performance: The model achieves an accuracy of 98.25%, surpassing the results obtained with original and Gaussian filtered images. This indicates that median filtering significantly enhances the model's performance.

4.4.3 Training on VGG16 Model

Experiment I: Training on VGG16 Original Images

Figure 4.8 depicts the VGG16 model's training on original, unfiltered images over 30 epochs. The figure shows,

Training and Validation Accuracy: Continuous improvement over epochs.

Training and Validation Loss: Decreasing trend, indicating error reduction. This figure provides insights into the model's performance and training progress without noise removal.

```
Epoch 1/30
130/130 [==============================] - 3372s 26s/step - loss: 1.9996 - accuracy: 0.6570 - val_loss: 0.6191 - val_accuracy:
0.8010
Epoch 2/30
130/130 [==============================] - 3655s 28s/step - loss: 0.6731 - accuracy: 0.7913 - val_loss: 0.6926 - val_accuracy:
0.7462
Epoch 3/30
130/130 [==============================] - 3530s 27s/step - loss: 0.5765 - accuracy: 0.8087 - val_loss: 0.5297 - val_accuracy:
0.8385
Epoch 4/30
130/130 [==============================] - 3595s 28s/step - loss: 0.3813 - accuracy: 0.8788 - val_loss: 0.6912 - val_accuracy:
0.8500
Epoch 5/30
130/130 [==============================] - 3661s 28s/step - loss: 0.3697 - accuracy: 0.8969 - val_loss: 0.5518 - val_accuracy:
0.8500
```

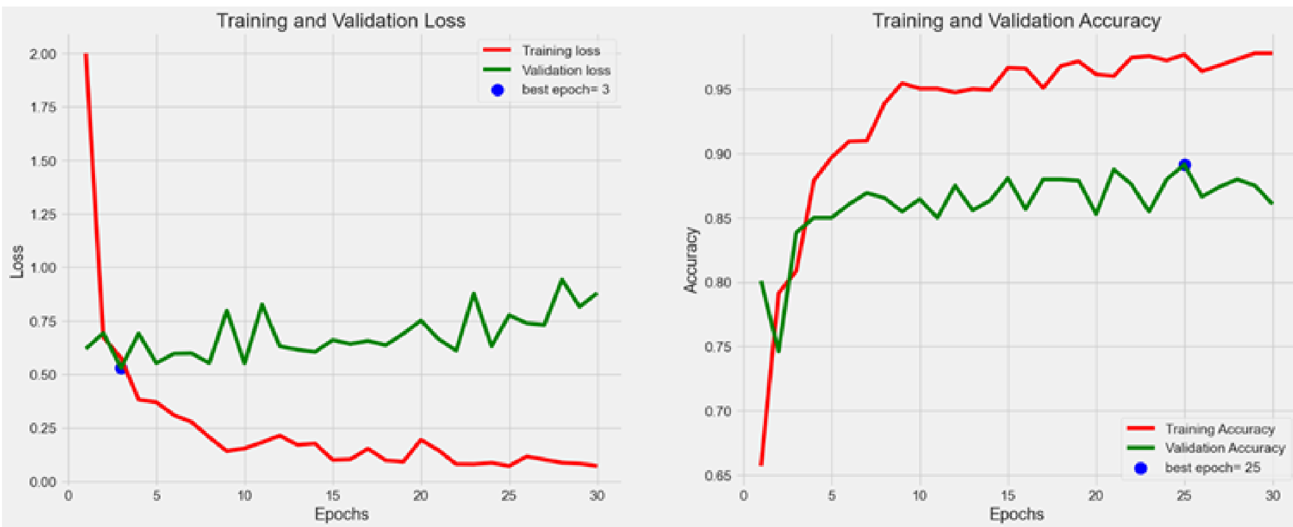


Figure 4.8: Training on VGG16 Original Images

Figure 4.8 shows that the VGG16 model's training and validation accuracy increase inconsistently, with both training and test loss decreasing but displaying high variance over 30 epochs. The model achieved an accuracy of 86.06%, indicating suboptimal performance. It is advisable to explore alternative methods to enhance the model's performance.

Experiment II: Training onVGG16 after Applying Gaussian Filtered

The VGG16 model was trained on images that had undergone Gaussian filtering for noise removal. This training process spanned 30 epochs, with the details and progress illustrated in figure 4.9 below.

