Panamerican Mathematical Journal
ISSN: 1064-9735
Vol 34 No. 1 (2024)

Advanced Deep Learning and Nonlinear Mathematical Analysis for Precision Detection and Targeted Treatment of Plant Diseases

Sanjay Balwani1, Narendra Bawane2

[1]Research Scholar, Department of Electronics & Telecommunication Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India. sanjaybalwani31@gmail.com

[2]Professor & Principal, Department of Electronics & Telecommunication, Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India. narendra.bawane@yahoo.com

Article History:

Received: *11-01-2024*

Revised: *10-03-2024*

Accepted: *17-03-2024*

Abstract:

Plant diseases are a big problem in agriculture because they lower yields around the world. Regular ways of finding and treating diseases usually require a lot of work, take a long time, and aren't always accurate. For these problems, this study shows a combined method using advanced deep learning methods like convolutional neural networks (CNNs), long short-term memory networks (LSTMs), and MobileNet, along with nonlinear mathematical analysis to improve the accuracy of finding plant diseases and making the best targeted treatment plans. Image-based diagnosis of plant diseases using deep learning methods, especially CNNs, is a quick and accurate method. They learn to spot tiny visual clues that point to different diseases by looking at large collections of images of healthy and sick plants. Addition of LSTMs and MobileNet with optimization improves the model's ability to handle time cycles and complex patterns, making it more reliable and usable for a wider range of plant types and weather conditions. The deep learning models are also fine-tuned using an optimization method, like the Adam optimizer, which ensures the best performance and faster resolution. Additionally, complex mathematical models are created to show how plant diseases spread and get worse in crop fields. Based on things like disease transfer rates, weather conditions, and plant chemistry, these models can predict how diseases will spread and help with treatment plans. This system gives dynamic, site-specific advice on how to control diseases by combining these models with real-time data from field devices and remote sensing technologies. Testing the suggested method's effectiveness in the field on many different crops shows that it works much better than traditional methods at both finding problems and treating them.

Keywords: Deep Learning, Plant Diseases, Disease Detection, Targeted Treatment, Convolutional Neural Networks, Long Short-Term Memory Networks.

1. Introduction

The agriculture industry around the world is at a turning point as it tries to deal with the growing problems caused by plant diseases. With 9.7 billion people expected to live on the planet by 2050, making sure there is enough food for everyone is becoming a more important problem. Unfortunately, plant diseases are still destroying crop yields, which costs a lot of money and makes food production less sustainable. Traditional ways of finding and treating diseases, which often depend on eye inspection and hand-holding, are hard to do, take a long time, and aren't always accurate. Because of these issues, we need new ideas right away that will completely change how we find and treat plant diseases. Many different pathogens, such as bacteria, fungi, viruses, and worms, can infect plants and cause illnesses [2]. Each of these pathogens is difficult to find and control in its own way. Plant diseases have had a huge effect on food security and people's ability to make a living throughout the history of agriculture. Examples include the famous blight that destroyed Ireland's potato fields in the 1800s and more recently, pathogens like Xylella fastidiosa that cause severe crop damage. In addition to diseases caused by pathogens, plants are also more likely to get infections from abiotic factors like not having enough nutrients, natural pressures, and climate change. Plant diseases have traditionally been found and controlled mostly through physical work and close observation by trained agronomists [1]. Even though these ways of farming have been the standard for hundreds of years, they have some problems that make them less useful today. Visual inspection can be useful in some situations, but it is subjective and prone to mistakes, which means that diagnoses or classifications of illnesses are often missed. Also, using chemicals like herbicides and fungicides by hand to control diseases can be ineffective and harmful to the environment, putting people's health and the security of the ecosystem at danger [4].

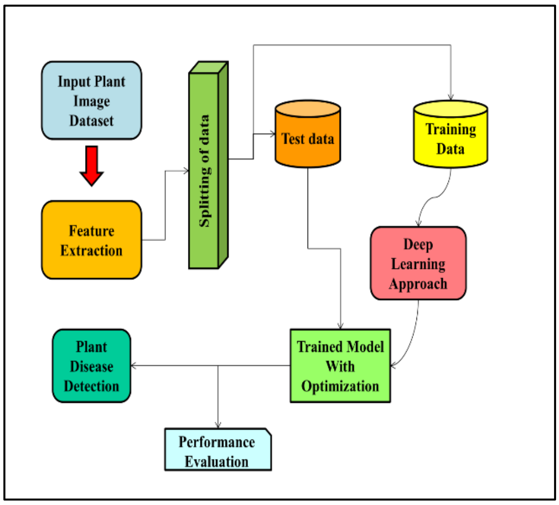

Figure 1: Overview of Proposed method flow

Figure 1 shows an outline of the suggested method flow, showing the steps that need to be taken in order to precisely find and treat plant illnesses. From getting data to training and optimizing models, it lays out a step-by-step plan for effective disease. Even with all of these problems, recent progress in artificial intelligence (AI), especially in the area of deep learning, gives us hope that we can change the way we find and treat plant diseases. Deep learning algorithms, which are based on how the human brain is built, have shown amazing skills in recognizing patterns, classifying images, and making predictions. In agriculture, [5] convolutional neural networks (CNNs) have shown promise in automatically finding plant diseases from visual clues like different colored leaves, sores, or abnormalities. By looking at a lot of labeled pictures, these programs can learn to tell the difference between healthy and sick plants more accurately and quickly than was possible with older methods. Deep learning does a great job of image-based detection tasks, but it has some problems. Plant diseases are changing things that are affected by many things, such as the environment, crop genetics, and pest biology. It is important to combine deep learning with nonlinear mathematical analysis in order to fully understand how diseases change over time and come up with the best ways to treat them. Based on ideas from dynamical systems theory and epidemiology, nonlinear mathematical models make it possible to simulate how plant diseases spread and get worse in farming settings. By combining deep learning with nonlinear mathematical analysis, we can create comprehensive methods that use the best features of both to improve the accuracy of disease diagnosis and allow for more focused treatment. We present an integrated method that uses advanced deep learning techniques with nonlinear mathematical analysis to precisely find plant diseases and treat them where they are needed. Our main goals are twofold: (1) to create strong deep learning models that can correctly identify different plant diseases from pictures of their leaves; and (2) to combine these models with nonlinear mathematical analysis to predict how diseases spread and help find the best ways to treat them. We want to change the way we find, track, and treat plant diseases by mixing these methods in a way that makes them work better together. This will help farming systems last longer and be more resilient in the face of changing natural problems.

