PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
UNIVERSITÉ KASDI MERBAH - OUARGLA
FACULTY OF MODERN SCIENCES OF INFORMATICS AND COMMUNICATIONS
DEPARTMENT OF ELECTRONICS AND COMMUNICATIONS

Master Thesis

Domain: Science and Technology

Sector: Automatic

Specialty: Automatic and Systems

THEME

Plant Leaves Disease Detection With VIT

Presented by:

Boublal Nada Rahil

Herrouz Salah Eddine

Evaluation Date : 01/06/2025

In front of the Jury :

Dr. Benlamoudi Azzedine	President	UKM Ouargla
Dr. Bensid Khaled	Supervisor	UKM Ouargla
Dr. Bagua Hakim	Examiner	UKM Ouargla

Academic year: 2024/2025

ACKNOWLEDGMENT

Above all, we thank Almighty God for giving us courage, will, patience, and health throughout these years, and thanks to whom this work has been possible.

we also wish to express our sincere gratitude to our supervisor, DR. Bensid Khalde, for his follow-up throughout the process of developing this work "thesis", and who never ceased to provide us with advice and feedback.

Heartfelt gratitude to my jury members for their interest in our work. We are grateful to Mr. Dr. Benlamoudi Azzedine and Dr. Bagua Hakim for agreeing to review this work.

We dare not forget to thank all the people of the faculty at Kasdi Merbah Ouargla University.

Furthermore, we sincerely thank all those who contributed directly and indirectly to the success of this research, Our thanks also go to all those who helped us, whether from near or far.

DEDICATE

I wish to dedicate this thesis to:

My dear parents.

My brother and sister, who have always supported me during my years of study, both financially and morally.

All my very dear friends, especially Amet Errahmane Hadjer.

All the students of my 2024-2025 graduating class of the Master's in Automatic and Systems.

All those who provided me with their help in the completion of this thesis.

Without forgetting my partner: Nada Rahil.

I wish to dedicate this thesis to:

All gratitude belongs to Allah, whose infinite mercy and guidance have illuminated my path through every trial.

To my family and dear ones, your unwavering support was my anchor in moments of doubt. Thank you for being my source of strength.

To the child I once was and the man l've grown to be - your resilience led us here

Salah Eddine Herrouz

ABSTRACT

Abstract

This work investigates the application of advanced deep learning techniques for the automated detection and classification of tomato plant diseases. The study begins with a comprehensive review of image processing fundamentals in agriculture, tomato plant pathology, and the core concepts of Artificial Intelligence, Machine Learning, and Deep Learning, focusing on architectures like CNNs (ResNet50) and Vision Transformers (ViT,DeiT, Swin Transformer). The core methodology involved utilizing the"New Plant Diseases" dataset , implementing data preprocessing and augmentation , and employing K-Fold cross-validation. Four pretrained models ResNet50 ,DeiT 3, SWIN V2, and a Combined ViT+ResNet50 were evaluated based on accuracy, precision, recall, and F1-score. Results indicated exceptional performance across all models, with DeiT 3 achieving the highest accuracy 99.93٪.The findings demonstrate the significant potential of deep learning, particularly transformer-based architectures, to advance precision agriculture by providing accurate and efficient tools for plant disease identification.

key words:Artficial Intelligence, Agriculture, Deit V3, Resnet50, Swin V3, VIT, Transformer

ملخص :

يتناول هذا العمل تطبيق تقنيات التعلم العميق المتقدمة للكشف والتصنيف الآلي لأمراض نبات الطماطم. تبدأ الدراسة بمراجعة شاملة لأساسيات معالجة الصور في الزراعة وأمراض نبات الطماطم، والمفاهيم الأساسية للذكاء الاصطناعي والتعلم الآلي والتعلم العميق مع التركيز على هياكل مثل الشبكات العصبية الالتفافية Reset50 **ومحولات الرؤية** (ViT,DeiT,Swin Transformer) **تضمنت المنهجية الأساسية استخدام مجموعة بيانات أمراض النباتات الجديدة الفرعية للطماطم، وتنفيذ المعالجة المسبقة للبيانات وتعزيزها، واستخدام التحقق المتقاطع** K-Fold **تم تقييم أربعة نماذج مدربة مسبقا** ViT+ResNet50 ,ResNet50 ,DeiT 3,SWIN V2 **بناء على الدقة والإحكام والاستدعاء، ومقياس** F1 **. أشارت النتائج إلى أداء استثنائي لجميع النماذج، حيث حقق** DeiT 3 **أعلى دقة٩٩.٣٩٪توضح النتائج الإمكانات الكبيرة للتعلم العميق وخاصة الهياكل القائمة على المحولات في تطوير الزراعة الدقيقة من خلال توفير أدوات دقيقة وفعالة لتحديد أمراض النباتات .**

كلمات مفتاحية:**الذكاء الاصطناعي، الزراعة،** Deit V3, Resnet50 , Swin V3, VIT, Transformer

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI Artficial Intelligence

ANN Artificial Neural Network

BERT Bidirectional Encoder Representations from Transformers

CNN Convolutional Neural Network

DBNs The Deep Belief Networks

DEIT Data-efficient Image Transformer

DL Deep learning

DT Decision Tree

FAO Food and Agriculture Organization

HIS Hue, Illumination, Saturation

HSV Hue, Saturation, Value

HLS Hue,Lightness,Saturation

HVC Hue, Value, Chroma

kNN k-Nearest-Neighbours

LN Layer Normalization

LR Logistic Regression

MLP Multi-Layer Perceptron

ML Machine Learning

MSA Multihead Self Attention

NB Naive Bayes

NLP Natural Language Processing

RGB Red , Green , and Blue

RF Random Forest

RL Reinforcement Learning

RNN Recurrent Neural Network

SL Supervised Learning

SVM Support Vector Machine

SWIN Shifted Window Transformer

VIT Vision Transformer

UL Unsupervised Learning

GENERAL INTRODUCTION

Agriculture is widely regarded as a foundational sector for any nation, evolving in parallel with the industrial revolution. Crops play a vital role in sustaining human life . However, plants are vulnerable to a range of diseases worldwide. Various factors such as climatic conditions, soil quality, and the presence of pests and pathogens contribute to the emergence and spread of these diseases. These issues can lead to serious economic losses and environmental challenges for both farmers and communities. Plant diseases pose a significant threat to agriculture and food security by reducing crop yields and compromising quality. As a result, the early detection and effective management of plant diseases are essential to limit their spread and safeguard plant health and food production[1]. Despite progress in technology across multiple fields, many farmers still depend on traditional, manual methods for detecting plant diseases, primarily through visual inspection of crops[2].

The proliferation of digital technologies has spurred increasing interest in autonomous systems that utilize computer vision and machine learning techniques for the identification and monitoring of plant diseases. Such systems offer rapid, precise, and cost-effective solutions amenable to large-scale agricultural applications [3]. The application of machine learning (ML) and particularly deep learning (DL) for plant disease detection represents an increasingly dynamic field of research. Within these methodologies, DL has exhibited superior performance, especially in image recognition tasks, attributable to its capacity for automatic feature extraction without manual intervention, followed by classification [4].Convolutional Neural Networks (CNNs) and pre-trained models are among the most extensively employed approaches, establishing themselves as foundational in visual recognition.Numerous research studies have explored this domain, For instance, one study [5] employing the EfficientNetB3-AADL model, augmented with Gaussian noise, reported an accuracy of 80.19%. Another investigation [4] utilized the Inception model, adopting a distinctive approach of identifying diseased regions within diminutive sections of plant leaves this achieved 94.04%accuracy on the PlantVillage dataset and 97.13%on novel datasets. Furthermore, research [1] comparing ResNet-50, VGG-16, and VGG-19 demonstrated that ResNet-50 yielded the highest accuracy at 98.98%.

More recently, attention-based deep learning models have surfaced as potent alternatives to conventional CNNs, presenting distinct mechanisms for processing and interpreting visual data [6] ,The present study undertakes an examination of several prominent models: ResNet; Vision Transformer (ViT), which incorporates a transformer-based attention architecture; Data-efficient Image Transformer (DeiT), noted for its amalgamation of high accuracy with efficient data utilization; and Swin Transformer (Swin), which implements local attention via a sliding window mechanism.

Our primary focus is the identification of plant diseases. These diseases represent a significant worry for all farmers, largely due to challenges in early detection and a shortage of means to monitor all agricultural crops. Consequently, they continue to depend on traditional, manual methods for discovery.

The primary aim of this study is to investigate and apply new deep learning techniques to diseased tomato crops, given that tomatoes are one of the most widely cultivated crops with considerable global economic value, with the goal of obtaining improved results compared to prior research.

This thesis is composed of three chapters:

Chapter 1 presents a general introduction to the basic concepts of image processing, outlines the various types of diseases affecting tomato crops, and reviews related literature on disease detection methods.

Chapter 2 explores the fundamental principles of artificial intelligence and its various learning branches, emphasizing the specific techniques planned for implementation.

The last chapter outlines the adopted methodology, showcases the results achieved, and provides an in-depth discussion of the findings, and using it in the CROPX agricultural robot system.

In conclusion, the study ends with a summary that highlights the key findings achieved throughout the work.

CHAPTER 1

BASICS OF IMAGE PROCESSING FOR DETECTING TOMATO PLANT DISEASES

1.1 Fundamental Concepts of Images

1.1.1 Definition of Digital Image Processing

An image can be described as a two-dimensional function, f(x, y), where x and y represent spatial (plane) coordinates. The value of f at any given coordinate pair (x, y) denotes the intensity or gray level of the image at that specific point. When x, y, and the intensity values of f are all finite and discrete, the image is referred to as a digital image[7].

1.1.2 Structure of a Digital Image

1. **Pixel:** the smallest component of a digital image is a pixel, or picture element. Each pixel is assigned a specific color or intensity value, representing a distinct point within the image. The entire image consists of a grid of pixels, where each pixel's value corresponds to the luminance or color at that particular location [8], As the following figure 1.1 show.

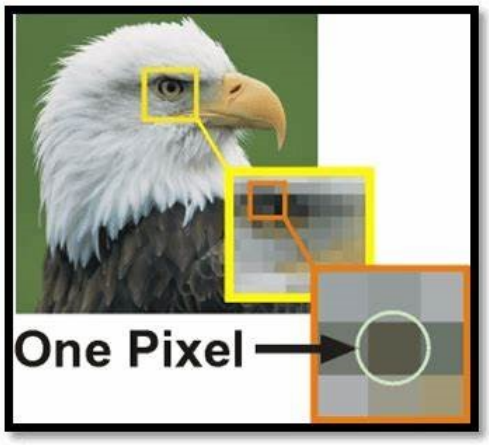


Figure 1.1: Representative image intended to elucidate the concept of pixels[8].

2. **Resolution:** the number of pixels in an image serves as the standard measure of resolution, indicating the level of detail it holds. It is commonly expressed as width × height (e.g., 1920 × 1080 pixels). Higher resolution images have more pixels, providing greater detail, while lower resolution images contain fewer pixels, resulting in less detail[8] .

3. **Bit depth:**refers to the number of bits used to represent the color or intensity of each pixel. For example, a 24-bit color image can display 16.7 million colors (8 bits per channel × 3 channels), while an 8-bit image offers 256 intensity levels(28) A higher bit depth allows for more accurate color or intensity representation[8].

1.1.3 Types of Digital Image

Before an image can be analyzed, it must first be converted into a digital format. Based on color and grayscale properties, images are generally classified into four main types[9].

❖ **Binary image.**

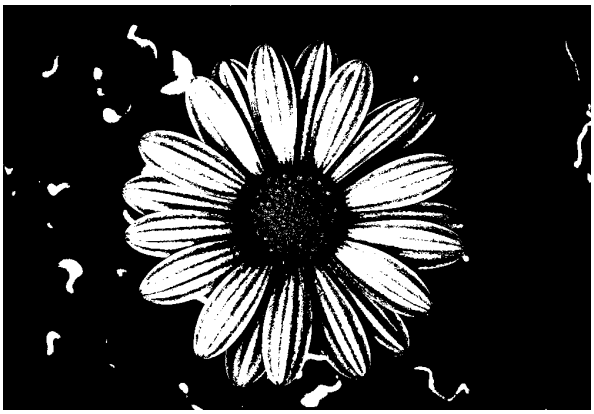


Figure 1.2: Binary image:

A binary image is the most basic type of digital image, where each pixel is either completely black or completely white[10],figure 1.2 illustrates that.

❖ **Grayscale:**

Figure 1.3: Grayscale image.

Grayscale images are two dimensional arrays in which each pixel is assigned a single numerical value representing its intensity at that specific point .Each pixel has a brightness value ranging from 0 to 255, where 0 is black and 255 is white[9],figure 1.3 illustrates that.

❖ **Indexed color image:** contains a limited set of colors, such as 256 or fewer, it can be efficiently stored and managed using a color map or palette. Each pixel is assigned an index that corresponds to a specific color within the map[9].

❖ **true colour images:**

- RGB in the red, green, blue (RGB) color model, each pixel's color is determined by the intensity levels of red (R), green (G), and blue (B). Each of these colors has a range from 0 to 255. A true color image is represented as a stack of three matrices where each matrix corresponds to the R, G, and B values of every pixel,figure 1.4 shows that. As a result, each pixel is defined by three distinct values[9].

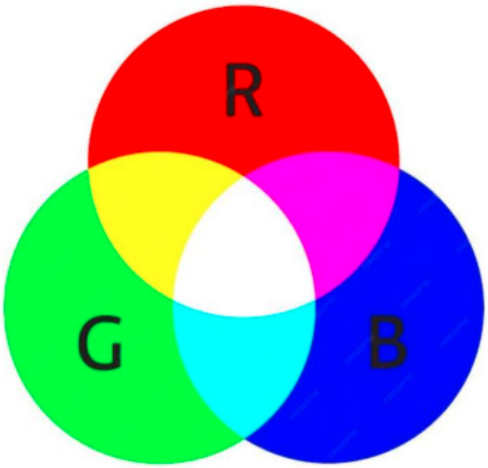


Figure 1.4: RGB Image.

- HIS, HSV, and other color models are used to define the color space as perceived by the human eye. Various color vision models have been developed, including HSV (hue, saturation, value), HIS (hue, illumination, saturation), HLS (hue, lightness, saturation), and HVC (hue, value, chroma)[9].

1.1.4 Image processing methods

✤ **Image acquisition:** refers to the process of capturing a digital representation of a scene. This representation, known as an image, consists of individual elements called pixels (picture elements). The electronic device responsible for capturing the scene is referred to as an imaging sensor[11].

✤ **Image Preprocessing:**

(a) **Low-level image processing (preprocessing):** consists of fundamental operations aimed at preparing an image for more advanced analysis, such as assessing corrosion levels. These initial steps may include basic techniques like noise reduction, image restoration, contrast enhancement, segmentation, edge detection, and sharpening. In these processes, both the input and output remain in the form of images[9].

- Image Noise and Filtering during the process of digitising images, the generation of noise is a possibility. The potential sources of noise include: (1) external noise arising from voltage instability or atmospheric electric (magnetic) explosions that may occur during image collection; (2) internal noise, such as shot noise (also termed salt and pepper noise, binary noise, or impulse noise), quantisation noise. The elimination of noise can be achieved through the implementation of an image addition (averaging) algorithm, spatial filtering (e.g. average filtering, Gaussian filtering, median filtering, or bilateral filtering), or frequency domain filtering[9].
- Contrast enhancement techniques differ depending on the application, as there is no universal theory of image enhancement. One common method involves converting an image into a binary (black and white) format to improve contrast. This can be achieved using either a single threshold or double threshold segmentation algorithm to transform a non binary image into a binary one[12].

- **Image segmentation:** plays a crucial role in various image processing applications, including pattern recognition, image retrieval, and surveillance. The primary purpose of segmentation is to enhance image content understanding and facilitate visual entity recognition by identifying regions of interest. It involves dividing an image into meaningful segments for better analysis. Image segmentation can be categorized into different types[13].
  - Edge Detection Technique is used to identify the boundaries of a leaf within an image by detecting intensity differences at the edges. It segments the image by highlighting variations in intensity along the borders. Common edge detection methods include Sobel, Canny, Laplacian, and fuzzy logic[13].
  - Threshold Technique is one of the simplest image segmentation methods, relying on a predefined threshold value to convert an image into a binary format. This technique includes global and local thresholding, with notable methods such as Otsu's global thresholding and adaptive local thresholding[13].
  - Region Based Segmentation the primary objective of this methodology is the division of an image into disparate regions of analogous nature. Pixels of congruent nature are identified and amalgamated into homogeneous regions. The primary techniques in region based segmentation include Region Growing, Region Splitting, and Region Merging[13].
- **Image Feature Extraction:** identifies the essential shape characteristics within a pattern, simplifying the classification process through a structured approach. In image processing, it serves as a specialized method of dimensionality reduction. The primary goal of feature extraction is to extract meaningful information from the original image and represent it in a lower dimensional space. There are three main types of features [14]:

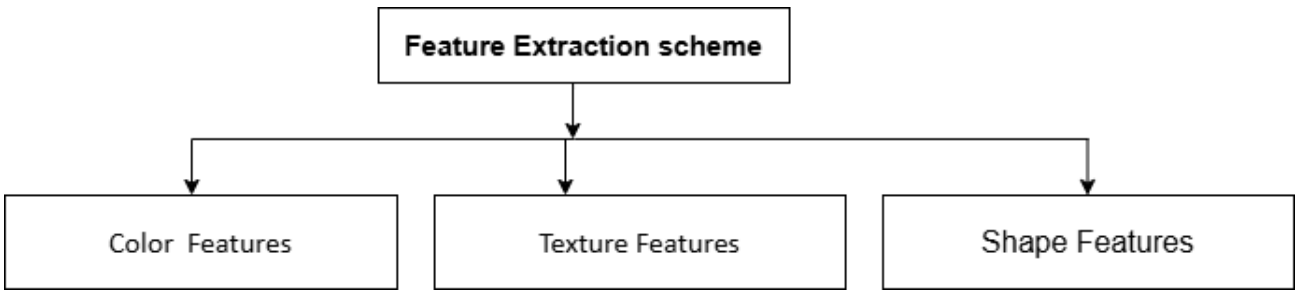


Figure 1.5: The main types of feature extraction.

  - Color Features :refers to one of the most essential attributes of an image. Color features vary depending on the chosen color space or model. Once the color space is determined, relevant color features can be extracted from the image or specific regions,such as RGB, HSV [15].
  - Texture Features :refers to highly valuable characteristic for classifying various types of images. It is widely accepted that human vision relies on texture for interpretation and recognition. While color is a property of a pixels, while texture can only be estimated from a group of pixels these features are especially useful in medical imaging, remote sensing ,such as :Silk ,Wool[14].
  - Shape Features :refers to fundamental cue that enables humans to recognize and identify real-world objects by encoding simple geometric forms, such as straight lines in various directions. Shape feature extraction methods are categorized into two main types :Contour based methods and Region based methods,such as :identifying a fruit based on its shape and size in an image[16].