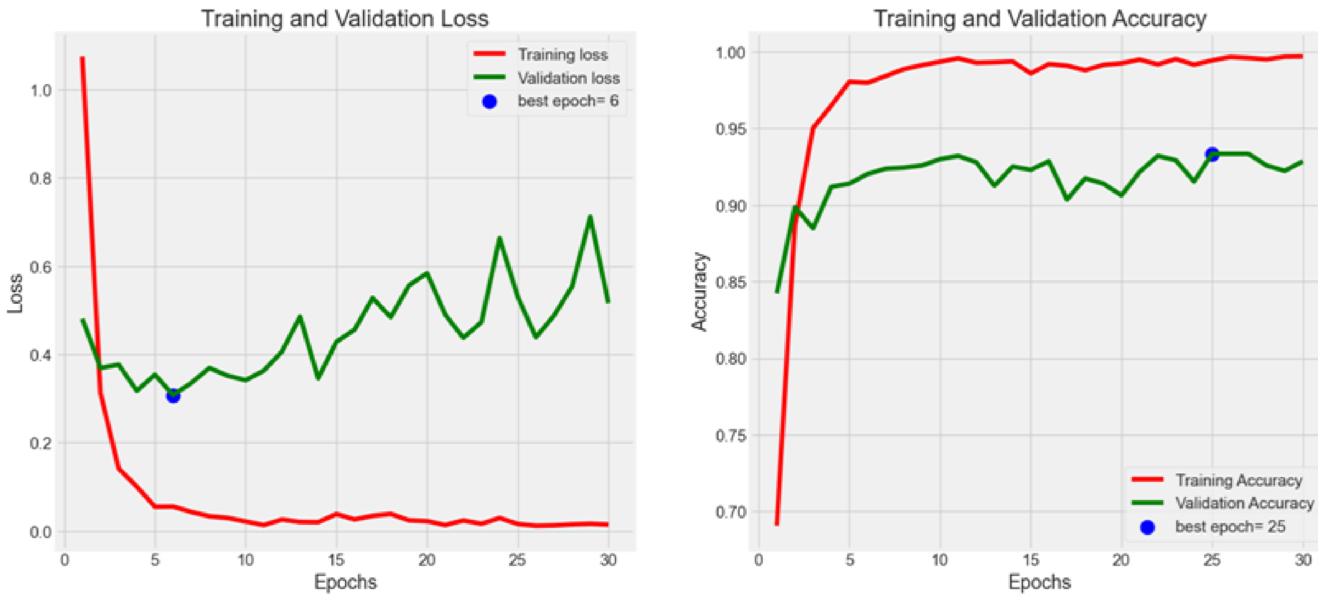


Figure 4.9: Training on VGG16 after Applying Gaussian Filtered

As illustrated, the VGG16 model was trained on Gaussian-filtered images. Throughout the 30 epochs, both the training and validation accuracy of the model consistently increased, while the training and test loss decreased with significant variance. The model achieved an accuracy of 99.39% using the Gaussian-filtered images, surpassing the performance of the previous model.

4.5 Evaluation of the Model

Soybeans are a crucial crop in Ethiopia, contributing to food security and economic stability. Accurate classification of soybean seed types is essential for improving yield, quality control, and marketability. This evaluation explores the performance of three deep learning models Custom Convolutional Neural Network (CNN), Inception V3, and VGG16in classifying Ethiopian soybean seed types.

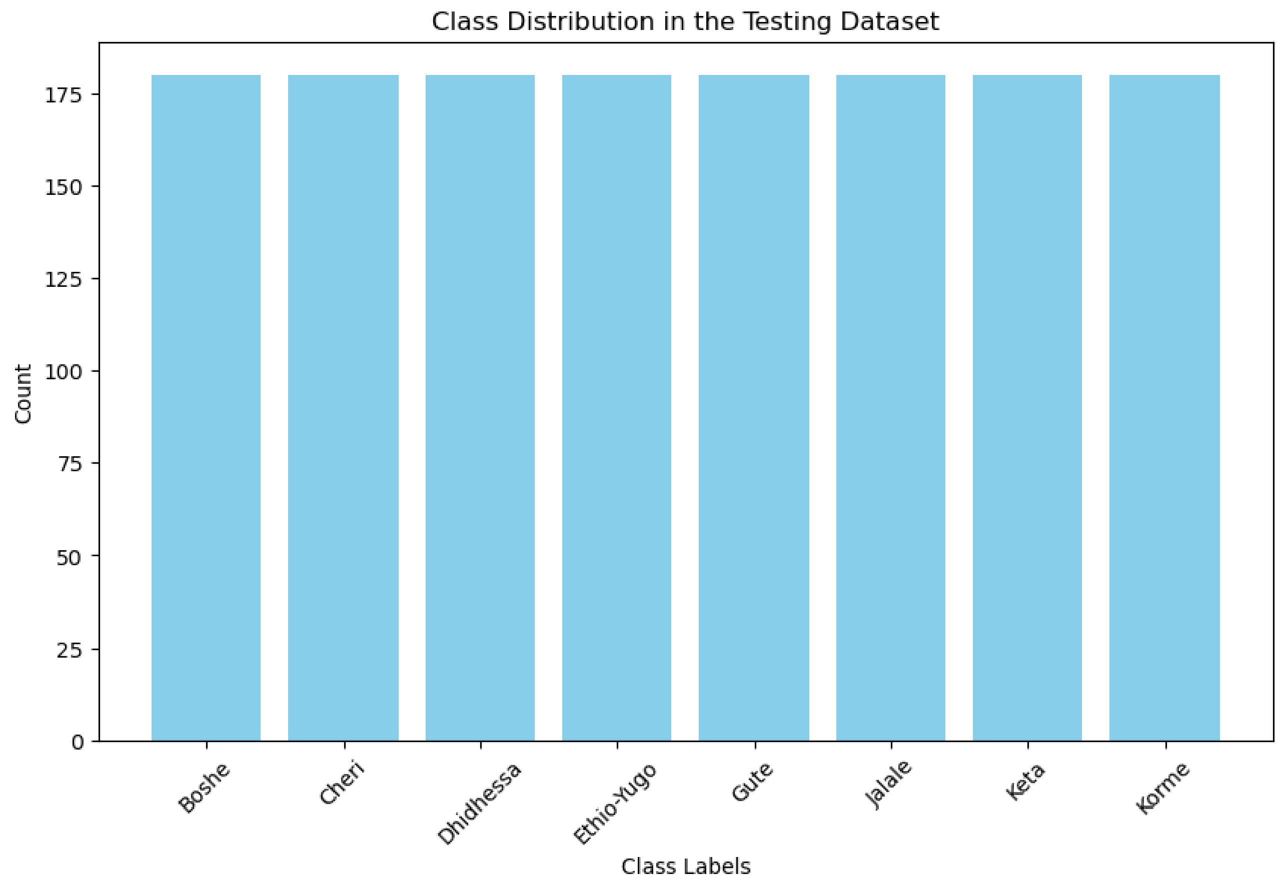


Figure 4.10: Class distribution in the testing dataset

4.5.1 Evaluation of Custom CNN

The figure 4.11 and table 4.1 offers a detailed depiction of the CNN evaluations, highlighting correctly and incorrectly classified instances from the provided image datasets. Custom CNN achieved perfect accuracy, with both training and testing accuracy at 100%. The diagram, along with the confusion matrix, covers eight classes of soybean seed varieties. Out of a total of 1440 testing sessions for these classes, there was one (1) misclassification in the Boshe Soybean type categories.