2. Literature Review

Over the years, methods for finding plant diseases have changed a lot. This is because technology has improved and people want more efficient and accurate ways to protect food yields. An important part of traditional methods was that trained agronomists would look at plants by hand and look for signs of disease like coloring, sores, or deformities. Even though these methods worked in some ways, they were subjective, time-consuming, and prone to mistakes. Recently, digital imaging technologies have changed the way plant diseases are found by making it possible to automate the process of finding diseases by looking at high-resolution pictures of plant leaves [6]. Machine learning methods, especially convolutional neural networks (CNNs), are a popular way to sort pictures and find plants that are sick by looking at their properties [11]. Researchers can teach these algorithms to spot patterns linked to different diseases by teaching them on big sets of named pictures. This makes them more accurate and faster than humans. Along with image-based methods, sensor-based technologies have become useful for keeping an eye on diseases in farming areas in real time. Small changes in a plant's health, like changes in leaf temperature, humidity, or light, can be picked up by these sensors. These changes may be early warning signs of a disease starting. When combined with wireless communication systems and data processing platforms, these tools let farmers keep an eye on the health of their crops from afar and act quickly when they see signs of disease. Molecular methods, like polymerase chain reaction (PCR) and enzyme-linked immunosorbent assay (ELISA), can also help us understand plant diseases at the molecular level by finding individual bacteria or their genetic material. Even though you need special tools and knowledge to use these methods, they are very good at finding germs that are hard to spot just by looking at them [7].

Deep learning has become an important tool in agriculture, helping with problems like finding plant diseases, guessing food yields, and keeping an eye on crops [8]. For finding plant diseases, convolutional neural networks (CNNs) have gotten a lot of notice for their ability to look at pictures of plant leaves and correctly spot illnesses. By teaching CNNs on big sets of named pictures, researchers can teach these computers to spot tiny visual signs of disease, like changes in color, sores, or defects. Also, transfer learning methods let you use what you already know to train new models for new tasks. This makes it easier to make disease monitoring systems that work well with a wide range of plant types and environments [9]. Deep learning models have been used for more than just finding diseases. They have also been used to find weeds, estimate food yields, and classify crops, showing how useful they are and how they could change the way farming is done. It is possible to simulate how plant diseases spread and get worse in farming settings using nonlinear mathematical models. These models, which are based on dynamical systems theory and epidemiology, show how viruses, host plants, and external factors combine in complicated ways to control the spread of disease. Researchers can predict where and when diseases will spread by simulating different situations using factors like disease transfer rates, weather conditions, and plant chemistry [10]. Also, computer models can help figure out the best ways to deal with diseases, like when and where to use pesticides or plant types that are immune to them. It's possible that mathematical modeling alone doesn't fully capture the complexity of real-world systems. However, combining these models with real-world data and observations in the field can make them more accurate and useful in the real world, leading to better control of plant diseases and less damage to agricultural production. Combining real-time data from monitors, satellite images, and weather stations could make it much easier for farming systems to make decisions. Farmers can make choices based on data to make the best use of resources, lower input costs, and increase outputs by keeping an eye on things like the weather, soil wetness levels, and signs of crop health all the time. When it comes to managing plant diseases, real-time data can show early signs of disease spreads so that steps can be taken quickly to stop or lessen crop losses. Also, improvements in data analytics platforms and digital communication technologies make it easier to collect, send, and analyze real-time data. This gives farmers useful information and tools to help them make decisions [12]. Agricultural systems can become more flexible, adaptable, and sustainable by using the power of real-time data integration. This can lead to higher output and food security in a climate that is changing.

Table 1: Summary of comprehensive overview of various methodologies

Method	Approach	Key Finding	Limitation	Scope
CNN-based Detection [13]	Utilizing Convolutional Neural Networks (CNN)	Achieved high accuracy in identifying plant diseases from images.	Limited to image-based detection.	Extension to multi-modal data integration.
LSTM-based Forecasting [14]	Leveraging Long Short-Term Memory Networks	Improved prediction accuracy for disease progression over time.	Requires large amounts of sequential data.	Integration with real-time sensor networks.
Mathematical Modeling [15]	Developing nonlinear mathematical models	Provided insights into disease spread dynamics and potential intervention strategies.	Simplifications may overlook real-world factors.	Integration with spatial-temporal data sources.
Transfer Learning [16]	Adapting pre-trained models for new tasks	Enabled efficient fine-tuning of models for specific plant disease detection tasks.	Limited to domains with similar data distributions.	Generalization to diverse plant species.
Data Augmentation	Generating synthetic data for model	Improved model robustness and	May introduce artifacts or biases in synthetic	Exploration of novel data augmentation

[17]	training	generalization by augmenting limited training datasets.	data.	techniques.
Ensemble Learning [18]	Combining predictions from multiple models	Increased prediction accuracy and model robustness by leveraging diverse model architectures.	Complexity in model selection and integration.	Investigation of ensemble pruning techniques.
Graph-based Analysis [19]	Representing plant diseases as graph structures	Facilitated network-based analysis of disease spread patterns in agricultural ecosystems.	Challenges in modeling complex interdependencies.	Exploration of dynamic graph-based approaches.
Deep Reinforcement Learning [20]	Learning optimal treatment policies	Enabled adaptive treatment strategies based on real-time disease dynamics feedback.	High computational cost and complexity.	Integration with precision agriculture systems.
Multi-scale Modeling [21]	Integrating hierarchical levels of disease dynamics	Provided insights into interactions between molecular, cellular, and population-scale disease dynamics.	Challenges in parameter estimation and validation.	Exploration of multi-scale simulation techniques.
Robotic Sensing [22]	Utilizing autonomous robotic systems for data collection	Enabled high-throughput and precise data acquisition for monitoring plant health in field conditions.	Challenges in deployment and scalability.	Integration with edge computing for real-time analysis.