1.1.5 Image Processing application in agriculture

Image processing plays a crucial role in detecting plant diseases, utilizing artificial intelligence to analyze images, the proposed method comprises multiple steps, including image acquisition, preprocessing, segmentation, feature extraction, classification, and notification. The process begins with capturing and collecting digital images from tomato fields, preprocessing enhances image quality and ensures more reliable feature extraction using techniques such as leaf image cropping, resizing, and enhancement. Once preprocessing is complete, segmentation is performed to isolate relevant regions for further analysis, during the feature extraction phase, texture based features are extracted and stored in a database. The extracted features are then used for classification, determining whether the plant is healthy or diseased, finally in the notification stage, the farmer receives information about the crop's condition, in case a disease is detected[17].

1.2 Tomato Plant Diseases

1.2.1 Introduction to Tomato Plant Diseases

Tomatoes are among the most widely cultivated and economically important crops worldwide. According to the Food and Agriculture Organization (FAO) [18]. Tomato plants are susceptible to various pathogens [19], including fungal, bacterial, viral, and insect borne diseases. Many of these diseases primarily affect the leaves, leading to substantial crop losses by reducing both yield and quality, early detection and accurate diagnosis are essential for several reasons: preventing the spread of diseases, reducing economic losses, enhancing the sustainability of tomato farming, and increasing crop yield while minimizing pesticide use. However, the identification of plant diseases remains a persistent challenge[18].

1.2.2 Classification of Tomato Plant Diseases

1. **Fungal Diseases:** Various abiotic and biotic factors play a role in the onset and progression of fungal infections in plants. Among the abiotic factors, climate and environmental conditions significantly affect the frequency and severity of fungal diseases. These influences can be limiting, inductive, or resistant, impacting disease development in different ways[20].

(a) Target Spot

(b) Late Blight

(c) Early blight

(d) Leaf mold

(e) Leaf mold

Figure 1.6: Images Tomato leaf Fungal Diseases [18].

- Early Blight caused by the fungus alternaria solani Sorauer, is a major disease affecting tomatoes. In severe cases, it can result in complete defoliation, making it one of the most damaging diseases for tomato plants[21], as shown in the figure 1.6c.
- Late Blight Phytophthora infestans (Mont) de bary, the pathogen responsible for late blight, is the most destructive disease affecting tomato plants worldwide. Unlike most phytophthora species, which typically cause soil borne root rot. Infestans is a specialized pathogen that primarily infects the foliage, stems, and fruit of tomato plants [22], As shown in the figure 1.6b.
- Target Spot Leaf lesions initially appear as small, dark brown spots, resembling early bacterial spot symptoms caused by xanthomonas spp. However, as target spot lesions grow, their centers turn light brown to gray, surrounded by dark concentric rings, sometimes accompanied by diffuse yellowing [23], as shown in the figure 1.6a.

- Leaf Mold Irregu-lar chlorotic spots develop on the upper (adaxial) surface of the lower tomato leaves, while a layer of mold forms on the underside (abaxial surface). Over time, the leaf edges gradually curl and wither [24], as shown in the figure 1.6d and 1.6e.

2. **Bacterial Diseases:** are seedborne pathogens mainly transmitted through contaminated seeds and transplants. Factors such as high plant densities, overhead irrigation, high humidity, and elevated temperatures create favorable conditions for their rapid spread [25].

Figure 1.7: Images Tomato leaf Bacterial Diseases [18].

- Bacterial spot BS is a significant threat to tomato production, impacting both fresh market and processing tomatoes. This disease is caused by at least four Xanthomonas species: X. euvesicatoria, X. vesicatoria, X. perforans, and X. gardneri [26], as shown in the figure 1.7.

3. **Viral Diseases:** most plant viruses rely on insect vectors for transmission, spreading persistently and circulatively [27]. Once a plant is infected, eliminating the virus becomes challenging, as viruses integrate into plant cells, spread rapidly, and make control difficult[18].

- Tomato mosaic virus TOMV a member of the genus tobamovirus in the virgaviridae family, primarily infects tomato plants. It causes mosaic patterns, stunted growth, and leaf distortion, resulting in 15%25%crop losses. If an infected seedling is transplanted, ToMV spreads rapidly through contact with workers hands or clothing. Beyond seeds, the virus can remain infectious for years in dead plant tissues, irrigation water, soil, or contaminated clothing from handling infected plants[28].

1.2.3 Challenges in Diagnosing Tomato Plant Diseases

Identifying tomato plant diseases based solely on visual characteristics poses significant challenges due to the following factors:

1. Symptom Overlap and Variability Symptom Similarity Many plant diseases and physiological stress conditions exhibit similar visual symptoms, such as mottling, chlorosis, or leaf curling. These similarities make it challenging to distinguish between viral infections, nutrient deficiencies, and fungal diseases. As a result, relying solely on visual detection methods can lead to misdiagnosis[18].

2. Environmental Factors variations in environmental conditions, including light, temperature, and humidity, can affect symptom expression, leading to inconsistencies in how symptoms appear. As a result, symptoms may vary even within the same plant or among plants infected with the same disease[18].

3. Latency and Subtle symptoms Latent Infections Certain viruses, such as ToMV, can infect plants without causing immediate or Noticeable symptoms. In such cases, the virus may spread undetected, making visual detection unreliable in the early stages of infection[18].

4. Subtle Symptoms in the early stages, symptoms may be too faint for human observers to detect accurately. Additionally, some image based detection systems may struggle to recognize these subtle signs, further complicating visual diagnosis[18].

1.2.4 Strategies for Preventing and Managing Tomato Plant Diseases

- Cultural practices refer to farming techniques designed to enhance both the quality and quantity of crop yield while minimizing the impact of pests and diseases. These methods involve environmental manipulation through non mechanical approaches to control plant pests and diseases. This includes adjusting farming practices to create unfavorable conditions for the growth and spread of disease causing pathogens and pests[29].

- Chemical control involves the application of chemicals to manage plant diseases (fungal, viral, bacterial, and nematode related), as well as pest infestations and weed growth[29].

- Biological control provides a more sustainable alternative to chemical use by utilizing natural antagonistic organisms to manage pests and suppress plant diseases. Plant Growth Promoting Rhizobacteria (PGPR), known for enhancing plant growth and reducing diseases, presents a promising approach for disease management through biological control[30].

1.3 Recent Research Review

Table 1.1: Recent Research

Year	The Author	Methods	database	Result
2025 [5]	M.Shetti and Oth.	EfficientNet-B3 model	Combination of the PlanDoc and web sourced datasets	80.19%accuracy
2025 [31]	K. Joshi and oth	YOLOv8 model	PlantDoc dataset	97%precision ,93.8%recall , 95%F1 score
2024 [4]	I. Bouacidaa and oth	Inception model	PlantVillage dataset and new datasets	94.04%accuracy and new 97.13%accuracy
2024 [32]	Ali and oth.	EfficientNetB3 model	New PlantVillage	99.89%accuracy
2023 [2]	FAIQA and oth	EfficientNetB3-AADL model	The dataset used in this research from Kaggle	98.71%accuracy
2023[1]	Md. Manow and oth	ResNet-50 ,VGG-16, VGG-19	plant-village dataset	accuracy: VGG-19:92.39%, VGG-16:96.15%,ResNet-50:98.98%

1.4 Conclusion

This chapter provides an overview of the fundamentals of digital image processing, its applications in agriculture, and a general look at tomato plant diseases and their detection using artificial intelligence, based on previous studies. The key ideas discussed include: Fundamentals of Digital Image Processing:Images are represented as pixel matrices, where resolution and bit depth determine quality and detail.Techniques such as preprocessing, segmentation, and feature extraction enhance image analysis for disease detection, and fungal, bacterial, and viral diseases in tomatoes pose a significant threat to crop yields.dentifying tomato plant diseases based solely on visual characteristics presents major challenges due to overlapping symptoms, environmental factors, and subtle early signs.
Recent research demonstrates the effectiveness of AI in detecting tomato diseases using deep learning models (CNNs, YOLOv8, EfficientNet) and classifying them accurately, the integration of image processing techniques and artificial intelligence provides a powerful solution for sustainable agriculture and improving the quality and health of crops through accurate detection.

CHAPTER 2

CORE CONCEPTS OF ARTIFICIAL INTELLIGENCE AND ITS LEARNING BRANCHES

2.1 Artficial Intelligence

2.1.1 Introduction to artficial intelligence

Artificial intelligence (AI) is a composite of the disciplines of computer science and physiology[33]. scientific discipline focused on studying and developing intelligent machines[34],the term 'artificial intelligence' (AI) is defined as the capacity of a computer program to execute processes analogous to those performed by human intelligence. Such processes include, but are not limited to, reasoning, learning, adaptation, sensory understanding and interaction [35]. AI is a broad field that encompasses computer science, engineering, psychology, philosophy, ethics, and other disciplines. Designing technology to perform highly specialized tasks, such computer vision, speech processing, and pattern analysis and prediction, is one of artificial intelligence's objectives[36],the figure 2.1 show types of AI.

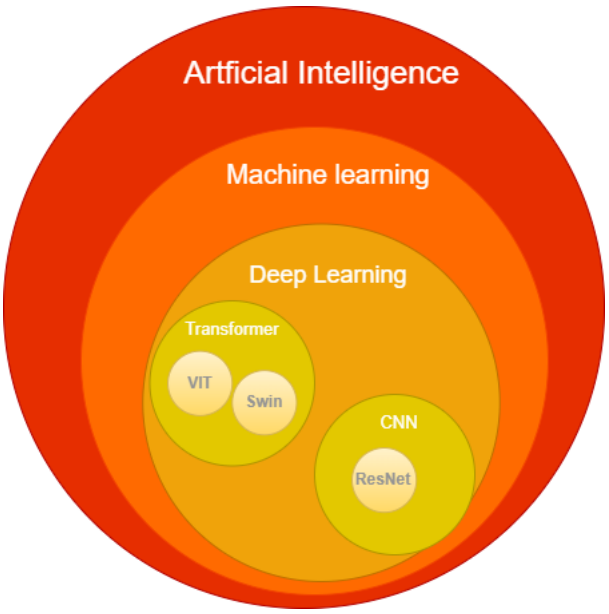


Figure 2.1: AI Technology Hierarchy[37].

2.1.2 The History and Development of AI

The concept of creating machines capable of performing tasks autonomously has intrigued humanity long before digital computers existed. In the 1930s, vannevar bush introduced a set of rules to solve differential equations automatically.A few years later, british mathematician alan turing proposed the idea of a machine capable of solving any algorithmic problem now famously known as the turing machine[38]. The invention of programmable computers in 1944 paved the way for the practical development of AI. Turing's influential work on evaluating machine intelligence laid the foundation for the field, particularly through the introduction of the turing Test [39], which aimed to determine whether a machine could mimic human responses in conversation. In 1952, the first AI program for playing checkers demonstrated that computers could learn, and this capability was significantly enhanced by Arthur Samuel in 1959, allowing the program to compete against skilled players. Following this, the development of the logic theorist applied fundamental AI principles to prove various geometric theorems [38].

The formal inception of artificial intelligence as a field began with a landmark conference at Dartmouth College in 1956, where John McCarthy introduced the term "artificial intelligence". Following this, AI research advanced rapidly, leading to the development of Unimate the first industrial robot introduced in 1961 to perform tasks on assembly lines[38]. Additionally, James

Slagle created the first LISP based program called the Symbolic Automatic Integrator (SAINT), which used heuristic methods to solve calculus problems [40].

Despite significant progress in AI algorithm development and applications, even the most advanced systems at the time could only address a narrow range of problems. In the 1980s, the emergence of expert systems revitalized interest in AI. These systems operated using a knowledge base filled with domain specific facts and rules, along with an inference engine to process this information [41]. The goal was to replicate human expertise within specific fields. However, by the early 1990s, expert systems began to decline due to challenges in acquiring and analyzing knowledge in real time. Although these systems aimed to mirror human reasoning by capturing real world rules, it proved impossible to encode the full range of human skills and intuition. Additionally, expert systems lacked the ability to learn, adapt, or evolve through user interaction. As a result, interest in this area waned, and many researchers shifted focus, working instead under terms like "machine learning," "intelligent systems," and "knowledge-based systems."This rebranding helped AI persist and laid the foundation for clearer distinctions between its subfields. Over time, these sub branches matured, enabling AI to evolve from simple problem solving tools to powerful deep learning systems capable of handling increasingly complex tasks[38].

2.1.3 Types of Artificial Intelligence

The field of artificial intelligence is broadly categorised into three distinct classifications: super intelligent AI, general AI, and narrow AI. Researchers in this domain typically prioritise a comprehensive understanding of these three pivotal categories[42].

1. Super AI: refers to highly advanced and autonomous AI systems with intellectual abilities that exceed human capabilities across multiple domains. In literature, super intelligent AI is often considered an aspirational concept, representing a future goal rather than a current reality[43].

2. General AI: defines general artificial intelligence (AI) as the ability to carry out any general job that is posed to it. However, it remains in the developmental phase[42].

3. Narrow AI :refers to systems designed to perform specific tasks that involve one or more decision making processes, such as image based facial recognition[44].

2.1.4 Fundamental Technologies of AI

The term artificial intelligence (AI) refers to a wide range of methods intended to demonstrate intelligent behavior and carry out activities that have required human intellect. These methods use computational, statistical, and mathematical approaches to evaluate information, reach conclusions, and resolve challenging issues. Among the fundamental methods in AI are[45]:

- Machine learning techniques :are a subset of artificial intelligence that enable computers to learn from data and improve their performance over time without the need for explicit programming. As illustrated in Figure 2.2, machine learning can be categorised into distinct classifications[46].

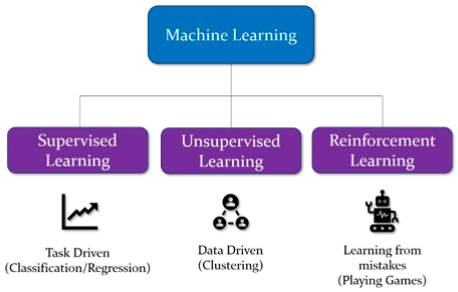


Figure 2.2: Machine Learning's Three Primary Methods[45].

- Deep learning :refers to a branch of machine learning that employs artificial neural networks with multiple layers (deep architectures) to capture and understand intricate patterns within data [47] ,The figure2.3 shows the mechanisms of deep learning.

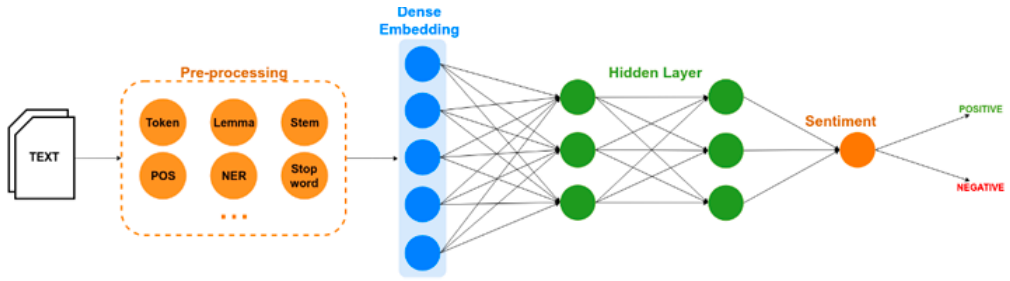


Figure 2.3: Deep learning mechanisms[45].

- Natural Language Processing (NLP):refers to a diverse field within artificial intelligence (AI) and computational linguistics, focused on enabling computers to comprehend and process human language in different forms, such as written text, spoken words, and gestures [48].

- Computer Vision :refers to a domain of artificial intelligence that enables machines to perceive, interpret, and analyze visual data from the real world. It includes various techniques such as Image Classification, Object Detection, and Semantic Segmentation. Image Classification involves assigning images to predefined categories based on their visual features[49].

2.2 Machine Learning

2.2.1 Introduction to Machine Learning

Machine Learning (ML) is a fundamental discipline within artificial intelligence (AI) that focuses on enabling computers to learn from data without explicit programming[50]. It encompasses various subfields and applications, such as statistical learning techniques, neural networks, instance based learning, genetic algorithms, data mining, image recognition, natural language processing (NLP), computational learning theory, inductive logic programming, and reinforcement learning,at its core ML allows software or machines to enhance task performance through continuous exposure to data and experience[34]. The role of ML developers in influencing the speed and quality of machine learning (ML) is paramount. They do this by modifying external configuration settings, termed "hyperparameters," prior to the initiation of an algorithm's training process. During the training phase, data is provided to the algorithm,

enabling the computer to identify patterns, correlations, and boundaries within the specified data set. The parameters are defined as internal variables, with the weights of these parameters being adjusted by the algorithm in order to enhance the accuracy of the predictions made[37].

2.2.2 Types of Machine Learning

The learning approaches used by machine learning algorithms can be further divided into three categories: supervised, unsupervised, and reinforcement learning. Each of these approaches is appropriate for a variety of tasks and data kinds.

- **Supervised Learning (SL):** involves training algorithms on datasets that have been prelabeled, enabling them to classify information or make predictions based on known input output pairs[37]. When presented with new data, the model generates outcomes by relying on the patterns it learned from the training examples[51]. However, preparing and labeling large datasets can be time consuming and resource intensive. Additionally, there's a risk of the model overfitting resulting in poor performance when encountering new, unseen data[37].

- **Unsupervised Learning(UL):** refers to machine learning approach where algorithms analyze unlabeled data without predefined outputs or human guidance[52]. These models aim to discover hidden patterns, trends, and anomalies in datasets that may not be obvious to humans, common techniques include clustering, association, dimensionality reduction, and anomaly detection . Although unlabeled data is easier to obtain, interpreting the results can be more complex[37].

- **Reinforcement Learning (RL):** refers to type of machine learning where an autonomous agent learns to make decisions by interacting with its environment repeatedly[52]. In other words, the algorithm improves through trial and error, receiving rewards or penalties based on its actions to ultimately achieve the optimal outcome[37].

2.2.3 Type of Machine Learning Tasks

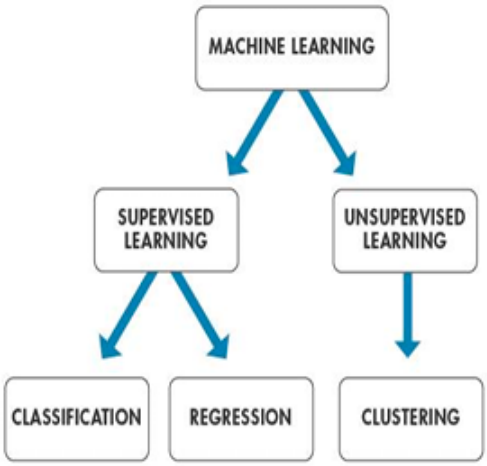


Figure 2.4: The Different Types of Machine Learning[53].

- **Classification:**refers to supervised learning technique where a model predicts a class label for a given example[54]. It involves assigning items to one or more predefined categories.

Binary classification deals with two possible outcomes (e.g., "yes" or "no"), multiclass classification involves more than two possible labels, multi label classification extends multiclass classification by allowing examples to belong to multiple classes simultaneously, often organized hierarchically[55].

- **Regression:**refers to supervised learning task focused on predicting continuous output values, univariate regression predicts a single output, multivariate regression predicts multiple outputs at once[55].

- **Clustering:**refers to an unsupervised learning process that groups similar items into subgroups based on a defined similarity measure. It can also support classification by organizing labeled data efficiently[55].

