Potato Leaf Disease Detection Using VIT

Ms. Honey Joseph
Dept. of Computer Science
Amal Jyothi College of Engineering
(Autonomous)
Kottayam, India
honeyjoseph@amaljyothi.ac.in

Aaron Samuel Mathew
Dept. of Computer Science
Amal Jyothi College of Engineering
(Autonomous)
Kottayam, India
aaronsamuelmathew2025@cs.ajce.in

Adhil P
Dept. of Computer Science
Amal Jyothi College of Engineering
(Autonomous)
Kottayam, India
adhilp2025@cs.ajce.in

Alan Siby
Dept. of Computer Science
Amal Jyothi College of Engineering
(Autonomous)
Kottyam , India
alansiby2025@cs.ajce.in

Alwyn Joseph
Dept. of Computer Science
Amal Jyothi College of Engineering
(Autonomous)
Kottyam , India
alwynjoseph2025@cs.ajce.in

***Abstract*—**

Potatoes are important for global meals security, but are liable to diseases including fungi, nematodes, viruses, early Blight, and bug damage, which reduces yields and financial losses for farmers. The mission takes benefit of the imaginative and prescient transformer (VIT) to detect accurate and efficient potato disease by using collecting a diverse dataset of potato leaf pics and implementing preprocess and statistics expansion strategies for strong version education. A safe login, an administrator dashboard, and a consumer with a chatbot for real - time conversation -will increase growth, whilst the integrated climate forecast will help farmers to estimate the outbreak of the sickness. Better photograph category abilities of VIT models will be evaluated with non-stop improvement to growth performance, accuracy, recollect and F1-rating. With practical treatment and practical gadget consisting of climate forecasts, this system offers rights to support global food safety, reduce pesticides and promote productivity.

***Index Terms*—Vision Transfomers (VIT),Image Classification, Convolutional Neural Networks (CNN) , Artificial Intelligence (AI) , Deep Learning , Machine Learning, Agricultural Technology .**

I. Introduction

A. Background:

Potatoes play a key role in world food security, but are extremely susceptible to diseases such as early plague, late pests, fungal disease, nematodes, viruses and damage to pests, resulting in tremendous performance losses and economic losses for these farmers. The conventional diagnosis of diseases depends on the visual examination, which consumes time and usually inaccurate. To achieve this, the project employs Vision's Transformers (VIT), ie the Google/Vit-Base-Patch16-224-In21k model, known for its best image classification features. By gathering a diverse variety of potato leaves images, including sick and healthy specimens and employing advanced pre-processing and increase data, the model seeks to detect and properly classify different potatoes of potatoes. This strategy improves early diagnosis and efficient disease control, allowing farmers to minimize crop losses, reduce pesticide use and promote sustainable agriculture, leading to global food safety.

B. Problem Statement:

Potato crops (stand of potato crops) are very susceptible to diseases (early pests, late pests, fungal diseases, nematode, virus, pest injuries) which causes considerable income loss. The current disease detection methods, which are manual inspection-based and time consuming with intensive work and relatively low accuracy, lead to difficulties especially in crop-based large-scale agriculture. It is essential to identify these diseases as early and as accurately as possible in order to maximize control and reduce yield loss caused by harvest injury. The goal of this project is to address this challenge via Vision Transformers (VIT), in particular the Google/Vit-Base-Patch16-224-In21k model in order to create an effective and precise potato disease detection system. By training in a diversified set of potato sheets images that represent multiple diseases and healthy samples and, using advanced pre - processing and data increase, the model seeks to improve early diagnosis of diseases, reduce the use of unnecessary pesticides and Promote sustainable agriculture, increase productivity and productivity support and global support for food security.global food security.

C. Objectives:

The purpose of this project centers on creating a system that uses artificial intelligence to find and sort diseases affecting potato leaves. It includes detection of early blight, late blight, fungi, nematodes, viruses, pest harm as well as normal leaf conditions.

1) Build an Accurate Detection Model: Through a combination of the google/vit-base-patch16-224-in21k vision transformer and extensive potato leaf images, this project aims to create a detection model that identifies diseases. The system

sorts each condition into categories allowing farmers to spot problems at their start. Such early detection helps prevent crop damage through immediate action.