Table 4.2: Summary of CNN Classification Performance

Soybean seed Varieties	Correctly Identified (100%)	Incorrectly Identified (100%)
Boshe	99.44	0.56
Cheri	100	0
Dhedhessa	100	0
Ethio-Yugo	100	0

Gute	100	0
Jalale	100	0
Keta	100	0
Korme	100	0

Model Classification testing results report,

	precision	recall	f1-score
0	1.00	1.00	1.00
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
4	1.00	1.00	1.00
5	1.00	1.00	1.00
6	1.00	1.00	1.00
7	1.00	1.00	1.00
accuracy			1.00
macro avg	1.00	1.00	1.00
weighted avg	1.00	1.00	1.00

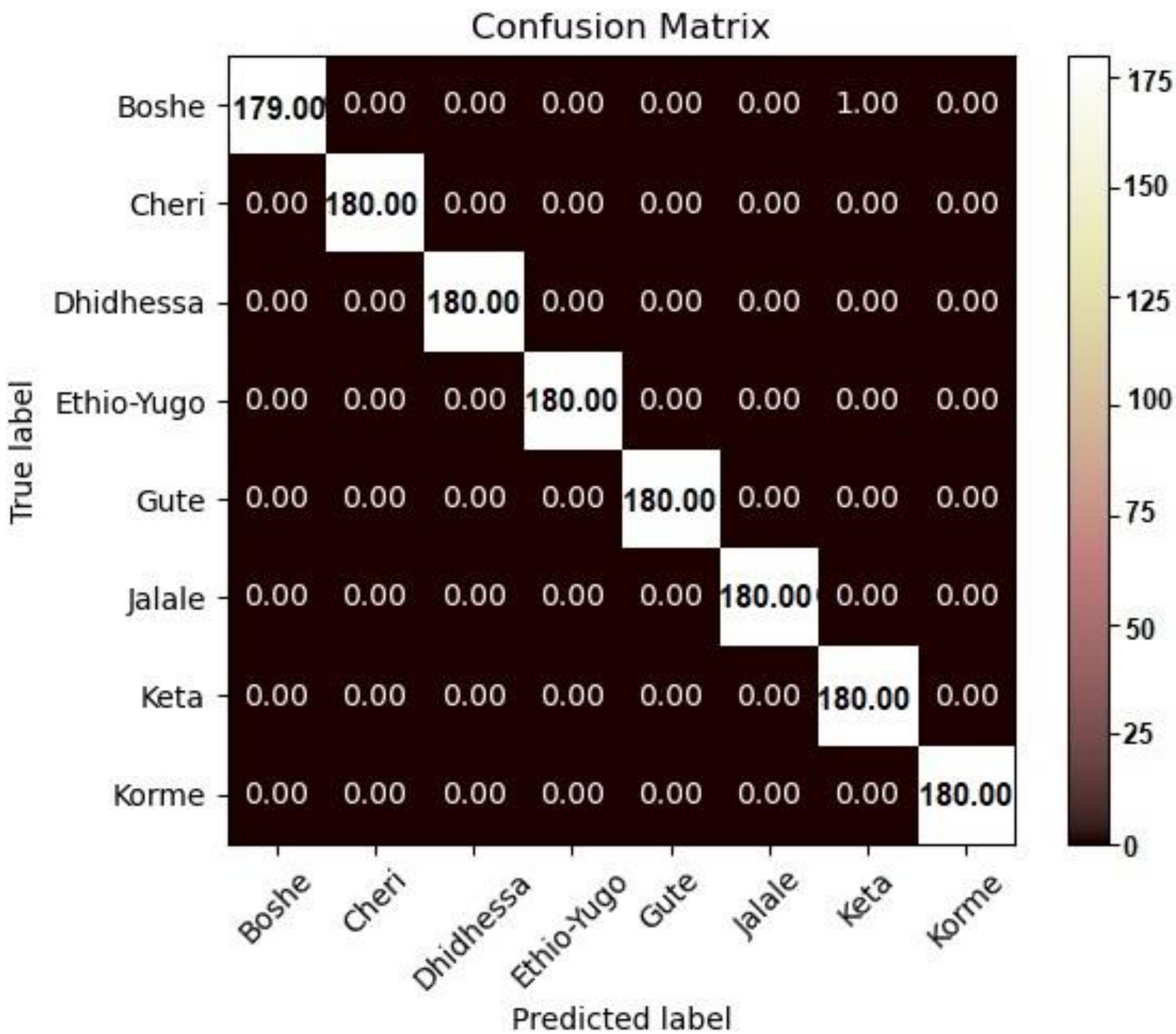


Figure 4.11: Classification Report and Confusion Matrix of Custom CNN

4.5.2 Evaluation of InceptionV3 Model

The figure 4.12 below clearly depicts the InceptionV3 evaluations, showing both correctly and incorrectly classified instances from the given image datasets. InceptionV3 achieved a training accuracy of 96.56% and a testing accuracy of 98.25%.

The confusion matrixes cover eight classes of soybean seed varieties. Out of 1440 testing sessions for these classes, there were misclassifications in each class: Boshe had 2 Cheri had 37, Dhidhessa had 0, Ethio-Yugo had 0, Gute had 2, Jalale had 24, Keta had 21, and Korme had 5 misclassifications,. Generally, InceptionV3 correctly classified 1349 images and misclassified 91 images.

Table 4.3: Summary of InceptionV3 Classification Performance

Soybean seed Varieties	Correctly Identified (100%)	Incorrectly Identified (100%)
Boshe	98.89	1.11
Cheri	79.44	27.56
Dhedhessa	100	0
Ethio-Yugo	100	0
Gute	98.89	1.11
Jalale	86.67	13.37
Keta	88.33	11.67
Korme	97.22	2.78

Testing classification results report of the InceptionV3 model,

	precision	recall	f1-score
0	1.00	0.93	0.96
1	0.94	1.00	0.97
2	0.99	1.00	0.99
3	1.00	1.00	1.00
4	1.00	0.99	1.00
5	1.00	1.00	1.00
6	0.95	0.99	0.97
7	0.99	0.95	0.97
accuracy			0.98
macro avg	0.98	0.98	0.98
weighted avg	0.98	0.98	0.98