2.2.4 Core Machine Learning Algorithms

✶ ***Naive Bayes (NB):*** is based on Bayes' Theorem and assumes feature independence it is efficient, handles noisy data well, and works effectively for tasks like text classification and spam detection. Although it performs quickly with little training data. However, its performance can suffer in situations where the assumption of feature independence doesn't hold true[54].

✶ ***k-Nearest-Neighbours(k-NN)*:** is a non-parametric algorithm that classifies data based on the closest neighbors in the training set. It is simple and effective but sensitive to noisy or irrelevant features. A larger k value reduces noise sensitivity but increases computation. k-NN is slower on large datasets and requires careful feature weighting for optimal performance[56].

✶ ***Support vector machine (SVM)* :** is a versatile machine learning method used for classification, regression, and more[57]. It builds hyperplanes to separate classes with the largest possible margin, improving generalization. SVM works well in high dimensional spaces and depends heavily on the chosen kernel . However, its performance drops when the dataset has additional noise or overlapping classes[54].

✶ ***Decision tree (DT)*:** is a popular non-parametric supervised learning method used for classification and regression[54]. A Decision Tree uses a divide and conquer strategy, recursively splitting data based on categorical variables in classification tasks until all features are utilized[56].

✶ ***Random forest (RF)*:** is a widely recognized ensemble classification technique commonly used in machine learning and data science across various domains. This method applies parallel ensembling, where multiple decision tree classifiers are trained simultaneously on different subsets of the dataset[54].

✶ ***Logistic regression (LR)*:** is a widely used probabilistic statistical model for addressing classification problems in machine learning . It typically utilizes a logistic function, also known as the sigmoid function, to estimate probabilities. Logistic regression is effective for datasets that can be linearly separated, but it may overfit high dimensional datasets[54].

✶ ***Artificial Neural Network (ANN)*:** commonly known as a neural network, is a mathematical framework designed for pattern recognition and machine learning[58]. It draws inspiration from the structure and functioning of the human brain [59], but is significantly simpler in complexity and does not mimic higher level brain functions [58].

ANNs consist of interconnected processing units called neurons that transmit information among themselves. These networks are typically structured into three layers: an input layer that receives data, hidden layers where the data is processed, and an output layer that produces the final result. The connections between these layers are defined by weighted links, and during the training process, the model adjusts these weights in order to learn how to map inputs to outputs effectively. Initially, the model updates the weights in the hidden layer based on input features, and then uses the transformed data to calculate the output through the output layer[59].

2.3 Deep Learning

2.3.1 Introduction to Deep Learning

DL is a subset of machine learning techniques that analyzes and learns from data, including texts, sounds, and images [60], using multilayered (or "deep") neural networks. The hidden layers, which are the inner layers between the input and output, have nodes that function similarly to biological neurons. In order to reduce prediction errors, each node, or perceptron, independently modifies its parameters after processing inputs from the previous layer and sending outputs to the next. More hidden layers and parameters aid in the network's recognition of more complex patterns, but they come at a larger cost in terms of training time and energy consumption [37].

Deep learning is making significant strides in addressing issues that have long eluded the artificial intelligence community's best efforts. It is applicable to many areas of science, business, and government since it has proven to be particularly effective at identifying complex structures in high dimensional data. Apart from surpassing records in image recognition and speech recognition, it has also outperformed previous machine learning methods in predicting the activity of possible medicinal molecules, analyzing particle accelerator data and rebuilding brain circuits and forecasting how non coding DNA mutations affect disease and gene expression. deep learning has shown very promising results for a variety of natural language understanding tasks, including sentiment analysis, subject classification, question answering and language translation[61].

2.3.2 Key Differences Between ML and DL

- Data dependencies: Deep learning performs better with large datasets, while traditional machine learning is more effective with smaller datasets and established rules[61].
- Hardware dependencies: Deep learning algorithms rely on GPUs to efficiently perform matrix operations, requiring high-performance machines with GPUs, unlike machine learning algorithms[63].

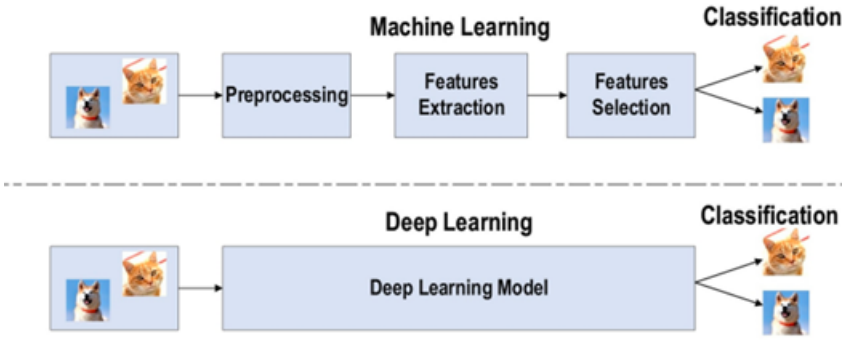


Figure 2.5: The difference between deep learning and traditional machine learning[62].

- Feature processing: In machine learning (ML), most of the characteristics of an application must be determined by an expert and then encoded as a data type. The performance of most machine learning (ML) algorithms depends on how accurate the features are. One big difference between DL and machine-learning algorithms is that DL can get high-level features from data, which is not possible with traditional methods[60].
- Approach to problem solving : When using machine learning algorithms to solve problems, the problem is divided into multiple smaller problems, which are to get the final answer. Deep learning, on the other hand, promotes direct end to end issue solutions[60].
- Time of execution: Deep learning algorithms take longer to train due to having many parameters, while machine learning training is faster (seconds to hours). However, during testing, deep learning algorithms are quick, while some machine learning algorithms may take longer as data volume increases[60].

2.3.3 Types of Deep Learning Architectures

Deep learning uses several processing layers with intricate architecture in an effort to model enormous amounts of data. Consequently, it differs from shallow machine learning architectures. Deep architectures come in a vast array of variations, and various structures can be utilized to represent various data sources. For instance, recurrent neural networks are better suited for sequential tasks like handwriting or speech recognition, while convolutional neural networks are the most often used design for picture recognition and in several applications among other networks. We shall present several designs in this part and discuss it with a focus on CNN and transformer in the explanation :

1. ***Recurrent Neural Network (RNN)*:** are designed for sequence and time series data, where each data point depends on previous ones. Unlike regular feedforward neural networks, RNNs use "memory" to remember past information, making them ideal for tasks like text, speech, and handwriting recognition[64].

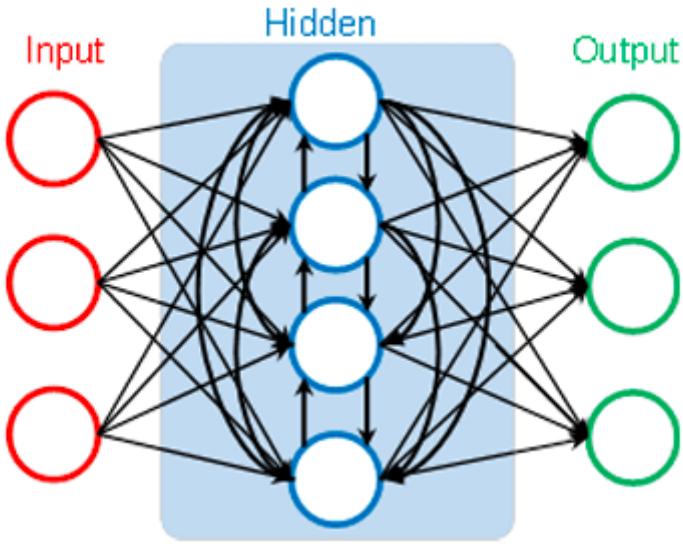


Figure 2.6: Recurrent Neural Network Architecture[64].

2. ***The Deep Belief Networks (DBNs):*** are powerful models capable of not only recognizing and classifying data but also generating new data. A DBN is structured with multiple layers of neurons, which are divided into visible units and hidden units.DBNs are built using stacked layers of Restricted Boltzmann Machines (RBMs).undergo two training steps: pre-training and fine-tuning [64].

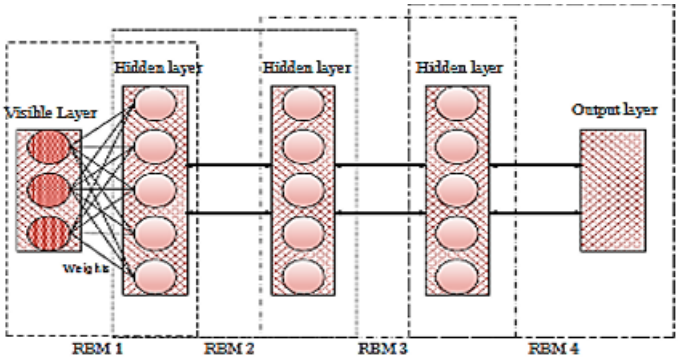


Figure 2.7: Deep belief networks Architecture [64].

3. ***Convolutional Neural Network (CNN)*:** are a widely recognized deep learning architecture, especially effective in areas such as image recognition, natural language processing (NLP), speech analysis, and computer vision. Their structure is inspired by the neural mechanisms in human and animal brains[65]. A typical CNN is composed of three main types of layers: convolutional layers, pooling layers, and fully connected layers [64]. CNNs are known for three major strengths parameter sharing, sparse interactions, and equivalent representations [65], CNN's primary advantage over its predecessors is its ability to recognize pertinent elements automatically without human oversight and the employment of shared weights and local connections in the CNN is instrumental in optimising the utilisation of 2D input data structures, such as image signals. This operation utilises an extremely small number of parameters, which both simplifies the training process and speeds up the network. This approach is analogous to that observed in visual cortex cells. It is important to note that these cells are selective in their sensing,

responding only to specific regions of a scene rather than the entire scene[62]. Some of the most commonly used CNN architectures include AlexNet, VGGNet, ZFNet, GoogleNet, and ResNet[64].

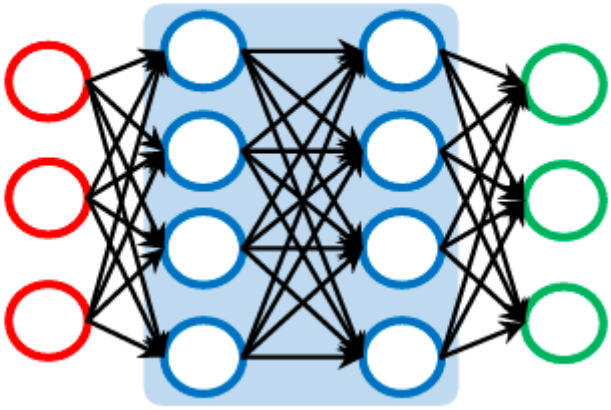


Figure 2.8: Convolutional neural networks Architecture[64] .

(a) **CNN layer types**

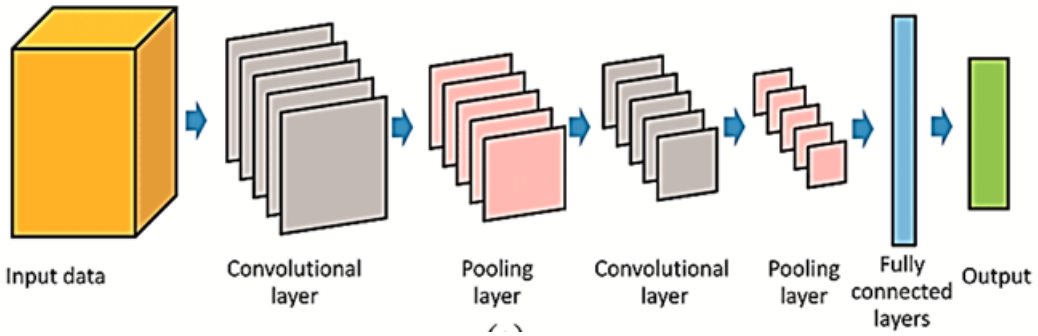


Figure 2.9: Layer-Wise Architecture of a Convolutional Neural Network (CNN)[64].

- Convolutional Layer :is a key component of Convolutional Neural Networks (CNNs), responsible for most of the computation. It focuses on learnable kernels, which are applied across the input data's spatial dimensions to produce a 2D activation map. The layer calculates the scalar product between the filter weights and the corresponding input area to generate the output[66].
- Pooling Layer :reduces the size of feature maps produced by the convolutional layer through sub sampling or down sampling. This process helps shrink large feature maps while retaining important features. Pooling uses predefined kernel sizes and strides, with different techniques suited for various tasks[62].
- Fully Connected Layer :is a key component of Convolutional Neural Networks (CNNs), typically located at the end of the network. In this layer, each neuron is connected to all neurons in the previous layer, acting as the CNN's classifier. The input comes from the flattened feature maps of the preceding pooling or convolutional layer, and the output represents the final CNN result[62].

(b) **CNN model architectures**

CNN's model architecture is essential for enhancing the functionality of various applications. Since 1989, the CNN architecture has seen a number of changes such as regularization, parameter optimization, and structural reformulation are a few examples of these changes[67]. We examine the most widely used CNN architectures in this section.

- ❖ LeNet :was first suggested by LeCun in 1998 . Its historical significance stems from the fact that it was the first CNN to demonstrate cuttingedge performance on tests involving hand digit recognition. Small distortions, rotation, and changes in location and scale can't influence its capacity to classify digits. Five alternating convolutional and pooling layers make up the feed forward neural network (NN) LeNet, which is followed by two fully connected layers[68].
- ❖ AlexNet :is well respected in deep CNN architecture because it produced ground breaking outcomes in image recognition and classification. AlexNet was initially proposed by Krizhevesky et al, who then enhanced CNN's learning capacity by deepening it and applying a number of parameter optimization techniques[62].
- ❖ VGG :Following CNN's success in image identification, Simonyan and Zisserman developed a simple and successful design principle for the network. They termed this innovative design Visual Geometry Group (VGG). Compared to AlexNet, this multilayer model has 19 additional layers to replicate the relationships of the network representational capacity in depth[69].

- ❖ GoogleNet :is also known as Inception V1. The aim of the GoogleNet architecture is to achieve high accuracy with low computational cost[62]. GoogleNet controls the computations by adding a bottleneck layer of 1x1 convolutional filters, before using large size kernels. It also used sparse connections to overcome the problem of redundant information and reduced cost by omitting feature maps that were not relevant[67].
- ❖ ResNet :short for Residual Network, was introduced by He et al. to overcome the vanishing gradient problem in very deep neural networks. Their goal was to create extremely deep architectures without the degradation issues found in earlier models. Various ResNet versions were developed, ranging from 34 to 1202 layers, with ResNet50 being one of the most widely used. ResNet50 includes 49 convolutional layers and one fully connected (FC) layer, the total number of network weights was 25.5 million, while the total number of MACs was 3.9 million.A key innovation of ResNet is the bypass or skip connection, inspired by Highway Networks, which allows the model to train deeper layers more effectively by directly passing information across layers[62].

4. ***Transformer***

The Transformer model, introduced by Vaswani et al. in 2017, addresses several limitations found in traditional deep learning methods. One of its key strengths lies in the attention mechanism, which enables every token in the input sequence to influence the weight of every other token. This ability to capture long range dependencies allows the model to understand the full context of the sequence, significantly improving performance [70],thanks to this contextual awareness transformer models generate more accurate sequence embeddings .Additionally, the model's architecture with direct pathways between distant tokens enhances training efficiency. Since Transformers are built using straightforward components like attention layers and feedforward layers, they are not only easier to train but also highly parallelizable, making them computationally efficient and scalable[71].

(a) **The Transformer architecture:** introduced by Vaswani et al.[70], incorporates multihead attention mechanisms specifically, eight parallel attention heads along with fully connected feedforward networks, which essentially function as multilayer perceptrons (MLPs) within the model's internal structure. Both the encoder and decoder components consist of six stacked layers. To create input embeddings, the model utilizes either byte-pair encoding or a word piece vocabulary strategy, depending on the dataset used for training. Each token in the input is represented as a 512 dimensional contextualized embedding. The multihead attention mechanism across the layers of each network block allows the model to capture valuable and complex representations by analyzing relationships between tokens located at various positions within the input[71].

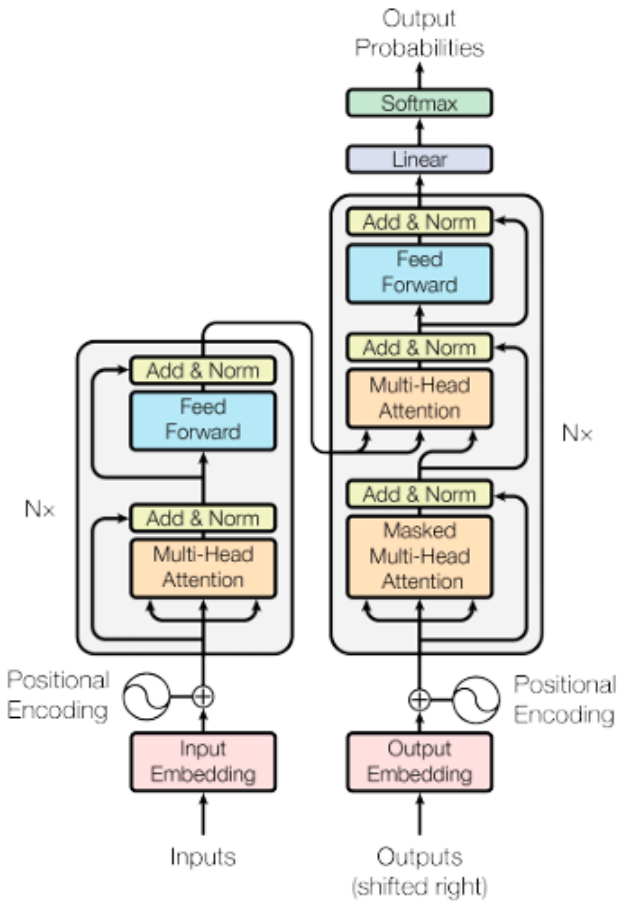


Figure 2.10: Transformer Model Architecture [70]

(b) **Types of Transformers:**

i. **NLP Transformers :** The Transformer architecture was initially widely used because it is very good at sequence modelling and machine translation. In fact, it became the most common deep learning model for natural language processing (NLP). Transformers have been a key part of making large language models like GPT-3 and GPT-4 [72], and BERT, a special type of language model that works by encoding information. BERT has shown very good results in many NLP tasks, like machine translation and answering questions[71].

ii. **Vision Transformer:** Introduction of the Vision Transformer (ViT) by Dosovitskiy et al.[73] in late 2020 precipitated a paradigm shift within the research field. In order to adapt the transformer for image tasks, the authors applied a standard transformer to images by splitting them into patches and providing the sequence of linear embeddings of the patches as the input to the transformer [72].

- **The Vision Transformer (ViT) architecture :**The image is first converted into a series of fixed sized patches, and then subsequently flattened into a set of vectors. These vectors are then passed through a trainable linear projection layer that maps them into N vectors. The dimensionality of these N vectors is D $\times$ N, which is the number of patches. The outputs of this stage are referred to as patch embeddings. In order to preserve the positional information present within each patch, positional embeddings are added to the patch embeddings. Furthermore, a trainable class embedding is appended to the patch embeddings prior to their processing by

the transformer encoder. The Transformer encoder is made up of several identical blocks, each containing a multihead self attention (MSA) layer and a multi-layer perceptron (MLP). Before the data enters these layers, it is first normalized using Layer Normalization (LN). Additionally, residual (skip) connections are applied before normalization, allowing the original inputs to be added to the outputs of the MSA or MLP layers. Finally, an MLP classification head is used to transform the output into the final class prediction[74].