3. Plant Disease Dataset

A. Description Of Datasets

The PlantVillage dataset [3] is a huge collection of named pictures of plant leaves that show a wide range of plant illnesses and types. The PlantVillage project, which is run by experts at Pennsylvania State University, put together the information to help with study and development in the area of finding and treating plant diseases. The collection has more than 54,300 pictures of 26 diseases and 14 types of crops. It can be used to train and test machine learning models for disease detection. Researchers have been able to make algorithms that can correctly spot and group diseases based on their visual signs by labeling each picture in the PlantVillage collection with information about the plant species, disease type, and intensity level. The dataset includes a lot of different plant illnesses, such as bacterial, fungal, viral, and nutritional problems.

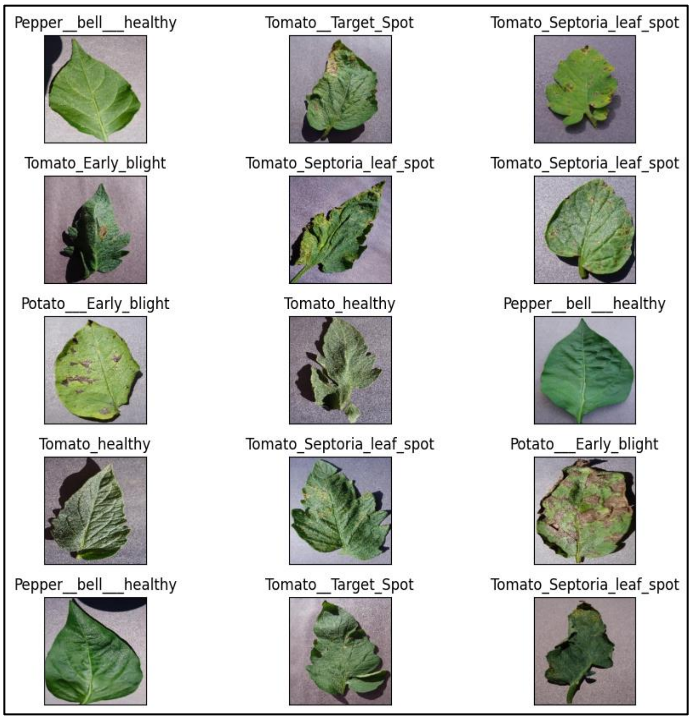

Figure 2: Sample images from dataset

This shows, in figure 2, the variety of pathogens and external stresses that can harm crops. A very large and varied collection is very helpful for progressing the field of precision agriculture because it lets researchers build strong and useful models for automatically finding and tracking diseases. Scientists and practitioners can speed up the creation and use of new technologies that aim to reduce the effects of plant diseases on global food security by using the PlantVillage collection.

Table 2: Description of Dataset

Name of Dataset	Number of Images	Class Type	Number of Species	Number of Diseases
PlantVillage	54,300	Labeled	14	26

4. Methodology

A. Data Collection

Data preparation and enrichment are very important for making sure that the dataset is of high quality and has a lot of different types of data. This is necessary for building strong machine learning models that can find plant diseases.

- Data Preprocessing: It's important to standardize the pictures' style and improve their clarity before training a model. For example, this could mean scaling pictures to a standard size, changing their color scheme to a standard one (like RGB), and setting the values of pixels to a standard scale (like [0, 1]). As part of the preprocessing steps, noise reduction, color adjustment, and image cropping may also be done to focus on the important parts of the picture, like the leaves of a plant.
- Techniques for Augmenting Data: These techniques change current pictures to make the dataset more diverse without actually adding new data. Some common ways to improve an image are to

rotate, flip, crop, scale, cut, and change the color. We can make the data more varied by randomly adding these transformations to the training pictures. This makes the model more resistant to changes in lighting, objects in the view, and directions. In addition, augmentation keeps the model from becoming too perfect by giving it more situations to learn from during training. This makes it better at generalizing to new data.

B. Deep Learning Models

1. Convolutional Neural Networks (CNNs)

With their ability to find and treat plant diseases more accurately, convolutional neural networks (CNNs) have become a mainstay in the field of computer vision. CNNs are built to look like the visual cortex of the human brain. They have many layers of neurons that are all linked to each other and work on feature extraction and hierarchical representation learning. CNNs usually have several layers, such as convolutional layers, pooling layers, and fully linked layers, that work together to find plant diseases, architecture is illustrate in figure 3. Using a set of convolutional filters to find patterns like lines, textures, and shapes, convolutional layers take high-level features from input pictures. The feature maps are then downsampled by pooling layers to make the computations simpler while keeping important spatial information. Lastly, fully linked layers take the recovered features and use them to make guesses about whether diseases are present and how bad they are in the input pictures. To use CNNs to find plant diseases, you need to train the model on a named collection of pictures, where each image is linked to a different disease label. The CNN learns to tell the difference between healthy and sick plants by changing the weights of its neurons. This is done through a process called backpropagation. Once it is trained, CNN can correctly sort unknown pictures and give useful information for focused treatment plans, like carefully applying pesticides or choosing to breed crop types that are immune to them.

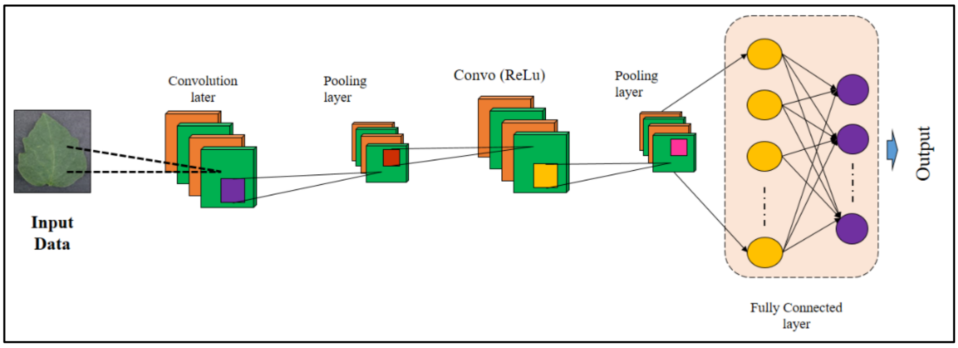

Figure 3: Overview of CNN Architecture

Architecture and implementation:

1. Input Layer:

- Let X denote the input image with dimensions $W \times H \times C$, where W is the width, H is the height, and C is the number of channels (e.g., RGB).