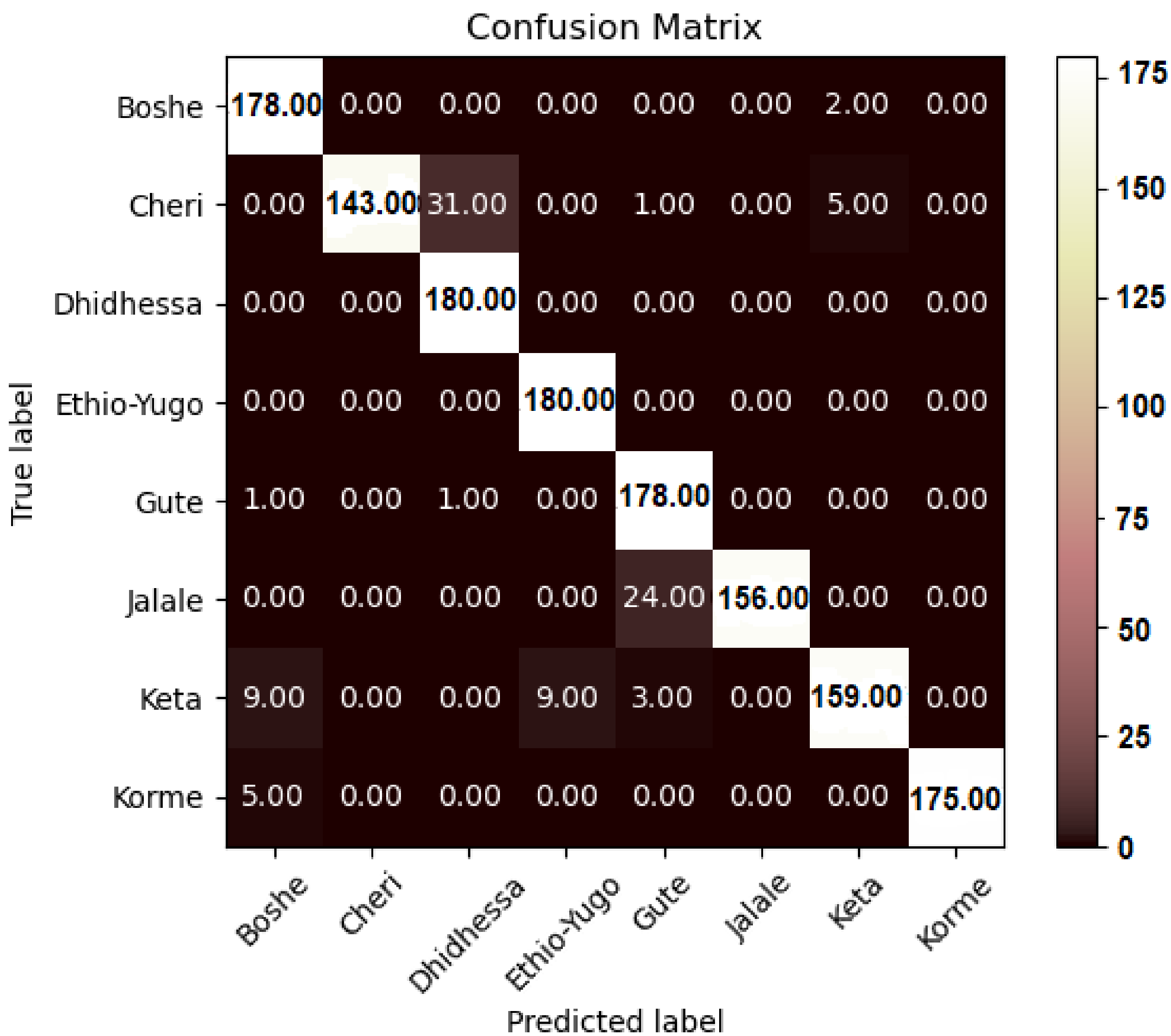


Figure 4.12: Classification Report & Confusion Matrix of InceptionV3

4.5.3 Evaluation of VGG16 Model

Figure 4.13 clearly illustrates the VGG16 evaluations, showing the correctly and incorrectly classified instances from the given image datasets. VGG16 achieved a training accuracy of 99.25% and a testing accuracy of 99.34%. The confusion matrix represents eight classes of soybean seed varieties. Specifically, Boshe had 0 misclassifications, Cheri had 0, Dhedhessa had 9, Ethio-Yugo had 0, Gute had 0, Jalale had 0, Keta had 26, and Korme had 0. In general, out of 1440 images, VGG16 correctly classified 1405 images and misclassified 35 images.

Table 4.4: Summary of VGG16 Classification Performance

Soybean seed Varieties	Correctly Identified (100%)	Incorrectly Identified (100%)
Boshe	100	0
Cheri	100	0
Dhedhessa	95	5
Ethio-Yugo	100	0
Gute	100	0
Jalale	100	0
Keta	85.56	14.44
Korme	100	0

Testing results of the model and classification reports,

	precision	recall	f1-score
0	0.96	1.00	0.98
1	1.00	1.00	1.00
2	1.00	0.99	0.99
3	1.00	1.00	1.00
4	1.00	1.00	1.00
5	1.00	1.00	1.00
6	1.00	0.96	0.98
7	1.00	1.00	1.00
accuracy			0.99
macro avg	0.99	0.99	0.99
weighted avg	0.99	0.99	0.99