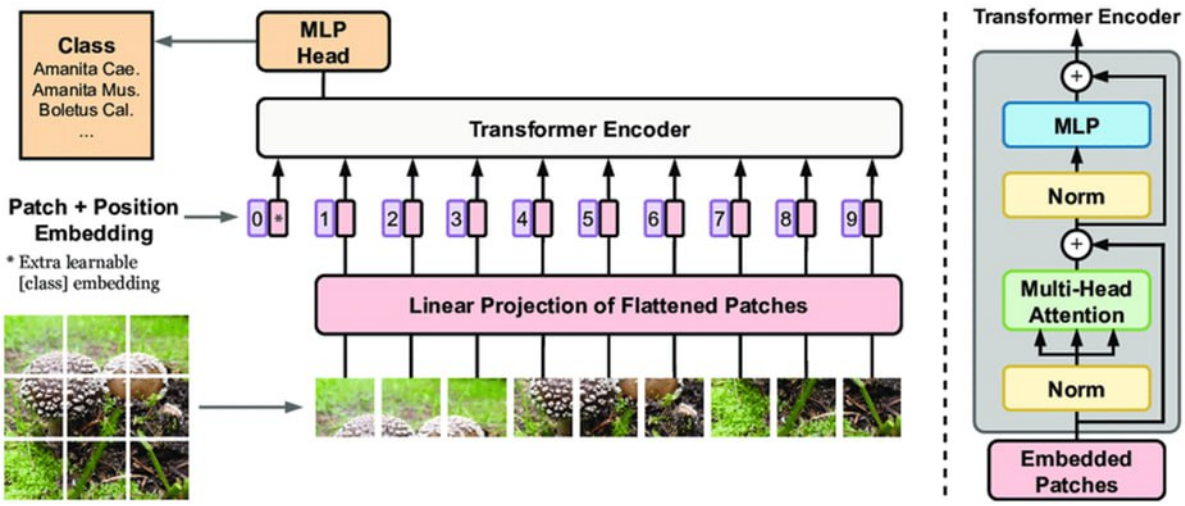


Figure 2.11: The Vision Transformer (ViT) architecture[74].

iii. **Swin Transformer:**is a versatile backbone model for computer vision that delivers high performance across different levels of recognition tasks, including object detection (at the region level), semantic segmentation (at the pixel level), and image classification (at the image level). Its core innovation lies in incorporating key visual principles into the standard transformer architecture such as hierarchical structure, locality, and translation invariance. This approach effectively combines the powerful modeling ability of transformers with visual features that are crucial for success in a wide range of visionrelated applications[75].

- **The Swin Transformer's architecture:**
  Input Processing: The image is divided into non-overlapping patches (e.g., 4×4 pixels). Each patch is treated as a token, where its initial feature vector is of size 48, derived from the RGB values (4×4×3).
  Stage 1: Patch features are transformed through a linear embedding layer to a higher-dimensional space.A series of Swin Transformer blocks, using shifted window self-attention, are applied at this stage while preserving the spatial resolution (H/4 × W/4).
  Stage 2: Adjacent 2×2 patches are merged, reducing the total number of tokens by a factor of four. The resulting feature dimension is increased to 2C. The output then undergoes further processing with Swin Transformer blocks at a resolution of H/8 × W/8.
  Stages 3 and 4: This merging and processing pattern continues in deeper layer, Resolution is reduced to H/16 × W/16, Further reduced to H/32 × W/32
  Final Output: The architecture creates a hierarchical and multiscale representation, enabling strong performance in tasks like image classification,

object detection, and semantic segmentation[76].

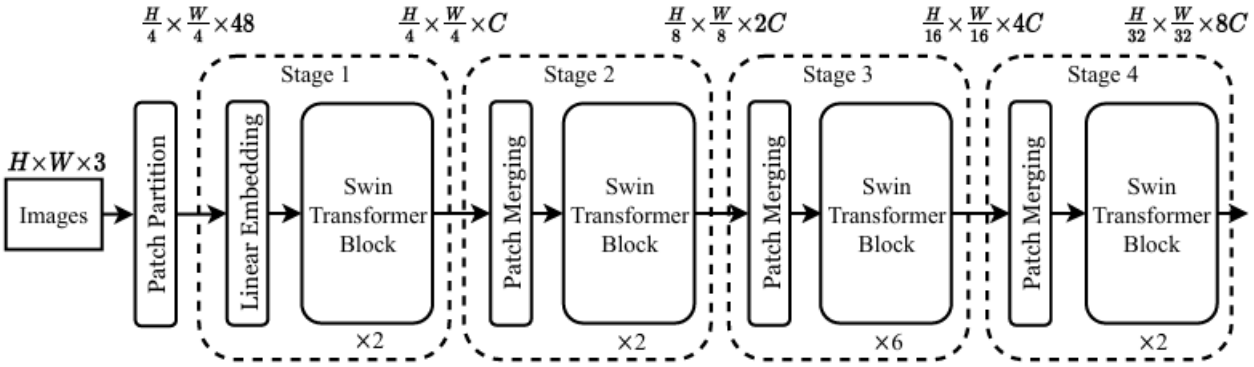


Figure 2.12: The architecture of a Swin Transformer[76].

iv. **DeiT model :** are image transformers notable for their ability to be trained effectively even without massive datasets, thanks to advanced training methods, especially a new distillation technique[77].

- **DeiT architecture:** The DeiT model starts with the ViT architecture and enhances it by integrating a feed-forward network (FFN), composed of two linear layers separated by GELU activation, on top of the Multi-head Self-Attention (MSA) layer. Furthermore, as depicted in Figure 2.13, a unique distillation token is included as an input to allow the DeiT model to learn from the output of a teacher model[78].

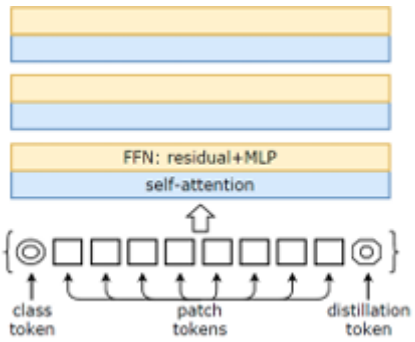


Figure 2.13: DeiT architecture [78].

2.4 Conclusion

The integration of artificial intelligence, machine learning, and deep learning represents a pivotal advancement in modern technology, driving profound transformations across various industries. This study explores the foundational principles, evolution, and practical applications of these technologies, highlighting their ability to mimic human intelligence, adapt to complex environments, and solve intricate problems. As subsets of artificial intelligence, machine learning and deep learning provide specialized tools for data-driven insights, automation, and predictive modeling, pushing the boundaries of innovation.

CHAPTER 3

METHODOLOGY, RESULT AND DISCUSSION

3.1 Proposed methodology

This study presents a robust and scalable methodology for the detection of plant leaf diseases using convolutional neural network (CNN) and Vision Transformer (ViT) architectures. The suggested design, which uses deep learning algorithms, is scalable and modular, Figure 3.1 illustartes the system consists of four integrated modules, each contributing in the detection pipeline. The first is a data preprocessing module which involves preparing and cleaning the input data. This is followed by the deep learning module for model training and fine-tuning. As part of this, the models are trained and evaluated to assess their performance using various metrics, ensuring both effectiveness and generalizability.

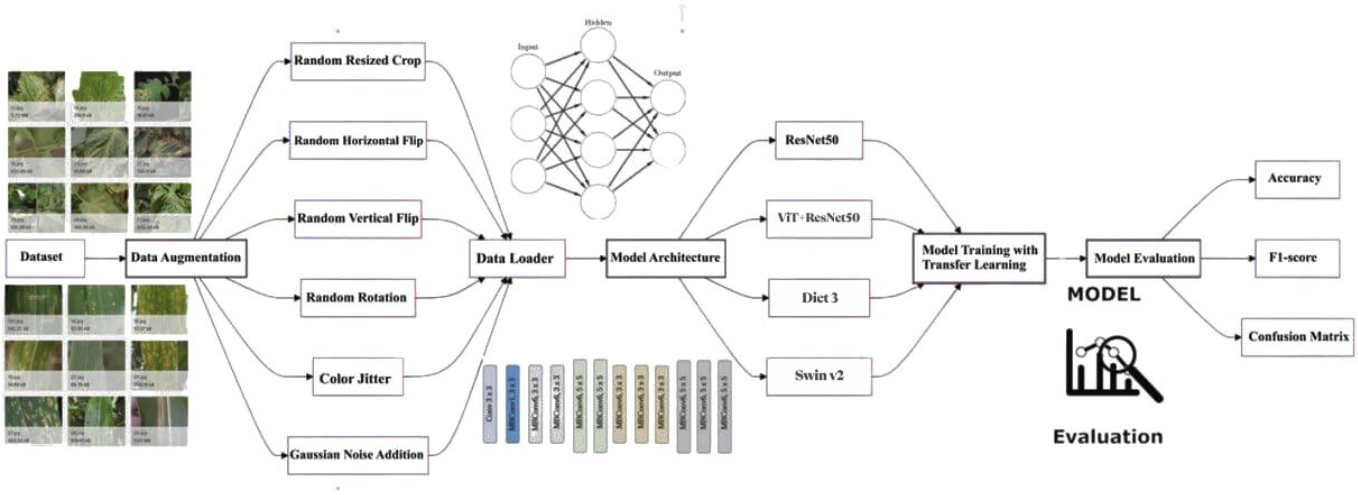


Figure 3.1: The proposed architecture of the plant disease detection system, illustrating the modular design with four key components: data preprocessing, model architecture, training, and evaluation modules

3.1.1 Dataset

In this work a dataset was New Plant Diseases used that contains 87000 rgb images of healthy and diseased leaves of various crops which is categorized into 38 different classes. The tomato crop was selected, which consists of 8 diseased classes and one healthy class and to train and evaluate the performance of the proposed model the dataset was divided into two separate sets by 80/20: a training set containing 29960 images and a testing set containing 7495 images, the following figure (3.2 and 3.3) show the categories of plant leaf diseases and the percentage of each class with 0 common files indicating that there are no duplicate images between the training and testing sets.

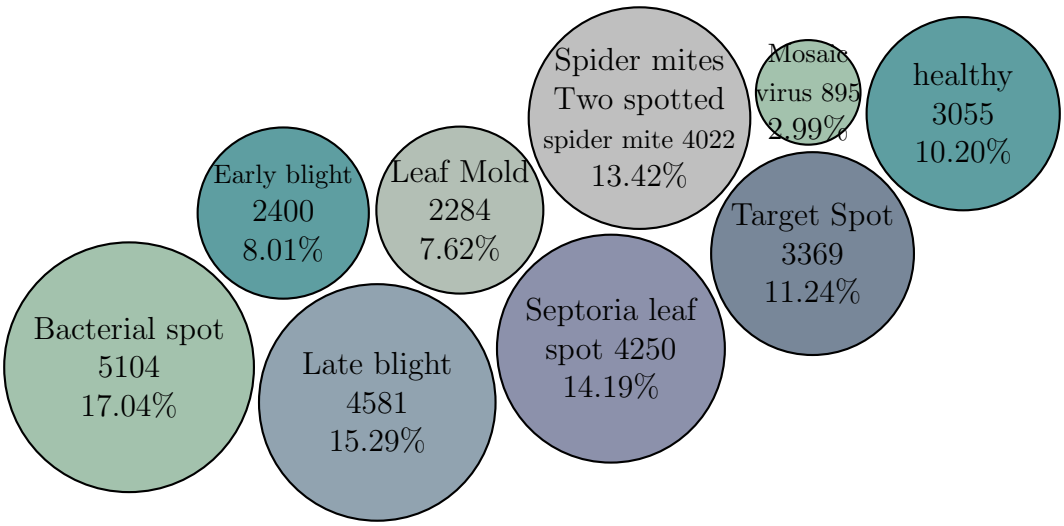


Figure 3.2: Training Set Class Breakdown

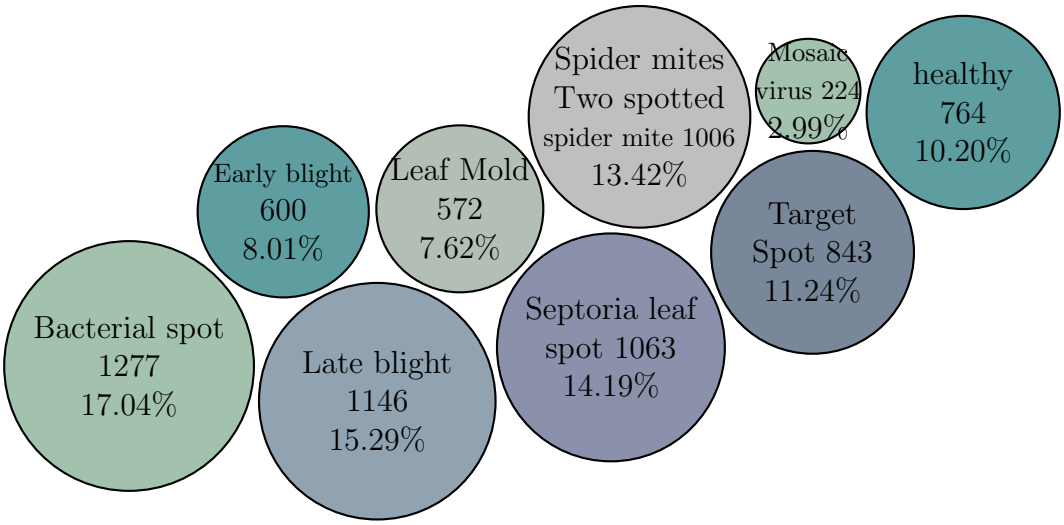


Figure 3.3: Test Set Class Breakdown

3.1.2 Experimental Setup and Configuration

The Kaggle cloud platform which offers a reliable and high-performance environment ideal for deep learning workflows, was used for all training and evaluation tasks in order to carry out the experiments effectively. The following table 3.1 describes the precise hardware and software setup.

Table 3.1: Experimental Environment Configuration

Component	Details
Platform	Kaggle Notebooks (Kernel)
GPU	NVIDIA Tesla P100 (16 GB VRAM)
CPU RAM	16 GB
Software Stack	Python 3.x PyTorch OpenCV (image preprocessing) Scikit-learn (evaluation metrics) Matplotlib and Seaborn (visualization)

3.1.3 Data Preprocessing

Pre-processing image datasets specifically resizing photos and normalizing pixel values is important before using deep learning algorithms on them. resizing photos to the same size (a width/ height of 224 × 224 pixels) which standardizes image dimensions across the dataset. specifically compatible with pre-trained models like ResNet and DeiT 3 size to sizes are crucial to achieve the best results when training a convolutional neural network (CNN) on images. plus normalizing pixel values into the value between 0 to 1 if Example when we divide all our pixels by dividing over (255) The purpose of this normalization is to accelerate the speed of convergence for neural network while training, ensuring that input scores have a similar range so as not only stabilize and also enhance how well we can train our model. these processes enhance the quality and efficiency of our dataset, thereby paving way for creation of more accurate, precise models.

3.1.4 Data augmentation

Data augmentation were essential in enhancing the model's resilience and capacity for generalization. like Random resized cropping was applied to simulate variability in leaf positioning and camera distance thereby increasing the model's adaptability. random horizontal and vertical flips were included as plant diseases are generally invariant to orientation, further diversifying the dataset without compromising realism. random rotations of up to 30 degrees mimicked natural variations while preserving discernible disease features Colour jitter with modest adjustments to account for lighting fluctuations was imple mented. gaussian noise with a mean of 0 and a standard deviation of 0.05 was added to simulate real-world image degradation striking a balance between image degradation and the preservation of critical features,for example figure 3.4.

Figure 3.4: Tomato plant diseases (combined dataset).

3.1.5 Dataset Splitting and K-Fold Cross Validation

K-fold cross validation represents a robust technique for evaluating predictive models in deep learning. This method divides the dataset into K equally sized subsets (folds), with each fold serving as the validation set exactly once while the remaining K-1 folds form the training set. K distinct model assessments are produced once the procedure is repeated K times.

1. **Advantage of K-Fold Cross Validation:** it offers several advantages that make it an essential technique in deep learning model development:
   First, it prevents overfitting by ensuring the model doesn't simply memorize training examples. by training and testing on different data combinations K-fold cross validation confirms that the model captures genuine patterns rather than noise or anomalies specific to a particular dataset segment.
   second it provides reliable performance assessment by averaging metrics across all K iterations. measures like accuracy, precision, and recall become more trustworthy when calculated as the mean of multiple evaluations, reducing the impact of lucky or unlucky data splits.
   third K-fold cross validation enables efficient data utilization. Every data point participates in both training and validation phases, maximizing the value extracted from limited

datasets. this aspect proves particularly valuable in specialized domains like plant pathology, where acquiring labeled data often requires significant resources and expertise.

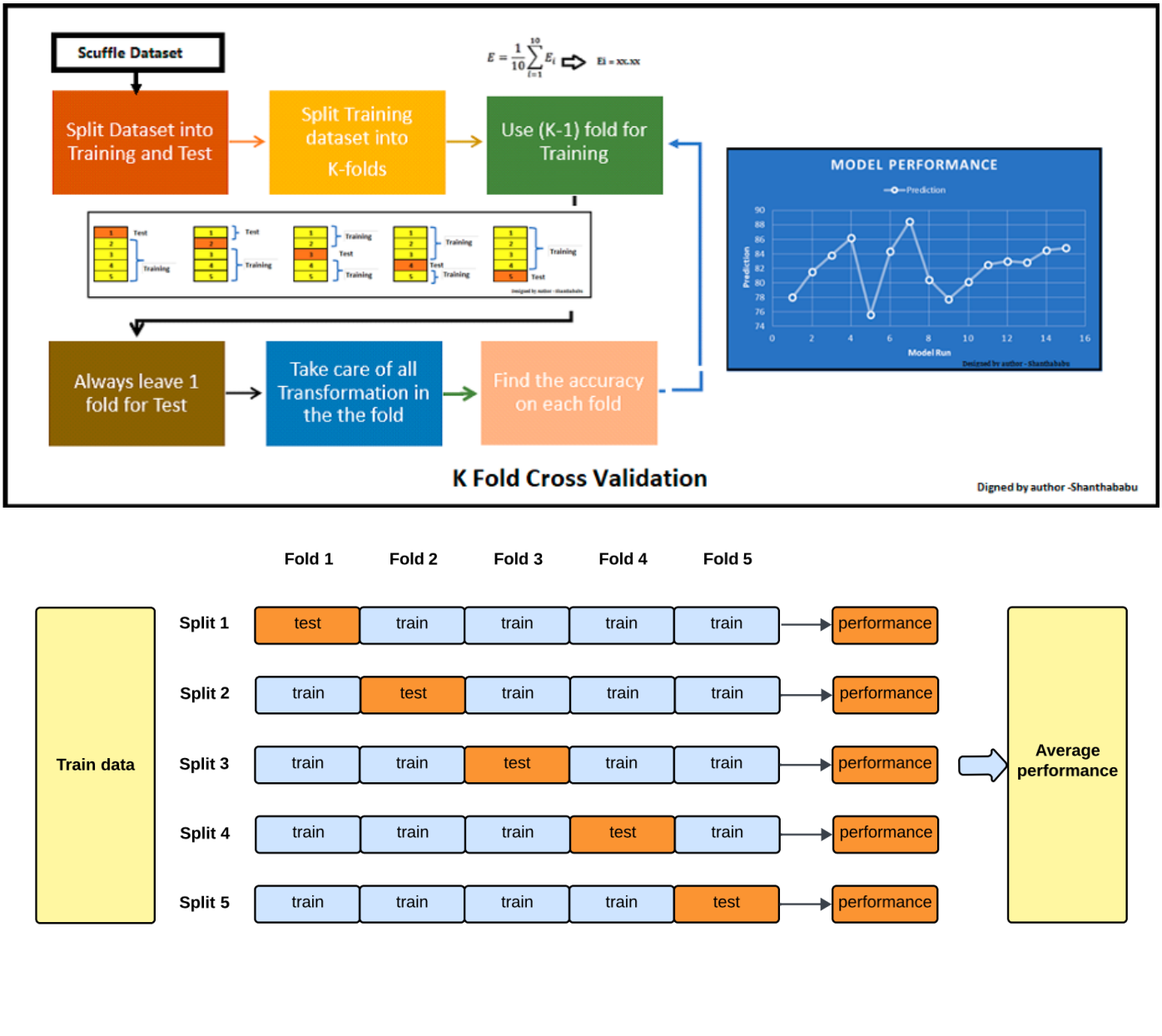


Figure 3.5: Visual Workflow of K-Fold Cross-Validation.