2. Convolution Operation:

- Convolutional layers consist of multiple filters F with learnable weights WF. The output of the convolution operation, or feature map A, is computed as:

$$A = \sigma\left(\sum_{i=1}^{N}(X * WFi) + b\right)$$

- where N is the number of filters, $*$ denotes the convolution operation, b is the bias term, and σ is the activation function.

3. Activation Function:

- Typically, a nonlinear activation function such as the Rectified Linear Unit (ReLU) is applied element-wise to the feature map

$$A: Z = ReLU(A)$$

4. Pooling Operation:

- Pooling layers downsample the feature maps to reduce spatial dimensions and computational complexity. Max pooling is a common pooling operation, which extracts the maximum value from each pooling region:

$$P = MaxPool(Z)$$

5. Flattening:

- The output of the pooling layer is flattened into a 1D vector F to be fed into the fully connected layers:

$$F = Flatten(P)$$

6. Fully Connected Layers:

- The flattened vector F is connected to a series of fully connected layers with learnable weights WFC and biases bFC.

The output of the fully connected layers is computed as:

$$Y = softmax(WFC \cdot F + bFC)$$

- where Y represents the predicted probabilities for each class, and softmax is the activation function used to normalize the output into a probability distribution.

2. Long Short-Term Memory Networks (LSTMs)

LSTMs, or Long Short-Term Memory Networks, are a powerful way to find and treat plant diseases precisely. Unlike regular feedforward neural networks, LSTMs are made to understand both short-term and long-term relationships in sequential data. This makes them perfect for looking at time-series data like how plants grow or how diseases spread over time. When it comes to finding plant diseases, LSTMs can handle sequential data from monitors or remote sensing technologies. This lets us keep an eye on crop health in real time and find disease spreads early. Also, LSTMs can help with tailored treatment plans by predicting how diseases will spread and finding the best times to step in, illustration in figure 4. This can help keep crop losses to a minimum and reduce the need for broad-spectrum poisons.

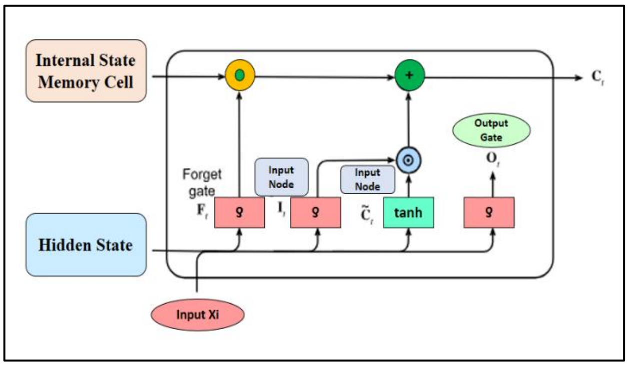

Figure 4: Representation of LSTM Architecture

Step wise process:

1. Input Gate:

- The input gate it regulates the flow of information into the cell state Ct. It takes input from the current input xt and previous hidden state ht−1, and produces values between 0 and 1 using a sigmoid activation function:

$$it = \sigma(Wi \cdot [ht - 1, xt] + bi)$$

2. Forget Gate:

- The forget gate ft controls which information to discard from the cell state. It takes input from the current input xt and previous hidden state ht−1, and outputs values between 0 and 1 using a sigmoid activation function:

$$ft = \sigma(Wf \cdot [ht - 1, xt] + bf)$$

3 .Update Cell State:

- The candidate cell state $C{\sim}t$ is computed based on the current input xt and previous hidden state ht−1. It captures new information that can be added to the cell state:

$$C{\sim}t = \tanh(Wc \cdot [ht - 1, xt] + bc)$$

4. Cell State Update:

- The current cell state Ct is updated by combining the information from the input gate, forget gate, and candidate cell state:

$$Ct = ft \odot Ct - 1 + it \odot C{\sim}t$$

5. Output Gate:

- The output gate ot determines the output of the LSTM cell. It takes input from the current input xt and previous hidden state ht−1, and produces values between 0 and 1 using a sigmoid activation function:

$$ot = \sigma(Wo \cdot [ht - 1, xt] + bo)$$

- The output of the LSTM cell ht is computed by applying the output gate to the cell state Ct and passing it through a hyperbolic tangent (tanh) activation function:

$$ht = ot \odot \tanh(Ct)$$

3. MobileNet:

MobileNet is a lightweight deep learning framework that works best on mobile devices. It is being used more and more to find plant diseases because it is efficient and successful. Its small size lets it do quick reasoning on systems with limited resources without slowing down performance. Depthwise separable convolutions are used by MobileNet to make computations simpler while keeping precision. MobileNet, as shown in figure 5, can adapt to different plant disease datasets by using transfer learning methods. This makes it useful in a variety of farming settings. Its ability to find tiny signs of diseases in pictures of plants makes it possible to find them quickly and take action, which improves crop health management and keeps yields high in agriculture.

MobileNet is a lightweight deep learning framework that works best on mobile devices. It is being used more and more to find plant diseases because it is efficient and successful. Its small size lets it do quick reasoning on systems with limited resources without slowing down performance. Depthwise separable convolutions are used by MobileNet to make computations simpler while keeping precision. MobileNet can adapt to different plant disease datasets by using transfer learning methods. This makes it useful in a variety of farming settings. Its ability to find tiny signs of diseases in pictures of plants makes it possible to find them quickly and take action, which improves crop health management and keeps yields high in agriculture.