2) Enable Early Detection: This objective aims to enable early detection of potato leaf diseases, allowing farmers to take timely action and minimize crop damage, ultimately improving yield and reducing financial losses.

3) Integrate with Farming Technology: The system brings disease detection into current farming methods through an interface anyone can navigate with ease. A digital assistant responds to farmers in real time helping them find plant diseases and suggesting specific treatments for their crops. These additions make the complete package straightforward to use and turn complex data into clear steps for better plant care.

4) Ease of use: The system puts users in control through a simple interface that includes protected entry points and clear navigation screens. By keeping every feature in familiar spots, farmers enter plant photos, get disease results along with find matching solutions without complications. Such straightforward design lets people across different experience levels manage their farming tasks through the platform without needing advanced computer understanding.

II. Related Works

Recent studies of leaf disease detection demonstrate the advantages of Vision Transformer (ViT) systems over standard Convolutional Neural Networks (CNNs) in image sorting tasks. Through an examination of Java Plum leaf diseases, researchers applied a modified ViT system that adjusted core elements like image dimensions, patch measurements along with attention points to improve sorting precision. The results indicated a 97.51 % precision rate, which validates ViT's effectiveness in finding plant diseases. But the research identified drawbacks regarding ViT's heavy processing requirements and the necessity for specific adjustments to get the best results.[1]

The architecture known as EfficientRMT-Net combines ResNet-50 with ViT networks to detect diseases in potato leaves. This system takes advantage of ResNet-50's detail extraction alongside ViT's pattern recognition in images. By incorporating depth wise convolution, the model reduces its computation requirements. Through its stage block design, the network adapts to diverse plant image collections. For potato leaf disease classification, EfficientRMT-Net achieved 99.12 % accuracy exceeding previous detection methods. The results demonstrate that integrating ViT and CNN architectures creates more precise disease detection. But the combination serves practical purposes in farming operations that need exact plant condition analysis.[2]

Developments in Agricultural Technology: Classification of Diseases in Potato Leaves Using Vision Transformer Technology Smita Adhikari.[3]) Another researcher focused on a classification problem regarding potato leaf disease using ViT B 16 architecture. The model was enhanced on a more comprehensive dataset available in Kaggle, containing 2152 images divided into three classes: early blight, late blight, and healthy leaves. With independent training, testing, and validation splits, the model's accuracy reached an impressive 99.55%. This study pinpointed the amazing ability of ViT to deliver accurate and reliable disease detection insights, thereby augmenting chances of an accurate early diagnosis and improving agricultural plans. It also focused on the need for achieving model performance and generalization across many situations by fine-tuning ViT with custom datasets13.

III. Methodology

A. Data Collection and Preprocessing

The initial step in creating a robust potato leaf disease detection model is the collection of a good dataset comprising images of diseased and healthy potato leaves. The dataset, as a matter of fact, should be comprehensive enough to encompass multiple environmental conditions, light conditions, and angles so that the model is ready for actual deployment. Publicly available datasets like Proposed dataset are good points to begin with, but other images can be captured by using mobile phones, drones, or farm-sensing sensors. In image acquisition, the images are annotated appropriately, classifying them as healthy, early blight, late blight, Nematode, Fungi, pests, virus. In case a small dataset is given, methods of crowdsourcing or expert annotating can be employed to acquire correct labels. The dataset must also be balanced so that each class has a sufficient number of examples in order to avoid the model being biased toward one category.