2. **Implementation Process:** This process follows a structured approach as visualized in the accompanying diagram figure 3.5: Initially, we split the complete dataset into training and test sets, typically using an 80%/20%ratio this separation ensures we maintain a completely independent test set for final model evaluation.
   Next we divide the training dataset into K equal folds, the choice of K typically ranges between 5 and 10, balancing computational demands with validation thoroughness.
   The iterative validation phase then begins. For each of the K iterations, one fold is reserved for validation while the remaining K-1 folds are used for training, this process ensures that each data point serves exactly once as validation data and K-1 times as training data, the model's performance is carefully evaluated on each validation fold.
   Performance calculation follows where accuracy metrics from each fold are recorded, these metrics are then averaged to obtain a comprehensive assessment of the model's capabilities across different data configurations.

Finally the test set evaluation occurs, after completing the K-fold process and finalizing model parameters, we evaluate performance on the previously untouched test set, this step provides the ultimate measure of the model's generalization ability.

3. **Practical Application:** In our plant leaf classification research, applying K-fold cross validation helps ensure the robustness of disease identification models. For example with K=5 we would conduct five training iterations each using 80%of the training data for model building and 20%for validation. During each training cycle we maintain consistent preprocessing and transformation procedures applying them independently within each fold to prevent data leakage. This meticulous approach ensures that validation results accurately reflect the model's performance on truly unseen data.

   by examining performance across all folds we can assess model stability and identify potential issues like high variance (performance fluctuating significantly between folds) or high bias (consistently poor performance across all folds) these insights guide further refinements to model architecture, feature selection, or hyperparameter tuning.
   K-fold cross validation represents an essential methodology in our machine learning workflow for plant disease identification systems, by implementing this rigorous validation technique, we can develop models that not only perform well during testing but also maintain their effectiveness when deployed in real-world agricultural applications.

3.1.6 Proposed Frameworks

In this experiment, we evaluated four pre-trained models ResNet50, DeiT V3, SWIN V2, and a Hybrid ViT+ResNet50 architecture on New Plant Diseases.

1. **Performance metrics:** The following evaluation metrics are employed to assess the performance of deep learning models.

   - **Accuracy:** metric quantifies the ratio of correctly predicted classes to all samples analyzed[2] .
     $$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
   - **Recall:** is defined as the percentage of correctly predicted disease-affected leaves relative to the total positive instances of the test case [1].
     $$Recall = \frac{TP}{TP+FN}$$
   - **Precision:** The ratio of correctly predicted disease-affected leaves to all positively predicted leaves by the model is known as precision [1].
     $$Precision = \frac{TP}{TP+FP}$$
   - **F1-score:** is a metric that combines precision and recall in a single metric [32].
     $$F1score = 2 * \frac{Precision * recall}{Precision + recall}$$

     TP stands for the number of samples that were correctly predicted to be positive.
     FP stands for the number of samples that were falsely pre dicted to be positive.
     FN stands for the number of samples that were falsely predicted to be negative.
     TN stands for the number of samples that were correctly predicted to be negative

2. **Experiments and Results** This table 3.2presents a comparative analysis of essential architectural and training attributes for four distinct neural network models: ResNet-50, DeiT 3, SwinV2-Tiny, and Hybrid ViT+ResNet-50.

Table 3.2: Comparison of Deep Learning Models

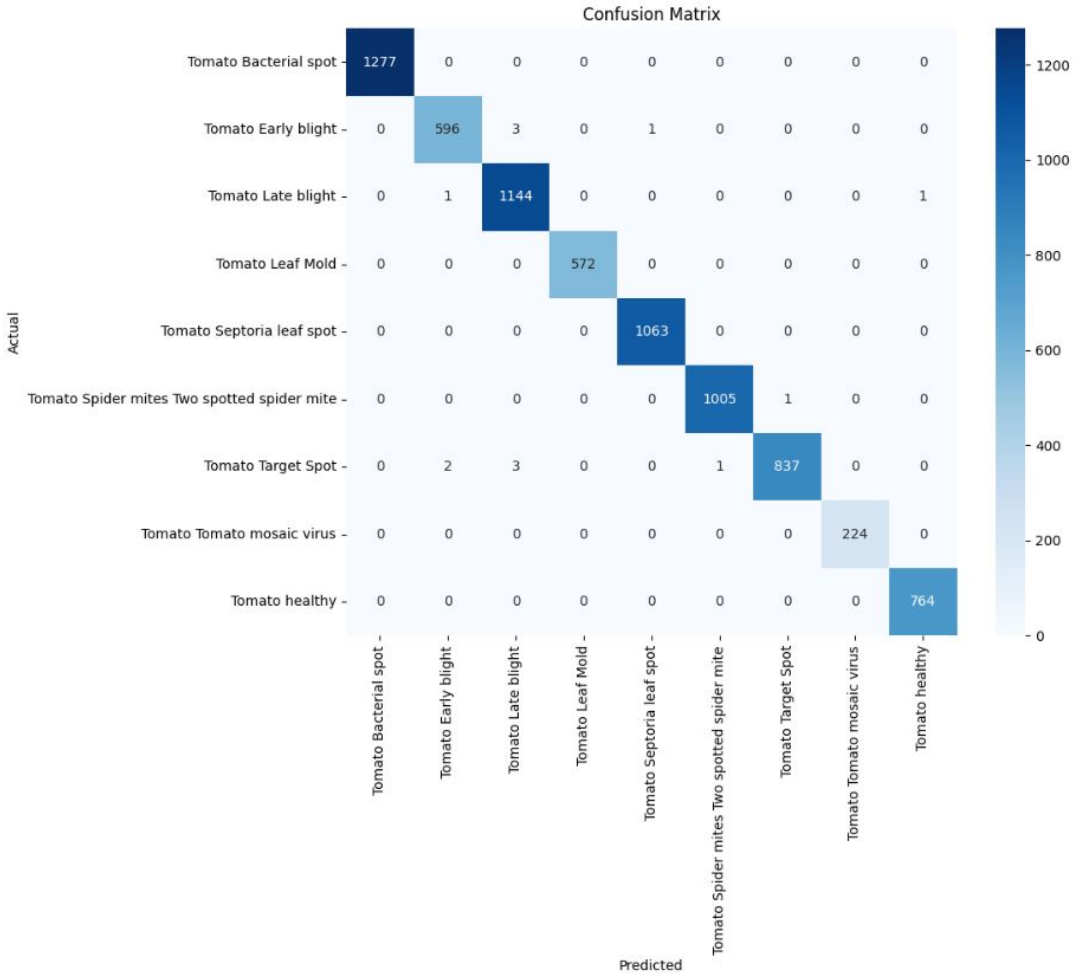
Model	**Image Size**	**Model Size**	**Total Layers**	**Pre-train Images**	**Total Parameters**
ResNet-50	224x224	98 MB	50	∼14M(ImageNet)	∼25.6M
DeiT 3	224x224	∼86MB (base)	12	∼1.28M (ImageNet-1k)	∼86M (base)
SwinV2-Tiny	256x256	∼ 110MB	24	∼1.28M (ImageNet-1k)	∼28M
Hybrid ViT+ResNet-50	224x224	120MBto 150MB (est.)	∼62 (est.)	∼14M (ImageNet, est.)	50M to 100M(est.)

- **Resnet-50 :** Our evaluation of the ResNet50 model on the New Plant Diseases revealed insightful performance characteristics. ResNet-50, with its 50-layer architecture and distinctive residual connections, was specifically configured to address the challenging task of plant disease classification.

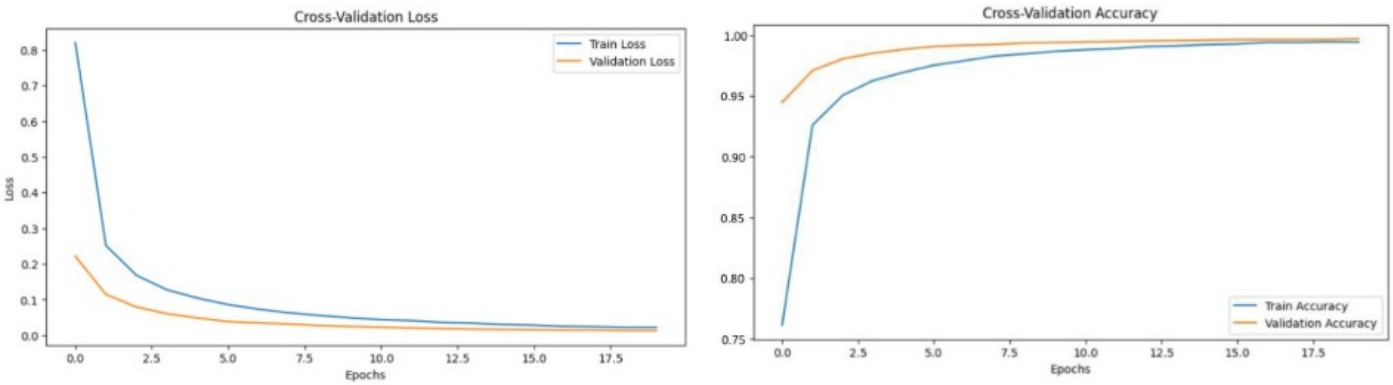
  We implemented ResNet50 with its signature architecture featuring residual connections designed to mitigate the vanishing gradient problem that typically plagues deep networks.

  The network's backbone consists of bottleneck blocks structured with a sequence of convolutions: a 1×1 convolution that reduces channel dimensions, a 3×3 convolution capturing spatial features, and another 1×1 convolution restoring channel size. For optimal performance, we incorporated batch normalization after each convolution layer, significantly enhancing training stability and convergence speed. The architecture employs global average pooling in place of traditional fully connected layers, effectively reducing parameter count and computational demands.

  The network follows a systematic organization with four residual stages of varying complexity: Layer1 with 3 bottleneck blocks, Layer2 with 4 blocks, Layer3 with 6 blocks, and Layer4 with 3 blocks. These are followed by global average pooling and a customized fully connected layer tailored to match our specific plant disease classes.



(a) Confusion matrix



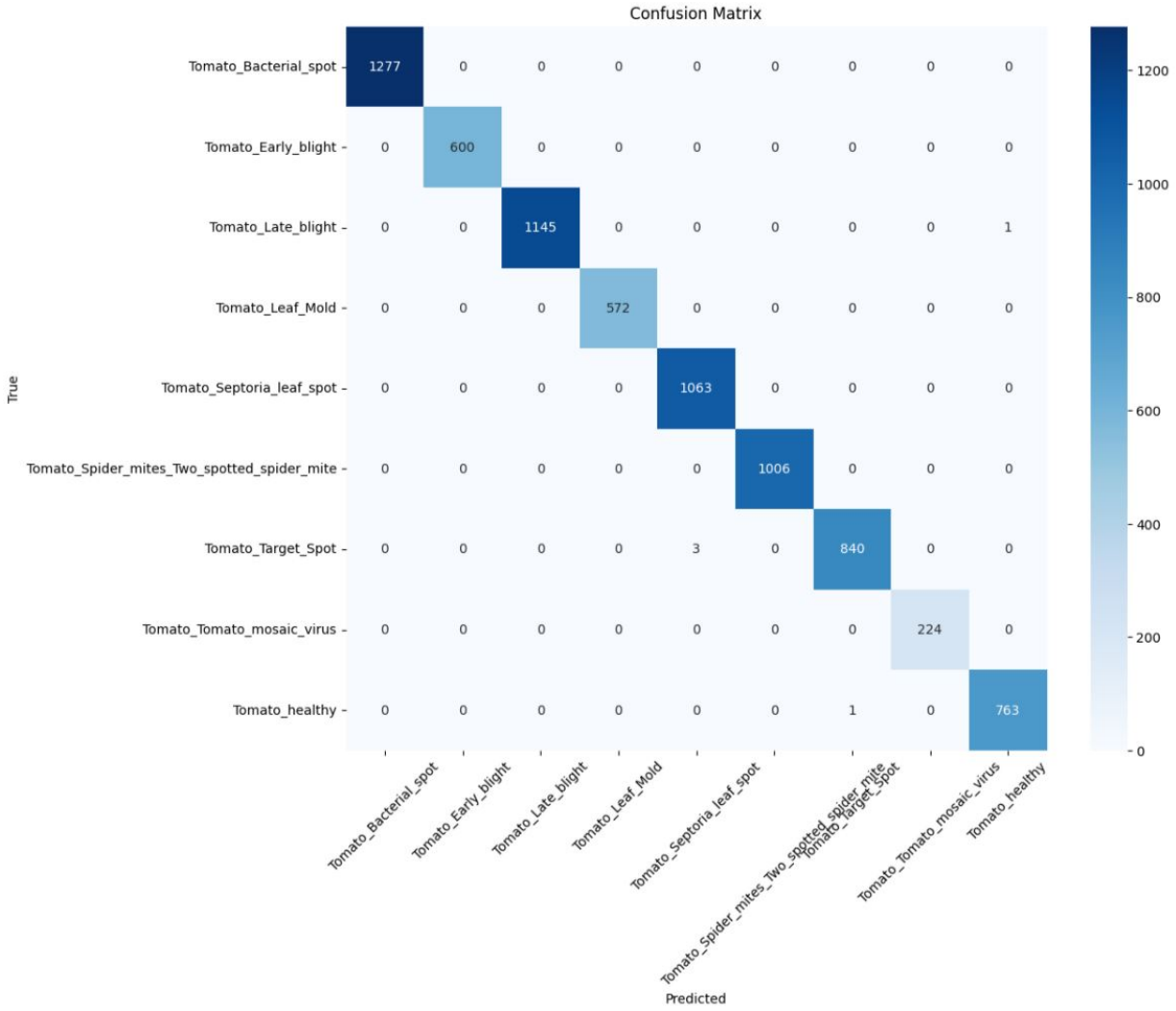
(b) Training and Cross-Validation Curves

Figure 3.6: Model Validation Results Resnet-50.

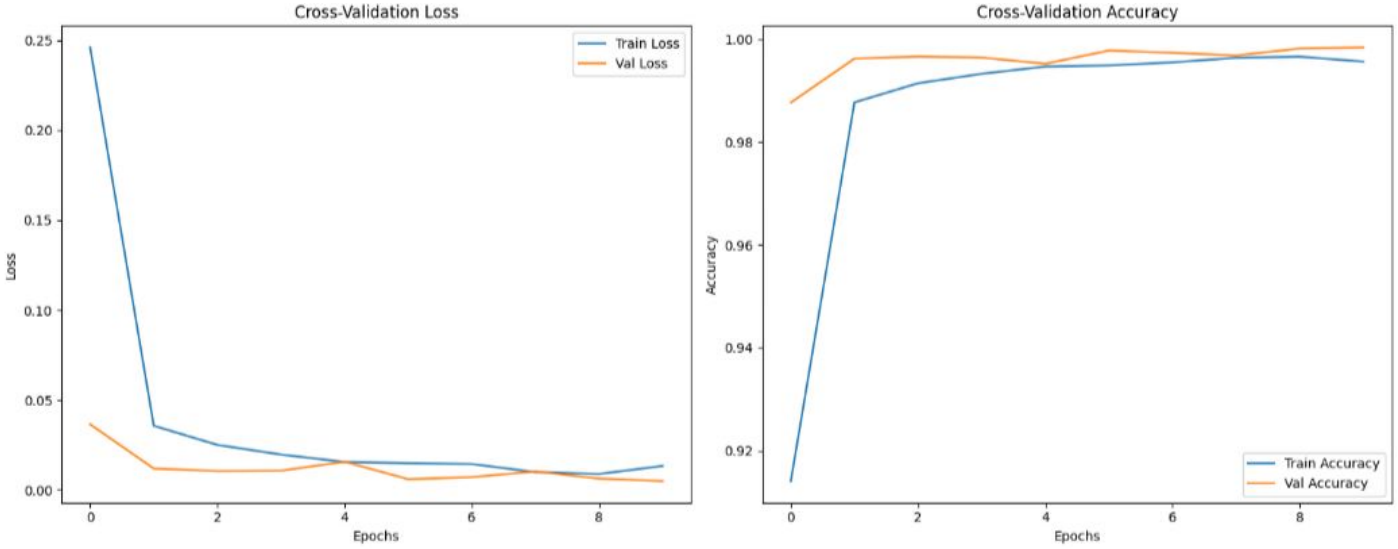
i. **Discussion:**The model exhibits exceptional efficacy in classifying tomato diseases. The cross-validation curves figure 3.6b indicate effective training, since both training and validation losses decline steadily and converge at minimal levels. Correspondingly, training and validation accuracies swiftly ascend and stabilize at exceptionally elevated levels, with validation nearing or surpassing 99%. The good correlation between training and validation sets across epochs strongly suggests effective learning without much overfitting, indicating the model generalizes effectively to novel data.
This elevated performance is validated at a class-specific level by the confusion

matrix in figure 3.6a , it exhibits a pronounced diagonal, indicating elevated true positive rates across all nine categories of tomato diseases and healthy specimens. Off-diagonal elements are highly scarce and exhibit negligible values, signifying minimal misclassifications. The limited errors identified are trivial and frequently occur within visually analogous disease categories, which is a reasonable result even for high-performing algorithms.