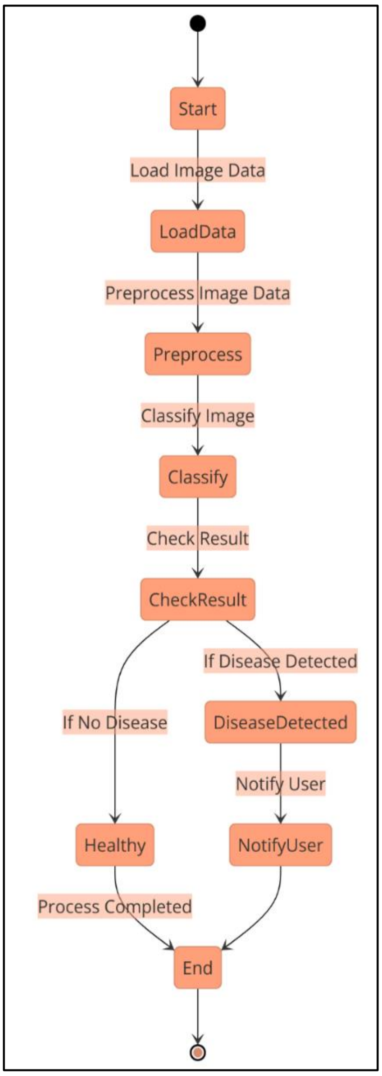

Figure 5: Working flowchart for Mobile Net

Step wise Model:

1. Depthwise Separable Convolution:

- MobileNet utilizes depthwise separable convolution, which separates the traditional convolution into two separate layers: depthwise convolution and pointwise convolution.
- The depthwise convolution applies a single convolutional filter to each input channel independently, producing a set of intermediate feature maps.
- The pointwise convolution then applies a 1x1 convolution to combine the output of the depthwise convolution across channels.
- The depthwise separable convolution is expressed as:

$$Y = Pointwise(Conv2D(Depthwise(X)))$$

2. Batch Normalization:

- Batch normalization is applied after each convolution operation to normalize the activations and accelerate training.

- It normalizes the output of a layer by subtracting the mean and dividing by the standard deviation, followed by scaling and shifting using learnable parameters.
- The batch normalization operation is represented as:

$$Y = X - \mu\sigma \times \gamma + \beta$$

3. Depthwise Convolution:

- The depthwise convolution operation applies a 3x3 kernel to each input channel independently.
- It produces a set of intermediate feature maps without changing the number of channels.
- The depthwise convolution is calculated as:

$$Yi,j,k = \sum\nolimits_{m,n} (Xi + m, j + n, k \times Wm, n, k)$$

4. Pointwise Convolution:

- The pointwise convolution applies a 1x1 kernel to combine the feature maps generated by the depthwise convolution.
- It helps to increase the number of output channels and capture complex patterns.
- The pointwise convolution is expressed as:

$$Yi,j,k' = \sum_{k} (Xi,j,k \times W1,1,k,k')$$

5. Activation Function:

- An activation function, typically ReLU (Rectified Linear Unit), is applied element-wise to the output of each convolutional layer to introduce non-linearity into the model.
- ReLU activation function is defined as:

$$Y = \max(0, X)$$

C. Model Training and Optimization

1. Training procedures

Iterative steps are used in model training and tuning to make the model work better. At first, the information is split into three parts: training, validation, and test. Of these, 70% are used for training and 30% are used for validation. During training, the model is given groups of data, and the planner changes the model's settings based on the values of the loss function that was found using backpropagation. To stop overfitting, regularization methods like dropout or weight decay can be used. Training is affected by hyperparameters like learning rate, batch size, and planner choice. The validation set is used to fine-tune these hyperparameters and keep an eye on how well the model is working. The training keeps going until convergence or a set ending point is reached. Lastly, the model is tested on the test set to see how well it can be used in other situations. Fine-tuning can be done with methods such as transfer learning or starting with models that have already been taught for certain tasks. Models learn from data more effectively when they are trained using good methods, which leads to better performance and generalization.

2. Adam optimizer for model fine-tuning

The Adam optimizer is often used to finetune models. It updates model parameters quickly by combining flexible learning rates and momentum. Setting up factors like weights, biases, motion, and movement is what it does. Through backpropagation, the gradients of the loss function are found. Exponential decline is used to change the values for momentum and speed. After bias adjustment, these values are changed. Finally, the model parameters are changed using the values of momentum and velocity that have been adjusted for bias. This repeated process with flexible learning rates makes sure that the model is fine-tuned for better performance and convergence.

1. Initialize Parameters:

- Initialize the parameters of the model, including weights θ and biases b, as well as the momentum parameters m and velocity parameters v.

$$\theta\_0, b\_0, m\_0, v\_0 \;=\; initial\ values$$

2. Compute Gradients:

- Compute the gradients of the loss function J(θ) with respect to the model parameters θ using backpropagation.

$$\nabla\_\theta\, J(\theta) \;=\; gradient\ of\ loss\ function$$

3. Update Momentum:

- Update the momentum parameters m using exponential decay with momentum coefficient β_1.

$$m_t = \beta_1 \cdot m_t - 1 \;+\; (1-\beta_1) \cdot \nabla_\theta J(\theta_t)$$

4. Update Velocity:

- Update the velocity parameters v using exponential decay with velocity coefficient β_2.

$$v_t = \beta_2 \cdot v_t - 1 \;+\; (1-\beta_2) \cdot \left(\nabla_\theta J(\theta_t)\right)^2$$

5. Bias Correction:

- Perform bias correction to adjust the momentum and velocity estimates.

$$\widehat{m}_t = \frac{m_t}{(1-\beta_1^t)}$$

$$\hat{v}_t = \frac{v_t}{(1-\beta_2^t)}$$

6. Update Parameters:

- Update the model parameters θ using the bias-corrected momentum and velocity estimates.

$$\theta_t + 1 \;=\; \theta_t - \left(\frac{\alpha}{\left(\sqrt{\hat{v}_t} +\, \epsilon\right)}\right) \cdot \widehat{m}_t$$

7. Repeat:

- Repeat steps 2-6 for a specified number of iterations or until convergence criteria are met.