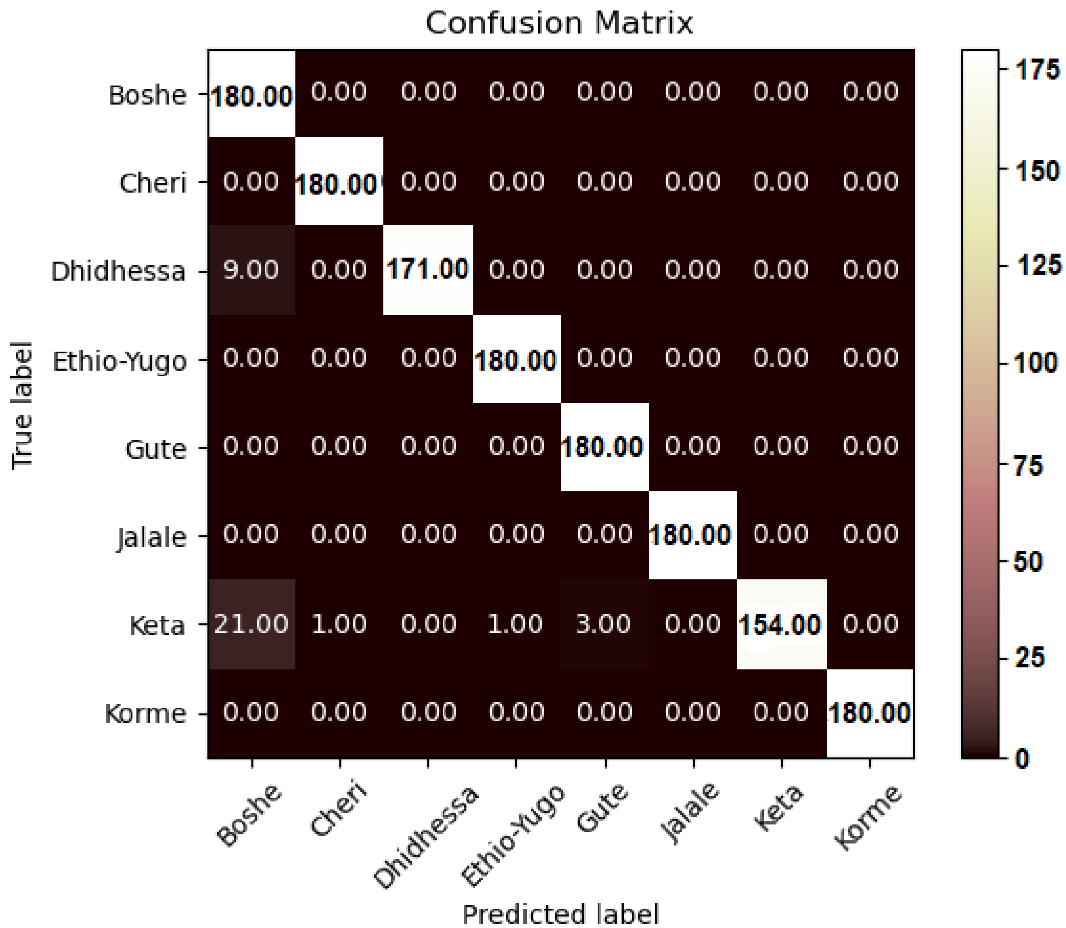


Figure 4.13: Classification Report and Confusion Matrix of VGG16

Table 4.5: Experimental results summary of the models

Models	Experimental Results of Models (%)	
	Training Accuracy (%)	Testing Accuracy (%)
Proposed Custom CNN	**100**	**100**
VGG16	99.29	99.39
InceptionV3	98.35	98.25

4.6 Discussion of the Results

This study focused on identifying the various types of Ethiopian soybean seed varieties. To achieve this, the following research questions were addressed:

Q1: Developing a Model for Ethiopian Soybean Seed Varieties Classification Using a Deep Learning Approach

To tackle the first research question, a dataset comprising 7,200 images of soybean seeds was gathered, with each of the eight classes containing 900 images. Noise removal techniques like Median and Gaussian filtering were applied to maintain the dataset's quality. Additionally, all images were standardized to a resolution of 227x227 pixels and saved in the .bmp format. To prevent overfitting and enhance the diversity of the training data, data augmentation techniques were used. These techniques generated additional image variations, enhancing the model's generalization capability. The dataset was divided into 80% training and 20% testing to ensure sufficient training data while maintaining a separate evaluation set. Three models custom CNN model, InceptionV3, and VGG16 were trained using the Adam optimizer for 30 epochs with a batch size of 32 and a learning rate of 0.001. A SoftMax activation function was used to assign class probabilities, and categorical cross-entropy loss measured performance. These steps aimed to develop accurate models for classifying different soybean seed varieties based on their images.

Q2: Which deep learning architectures are better suited for accurately classifying Ethiopian soybean seed types based phenotype features like colour , size, shape etc.?

The proposed custom CNN model achieved 100% validation accuracy in classifying Ethiopian soybean seed varieties. This demonstrates the model's effectiveness in accurately differentiating seed classes. The use of three models custom CNN, InceptionV3, and VGG16 further enhances performance and reduces errors. These methods show strong potential as a decision support system for identifying and classifying different soybean seed varieties. High-quality soybean seeds have the potential to result in increased crop yield. Quality seeds have a higher resistance to common soybean diseases and pests. High-quality seeds often contain desirable genetic traits, such as tolerance to specific environmental conditions (e.g., drought or heat tolerance).

Q3: What strategies can be applied to measure the performance of deep learning models on the collected dataset?

Several strategies were employed to enhance the accuracy of the soybean seed varieties classification model. Initially, the original dataset of 5,200 soybean seed images was augmented by adding 250 images to each class, resulting in 2,000 additional images and a total of 7,200 images. Three models custom CNN model, VGG16, and InceptionV3 were selected for classification. Pre-processing techniques, including Gaussian and Median filters and resizing, were used to remove noise and improve accuracy, with Median filtering yielding the better results. Additionally, dropout regularization at a rate of 25% and a learning rate of 0.001 were incorporated to prevent overfitting and control the learning step size. These combined approaches data augmentation, model selection, pre-processing techniques, and regularization significantly improved the model's accuracy. Median filtering, in particular, was highly effective, enhancing classification performance across all three models.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Soybeans are a crucial oil crop, rich in protein (over 40%) and oil (over 20%), and play a significant role in food commodities and foreign exchange income. High-quality seeds are essential for optimal crop production, but the increasing global population and shrinking agricultural land threaten food production. Enhancing agricultural productivity is vital to balance production and consumption. Currently, soybean variety identification relies on visual analysis by experts, which is time-consuming and prone to error. Given the growing demand for soybeans, a more efficient and reliable classification method is needed.

Deep learning (DL) offers an effective solution for precise seed classification by analyzing the unique properties of Ethiopian soybeans. DL can accurately identify soybean types, leading to improved agricultural practices and yields. The proposed DL-based model aims to classify eight Ethiopian soybean varieties Boshe, Cheri, Dhedhessa, Ethio-Yugo, Gute, Jalale, Keta, and Korme based on their physical and morphological characteristics. This study employs an experimental approach involving dataset preparation, model implementation, and evaluation.

The dataset, consisting of 7,200 soybean seed images, was split into 80% training and 20% testing sets. Three models custom CNN model, VGG16, and InceptionV3 were trained using various pre-processing techniques, including noise removal (Median and Gaussian filters) and image resizing. Data augmentation, such as horizontal flipping, increased the training data size and improved model generalization.

The classification models were designed using a custom CNN model and pre-trained models (VGG16, InceptionV3). Features extracted from the training and testing datasets were fed into the models for classification using a Softmax classifier. The custom CNN model architecture included convolutional layers, pooling layers, and fully connected layers for feature extraction and classification. The models were evaluated on the validation dataset to measure performance and accuracy. The experiments showed that Median filtering yielded the better results. The custom CNN model achieved 100% accuracy on Median filtered images. InceptionV3 achieved 98.25% accuracy, and VGG16 reached 99.39% accuracy with Median filtered images. These results indicate that Median filtering significantly enhances model performance. In general, this study provides promising results for soybean seed variety identification. The proposed model can assist agricultural research institutes and experts in accurately classifying soybean varieties, benefiting ECX trades and farmers. However, some misclassification errors were due to non-uniform image sizes affecting feature extraction. Improving noise removal techniques can further enhance classification accuracy.