- **DeiT 3:** To address the constraints of limited training samples inherent in specialized domains such as plant pathology, we implemented transfer learning with the DeiT V3 (Data-efficient image Transformer) model to overcome the limited sample availability in the plant leaf dataset. This approach significantly reduced training requirements while enhancing generalization capability and mitigating overfitting risks.
  The model was fine-tuned by incorporating a dedicated prediction head upon the final hidden state of the class token, configured for nine-class disease classification. The architecture, pre-trained on ImageNet using 224×224 pixel inputs and a batch size of 16 , incorporates 12 self-attention heads that combine to generate comprehensive attention profiles. These attention mechanisms effectively highlight diagnostically significant regions within leaf images, simultaneously capturing global leaf morphology and localized disease manifestations. This distinctive attention distribution, characteristic of transformer-based architectures, fundamentally differentiates DeiT 3 from conventional convolutional approaches in visual information processing and feature prioritization.
  The model's attention maps provide interpretable visualization of classification decision pathways, enhancing transparency in the diagnostic process while delivering superior performance metrics compared to traditional convolutional neural networks for agricultural pathology identification.



(a) Confusion matrix



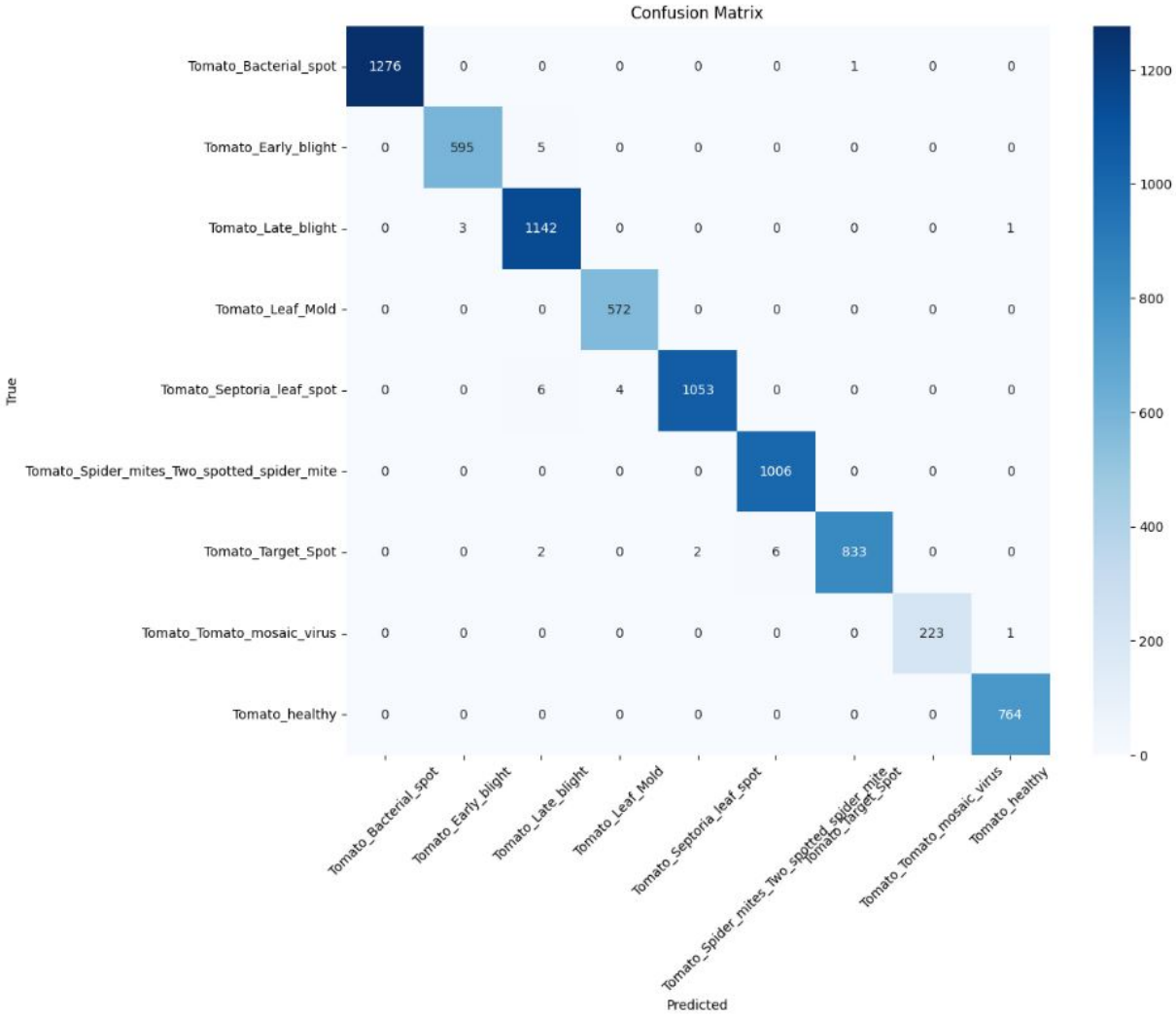
(b) Training and Cross-Validation Curves

Figure 3.7: Model Validation Results DeiT 3.

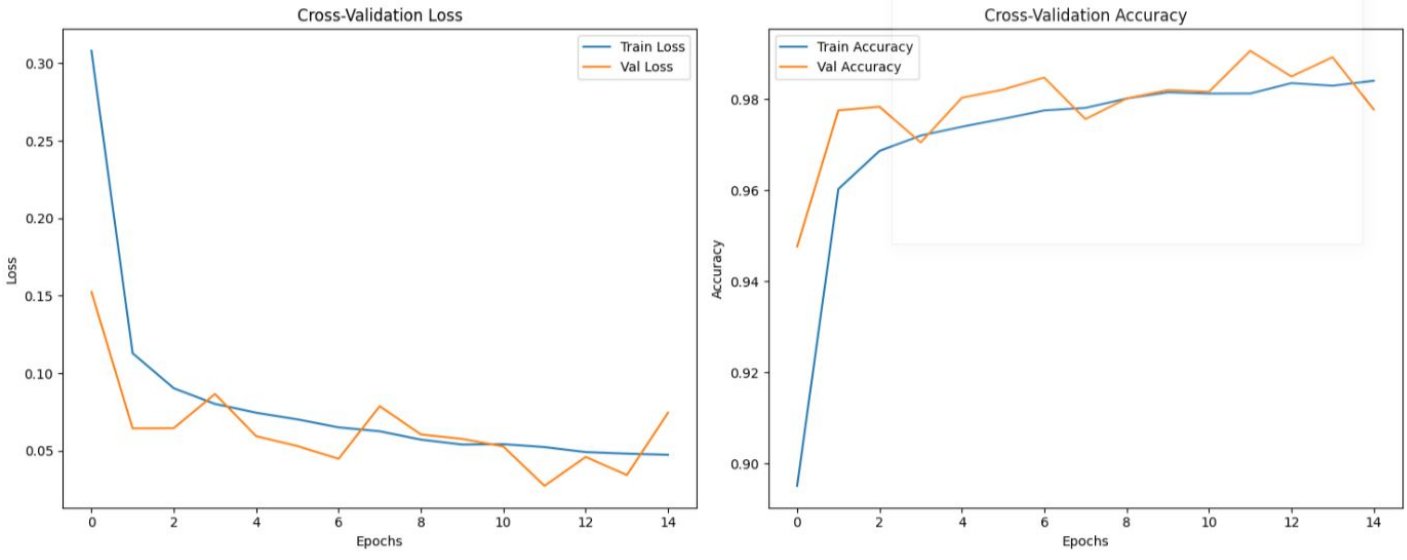
i. **Discussion:**The cross-validation curves figure 3.7b show that the training was quite good. The losses for both training and validation go down quickly and come together at very low values. The validation loss is generally lower than

or very close to the training loss. In the same way, both training and validation accuracies quickly get close to perfect (validation accuracy is always above 99.5%and often gets close to 99.9%).Validation measures that consistently outperform or closely follow each other, with no major gaps, strongly suggest that the model generalizes well, the model learns strong features that can be used with new data.
The confusion matrix figure 3.7a backs up this great performance. It has a nearly fully dominant diagonal, which means that the actual positive rates are quite high for all nine categories of tomato disease and healthy plants. There are very few off-diagonal elements, which means that there are only a few small and very likely misclassifications.

- **SWIN V2-Tiny:** (Shifted Window Transformer V2) implementation in our study features three key technical enhancements.
  First, the network employs a specialized Transformer Stem with multi-scale convolutional layers (1×1, 3×3, and 5×5) that replace the original Patch Partition and Linear Embedding modules, enabling more effective capture of local structural information.
  Second, a Dual-Branch Downsampling structure supersedes the conventional Patch Merging module, utilizing parallel pathways of max pooling with convolution and strided grouped convolution to efficiently reduce spatial dimensions while preserving critical feature information.
  Third, the architecture integrates convolutional operations within transformer blocks, incorporating average pooling in the self-attention mechanism and replacing the standard MLP with an Inverted Residual Feed-Forward Network composed of convolutional layers.
  Our implementation utilized the SWIN V2-Tiny variant (swinv2-tiny-window8-256.ms-in1k) with progressive downsampling at factors of 4, 8, 16, and 32 across successive stages. This hierarchical design creates multi-scale representations that effectively capture disease manifestations at various granularities. This hybrid approach addresses fundamental limitations of pure transformer architectures by introducing inductive bias through strategic convolutional operations, thereby preserving critical 2D structural information in plant leaf images while maintaining the global context modeling capabilities essential for accurate disease classification.



(a) Confusion matrix



(b) Training and Cross-Validation Curves

Figure 3.8: Model Validation Results SWIN V2-Tiny.

i. **Discussion:**Efficient training resulted in successful convergence, with validation measures initially exceeding training metrics. Subsequent epochs demonstrated less validation, although the model sustained high accuracy (about 98-98.5%on

validation). The confusion matrix in figure 3.8a indicates elevated accuracy at the level of individual classes. A prominent diagonal with significant true positive counts across all nine categories of tomato diseases and healthy specimens signifies effective classification. Off-diagonal elements are sparse and indicate modest, frequently reasonable, misclassifications among visually comparable disorders.

- **Combined (VIT+Resnet):**The combined ViT+ResNet50 model employs a multi-learning ensemble approach designed to surpass the performance limitations of individual architectures. This strategic integration combines the complementary strengths of transformer-based Visual Transformer (ViT) and convolutional ResNet50 architectures to enhance disease classification accuracy.

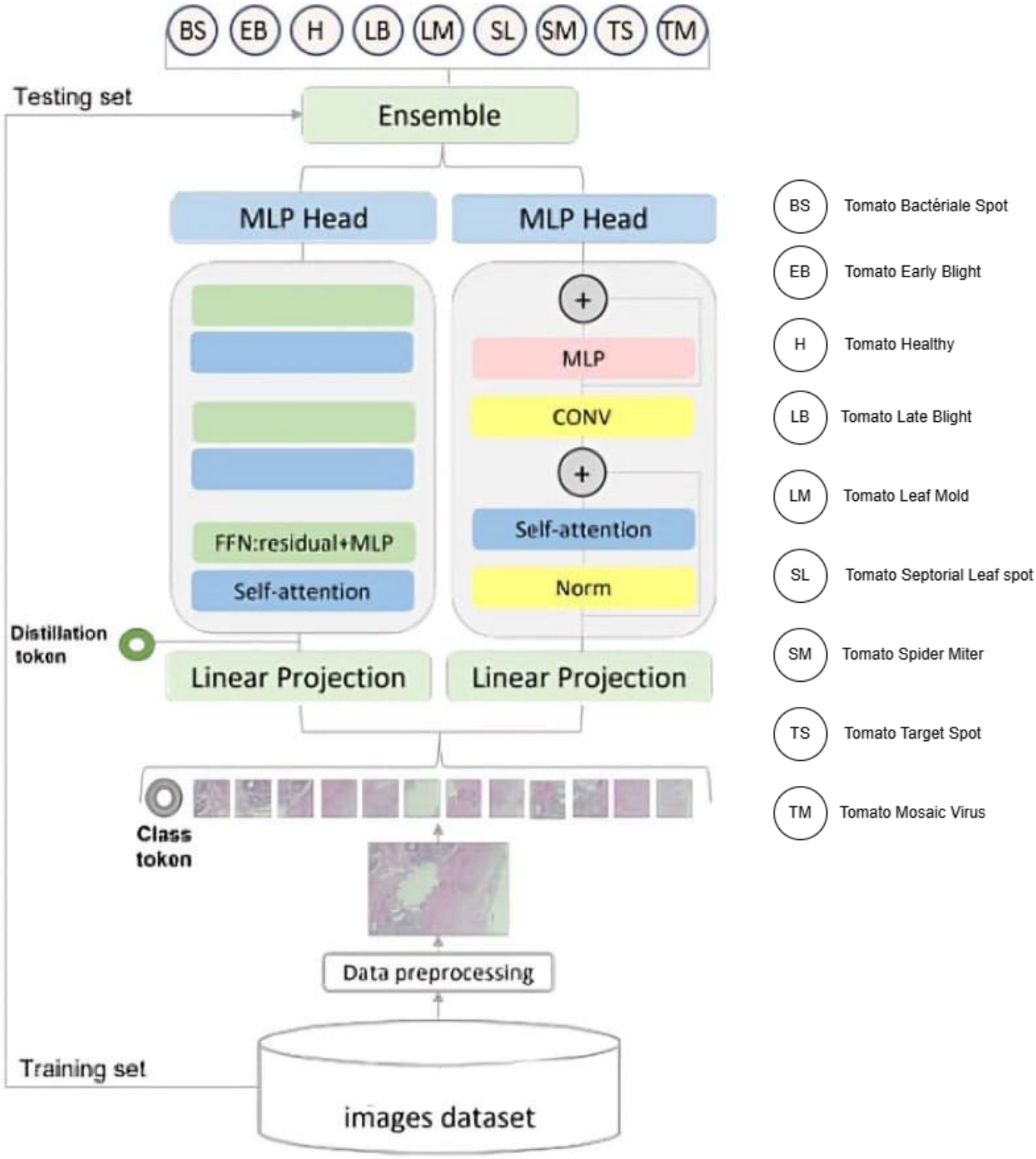
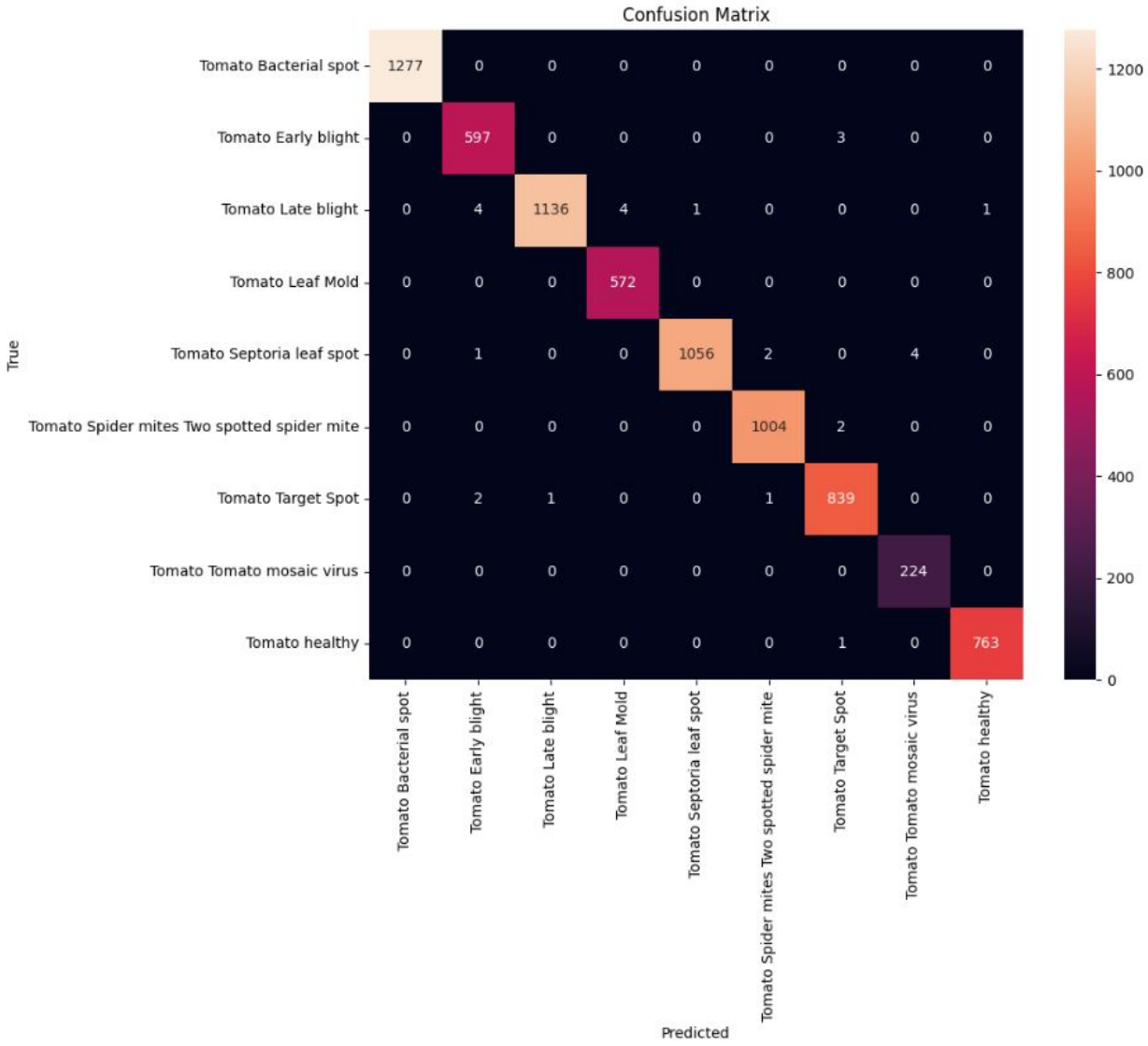


Figure 3.9: Structure of the proposed ViT–ResNet50 model

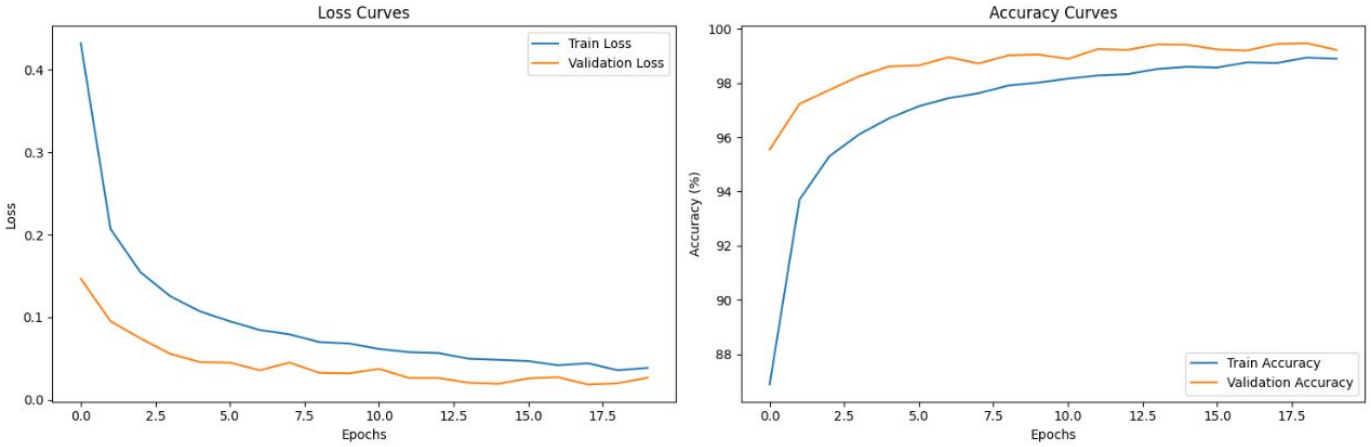
The ensemble architecture, illustrated in Figure 3.9 , leverages the distinctive processing mechanisms of both constituent models. The ResNet50 component contributes hierarchical feature extraction through its convolutional layers, while the ViT component provides global contextual understanding through self-attention mechanisms. A key enhancement in this implementation is the incorporation of a distillation token within the ViT framework, which facilitates effective knowledge

transfer from the ResNet50 teacher model. This distillation token undergoes optimization through backpropagation, systematically interacting with both class and patch tokens across the transformer layers. This interaction enables the model to benefit from the ResNet50's convolutional inductive biases while maintaining the ViT's capacity for capturing long-range dependencies within leaf images. The integration methodology employs soft voting as the ensemble technique, wherein each constituent model generates probability distributions across the disease categories for each input image. These probabilities undergo aggregation through a averaging process, with the sum of probabilities for each class divided by the number of classifiers. The class associated with the highest resultant probability value is designated as the predicted label, as formalized in equation 3.1

$$\hat{y} = \underset{i}{\operatorname{argmax}} \left\{ \frac{1}{N} \sum_{j=1}^{n} p_{ij} \right\} \tag{3.1}$$



(a) Confusion matrix



(b) Training and Cross-Validation Curves

Figure 3.10: Model Validation Results Combined.

i. **Discussion**:The loss and accuracy curves figure 3.10b indicate exceptionally effective training. The training and validation losses continually decline and converge at minimal levels, with the validation loss consistently lower than or nearly equal to the training loss. This signifies that the model is acquiring knowl-

edge effectively from the training data. Both training and validation accuracies similarly exhibit a rapid increase, ultimately plateauing at exceptionally high levels, with validation accuracy constantly around or exceeding 99%. The meticulous monitoring and frequently higher performance of validation measurements relative to training metrics over the epochs strongly indicate exceptional generalization.
This exceptional performance is validated at a class-specific level by the confusion matrix in figure 3.10a. It exhibits a pronounced diagonal, indicating elevated true positive rates across all nine categories of tomato diseases and healthy specimens. Off-diagonal elements are highly scarce and exhibit negligible values, signifying minimal misclassifications.