Once the dataset has been collected, it must be preprocessed before its input into the Vision Transformer (ViT) model. This is because raw images contain noise, varying resolutions, and inconsistencies that could affect the performance of the model. Image augmentation is the initial preprocessing process, and it artificially increases the dataset size using operations such as rotation, flipping, scaling, brightness adjustment, and addition of Gaussian noise. The operation makes the model generalize more effectively by forcing the model to learn multiple instances of the same leaf. Mathematically, augmentation can be represented as applying a transformation T to the original image I, such that the augmented image I' is given by:

$$I' = T(I) = \text{Flip}(\text{Rotate}(I, \theta)) + N \tag{1}$$

where θ is the rotation angle, and $N \sim \mathcal{N}(0, \sigma^2)$ represents Gaussian noise with mean 0 and variance σ^2. The next step is image resizing, as ViT requires a fixed input size. Typically, images are resized to 224 × 224 pixels to match the model's architecture. Since ViT processes images as sequences of patches, each image is divided into non-overlapping square patches (e.g., 1616 pixels). The number of patches per image, N, is calculated as:

$$N = \frac{H \times W}{P^2} \tag{2}$$

where H and W are the image height and width, and P is the patch size. Each patch is then flattened and projected into a high-dimensional space using a learnable weight matrix:

$$X_p = \text{Flatten}(P) \cdot W_E \tag{3}$$

where W_E is a trainable embedding matrix. After resizing and patch embedding, image normalization is applied to standardize pixel values. This involves subtracting the mean μ and dividing by the standard deviation σ to ensure zero mean and unit variance:

$$I_{\text{norm}} = \frac{I - \mu}{\sigma} \tag{4}$$

Normalization aids in faster convergence of the model during training as well as avoids issues related to varying brightness levels and contrast settings of images. Transformation of color space can also be done if images are not RGB-stored as ViT models are trained against pre-trained RGB images. Once preprocessed, the dataset is divided into test, validation, and training sets in the proportion 80:10:10 for efficient evaluation of the model. The whole preprocessing pipeline guarantees that the images are in the correct format for correct learning and best performance upon input to the ViT model. The proposed dataset consists of 3070 images which is then further augmented[6]

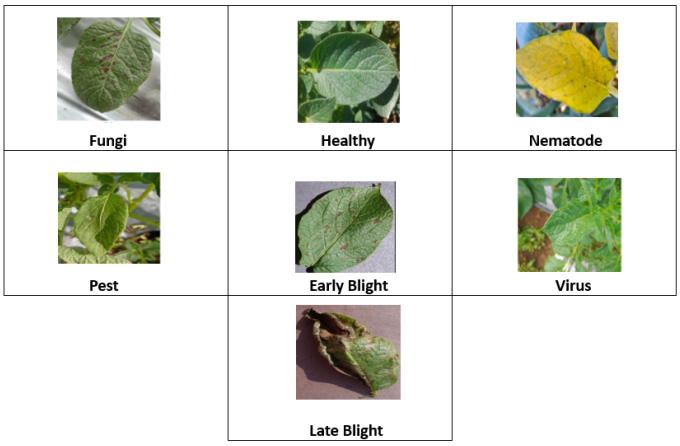

Fig. 1: Proposed Dataset

B. Architecture

Vision Transformer (ViT) architecture breaks through image recognition through the use of the Transformer model, developed for NLP, as is, on image patch sequences. The method does away with the convolutional neural networks (CNNs) since it processes image patches as tokens, similar to the case of words in NLP, enabling efficient learning of global dependencies.

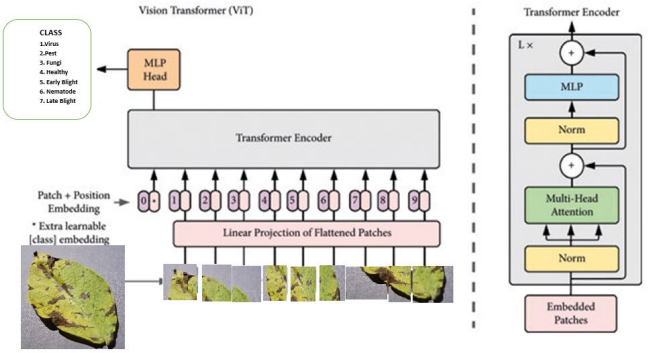

Fig. 2: Model Overview:Image patches are embedded, positionally encoded, and processed by a Transformer for classification.