5.2 Recommendations and Future Works

We recommend that future researchers compile comprehensive image datasets for each soybean variety to enhance model performance. Additionally, it would be beneficial to work on identifying soybean grading, maturity dates, and oil content for these varieties. Policymakers should support integrating and deploying this model in mobile environments, allowing domain experts, merchants, and farmers to quickly and accurately classify soybean seed varieties.

It is also recommended to explore other deep learning architectures to compare their performance and potentially achieve higher accuracy. Investigating advanced image processing methods may lead to more accurate and reliable results. Future research should focus on improving model performance and addressing misclassification issues for specific seed varieties. Exploring transfer learning, which leverages knowledge from pre-training on large datasets, could enhance performance on smaller target datasets and reduce the required training data. Extending this approach to other crops could further improve agricultural practices and promote food security. By addressing these recommendations, the classification model can be refined, benefiting the agricultural industry by enhancing productivity and contributing to sustainable food production.

REFERENCES

Abraham, E. R., Dos Reis, J. G. M., Vendrametto, O., Neto, P. L. de O. C., Toloi, R. C., de Souza, A. E., & Morais, M. de O. (2020). Time series prediction with artificial neural networks: An analysis using Brazilian soybean production. *Agriculture (Switzerland)*, *10*(10), 1–18. https://doi.org/10.3390/agriculture10100475

Abuhayi, B. M., & Mossa, A. A. (2023). Coffee disease classification using Convolutional Neural Network based on feature concatenation. *Informatics in Medicine Unlocked*, *39*(February), 101245. https://doi.org/10.1016/j.imu.2023.101245

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. In *Journal of Big Data* (Vol. 8, Issue 1). Springer International Publishing. https://doi.org/10.1186/s40537-021-00444-8

Amiriebrahimabadi, M., Rouhi, Z., & Mansouri, N. (2024). A Comprehensive Survey of Multi - Level Thresholding Segmentation Methods for Image Processing. In *Archives of Computational Methods in Engineering* (Issue 0123456789). Springer Netherlands. https://doi.org/10.1007/s11831-024-10093-8

Archana, R., & Jeevaraj, P. S. E. (2024). Deep learning models for digital image processing: a review. In *Artificial Intelligence Review* (Vol. 57, Issue 1). Springer Netherlands. https://doi.org/10.1007/s10462-023-10631-z

Aung, S. W. Y., Khaing, S. S., & Aung, S. T. (2019). Multi-label land cover indices classification of satellite images using deep learning. In *Advances in Intelligent Systems and Computing* (Vol. 744, Issue January). Springer Singapore. https://doi.org/10.1007/978-981-13-0869-7_11

Ayele Nigus Asres, H. K. (2020). Developing Classification Model for Chickpea Types using Machine Learning Algorithms. *International Journal of Innovative Technology and Exploring Engineering*, *10*(1), 5–11. https://doi.org/10.35940/ijitee.a8057.1110120

Beyene, D., & JPAA, Z. J. (2022). Diversity of soybean (Glycine max L.) genotypes based on agro morphological parameters. *Ojs.Aiou.Edu.PkDG Beyene, Z JalataJPAA, 2022•ojs.Aiou.Edu.Pk*, *7*(2), 30–37. https://ojs.aiou.edu.pk/index.php/jpaa/article/download/994/861

Butuner, R., Cinar, I., Taspinar, Y. S., Kursun, R., Calp, M. H., & Koklu, M. (2023). Classification of deep image features of lentil varieties with machine learning techniques. *European Food Research and Technology*, *249*(5), 1303–1316. https://doi.org/10.1007/s00217-023-04214-z

Çetin, N. (2022). Machine Learning for Varietal Binary Classification of Soybean (Glycine max (L.) Merrill) Seeds Based on Shape and Size Attributes. *Food Analytical Methods*, *15*(8), 2260–2273. https://doi.org/10.1007/s12161-022-02286-3

Cong, S., & Zhou, Y. (2023). A review of convolutional neural network architectures and their optimizations. In *Artificial Intelligence Review* (Vol. 56, Issue 3). Springer Netherlands. https://doi.org/10.1007/s10462-022-10213-5

Dastres, R., Soori, M., Neural, A., Systems, N., & Journal, I. (2021). Artificial Neural Network Systems To cite this version : HAL IAd : hal-03349542. *International Journal of Imaging and Robotics (IJIR)*, *21*(2), 13–25.

de Medeiros, A. D., Capobiango, N. P., da Silva, J. M., da Silva, L. J., da Silva, C. B., & dos Santos Dias, D. C. F. (2020). Interactive machine learning for soybean seed and seedling quality classification. *Scientific Reports*, *10*(1), 1–10. https://doi.org/10.1038/s41598-020-68273-y

Delele, T. A., Adugna, A. G., & Gelaw, B. M. (2022). Determinants of soybean (Glycine max.) market supply in Northwestern Ethiopia. *Cogent Economics and Finance*, *10*(1). https://doi.org/10.1080/23322039.2022.2142313

Dönmez, E. (2024). Hybrid convolutional neural network and multilayer perceptron vision transformer model for wheat species classification task: E-ResMLP+. *European Food Research and Technology*, *0123456789*. https://doi.org/10.1007/s00217-024-04469-0

Gudina, G., Tesfaye, A., Nebiyu, A., & Bekele, G. (2023). Genetic Variation of Recombinant Inbred Lines Soybean (<i>Glycine max </i>L. Merrill) from USA at Jimma, Ethiopia. *Cell Biology*, *January*. https://doi.org/10.11648/j.cb.20231102.12

Gulzar, Y., Hamid, Y., Soomro, A. B., Alwan, A. A., & Journaux, L. (2020). A convolution neural network-based seed classification system. *Symmetry*, *12*(12), 1–18. https://doi.org/10.3390/sym12122018

Han, M., Wu, H., Chen, Z., Li, M., & Zhang, X. (2023). A survey of multi-label classification based on supervised and semi-supervised learning. *International Journal of Machine*