3. **General Discussion:** The performance metrics presented in Table 3.3 demonstrate exceptional results across all four evaluated models on New Plant Diseases (9 classes). The comparative analysis reveals:

Table 3.3: Performance comparison of different models across datasets

Name of the Model	Accuracy	Precision	Recall	F1-score
New Plant Diseases (9 Classes)				
ResNet 50	99.83	0.9977	0.9977	0.9988
DeiT V3	99.93	0.9994	0.9993	0.9993
SWIN V2-Tiny	99.59	0.9961	0.9957	0.9959
Combined ViT+ResNet 50	99.64	0.9944	0.9966	0.9966

- **DeiT V3:** achieved the highest overall performance with 99.93%accuracy, 0.9994 precision, 0.9993 recall, and 0.9993 F1-score, establishing it as the superior model in this evaluation.
- **ResNet50:** followed closely with 99.83%accuracy and a strong F1-score of 0.9988, demonstrating the continued effectiveness of well-optimized CNN architectures.
- **Combined ViT+ResNet50:** secured 99.64%accuracy with balanced precision (0.9944) and recall (0.9966) metrics.
- **SWIN V2-Tiny:** delivered 99.59%accuracy with consistent performance across all evaluation metrics.

3.1.7 Comparative Analysis with Recent Research

In this section, a comparison with the State-of-art algorithms is presented. The main metric used in the comparison is the accuracy of the models, as accuracy is a simple metric, it is easy to interpret.

Table 3.4: Comparison with Recent Work

Study	Model Used	Database	Accuracy
M.Shetti and Oth.2025 [5]	EfficientNet-B3 model	Combination of the PlanDoc and web sourced datasets	80.19%
K. Joshi and oth 2025 [31]	YOLOv8 model	PlantDoc dataset	96.5 %
I. Bouacidaa and oth 2024 [4]	Inception model	new datasets	97.13%
Ali and oth. 2024 [32]	EfficientNetB3 model	New PlantVillage	99.89%
FAIQA and oth 2023 [2]	EfficientNetB3-AADL model	The dataset used in this research from Kaggle	98.71%
Md. Manow and oth 2023[1]	ResNet-50 ,VGG-16, VGG-19	plant-village dataset	ResNet-50:98.98%, VGG-16:96.15%, VGG-19:92.39%
Our proposed	ResNet 50	New Plant Diseases	99.83 %
	DeiT V3		99.93%
	SWIN V2-Tiny		99.59 %
	Combined ViT+ResNet 50		99.64 %

Table 3.4 shows the comparison Our highest-performing model DeiT V3 at 99.93%accuracy surpasses most recent approaches in plant disease classification. Compared to FAIQA and team. [2] (2023) using the EfficientNetB3-AADL model and K.Joshi and others. [31] (2025) using the YOLOv8 model with an accuracy of 96.5%, our models show significant improvements and performance differences. Furthermore, our implementation outperforms the Inception model with an accuracy of 94.04%reported by Bouacidaa's group [4] (2024). The performance of the ResNet50 implementation with an accuracy of 99.83%also significantly exceeds the results reported by Md.Manow and oth. [1] (2023) for the same architecture with an accuracy of 98.98%, this improvement is likely due to our improved hyperparameter strategy and optimization approach. It is also worth noting that our results closely approximate the 99.89%accuracy achieved by Ali and oth team. [32] (2024) using the EfficientNetB3 model, although they implemented them using a different Kaggle dataset source.

3.1.8 Significance of Findings

The exceptional performance across all evaluated models, particularly DeiT V3, demonstrates the effectiveness of transformer-based architectures for plant disease classification tasks. While all models achieved accuracy exceeding 99.5%, the subtle performance differences highlight the relative advantages of different architectural approaches.

The consistent high performance across diverse architectural paradigms (CNN-based ResNet50, transformer-based DeiT V3 and SWIN V2, and Combined ViT+ResNet50) indicates that New Plant Diseases dataset features distinct disease characteristics that are effectively captured by multiple architectural approaches. This suggests that future research might benefit from focusing on model efficiency, interpretability, and deployment considerations rather than marginal accuracy improvements.

Our implementation advances the state-of-the-art in plant disease classification, with particu-

lar emphasis on the transformative potential of transformer-based architectures for agricultural applications.

3.1.9 Integration of AI in Robotics for Plant Disease Detection

The convergence of artificial intelligence and robotics represents a transformative approach to agricultural automation, particularly in the critical domain of plant disease detection and management.Modern agricultural challenges demand sophisticated solutions that can operate autonomously while maintaining high accuracy and efficiency in disease identification and treatment.

1. **AI in Robotics for Agricultural Applications**
   Artificial intelligence enhances robotic capabilities by providing intelligent decision-making processes, pattern recognition, and adaptive responses to environmental conditions.In agricultural robotics, AI integration enables systems to process complex visual data, identify disease patterns with high precision, and execute targeted interventions. This synergy between AI and robotics addresses the limitations of traditional manual inspection methods, which are constrained by workforce availability, sampling resolution, and the physical demands of comprehensive field monitoring.
   The implementation of AI-driven robotics in agriculture offers several key advantages: improved accuracy in disease detection through advanced image processing algorithms, enhanced efficiency via autonomous operation and real-time data processing, and increased adaptability to varying environmental conditions and crop types.These capabilities are particularly valuable in greenhouse environments, where controlled conditions can be optimized while simultaneously monitoring for potential disease threats.

2. **CROPX Agricultural Robot System Architecture**
   Our developed autonomous agricultural robot,represents an integrated solution for plant disease detection and targeted treatment.The system combines multiple technological components to create a comprehensive agricultural monitoring platform capable of autonomous operation in greenhouse and field environments.

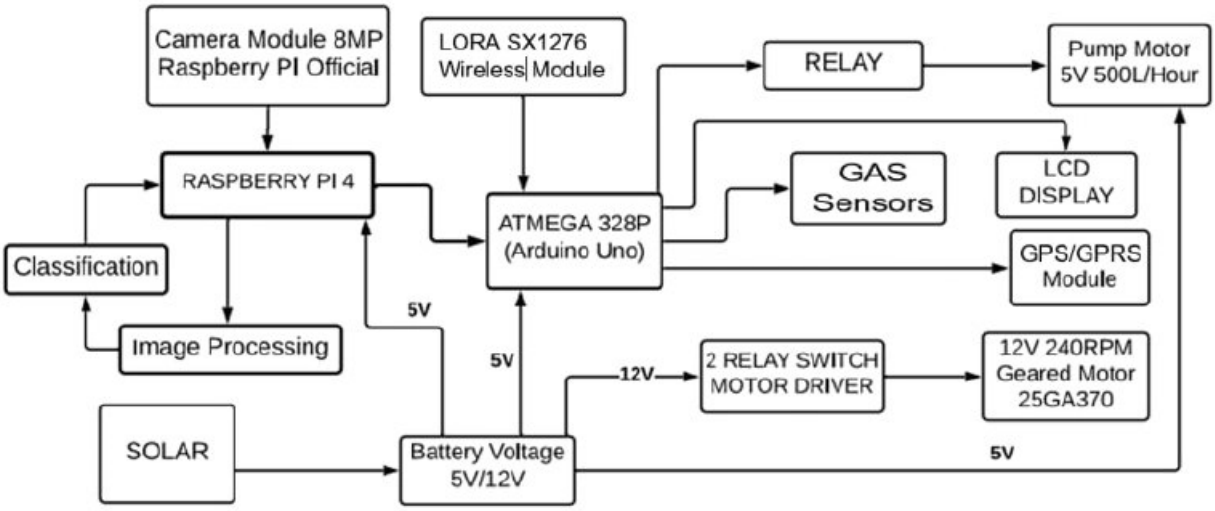


Figure 3.11: System architecture .

- **Hardware Configuration and Power System:** The robot employs a dual-processing architecture featuring a Raspberry Pi 4 as the primary processing unit for image analysis and AI computations, supported by an Atmega328p (Arduino Uno) microcontroller for robotic movement control and sensor coordination.This distributed processing approach ensures optimal performance allocation between high-level AI tasks and real-time control operations 3.11.
  Power sustainability is achieved through an integrated solar energy system complemented by lithiumion battery backup.The solar panel configuration enables approximately 7 hours of continuous daily operation, with the photovoltaic cells converting sunlight into electrical energy that is regulated through a solar charge controller to prevent battery damage from voltage fluctuations. Power distribution is managed through a DC-DC buck converter system, providing 12V for motor drivers and 5V for processing units and auxiliary components.
  The mobility platform consists of four 240 RPM gear motors designed for smooth operation across uneven greenhouse surfaces.The compact robot design, smaller than its solar panel charging station, allows for flexible positioning while maintaining energy independence.
- **Sensor Integration and Data Acquisition:** The robot incorporates multiple sensing modalities for comprehensive environmental monitoring. The primary visual sensor is a Raspberry Pi camera module configured for real time image processing, synchronized with the robot's movement speed to minimize motion blur and ensure high-quality image capture. Additional sensors monitor temperature, humidity, and CO2 levels, providing contextual environmental data that supports disease detection accuracy.
  Navigation and positioning are facilitated through GPS/GPRS modules for location tracking and data transmission,enhancing the robot's ability to comprehensively survey plant populations.
- **AI-Driven Disease Detection and Response System** The core AI functionality centers on the implementation of the DeiT V3 model, trained on the New Plant dataset for plant disease classification. When the robot captures plant images, the Raspberry Pi processes these images through the pre-trained model,comparing detected patterns against the established disease database.Upon positive disease identification, the system immediately triggers a coordinated response sequence.
  The detection workflow operates as follows: continuous image acquisition during autonomous movement, real-time AI processing for disease classification, immediate halt signal transmission to the movement controller upon disease detection, and automated pesticide application through a precision pump system. The treatment protocol delivers exactly 1ml of pesticide over a 3-second application period, ensuring targeted intervention while minimizing chemical usage.
- **Communication and Data Management** The robot maintains connectivity through multiple communication channels. LORA modules enable wireless communication between the robot and manual control interfaces, while GPS/GPRS connectivity facilitates data transmission to cloud-based storage systems. Detected disease information, including captured images and location data, is automatically uploaded to Google Firebase Cloud services for remote monitoring and analysis figure 3.12 shows that.

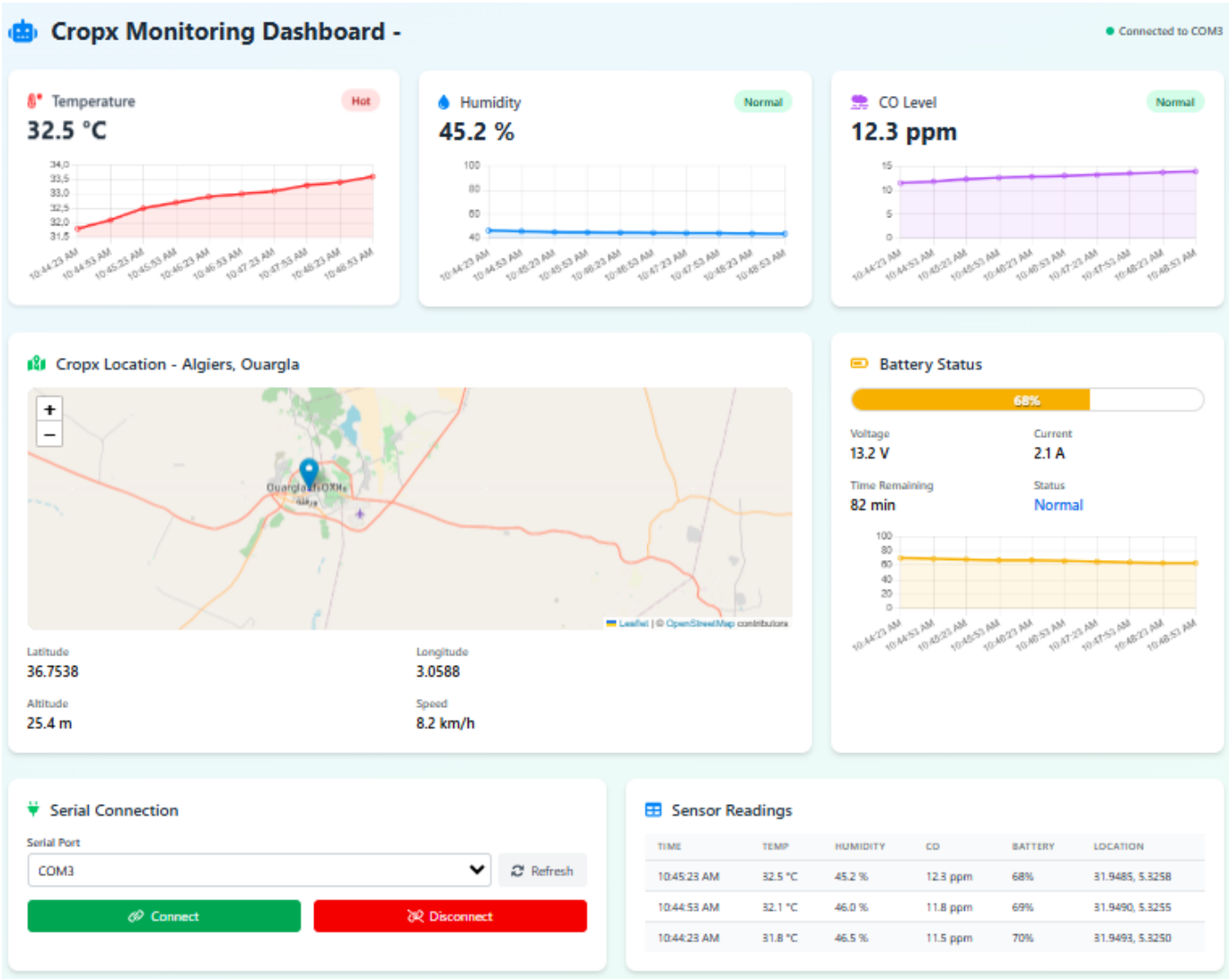


Figure 3.12: Monitoring Interface for the Cropx Agricultural Robot.

A local web-based control interface provides real-time monitoring capabilities, displaying battery voltage, solar power status, movement direction, Sensors reading, and spray system activation indicators. This comprehensive monitoring system enables operators to track robot performance and disease detection activities remotely.

- **Operational Methodology and Field Deployment** The CROPX system operates autonomously at constant speed along predetermined paths between plant rows. The synchronized camera operation ensures optimal image quality for disease detection, while the integrated AI processing provides immediate analysis results. Upon disease detection, the robot executes a programmed response sequence: immediate cessation of movement, precise positioning for treatment application, controlled pesticide delivery, and resumption of monitoring activities.
  This integrated approach addresses the critical limitations of traditional manual inspection methods, providing continuous monitoring coverage, immediate response to disease detection, and precise treatment application that minimizes pesticide usage while maximizing treatment effectiveness. The solar-powered operation ensures sustainable field deployment, while the AI-driven detection system maintains consistently high accuracy in disease identification and classification.

3.1.10 Conclusion

This chapter meticulously detailed the methodology, experimental setup, and subsequent results and discussion for the development of a robust system for tomato plant disease detection and use in the CROPX Agricultural Robot System. The approach began with the selection and preprocessing of the "New Plant Diseases" dataset,focusing on tomato leaf images, followed by comprehensive data augmentation techniques to enhance model generalization. A rigorous K-fold cross-validation strategy was implemented to ensure reliable performance assessment ,and the evaluation of four distinct pre-trained deep learning frameworks: ResNet50, DeiT 3, SWIN V2, and Combined ViT+ResNet50 architecture. Each model was subjected to the same training and evaluation pipeline, utilizing performance metrics such as accuracy, precision, recall, and F1-score.Then, the four results were compared with each other, along with previous studies, to determine the best outcome ,and all four evaluated models (DeiT 3, ResNet50, Combined ViT+ResNet50, SWIN V2) demonstrated ex ceptional performance (>99.5 ٪accuracy) in tomato disease classification. DeiT 3 emerged as the top performer, achieving 99.93 ٪accuracy, closely followed by ResNet50 (99.83٪), Combined (99.64٪), and SWIN V2 (99.59٪), The CROPX agricultural robot, designed for plant disease detection, with integrates a high-performing DeiT 3 AI model.

GENERAL CONCLUSION

This work's successfully investigated and demonstrated the efficacy of advanced deep learning techniques for the automated detection and classification of tomato plant diseases, aiming to improve upon existing methodologies and contribute to precision agriculture.

The research commenced with a foundational exploration of image processing principles relevant to agricultural applications, a detailed overview of common tomato plant diseases, their characteristics, and the challenges inherent in their visual diagnosis. This was complemented by a comprehensive review of Artificial Intelligence, Machine Learning, and specifically Deep Learning, detailing their core concepts, historical development, and various architectures pertinent to image recognition, such as Convolutional Neural Networks (ResNet50) and different Vision Transformer models (ViT, DeiT, Swin Transformer).