The input to the Vision Transformer is a 2D image denoted as x RH×W×C, where H and W represent the height and width of the image, and C is the number of channels (e.g., 3 for RGB images). The image is divided into fixed-size patches of resolution P×P, resulting in a sequence of patches[5]. The number of patches is calculated as:

$N = HW/P^2$

Each patch is flattened into a vector and linearly projected to a latent dimension D by a learnable projection matrix E. The patch embeddings are obtained as follows:

z = [xclass; xE; xE; ...; xNE] + Epos An additional learnable ”classification token” xclass is added to the beginning of the sequence, whose state at the output of the Transformer encoder is used as the image representation for classification.

The Vision Transformer input is a 2D image, where and are the height and width of the image, and is the number of channels (e.g., 3 for RGB images). The image is split into fixed-size patches of resolution. As a result, the number of patches is:

Each patch is flattened into a vector of size and then projected linearly onto a latent space with a learnable projection matrix :

The output of this projection forms the patch embeddings: z0 = [xclass; x1 E; x2 E; ...; xN E] + Epos

In order to maintain spatial information, learnable positional embeddings (Epos) are added to the patch embeddings. The positional embeddings provide information about the position of each patch in the input image such that the model can maintain spatial relationships between patches. This is necessary because the Transformer architecture is permutation-invariant by nature and does not possess the inductive biases present in CNNs, such as locality and translation equivariance. As a result, ViT can effectively capture long-range dependencies from the whole image. Positionally encoded patch embeddings are input into the Transformer encoder.

In order to maintain spatial information, learnable positional embeddings are added to the patch embeddings. This provides the input sequence

In this instance, a learnable "classification token" is prefixed to the sequence, just like the [CLS] token in BERT, whose output state is utilized as the image representation.

The Transformer encoder consists of layers, with Multi head Self Attention (MSA) and Multi Layer Perceptron (MLP). Before each block in the layer there is Layer Normalization (LN). After each block residual connections are included to help with gradient flow and maintain training stability. In the self attention process of the mechanism involves calculating attention weights based on similarities, between query (Q) and key (K).

Attention(Q, K, V) = Softmax(QK / D) V

Multi-Head Self-Attention (MSA) is calculated by applying several self-attention steps in parallel and combining their results through concatenation[7]. Each attention head's output is linearly transformed with WO. The residual connected MSA is defined as

z' = MSA(LN(z)) + z, for = 1, ..., L

The MLP block, which follows each MSA, consists of two fully connected layers with a GELU activation in between. The output of the MLP block with residual connection is calculated as:

z = MLP(LN(z')) + z', for = 1, ..., L

The Transformer encoder, as motivated by Vaswani et al. (2017), is composed of layers that consist of: Multi-Head Self-Attention (MSA) mechanism Multi-Layer Perceptron (MLP) block Layer Normalization (LN) and Residual Connections

The encoder performs LayerNorm after each block and residual connections following each block. The self-attention mechanism with residual connection is given as:

z'l = MSA(LN(z(l-1))) + z(l-1), l = 1, ..., L

where queries , keys , and values are obtained by linearly projecting the input sequence.

Multi-Head Self-Attention (MSA) is computed by running multiple self-attention operations in parallel and concatenating their outputs:

The output of the MLP block with residual connection is calculated as:

zl = MLP(LN(z'l)) + z'l, l = 1, ..., L

For classification, an MLP head is added to the classification token's final state. For pre-training, the MLP head has a single hidden layer, but for fine-tuning, it is replaced with a single linear layer outputting the class probabilities. The final layer normalization is performed on the classification token as follows:

y = LN(z)

This generalized output is then fed to obtain classification logits in the MLP head, which is used to calculate the cross-entropy loss in training time. This structure facilitates good convenience representation and accurate classification.

For classification functions, the final layer is done for generalization classification tokens: as follows: Y = LN (Z0l) pre-training involves MLP head with a hidden layer, while fine tuning replaces it with a single linear layer for output.

Unlike CNNS, the vision transformer clearly does not encoded the localism or translation equorians and instead depends on the self-revolution mechanism to learn spatial relations with data. This allows the model to learn long distance dependence and complex global patterns. But this also means that VITs require more data and prolonged training time than CNN, as it should learn these individual prejudices. Nevertheless, the VIT shows better performance on image classification functions by taking advantage of its global meditation system, which is very effectively able to interact between far-flung patches in the image. This optimity and ability to model global references distinguish it from traditional firm designs.