Learning and Cybernetics, *14*(3), 697–724. https://doi.org/10.1007/s13042-022-01658-9

Hunde Desissa, D. (2019). Soybean Research and Development in Ethiopia. *Acta Scientific Agriculture*, *3*(10), 192–194. https://doi.org/10.31080/asag.2019.03.0668

K. Achamyelh, Z. Shumeta, A. Tesfaye, and M. Hailemariam. (2020). Agricultural Sample Survey Report on Livestock and Livestock Characteristics. *Addis Ababa, Ethiopia:*, *8*(January), 1–10. https://doi.org/10.14662/IJEBM2019.200

K, P. V, Jayanth Srinivas, S., & Lokesh, S. (2020). Image Enhancement Using Gaussian Filters. *International Research Journal of Engineering and Technology*, *May*, 3268–3270. www.irjet.net

Kaesmetan, Y. R., & Overbeek, M. V. (2022). Digital Image Processing using Texture Features Extraction of Local Seeds in Nekbaun Village with Color Moment, Gray Level Co Occurance Matrix, and k-Nearest Neighbor. *Ultimatics : Jurnal Teknik Informatika*, *13*(2), 81–88. https://doi.org/10.31937/ti.v13i2.2038

Kim, J. H. (2022). Improvement of Inceptionv3 Model Classification Performance Using Chest X-Ray Images. *Journal of Mechanics in Medicine and Biology*, *22*(8), 1–11. https://doi.org/10.1142/S0219519422400322

Kumar, A., & Sachar, S. (2024). Deep Learning Techniques in Leaf Image Segmentation and Leaf Species Classification: A Survey. In *Wireless Personal Communications* (Vol. 133, Issue 4). Springer US. https://doi.org/10.1007/s11277-024-10873-2

Li, H. C., Lin, Z. X., Ma, T. Y., Zhao, X. Le, Plaza, A., & Emery, W. J. (2023). Hybrid Fully Connected Tensorized Compression Network for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, *61*, 1–16. https://doi.org/10.1109/TGRS.2023.3241193

Li, H., Zhang, L., & Sun, H. (2021). *Identification of soybean varieties based on hyperspectral imaging technology and one-dimensional convolutional neural network*. *April*, 1–14. https://doi.org/10.1111/jfpe.13767

Lin, W., Fu, Y., Xu, P., Liu, S., Ma, D., Jiang, Z., Zang, S., Yao, H., & Su, Q. (2023). Soybean image dataset for classification. *Data in Brief*, *48*. https://doi.org/10.1016/j.dib.2023.109300

Luo, T., Zhao, J., Gu, Y., Zhang, S., Qiao, X., Tian, W., & Han, Y. (2023). Classification of weed seeds based on visual images and deep learning. *Information Processing in*

Agriculture, *10*(1), 40–51. https://doi.org/10.1016/j.inpa.2021.10.002

Macuácua, J. C., Centeno, J. A. S., & Amisse, C. (2023). Data mining approach for dry bean seeds classification. *Smart Agricultural Technology*, *5*(April). https://doi.org/10.1016/j.atech.2023.100240

Markett, G. (2022). *Global Market Report*. *2020*, 1–5.

Nawrocka, A., Nawrocki, M., & Kot, A. (2023). Research study of image classification algorithms based on Convolutional Neural Networks. *Proceedings of the 2023 24th International Carpathian Control Conference, ICCC 2023*, 299–302. https://doi.org/10.1109/ICCC57093.2023.10178933

Nuthalapati, C. S., Kumar, A., Birthal, P. S., & Sonkar, V. K. (2024). Demand-side and supply-side factors for accelerating varietal turnover in smallholder soybean farms. *Journal of Cleaner Production*, *447*(February), 141372. https://doi.org/10.1016/j.jclepro.2024.141372

Pareek, C. M., Singh, N., Tewari, V. K., Dhruw, L. K., & Singh, H. D. (2023). Classification of Broken Maize Kernels Using Artificial Neural Network-Assisted Image-Processing Approach. *Journal of Biosystems Engineering*, *48*(1), 55–68. https://doi.org/10.1007/s42853-022-00173-7

Phung, V. H., & Rhee, E. J. (2018). A deep learning approach for classification of cloud image patches on small datasets. *Journal of Information and Communication Convergence Engineering*, *16*(3), 173–178. https://doi.org/10.6109/jicce.2018.16.3.173

Qu, X., Che, H., Huang, J., Xu, L., & Zheng, X. (2023). Multi-layered semantic representation network for multi-label image classification. *International Journal of Machine Learning and Cybernetics*, *14*(10), 3427–3435. https://doi.org/10.1007/s13042-023-01841-6

Rehima Musema, Samuel Diro. Beza Erko, Desalegn Teshale, Regasa Dibaba, Abush Tesfaye, L. Z. (2022). *Soybean Value Chain Analysis in Ethiopia : A Qualitative Study* (Issue Report Number: 134).

Soares, J. M., de Medeiros, A. D., Pinheiro, D. T., Rosas, J. T. F., da Silva, L. J., Machado, D. L. M., & Dias, D. C. F. D. S. (2023). Low-cost system for multispectral image acquisition and its applicability to analysis of the physiological potential of soybean seeds. *Acta Scientiarum - Agronomy*, *45*, 1–10. https://doi.org/10.4025/actasciagron.v45i1.57060

Sonmez, M. E., Sabanci, K., & Aydin, N. (2024). Convolutional neural network-support vector machine-based approach for identification of wheat hybrids. *European Food Research and Technology*, *0123456789*. https://doi.org/10.1007/s00217-024-04473-4

Tempola, F., Wardoyo, R., Musdholifah, A., Rosihan, R., & Sumaryanti, L. (2024). Classification of clove types using convolution neural network algorithm with optimizing hyperparamters. *Bulletin of Electrical Engineering and Informatics*, *13*(1), 444–452. https://doi.org/10.11591/eei.v13i1.5533

Teodoro, L. P. R., Silva, M. O., dos Santos, R. G., de Alcântara, J. F., Coradi, P. C., Biduski, B., da Silva Junior, C. A., Torres, F. E., & Teodoro, P. E. (2024). Machine learning for classification of soybean populations for industrial technological variables based on agronomic traits. *Euphytica*, *220*(3), 1–10. https://doi.org/10.1007/s10681-024-03301-w