The core of this study lay in the meticulously designed methodology. This involved the selection and preparation of the "New Plant Diseases" dataset, with a specific focus on tomato leaf images, followed by robust data preprocessing and augmentation strategies to enhance model generalization. A rigorous K-Fold cross-validation approach was employed for reliable performance assessment. Four distinct pre-trained deep learning frameworks ResNet50, DeiT 3, SWIN V2, and a proposed Combined ViT+ResNet50 architecture were systematically trained, evaluated, and compared and their performance was quantified using standard metrics like accuracy, precision, recall, and F1-score, with visual validation through loss/accuracy curves and confusion matrices .

The experimental results demonstrated exceptional performance across all evaluated models, achieving very high accuracy in classifying the nine targeted tomato disease classes from the New Plant Diseases dataset. benchmarked the findings against recent state-of-the-art research (Table 3.4), highlighting the advancements made by this study. The discussion indicated that DeiT 3 achieved the highest overall performance, closely followed by ResNet50, underscoring the potential of both transformer-based and well-optimized CNN architectures.The significance of these findings lies in advancing the state-of-the-art in plant disease classification, particularly emphasizing the transformative potential of transformer-based models for agricultural applications. The consistent high performance across diverse architectural paradigms suggests that these models effectively capture the distinct characteristics of tomato diseases.

In conclusion, this study makes a great contribution by creating and thoroughly testing a strong system for finding diseases in tomato plants and using it in the CROPX Agricultural Robot System. It gives strong evidence that modern deep learning models can be used to solve important problems in agriculture in a way that is accurate, efficient, and scalable. Our implementation not only improves the state of the art in plant disease, but it also shows that it can be used in the CROPX autonomous agricultural robot. This helps support sustainable farming practices and improve food security. The study not only met its main goal of building on previous research, but it also set the stage for more research into model efficiency, interpretability, and real-world use in agriculture with the CROPX Agricultural Robot System.

BIBLIOGRAPHY

[1] M. M. Islam et al., "DeepCrop: Deep learning-based crop disease prediction with web application," J. Agric. Food Res., vol. 14, no. March, p. 100764, 2023, doi: 10.1016/j.jafr.2023.100764.

[2] F. Adnan, M. J. Awan, A. Mahmoud, H. Nobanee, A. Yasin, and A. M. Zain, "EfficientNetB3-Adaptive Augmented Deep Learning (AADL) for Multi-Class Plant Disease Classification," IEEE Access, vol. 11, no. August, pp. 85426–85440, 2023, doi: 10.1109/ACCESS.2023.3303131.

[3] R. Bora, D. Parasar, and S. Charhate, "Plant Leaf Disease Detection using Deep Learning: A Review," 2022 IEEE 7th Int. Conf. Converg. Technol. I2CT 2022, pp. 1–24, 2022, doi: 10.1109/I2CT54291.2022.9824925.

[4] I. Bouacida, B. Farou, L. Djakhdjakha, H. Seridi, and M. Kurulay, "Innovative deep learning approach for cross-crop plant disease detection: A generalized method for identifying unhealthy leaves," Inf. Process. Agric., no. February, 2024, doi: 10.1016/j.inpa.2024.03.002.

[5] I. H. Sohan, S. Rahman, G. Chhabra, K. Kaushik, and R. Haque, "Plant Leaf Disease Identification Method Using Computer Vision and Machine Learning Algorithms," Int. Conf. Integr. Intell. Commun. Syst. ICIICS 2023, vol. 3, no. April, pp. 305–310, 2023, doi: 10.1109/ICIICS59993.2023.10421345

[6] U. Barman et al., "ViT-SmartAgri: Vision Transformer and Smartphone-Based Plant Disease Detection for Smart Agriculture," Agron. 2024, Vol. 14, Page 327, vol. 14, no. 2, p. 327, Feb. 2024, doi: 10.3390/AGRONOMY14020327

[7] R. C. . Gonzalez and R. E. . Woods, Digital image processing. Prentice Hall, 2002.

[8] K. Raveendra, Principles of Digital Image Processing. Academic Guru Publishing House, 2024

[9] D. H. Xia et al., "Review-material degradation assessed by digital image processing: Fundamentals, progresses, and challenges," Sep. 15, 2020, Chinese Society of Metals. doi: 10.1016/j.jmst.2020.04.033.

[10] Z. Wang, J. Wang, Y. Behnamian, Z. Gao, J. Wang, and D.-H. Xia, "Pitting growth rate on Alloy 800 in chloride solutions containing thiosulphate: image analysis assessment," Corros. Eng. Sci. Technol., vol. 53, no. 3, pp. 206–213, Apr. 2018, doi: 10.1080/1478422X.2018.1432738.

[11] F. Perez-Sanz, P. J. Navarro, and M. Egea-Cortines, "Plant phenomics: An overview of image acquisition technologies and image data analysis algorithms," Nov. 01, 2017, Oxford University Press. doi: 10.1093/gigascience/gix092.

[12] J. M. S. Prewitt, "Parametric and Nonparametric Recognition by Computer: An Application to Leukocyte Image Processing," Adv. Comput., vol. 12, no. C, pp. 285–414, Jan. 1972, doi: 10.1016/S0065-2458(08)60511-2.

[13] S. S. Lomte and A. P. Janwale, "Plant Leaves Image Segmentation Techniques: A Review," 2017

[14] P. V Kumaraguru and V. J. Chakravarthy, "An Image Feature Extraction and Image Representation Strategies for the Analysis of Image Processing," Indian J. Forensic Med. Toxicol., vol. 11, no. 2, p. 642, 2017, doi: 10.5958/0973-9130.2017.00202.x.

[15] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: An experimental comparison," Inf. Retr. Boston., vol. 11, no. 2, pp. 77–107, 2008, doi: 10.1007/s10791-007-9039-3.

[16] D. Zhang and G. Lu, "Review of shape representation and description techniques," Pattern Recognit., vol. 37, no. 1, pp. 1–19, Jan. 2004, doi: 10.1016/J.PATCOG.2003.07.008.

[17] S. U. Rahman, F. Alam, N. Ahmad, and S. Arshad, "Image processing based system for the detection, identification and treatment of tomato leaf diseases.," Multimed. Tools Appl., vol. 82, no. 6, pp. 9431–9445, Mar. 2023, doi: 10.1007/s11042-022-13715-0.

[18] A. Das, F. Pathan, J. R. Jim, M. M. Kabir, and M. F. Mridha, "Deep learning-based classification, detection, and segmentation of tomato leaf diseases: A state-of-the-art review," Jun. 01, 2025, KeAi Communications Co. doi: 10.1016/j.aiia.2025.02.006.

[19] A. H. Wani, "An overview of the fungal rot of tomato," 2011.

[20] Ali A Alsudani, "The most important diseases caused by fungi in tomato seeds: A review," GSC Adv. Res. Rev., vol. 22, no. 1, pp. 020–030, Jan. 2025, doi: 10.30574/gscarr.2025.22.1.0516.

[21] R. Chaerani and R. E. Voorrips, "Tomato early blight (Alternaria solani): The pathogen, genetics, and breeding for resistance," Dec. 2006. doi: 10.1007/s10327-006-0299-3

[22] E. M. Soylu, S. Soylu, and S. Kurt, "Antimicrobial activities of the essential oils of various plants against tomato late blight disease agent Phytophthora infestans," Mycopathologia, vol. 161, no. 2, pp. 119–128, Feb. 2006, doi: 10.1007/s11046-005-0206-z.

[23] K. Mackenzie, J. Chitwood, G. Vallad, and S. Hutton, "Target Spot of Tomato in Florida 1."

[24] T. Zhao et al., "Understanding the mechanisms of resistance to tomato leaf mold: A review," Hortic. Plant J., vol. 8, no. 6, pp. 667–675, Nov. 2022, doi: 10.1016/J.HPJ.2022.04.008.

[25] E. Osdaghi et al., "A centenary for bacterial spot of tomato and pepper," Mol. Plant Pathol., vol. 22, no. 12, pp. 1500–1519, Dec. 2021, doi: 10.1111/mpp.13125.

[26] P. Adhikari, T. B. Adhikari, F. J. Louws, and D. R. Panthee, "Advances and challenges in bacterial spot resistance breeding in tomato (Solanum lycopersicum L.)," Mar. 01, 2020, MDPI AG. doi: 10.3390/ijms21051734.

[27] S. N. Ong, S. Taheri, R. Y. Othman, and C. H. Teo, "Viral disease of tomato crops (Solanum lycopesicum L.): an overview," Dec. 01, 2020, Springer Science and Business Media Deutschland GmbH. doi: 10.1007/s41348-020-00330-0

[28] Y. Xu, S. Zhang, J. Shen, Z. Wu, Z. Du, and F. Gao, "The phylogeographic history of tomato mosaic virus in Eurasia," Virology, vol. 554, pp. 42–47, Feb. 2021, doi: 10.1016/j.virol.2020.12.009.

[29] C. F. Ajilogba and O. O. Babalola, "Integrated Management Strategies for Tomato Fusarium Wilt," 2013.

[30] K. Jetiyanon and J. W. Kloepper, "Mixtures of plant growth-promoting rhizobacteria for induction of systemic resistance against multiple plant diseases," Biol. Control, vol. 24, no. 3, pp. 285–291, Jul. 2002, doi: 10.1016/S1049-9644(02)00022-1.

[31] K. Joshi et al., "Precision diagnosis of tomato diseases for sustainable agriculture through deep learning approach with hybrid data augmentation," Curr. Plant Biol., vol. 41, no. January, 2025, doi: 10.1016/j.cpb.2025.100437.

[32] A. H. Ali, A. Youssef, M. Abdelal, and M. A. Raja, "An ensemble of deep learning architectures for accurate plant disease classification," Ecol. Inform., vol. 81, no. April, p. 102618, 2024, doi: 10.1016/j.ecoinf.2024.102618.

[33] K. Chhaya, A. Khanzode, and R. D. Sarode, "ADVANTAGES AND DISADVANTAGES OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING: A LITERATURE REVIEW," pp. 9–10, Accessed: Apr. 02, 2025. [Online].

[34] D. D. Luxton, "An Introduction to Artificial Intelligence in Behavioral and Mental Health Care," Artif. Intell. Behav. Ment. Heal. Care, pp. 1–26, Jan. 2016, doi: 10.1016/B978-0-12-420248-1.00001-5.

[35] L. Drukker, J. A. Noble, and A. T. Papageorghiou, "Introduction to artificial intelligence in ultrasound imaging in obstetrics and gynecology," Ultrasound Obstet. Gynecol., vol. 56, no. 4, pp. 498–505, Oct. 2020, doi: 10.1002/UOG.22122.

[36] R. Velik, "AI Reloaded: Objectives, Potentials, and Challenges of the Novel Field of Brain-Like Artificial Intelligence".

[37] "Special Section-Generative Artificial Intelligence, Part I Introduction to Generative Artificial Intelligence", doi: 10.5858/arpa.2024-0221-RA.

[38] F. H. Khan, M. A. Pasha, and S. Masud, "Advancements in Microprocessor Architecture for Ubiquitous AI—An Overview on History, Evolution, and Upcoming Challenges in AI Implementation," Micromachines 2021, Vol. 12, Page 665, vol. 12, no. 6, p. 665, Jun. 2021, doi: 10.3390/MI12060665.

[39] A. M. TURING, "I.—COMPUTING MACHINERY AND INTELLIGENCE," Mind, vol. LIX, no. 236, pp. 433–460, Oct. 1950, doi: 10.1093/MIND/LIX.236.433.

[40] J. R. Slagle, "A Heuristic Program that Solves Symbolic Integration Problems in Freshman Calculus," J. ACM, vol. 10, no. 4, pp. 507–520, Oct. 1963, doi: 10.1145/321186.321193.

[41] C. Nikolopoulos, "Expert systems: introduction to first and second generation and hybrid knowledge based systems," 1997, Accessed: Apr. 04, 2025. [Online]

[42] M. Damar, A. Özen, Ü. E. Çakmak, E. Özoğuz, and F. S. Erenay, "Super AI, Generative AI, Narrow AI and Chatbots: An Assessment of Artificial Intelligence Technologies for The Public Sector and Public Administration," J. AI, vol. 8, no. 1, pp. 83–106, 2024, doi: 10.61969/jai.1512906.

[43] P. S. Aithal, "Super-Intelligent Machines - Analysis of Developmental Challenges and Predicted Negative Consequences," SSRN Electron. J., Aug. 2023, doi: 10.2139/SSRN.4683700.

[44] M. M. Young, J. B. Bullock, and J. D. Lecy, "Artificial Discretion as a Tool of Governance: A Framework for Understanding the Impact of Artificial Intelligence on Public Administration", doi: 10.1093/ppmgov/gvz014.

[45] Y. C. Wong, Y. B. Lin, and M. S. Chen, "International Journal of Electrical Engineering: Foreword," Int. J. Electr. Eng., vol. 11, no. 4, 2004.

[46] J. Bell, "What Is Machine Learning?," Mach. Learn. City, pp. 207–216, May 2022, doi: 10.1002/9781119815075.CH18.

[47] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," Neurocomputing, vol. 187, pp. 27–48, Apr. 2016, doi: 10.1016/J.NEUCOM.2015.09.116.

[48] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: state of the art, current trends and challenges," Multimed. Tools Appl., vol. 82, no. 3, pp. 3713–3744, Jan. 2023, doi: 10.1007/S11042-022-13428-4/FIGURES/3.

[49] T. Ige, A. Kolade, and O. Kolade, "Enhancing Border Security and Countering Terrorism Through Computer Vision: A Field of Artificial Intelligence," Lect. Notes Networks Syst., vol. 597 LNNS, pp. 656–666, 2023, doi: 10.1007/978-3-031-21438-7-54.

[50] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," IBM J. Res. Dev., vol. 3, no. 3, pp. 210–229, Jul. 1959, doi: 10.1147/RD.33.0210.

[51] S. Anush Lakshman and D. Ebenezer, "Application of principles of a Artificial Intelligence in Mechanical Engineering," IOP Conf. Ser. Mater. Sci. Eng., vol. 912, no. 3, 2020, doi: 10.1088/1757-899X/912/3/032075.

[52] H. H. Rashidi et al., "Introduction to Artificial Intelligence and Machine Learning in Pathology and Medicine: Generative and Nongenerative Artificial Intelligence Basics," Mod. Pathol., vol. 38, no. 4, p. 100688, Apr. 2025, doi: 10.1016/j.modpat.2024.100688.

[53] D. Sharma, . . . N. K.-J. of A. R. in C., and undefined 2017, "A review on machine learning algorithms, tasks and applications," researchgate.net, Accessed: Apr. 06, 2025. [Online].

[54] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," SN Comput. Sci., vol. 2, no. 3, 2021, doi: 10.1007/s42979-021-00592-x.

[55] T. Jo, "Machine learning foundations: Supervised, unsupervised, and advanced learning," Mach. Learn. Found. Supervised, Unsupervised, Adv. Learn., pp. 1–391, Feb. 2021, doi: 10.1007/978-3-030-65900-4/COVER.

[56] N. Kumar, V. Maurya, and V. Kumar Maurya, "A REVIEW ON MACHINE LEARNING (FEATURE SELECTION, CLASSIFICATION AND CLUSTERING) APPROACHES OF

BIG DATA MINING IN DIFFERENT AREA OF RESEARCH JOURNAL OF CRITICAL REVIEWS A REVIEW ON MACHINE LEARNING (FEATURE SELECTION, CLASSIFICATION AND CLUSTERING) APPROAC," Artic. J. Crit. Rev., vol. 7, p. 2020, 2020, doi: 10.31838/jcr.07.19.322.

[57] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," Neural Comput., vol. 13, no. 3, pp. 637–649, Mar. 2001, doi: 10.1162/089976601300014493.

[58] J. B. O. Mitchell B.O., "Machine learning methods in chemoinformatics," Wiley Interdiscip. Rev. Comput. Mol. Sci., vol. 4, no. 5, pp. 468–481, Sep. 2014, doi: 10.1002/WCMS.1183.

[59] S. Jamal, S. Goyal, A. Grover, and A. Shanker, "Machine Learning: What, Why, and How?," Bioinforma. Seq. Struct. Phylogeny, pp. 359–374, Jan. 2018, doi: 10.1007/978-981-13-1562-6-16.

[60] Y. Xin et al., "Machine Learning and Deep Learning Methods for Cybersecurity," IEEE Access, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

[61] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539ï.

[62] L. Alzubaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," J Big Data, vol. 8, p. 53, 2021, doi: 10.1186/s40537-021-00444-8.

[63] I. M. Coelho et al., "A GPU deep learning metaheuristic based model for time series forecasting," 2017, doi: 10.1016/j.apenergy.2017.01.003.

[64] T. Endo, "Analysis of Conventional Feature Learning Algorithms and Advanced Deep Learning ModelsAnalysis of Conventional Feature Learning Algorithms and Advanced Deep Learning Models," 2023, doi: 10.53759/9852/JRS202301001.

[65] S. Pouyanfar et al., "92 A Survey on Deep Learning: Algorithms, Techniques, and Applications," A Surv. Deep Learn. Algorithms, Tech. Appl. ACM Comput. Surv, vol. 51, no. 5, p. 92, 2018, doi: 10.1145/3234150.

[66] S. Sakib, N. Ahmed, A. J. Kabir, and H. Ahmed, "An Overview of Convolutional Neural Network: Its Architecture and Applications," Feb. 2019, doi: 10.20944/PREPRINTS201811.0546.V4.

[67] A. Khan, A. Sohail, U. Zahoora, A. Q.-A. intelligence review, and undefined 2020, "A survey of the recent architectures of deep convolutional neural networks," SpringerA Khan, A Sohail, U Zahoora, AS QureshiArtificial Intell. Rev. 2020•Springer, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[68] Y. Lecun et al., "LEARNING ALGORITHMS FOR CLASSIFICATION: A COMPARISON ON HANDWRITTEN DIGIT RECOGNITION".

[69] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., Sep. 2014, Accessed: Apr. 16, 2025. [Online].

[70] A. Vaswani et al., "Attention Is All You Need".

[71] A. Chandra, L. Tünnermann, T. Löfstedt, and R. Gratz, "Transformer-based deep learning for predicting protein properties in the life sciences", doi: 10.7554/eLife.82819.

[72] A. Prati et al., "A historical survey of advances in transformer architectures," mdpi.comAR Sajun, I Zualkernan, D SankalpaApplied Sci. 2024•mdpi.com, 2024, doi: 10.3390/app14104316.

[73] A. Dosovitskiy et al., "AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE", Accessed: Apr. 19, 2025. [Online].

[74] R. Azad et al., "Advances in Medical Image Analysis with Vision Transformers: A Comprehensive Review", Accessed: Apr. 19, 2025. [Online]

[75] Z. Liu et al., "Swin Transformer V2: Scaling Up Capacity and Resolution", Accessed: Apr. 20, 2025. [Online]

[76] Z. Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows", Accessed: Apr. 20, 2025. [Online].

[77] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers and distillation through attention," Proc. Mach. Learn. Res., vol. 139, pp. 10347–10357, Dec. 2020, Accessed: May 21, 2025.

[78] A. Alotaibi et al., "ViT-DeiT: An Ensemble Model for Breast Cancer Histopathological Images Classification," 1st Int. Conf. Adv. Innov. Smart City, ICAISC 2023 - Proc., 2023, doi: 10.1109/ICAISC56366.2023.10085467.