Unlike CNNS, VIT does not naturally involve localism or translation equivalent and depends on the self-eclipse mechanism to achieve spatial relationships. This global attention allows VITs to perform better than traditional CNN on large dataset by learning high-level dependence directly from input data.

C. Training

This generalized output is then fed to obtain classification logits in the MLP head, which is used to calculate the cross-entropy loss on time training. This architecture allows for an accurate classification along with efficient facility representation.

In classification functions, the final layer generalization is done on the following classification tokens: Y = LN (Z0l) MLP is a hidden layer during pre-training and is replaced by a single linear layer in the output during fine-tuning.

Unlike CNNS, the vision transformer does not have the same as the terrain or translation, but instead uses the self-objective mechanism to learn spatial correspondence from data. This allows the model to be able to occupy long distance dependence and complex global patterns. But this also means that VITs require more data and prolonged training time than CNN because it has to learn these inductive prejudices from scratches. However, the VIT shows better performance on image recognition functions by taking advantage of its global meditation system, which enables it to interact in the image between a distant-up patches. This flexibility and ability to model global references distinguish it from standard firm structures.

Unlike CNNS, VIT mainly does not encoded localism or translation of Equareanus and depends on the self-eclipse mechanism to achieve spatial relationships instead. This global attention allows to cross the standard CNN on a large scale by learning complex dependence from data directly from data.

$$L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{5}$$

where y_i is the true label, $\hat{y}_i$ is the predicted probability for class i, and C is the total number of classes. The loss function guides the optimization process, allowing the model to adjust its parameters to improve classification performance.

For the optimal parameter tuning of the model, we use adamw optimizer, to avoid overfiting with a increased version

of adam with a increased version of adam. The learning rate is an important hyperpometer that controls how much the model adjusts its weight in each repetition. Instead of having a certain learning rate, we employ cosine decay scheduling, which reduces the learning rate over time to obtain stable convergence. The learning rate at step t is given by:

$$\eta_t = \eta_{\text{min}} + \frac{1}{2}(\eta_{\text{max}} - \eta_{\text{min}}) \left(1 + \cos\left(\frac{t}{T}\pi\right)\right) \quad (6)$$

where η_{max} is the initial learning rate, η_{min} is the minimum learning rate, t is the current training step, and T is the total number of training steps. This strategy ensures faster learning in the early stages while refining the model in later epochs.

To avoid overfiting, we use dropouts regularization, where some neurons are randomly disabled in training, and it reduces dependence on certain characteristics. Data growth, batch normalization and early restrictions are also used to improve generalization. The Google VIT-BASE-Patch16-224 is trained for 25 ages depending on the dataset size with a batch size of 7, providing a trade-closure between computational resources and model performance.

When we train, we track important matrix such as training loss, verification loss and accuracy. Training loss must be reduced over time and verification loss needs to be coordinated, which indicates that the model is learning well without overfiting. Whenever verification loss increases, while the training loss is reduced, it will indicate overfiting, so countercers can be taken as a decrease in learning rate or increase in dropouts.

After training, the model with maximum verification accuracy is saved for testing and signs. The Google VIT-BASE-Patch16-224 model trained is now ready to use potato leaf diseases in real-world applications, where it can be included in the flask-based web API or mobile app for easy use.

D. Evaluvation and Metrics

Once trained with Google VIT-BASE-Patch16-224, it is required to test its performance to verify that it is sufficiently normalizing the ignored data. The testing process meets the use of several matrix, to determine that the model is capable of distinguishing between different classes, including healthy leaves, early blits, and late blights. These measurements aid in diagnosis of potential problems such as overfiting, class imbalance and miscalitation trends. The major evaluation matrix used in this project is accuracy, accurate, recall, F1-score and confusion matrix, which is well explained below.