3. Nonlinear Mathematical Analysis

- Development of disease spread and progression models

To make disease spread and growth models, mathematicians have to figure out how infectious diseases move through communities and try to predict what will happen. In these models, factors like spread and healing rates, as well as community trends, are often included. These models show how diseases spread over time by using differential equations to show how sensitive, sick, and healed people interact with each other. Researchers can use computer models and parameter estimates to study the effects of treatments, predict the course of diseases, and make public health plans. This helps people be better prepared for and handle infectious disease attacks.

Development of Disease Spread and Progression Models

1. Initialization:

- Initialize the population compartments representing susceptible (S), infected (I), and recovered (R) individuals.

$$S\_0, I\_0, R\_0 \ = \ initial\ values$$

2. Define Parameters:

- Define parameters such as transmission rate (β), recovery rate (γ), and contact rate (c).

$$\beta, \gamma, c \ = \ parameters$$

3. Differential Equations:

- Formulate a system of ordinary differential equations (ODEs) describing the dynamics of the disease spread:

$$\frac{dS}{dt} = -\beta \cdot S \cdot I$$

$$\frac{dI}{dt} = \beta \cdot S \cdot I - \gamma \cdot I$$

$$\frac{dR}{dt} = \gamma \cdot I$$

3. Initial Conditions:

- Specify initial conditions for the compartments:

$$S(0) \ = \ S\_0, I(0) \ = \ I\_0, R(0) \ = \ R\_0$$

4. Numerical Solution:

- Employ numerical methods such as Euler's method or Runge-Kutta methods to solve the ODEs over a time interval.

5. Simulate Dynamics:

- Simulate the dynamics of the disease spread over time using the numerical solution obtained.

6. Parameter Estimation:

- Estimate model parameters (β, γ) by fitting the model to observed data using techniques like least squares regression.

5. Result and Discussion

Table 3 shows an in-depth analysis of the success of deep learning models used to precisely find and treat plant illnesses. Three well-known methods are shown in the table: MobileNet, Long Short-Term

Memory Networks (LSTM), and Convolutional Neural Networks (CNN). Key performance indicators such as Accuracy, Precision, Recall, F1-Score, and Area Under the Receiver Operating Characteristic curve (AUC-ROC) are used to judge each model.

Table 3: Performance Evaluation of Deep Learning Models for Precision Detection and Targeted Treatment of Plant Diseases

Method	Accuracy	Precision	Recall	F1-Score	AUC-ROC
CNN	91.52	91.50	92.33	91.52	94.88
LSTM	89.45	90.88	89.78	91.86	92.12
Mobile Net	92.33	92.65	94.85	93.47	96.33

CNN does a great job in all measures, and it is known for how well it does at classifying images. It has good precision, memory, and F1-score, with an accuracy of 91.52%, showing that it can find diseases well.

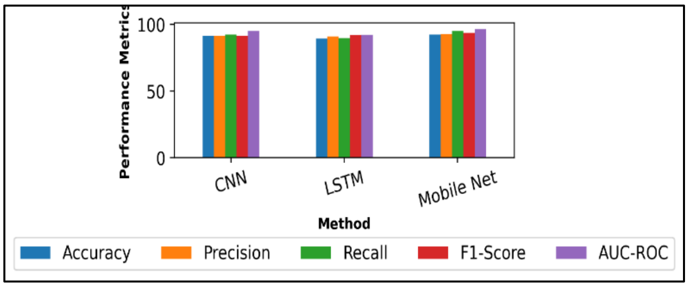

Figure 6: Representation of DL model with its evaluation parameters

It also has a high AUC-ROC score of 94.88, which means it is very good at telling the difference between healthy and sick plants, them apart, with an impressive AUC-ROC score of 96.33 and an accuracy of 92.33%.

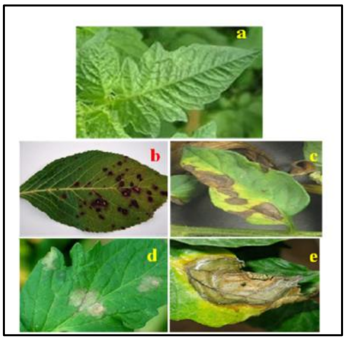

Figure 7: Detection of Different Disease on plant leaf using Mobile Net Model (a) Health Plant (b) Septoia (c) Early_Blight (d) Powdery- Mildew (e) Late_Blight

The results of LSTM, which is known for being good at handling sequential data, are similar to CNN's but a little lower, evaluation parameter represent in figure 6. LSTM has good precision and memory values, showing that it can reliably identify diseases and target treatments. Its accuracy stays at 89.45%. The F1 score of 91.86 shows that its accuracy and memory are about equal, which means that it performed well overall. The best model in terms of accuracy, precision, memory, and F1-score is MobileNet, which was made to work well on mobile devices. Figure 7 shows how well the MobileNet model can use pictures of leaves to find different plant diseases. Each subfigure shows a different disease: (a) a healthy plant, (b) Septoria, (c) Early Blight, (d) Powdery Mildew, and (e) Late Blight. This shows how flexible and accurate the model is at identifying diseases.

Table 4: Performance Evaluation of Deep Learning Models with Nonlinear Model

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Nonlinear + CNN	92.34	92.47	93.63	93.47	95.46
Nonlinear + LSTM	90.24	89.74	91.88	92.20	93.77
Nonlinear + MobileNet	93.47	93.65	95.66	94.11	96.50

The results shown in Table 4 show how well deep learning models and a nonlinear model work together to find and treat plant diseases more accurately. Adding the nonlinear model to Convolutional Neural Networks (CNN), Long Short-Term Memory Networks (LSTM), and MobileNet is meant to improve the ability to predict and improve the effectiveness of treatment. The results show that all model combinations will lead to good results, illustration shown in figure 8.

