EMPOWERING AGRICULTURE: RESNET 101 DRIVEN DIAGNOSIS FOR ENHANCED POTATO LEAF DISEASES DETECTION

Mr. T.Rajan Babu[1], S.Bharathapriyan[2], C.Savin Sangeerthan[3]and G. Parthasarathi[4]

[1] Asst. Professor, Rajiv Gandhi College of Engineering and Technology, Puducherry, India
[2,3,4,5]B. Tech (CSE Students), Rajiv Gandhi College of Engineering and Technology, Puducherry.

ABSTRACT:

Potato cultivation faces significant threats from various diseases, leading to substantial yield losses worldwide. Timely detection and effective prevention of these diseases are crucial for ensuring food security and sustainable agricultural practices. In this study, a novel approach is proposed for the automatic detection and prevention of potato plant diseases using a Convolutional Neural Network (CNN), specifically the ResNet-101 architecture. By analyzing leaf images, the ResNet-101 accurately identifies common diseases like late blight, early blight, and potato virus Y (PVY). Training on a large dataset with data augmentation enhances the model's generalization and robustness. Integrating the disease detection system with a preventive management framework enables real-time monitoring and early intervention. Extensive experiments validate the effectiveness of this approach, showing superior performance compared to traditional methods. These findings highlight the potential of CNN-based models in transforming potato plant disease management, contributing to sustainable agriculture and increased crop productivity.

Keywords: *CNN, Deep learning, Convolutional layer and Potato disease*

INTRODUCTION:

Potato diseases are a significant concern for farmers worldwide, often leading to considerable crop losses if not managed effectively. One of the most common and destructive diseases affecting potatoes is late blight, caused by the fungus Phytophthora infestans. Late blight thrives in cool, moist conditions, making it particularly problematic in regions with temperate climates. The disease manifests as dark, water-soaked lesions on leaves, stems, and tubers, eventually leading to rot and complete plant destruction if left unchecked. Late blight can spread rapidly through a potato field, facilitated by windborne spores, and is notoriously difficult to control once established (1-3).

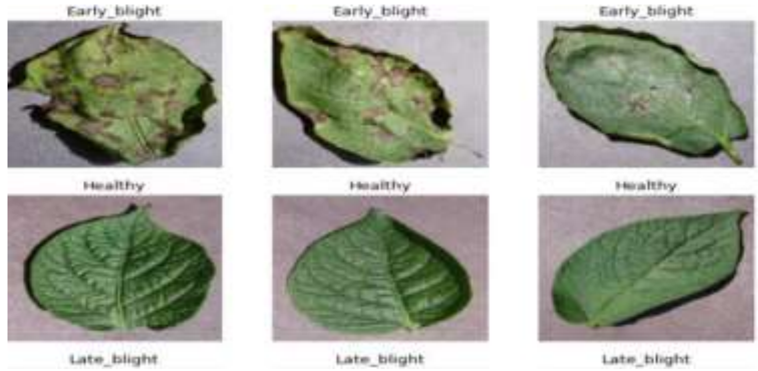

Figure 1. Potato leaf disease

Another notable potato disease is early blight, caused by the fungus Alternaria solani. Early blight typically appears as dark, concentric rings on potato leaves, eventually causing them to wither and die. Unlike late blight, which thrives in cooler conditions, early blight prefers warm, humid environments. Crop rotation, proper sanitation, and fungicide applications are commonly used strategies to manage early blight, but preventative measures are crucial as the fungus can survive in soil and plant debris for extended periods.

Additionally, bacterial wilt, caused by the bacterium Ralstonia solanacearum, poses a significant threat to potato crops. This disease affects the vascular system of the plant, causing wilting, yellowing of leaves, and eventual death. Bacterial wilt is challenging to control once established, as it can persist in soil for years. Implementing strict sanitation practices, using disease-resistant potato varieties, and employing cultural practices such as proper irrigation management are essential for managing bacterial wilt and preserving potato yields. Overall, a combination of preventative measures, early detection, and targeted management strategies is crucial for mitigating the impact of potato diseases on agricultural productivity (4-5).

MATERIALS AND METHODS

In the proposed system, the adoption of the ResNet101 algorithm for disease prediction in potato crops signifies a deliberate pursuit of accuracy and efficiency. Leveraging ResNet101's capabilities in real-time object detection, the system can swiftly and accurately identify diseases affecting potato crops, providing farmers with timely insights into the health of their plants. Beyond mere disease identification, the system goes a step further by offering actionable remedies tailored to the specific diseases detected. This holistic approach not only diagnoses issues but also empowers farmers with practical guidance on how to effectively address and mitigate the identified concerns. By equipping farmers with actionable solutions, the proposed system plays a pivotal role in enhancing agricultural productivity, reducing crop losses, and fostering the economic well-being of farmers (6).