1. Accuracy

Accuracy classification is one of the most popularly used matrix in tasks. This measures how pure the models are pure by calculating the correct ratio of the correct classified samples from the total samples. Mathematically, defined as accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

where: - TP (True Positive): Correctly predicted diseased leaf. - TN (True Negative): Correctly predicted healthy leaf. - FP (False Positive): Incorrectly predicted diseased leaf (when it's actually healthy). - FN (False Negative): Incorrectly predicted healthy leaf (when it's actually diseased).

A higher accuracy score (closer to 1 or 100%) indicates that the model correctly classifies most test samples. However, accuracy alone is not sufficient, especially if the dataset is imbalanced (e.g., more healthy leaves than diseased ones), as it may lead to misleading results.

2. Precision

Right, or positive prediction value (PPV), it suggests that positive examples are really correct. It is important in medicine and agricultural use, where unnecessary pesticides or incorrect choices may be as a result of false positivity (malnourishing a healthy leaf as a diseased). It is calculated as:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

A high precision score means that the model makes fewer false positive errors, ensuring that healthy leaves are not unnecessarily classified as diseased.

3. Recall (Sensitivity or True Positive Rate)

Remember estimates the ability of the model to identify real positive examples (diseased leaves). This informs us how many true diseased models leave correctly. This is particularly important in the identification of the disease, where failure to identify diseased cases (false negatives) can be harmful to crop health. Recall formula is:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

A high recall score means that the model effectively identifies diseased leaves, minimizing the risk of undetected infections spreading in a crop field

4. F1-Score

The F1-score means accurate and recall harmonic, which gives a well-balanced value in terms of unbalanced class distribution. This is particularly helpful when the data set is unbalanced (ie, when the number of healthy and sick leaf samples varies). F1- Score is the formula:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

F1- The score is between 0 and 1, where 1 indicates the correct memory and accuracy. The high F1-score implies that the model has a good ability to identify diseased leaves with minimum false positivity and false negative.

IV. Work Flow

The project initiates upon developing a model and preparing a dataset. To save on memory, the hugging face script loads the dataset in streaming mode using the Hugging Face 'Datasets' library. The VIT image processor is used to shape and normalize dataset and GOOGLE-VIT-BASE-PATCH16-224 model is pre-trained with certain number of classes of plant diseases. For training, the dataset is split into training, verification, and testing sets. These are converted into tensorflow and batched datasets. The model is built with the Adam optimizer with sparse ranked cross entropy and memory s aving features. The model checkpointing returns, restarts, learning rate decay and warm up are set for 25 training steps.

Post training, the model is being tested in the test database to measure the accuracy and loss. Predictions arrive in test images and metrics are evaluated using confusion matrix and classification report. The results, for example, claiming accuracy while training/verification, loss, and confusion matrix, are submitted as images. Furthermore, forecast function is applied to classify new images with trust score. For future estimation or deployment purposes, the final model weights is saved along with a classification report summarizing the model performance in a text file.

V. Results

Confusion Matrix displays great classification performance on all seven categories (bacteria, fungi, healthy, nematodes, insects, phetofthora and viruses). The diagonal elements of the matrix provide great correct classification, the strongest of detecting the insect with 70 correct classification. There were 56 correct classifications in both bacteria and healthy plants, 57 correct classification examples in each with fungi and nematodes. The model worked properly for phytopathora (35 correct) and virus (53 correct) categories. The only topical false positivity/negativity with misconducts showed that the model learned characteristics for each type of disease.