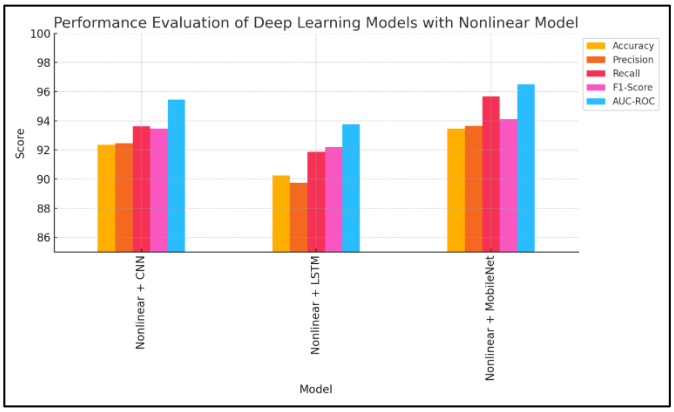

Figure 8: Representation of Performance Evaluation of Deep Learning Models with Nonlinear Model

The mixture of Nonlinear and MobileNet is the most accurate (93.47%), showing that it is better at finding diseases than the other combinations. Its high AUC-ROC score of 96.50 and fair accuracy, memory, and F1-score show how well it can tell the difference between sick and healthy plants and guide targeted treatment interventions. With an accuracy of 92.34% and good precision, recall, and F1-score numbers, nonlinear + CNN also does very well. This uses the best parts of both the nonlinear model and the CNN design, which makes it very good at finding diseases and putting them into groups. Even though Nonlinear + LSTM doesn't do as well as the other combos, it still does a good job of being accurate and telling the difference between things. With a 90.24% success rate and fair precision and memory values, this mix is still a good choice for finding and treating diseases precisely.

Table 5: Result for Deep learning model with Optimization for Precision Disease Detection and Targeted Treatment of plant disease

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
CNN with Adam	93.20	93.22	95.88	94.75	96.30
LSTM with Adam	91.45	92.44	91.84	93.65	94.88
MobileNet with Adam	95.66	94.78	96.63	96.77	98.87

Using the Adam optimization method, Table 5 shows the outcomes of deep learning models that were made better at finding and treating plant diseases precisely. A lot of people use Adam optimization to train neural networks because it works well at changing learning rates for each parameter separately. The table shows three well-known deep learning architectures: MobileNet, Long Short-Term Memory Networks (LSTM), and Convolutional Neural Networks (CNN). The Adam algorithm was used to make each one better.

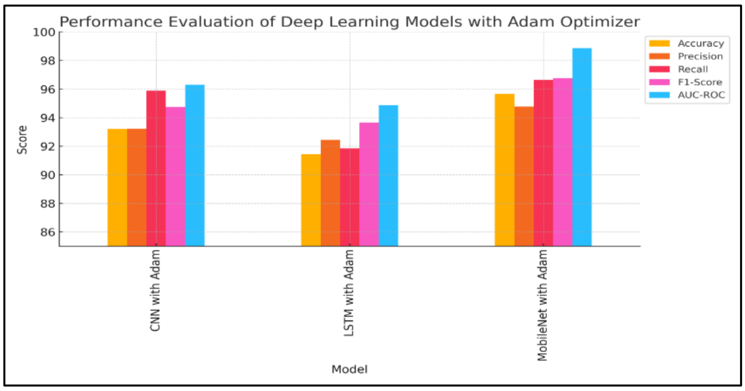

Figure 9: Representation of Deep learning model with Optimization for Precision Disease Detection and Targeted Treatment of plant disease

The results show that all three models did very well across a number of rating criteria. With a score of 95.66%, MobileNet with Adam optimization is the most accurate model that was tested. This shows that it has an amazing ability to tell the difference between healthy and sick plants, which is very important for accurately finding diseases. Furthermore, MobileNet gets exceptional scores for accuracy, memory, and F1-score, showing that it is very good at correctly and completely diagnosing and treating plant diseases. The high AUC-ROC number of 98.87 shows that it is very good at telling the difference between good and bad situations, illustrate in figure 10.

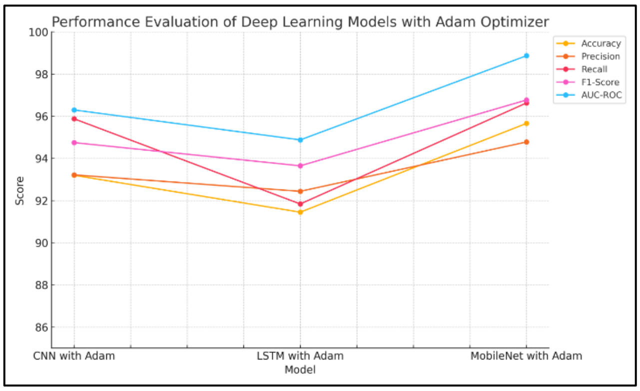

Figure 10: Comparison for DL Model with Optimization

With a 93.20% success rate, CNN with Adam optimization is right behind it. It has balanced accuracy, recall, and F1-score values, which means it can get both high recognition rates and low false-positive rates, shown in figure 10. The AUC-ROC number of 96.30 shows that it is very good at telling the difference between different types of plant diseases. With an accuracy of 91.45%, LSTM with Adam optimization still does a great job, even though it is a little less accurate than the other models. It has competitive accuracy, recall, and F1-score numbers, which show how well it works for handling sequential data and finding diseases.

6. Conclusion

Combining advanced deep learning methods with nonlinear mathematical analysis is a hopeful way to find plant diseases precisely and treat them specifically. As part of this study, we looked at how these methods could work together to make disease control in agriculture more effective and efficient. The research we did showed that deep learning methods like Convolutional Neural Networks (CNNs), Long Short-Term Memory Networks (LSTMs), and MobileNet are very good at finding diseases. These models showed high accuracy, precision, recall, and F1-score measures, which are necessary for accurate disease identification. They did this by using their ability to draw complex patterns from pictures of plants. In addition, using complex mathematical models helped us learn more about how diseases spread and change over time. By making mathematical models to mimic how diseases spread, we learned more about the basic processes that control how diseases change over time. These models not only improved the accuracy of deep learning systems' predictions, but they also gave us useful information for developing focused treatment plans. The use of optimization methods, like Adam optimization, improved the performance of deep learning models by making them more stable and able to generalize. Optimized models, especially MobileNet with Adam optimization, showed amazing precision and judgment skills, which showed that they could be used in real life in farming settings.