Tesfay, W. (2023). Analysis of Soy Bean Value Chain in North Western Ethiopia. *Journal of Marketing and Consumer Research*, *90*(2), 1–13. https://doi.org/10.7176/jmcr/90-01

Tesfaye, M. A. (2018). Progress of Soybean [Glycine max (L.) Merrill] Breeding and Genetics Research in Ethiopia: A Review. *Natural Sciences Research*, *8*(January 2018), 67–77. https://www.bing.com/search?q=scientific+name+of+banana&qs=n&form=QBRE&sp=-1&pq=scientific+name+of+banana&sc=8-25&sk=&cvid=FDF5839E0EBA4E9AAC0C95E86A177EAB

Umamaheswari, D., & Karthikeyan, E. (2019). Comparative analysis of various filtering techniques in image processing. *International Journal of Scientific and Technology Research*, *8*(9), 109–114.

USDA, U. S. D. of A. (2020). Ethiopia Oilseeds Report Annual. *Global Agricultural Information Network (GAIN)*, 1–21. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://apps.fas.usda.gov/newgainapi/api/Report/DownloadReportByFileName?fileName=Ethiopia Oilseeds Report Annual_Addis Ababa_Ethiopia_01-16-2020

Véstias, M. P. (2019). A survey of convolutional neural networks on edge with reconfigurable computing. *Algorithms*, *12*(8). https://doi.org/10.3390/a12080154

Wang, L., Pang, L., Yan, L., & Zhang, J. (2022). Nondestructive Rapid Identification of Soybean Varieties Using Hyperspectral Imaging Technology. *Journal of Applied Spectroscopy*, *89*(1), 84–91. https://doi.org/10.1007/s10812-022-01329-1

Wang, Z. J., Turko, R., Shaikh, O., Park, H., Das, N., Hohman, F., Kahng, M., & Polo Chau, D.

H. (2021). CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization. *IEEE Transactions on Visualization and Computer Graphics*, *27*(2), 1396–1406. https://doi.org/10.1109/TVCG.2020.3030418

Wei, Y., Li, X., Pan, X., & Li, L. (2020). Nondestructive classification of soybean seed varieties by hyperspectral imaging and ensemble machine learning algorithms. *Sensors (Switzerland)*, *20*(23), 1–12. https://doi.org/10.3390/s20236980

Wu, Q., Zhang, K., & Meng, J. (2019). Identification of Soybean Leaf Diseases via Deep Learning. *Journal of The Institution of Engineers (India): Series A*, *100*(4), 659–666. https://doi.org/10.1007/s40030-019-00390-y

Wu, Y. chen, & Feng, J. wen. (2018). Development and Application of Artificial Neural Network. *Wireless Personal Communications*, *102*(2), 1645–1656. https://doi.org/10.1007/s11277-017-5224-x

Xu, M., Yoon, S., Fuentes, A., & Park, D. S. (2023). A Comprehensive Survey of Image Augmentation Techniques for Deep Learning. *Pattern Recognition*, *137*, 109347. https://doi.org/10.1016/j.patcog.2023.109347

Yang, S., Zheng, L., He, P., Wu, T., Sun, S., & Wang, M. (2021). High-throughput soybean seeds phenotyping with convolutional neural networks and transfer learning. *Plant Methods*, *17*(1), 1–17. https://doi.org/10.1186/s13007-021-00749-y

Yu, X., Fu, X., Yang, Q., Jin, H., Zhu, L., & Yuan, F. (2024). Genetic and Phenotypic Characterization of Soybean Landraces Collected from the Zhejiang Province in China. *Plants*, *13*(3). https://doi.org/10.3390/plants13030353

Zhang, N., Zhang, E., & Li, F. (2023). *A Soybean Classification Method Based on Data Balance and Deep Learning*.

Zhu, Y., & Huang, C. (2012). An Improved Median Filtering Algorithm for Image Noise Reduction. *Physics Procedia*, *25*, 609–616. https://doi.org/10.1016/j.phpro.2012.03.133

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70-90.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in agriculture: A review. Sensors, 18(8), 2674.

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture, 145, 311-318.

APPNDEX

```
import tensorflow as tf
from tensorflow.keras.models import Model, Sequential
from tensorflow.keras.layers import Dense, Flatten, GlobalAveragePooling2D, Dropout
from tensorflow.keras.applications import VGG16
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.optimizers import Adam
from sklearn.metrics import classification_report
import numpy as np

# Load the VGG16 model pre-trained on ImageNet, without the top (classification) layers
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(227, 227, 3))

# Freeze the layers of the base model
for layer in base_model.layers:
    layer.trainable = False

# Create a new model on top of the pre-trained base
model = Sequential([
    base_model,
    GlobalAveragePooling2D(),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(64, activation='relu'),
    Dense(8, activation='softmax')  # Assuming 8 classes for seed identification
])
```

```
# Compile the model
model.compile(optimizer=Adam(learning_rate=0.001), loss='categorical_crossentropy',
metrics=['accuracy'])
# Summary of the model
model.summary()
# Data augmentation and data generators
train_datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range=40,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest'
)
val_datagen = ImageDataGenerator(rescale=1./255)
train_generator = train_datagen.flow_from_directory(
    'path/to/train/data',
    target_size=(227, 227),
    batch_size=32,
    class_mode='categorical'
)
validation_generator = val_datagen.flow_from_directory(
    'path/to/validation/data',
    target_size=(227, 227),
    batch_size=32,
    class_mode='categorical'
)
# Train the model
```

```
history = model.fit(
    train_generator,
    steps_per_epoch=train_generator.samples // train_generator.batch_size,
    validation_data=validation_generator,
    validation_steps=validation_generator.samples // validation_generator.batch_size,
    epochs=25
)
# Save the model
model.save('seed_identification_model.h5')
# Generate predictions
predictions4 = model.predict(validation_generator, steps=validation_generator.samples //
validation_generator.batch_size + 1)
# Get the true classes and predicted classes
true_classes = validation_generator.classes
predicted_classes4 = np.argmax(predictions4, axis=1)
# Get class labels
class_labels = list(validation_generator.class_indices.keys())
# Generate classification report
report = classification_report(true_classes, predicted_classes4, target_names=class_labels)
print(report)
```