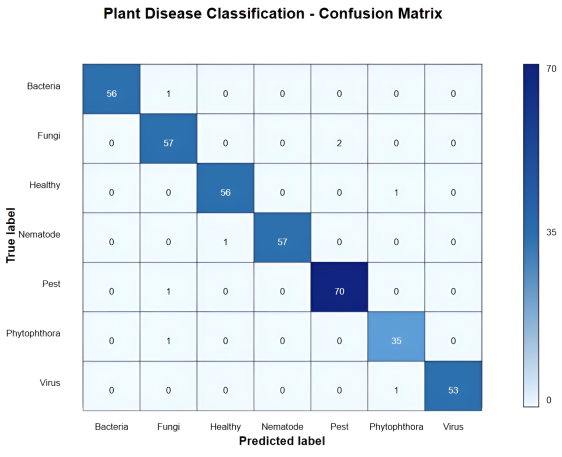

Fig. 3: Confusion Matrix

Training and verification losses decrease reveal the process of model learning in five ages. Training loss (blue line) shows a steady decline from about 1.2 to 0.25, reflecting successful learning from training data. Verification is a distinct trend of loss (orange line), which increases from 2.0 to 2.0 to about 2.4, then gradually increases to approximately 2.3 on the epocal 5. This division in training and verification loss indicates some overfiting, although overall model performance is still strong.

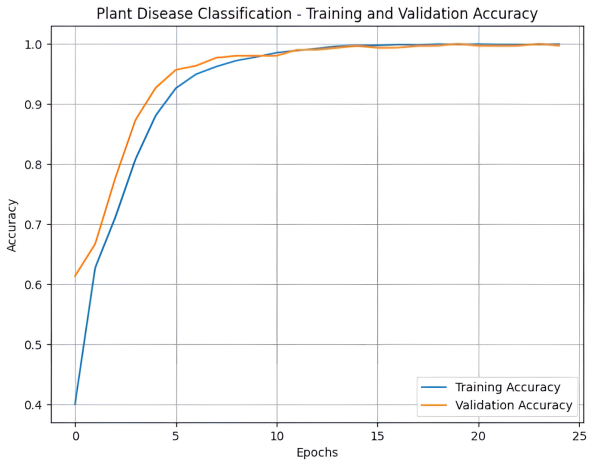

Fig. 4: Accuracy Graph

The accuracy declining training verbs the model's very good performance on data and highlights potential generalization issues. Training accuracy (blue line) increases rapidly and is about 95% for all ages, showing excellent learning ability. But verification accuracy remains up to 20% continuously, showing a major difference between training and verification accuracy. This incompatible indicates that although model is very good in handling the known pattern of data, it struggles with generalization for new examples, which needs to be improved in the most.

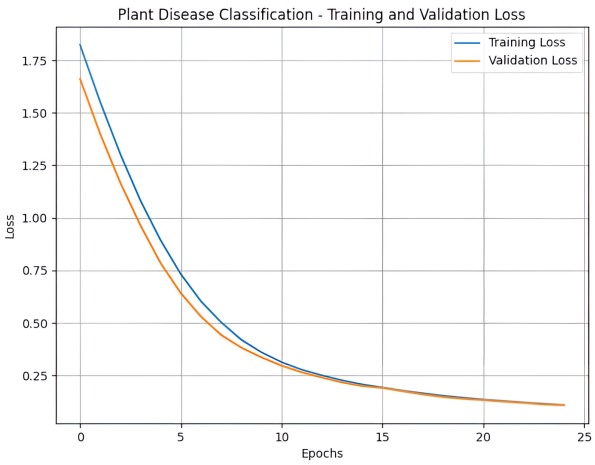

Fig. 5: Loss Graph

VI. Discussions and Analysis

My experiments show how the status of the dataset is strongly affected by the model performance. Trained and tested on the plant village dataset, RESNET50 performed with high accuracy of 95.17 %. This dataset is well made of light, high-quality and equally structured images, so it was easy for the model to identify the pattern and differentiate between classrooms. But when the Resnet50 was tested on an uncontrolled dataset, its accuracy rapidly fell to 68.12%. This deficiency is probably caused by real -world variability such as light difference, background noise, image orientation and plant states, which the model did not see during training.

In contrast, the Vision Transformer (VIT) model (Google/VIT-BASE-PATCH16-224-In21K) performed much better in relation to generalization, as it achieved 94.39% accurate on uncontrolled dataset. VITs are naturally able to learn global image dependence more effectively, so they are also more flexible to changes in environmental status. This means that VIT Baseline performs better with various datasets compared to RESNET50 like CNN. But VITs demand more calculations and require more data to train and get their optimal. Although the resnet50 is effective and good for a regular environment, in the real -world environment where the variations of images are fixed, VIT is the optimal option.