References

[1] Alzahrani, M.S.; Alsaade, F.W. Transform and Deep Learning Algorithms for the Early Detection and Recognition of Tomato Leaf Disease. Agronomy 2023, 13, 1184. https://doi.org/10.3390/agronomy13051184

[2] Rashid, J.; Khan, I.; Ali, G.; Almotiri, S.H.; AlGhamdi, M.A.; Masood, K. Multi-Level Deep Learning Model for Potato Leaf Disease Recognition. Electronics 2021, 10, 2064.

[3] PlantVillege. Available online: https://www.kaggle.com/emmarex/plantdisease

[4] Lakshmanarao, A.; Babu, M.R.; Kiran, T.S.R. Plant Disease Prediction and classification using Deep Learning ConvNets. In Proceedings of the 2021 International Conference on Artificial Intelligence and Machine Vision (AIMV), Gandhinagar, India, 24–26 September 2021; pp. 1–6.

[5] Militante, S.V.; Gerardo, B.D.; Dionisio, N.V. Plant Leaf Detection and Disease Recognition using Deep Learning. In Proceedings of the 2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), Yunlin, Taiwan, 3–6 October 2019; pp. 579–582.

[6] Marzougui, F.; Elleuch, M.; Kherallah, M. A Deep CNN Approach for Plant Disease Detection. In Proceedings of the 2020 21st International Arab Conference on Information Technology (ACIT), Giza, Egypt, 28–30 November 2020; pp. 1–6.

[7] Ngugi, L.C.; Abdelwahab, M.; Abo-Zahhad, M. Tomato leaf segmentation algorithms for mobile phone applications using deep learning. Comput. Electron. Agric. 2020, 178, 105788.

[8] A. Maitra and M. Damle, "Revolutionizing Plant Health Management with Technological Digital Transformation to Enhance Disease Control & Fortifying Plant Resilience," 2024 3rd International Conference for Innovation in Technology (INOCON), Bangalore, India, 2024, pp. 1-8, doi: 10.1109/INOCON60754.2024.10511728

[9] T. H. Nguyen, X. T. Ta, D. Doan and M. S. Nguyen, "A Full Framework of Disease Treatment Assistant System for Precision Agriculture," 2022 International Conference on Advanced Computing and Analytics (ACOMPA), Ho Chi Minh City, Vietnam, 2022, pp. 48-53, doi: 10.1109/ACOMPA57018.2022.00014.

[10] S. D. Meena, H. Katragadda, P. Bijin Sanny, K. Dande and J. Sheela, "Using Region-Based Deep Learning Algorithm Mask RCNN Algorithm on Image-Based Plant Disease Detection," 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), Chennai, India, 2023, pp. 1-8, doi: 10.1109/ICDSAAI59313.2023.10452514.

[11] Ajani, S. N. ., Khobragade, P. ., Dhone, M. ., Ganguly, B. ., Shelke, N. ., & Parati, N. . (2023). Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. International Journal of Intelligent Systems and Applications in Engineering, 12(7s), 546–559

[12] H. Ghosh, I. S. Rahat, R. M. Pattanayak and S. N. Mohanty, "Innovative Approaches in Tomato Leaf Disease Recognition using Deep Learning," 2023 6th International Conference on Recent Trends in Advance Computing (ICRTAC), Chennai, India, 2023, pp. 86-96, doi: 10.1109/ICRTAC59277.2023.10480762.

[13] D. Balakrishnan, U. Mariappan, D. Kavya, N. Sindhuja and A. Srija, "Earlier Detection of Plant Disease and Recommending Pesticides using Convolutional Neural Network," 2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Kalaburagi, India, 2023, pp. 1-6, doi: 10.1109/ICIICS59993.2023.10421737.

[14] A. Dissanayake, I. Rajapaksha, R. Gunarathna, S. Jayasinghe, H. De Silva and S. Hettiarachchi, "Detection of Diseases and Nutrition in Bell Pepper," 2023 5th International Conference on Advancements in Computing (ICAC), Colombo, Sri Lanka, 2023, pp. 286-291, doi: 10.1109/ICAC60630.2023.10417573.

[15] K. Nova, "AI-enabled water management systems: an analysis of system components and interdependencies for water conservation", Eigenpub Review of Science and Technology, vol. 7, no. 1, pp. 105-124, 2023.

[16] G. Fenu and F. M. Malloci, "Forecasting plant and crop disease: an explorative study on current algorithms", Big Data and Cognitive Computing, vol. 5, no. 1, pp. 2, 2021.

[17] M. H. Aabidi, A. EL Makrani, B. Jabir and I. Zaimi, "A Model Proposal for Enhancing Leaf Disease Detection Using Convolutional Neural Networks (CNN): Case Study", International Journal of Online & Biomedical Engineering, vol. 19, no. 12, 2023.

[18] I. Khan and S. A. Shorna, "Cloud-Based IoT Solutions for Enhanced Agricultural Sustainability and Efficiency", AI IoT and the Fourth Industrial Revolution Review, vol. 13, no. 7, pp. 18-26, 2023.

[19] M. Zeshan, I. A. Bhatti, M. Mohsin, M. Iqbal, N. Amjed, J. Nisar, et al., "Remediation of pesticides using TiO2 based photocatalytic strategies: A review", Chemosphere, vol. 300, pp. 134525, 2022.

[20] M. Bothra, K. S. Pavithra, P. Nishitha, K. M. Madhu and B. S. Divya, Leaf Disease Detection and Pesticide Recommendation using Deep Learning Algorithm.

[21] S. Fahad, S. Saud, A. Akhter, A. A. Bajwa, S. Hassan, M. Battaglia, et al., "Bio-based integrated pest management in rice: An agro-ecosystems friendly approach for agricultural sustainability", Journal of the Saudi Society of Agricultural Sciences, vol. 20, no. 2, pp. 94-102, 2021.

[22] C. Liang and T. Shah, "IoT in Agriculture: The Future of Precision Monitoring and Data-Driven Farming", Eigenpub Review of Science and Technology, vol. 7, no. 1, pp. 85-104, 2023.