Vision Transformer (ViT) is better for detecting potato leaf disease because it represents long-range dependencies through self-attention, unlike CNNs that rely on local filters. This allows ViT to focus on disease patches more easily even when symptoms are weak or scattered. ViT handles occlusions, changes in lighting, and high-resolution images much more effectively than CNNs and hence reports better and more accurate disease classification.

Model	Test Accuracy	Dataset
ResNet50	95.17%	Plant Village Dataset
ResNet50	68.12%	Uncontrolled Dataset
Google/vit-base-patch16-224-in21k	94.39%	Uncontrolled Dataset

TABLE I: Comparison Table

VII. Future Work

In the coming development of the system, model performance should be enhanced, features expanded, and farm-level usability improved. Identification of key areas for further development will enhance the effective applicability of our approach.

- **Enhancing Model Accuracy:** Enhanced classification performance of the vision transformer model using large, domain-specific pre-training datasets and advanced training techniques.
- **Real-Time Detection:** Mobile or web-based applications for real-time detection of diseases using smartphones and IOT-enabled cameras.
- **Integration with IoT and Smart Farming:** Integration with IOT sensors for environmental monitoring, soil health analysis, and automatic treatment recommendations.
- **Multi-Disease Classification:** Extension of models that can detect multiple crop diseases in varying species of the same plant.
- **Predictive Analytics and Early Warning System:** Historical data, weather conditions, and geographic coordinate tracking should provide input parameters for applying future models in the early detection of outbreaks.
- **Explainability and Interpretability:** AI-managed recommended treatment would take into account variable conditions based on an index of disease severity and environmental conditions.
- **Automated Treatment Suggestions:** Integrating AI to consider disease severity and environmental factors, providing farmers with valuable insights for farming.

Future improvements would include decisional support systems to assist farmers with managing the disease of crops proactively and sustainably.

VIII. Conclusion

Potato leaf disease detection using google/vit-base-patch16-224-in21k has shown promise as it can classify early blight, late blight, healthy and other diseases , all due to its sophisticated self-attention mechanism and global feature extraction capabilities. The Vision Transformer is more accurate and is able to generalize better than traditional CNNs because it captures complex patterns and long-range dependencies within leaf images. While the pruning and lightweight variants do help, the model does pose difficulties due to its lack of interpretability and high computational cost. In summary, this technique is effective for early onset disease detection, which ultimately helps with crop productivity and food supply security.

References

[1] Bhowmik, A. C., Ahad, D. M. T., Ahad, M. T., Emon, Y. R., Ahmed, F., Song, B., Li, Y. (2024). A customised Vision Transformer for accurate detection and classification of Java Plum leaf disease. *Smart Agricultural Technolog y

[2] EfficientRMT-Net—An Efficient ResNet-50 and Vision Transformers Approach for Classifying Potato Plant Leaf Diseases Kashif Shaheed 1, Imran Qureshi 2,* , Fakhar Abbas 3, Sohail Jabbar 2 and MuhammadZaheerSajid

[3] Advancements in Agricultural Technology: Vision Transformer-Based Potato Leaf Disease Classification Smita Adhikari.

[4] Mitali V. Shewale , Rohin D. Daruwala, ”High performance deep learning architecture for early detection and classification of plant leaf disease“.

[5] Sharma, Jatin ”Enhanced Rose Leaf Disease Classification Using Vision Transformer (ViT-B/16) Detecting Black Spot, Downy Mildew, and Healthy Leaves for Improved Plant Health Management”

[6] Nabila Husna Shabrina ,Siwi Indarti ,Rina Maharani , Dinar Ajeng Kristiyanti , Irmawati ,Niki Prastomo ,Tika Adilah M ”A novel dataset of potato leaf disease in uncontrolled environment”

[7] Wasi Ullah,Kashif Javed,Muhammad Attique Khan,Faisal Yousef Alghayadh, Mohammed Wasim Bhatt,Imad Saud Al Naimi, Isaac Ofori ”Efcient identifcation and classifcation of apple leaf diseases using lightweight vision transformer (ViT)”