Furthermore, the comprehensive and innovative integration of ResNet101 aligns closely with broader objectives of promoting global food security and advocating for sustainable agricultural practices. By harnessing the power of technology to address pressing challenges within the agricultural sector, the proposed system underscores the critical role of advancements in artificial intelligence in driving transformative changes. Through its robust disease prediction capabilities and provision of actionable solutions, the system not only supports farmers in optimizing crop yield and quality but also contributes to building resilience in food supply chains. In essence, the proposed system represents a forward-looking approach towards leveraging technology to address key challenges in agriculture and pave the way for a more sustainable and prosperous future.

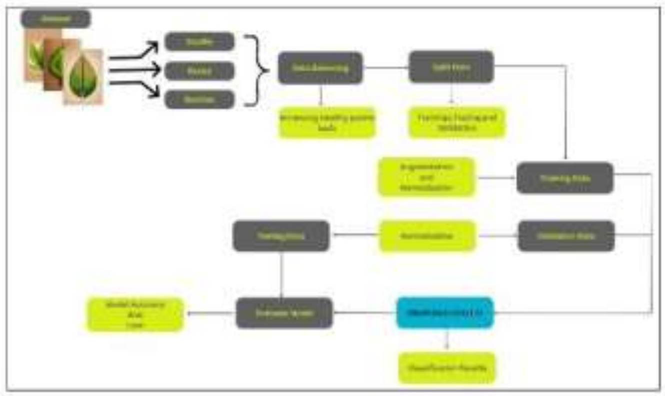

Figure 2. Proposed system architecture

Data collection: In proposed work, there are 4000 images in training dataset, 1000 images in validation dataset and 500 images in testing dataset. Out of 4000 training images, 1000 images belong to healthy category and 1000 images belong to each potato disease category described above. In validation set each class has 700 images and test set has 50 images in each class. For testing, we randomly picked 50 images from each class from training set and removed them from those folders. From remaining training dataset, we built our project training dataset by putting same number of images (1000) in each class. When the images in any class were less than 1000, we used data augmentation technique to generate some new images. Augmentation was done using Augmentor package of python and it helps to build similar new images by rotating, flipping, cropping and resizing the existing images. When images in any class in training dataset were more than 1000, we picked first 1000 images. We followed same process for validation dataset and made all classes have 700 images each. This process is necessary to prevent bias for any particular class during training of CNN. Size of all the images is 256 × 256 and format is jpeg.

Figure 3. Data collection of potato leaf disease

Pre-processing: Pre-processing is a fundamental step in data analysis and machine learning tasks, including the detection and classification of potato leaf diseases. It involves a series of operations applied to raw data to prepare it for further analysis by enhancing its quality, removing noise, and extracting relevant features. In the context of potato leaf disease detection, pre-processing typically begins with image acquisition, followed by operations such as resizing, normalization, and noise reduction. Resizing ensures uniformity in image dimensions, while normalization adjusts pixel values to a common scale, facilitating consistent analysis across images. Noise reduction techniques, such as Gaussian blurring or median filtering, help remove irrelevant details and artifacts from the images, improving their clarity and reducing the likelihood of misclassification. Overall, pre-processing plays a crucial role in optimizing data quality and preparing it for accurate and efficient analysis by subsequent machine learning algorithms.

Feature extraction: Feature extraction is a pivotal process within deep learning models like ResNet-101, particularly in tasks such as potato leaf disease detection. ResNet-101, a convolutional neural network architecture, is renowned for its depth and ability to extract intricate patterns from images. In feature extraction, the network undergoes a series of convolutional and pooling layers, which progressively extract abstract features from input images. In the case of potato leaf disease detection, ResNet-101 is trained on a dataset containing images of healthy and diseased potato leaves. During training, the network learns to identify distinctive characteristics indicative of different disease types. These characteristics may include color variations, texture irregularities, and structural abnormalities present in the leaves. By the end of the feature extraction process, ResNet-101 transforms the input images into high-dimensional feature vectors, encoding the learned representations of the input data. These feature vectors serve as inputs to subsequent classification layers, enabling the model to make accurate predictions regarding the presence and type of potato leaf diseases. Overall, feature extraction in the ResNet-101 algorithm plays a critical role in capturing meaningful patterns from input images, facilitating robust disease detection capabilities.

Model creation: ResNet-101, short for Residual Network with 101 layers, is a deep convolutional neural network architecture that introduces residual connections to address the vanishing gradient problem in very deep networks. The key formula used in ResNet-101 is the residual block:

$$\mathrm{Output} = \mathrm{Input} + \mathcal{F}(\mathrm{Input}, \mathrm{weights})$$

where:

- Input is the input feature map.
- $\mathcal{F}(\mathrm{Input}, \mathrm{weights})$ is the residual function representing the transformation applied to the input.
- $\mathrm{weights}$ are the parameters (e.g., weights and biases) learned by the residual function.
- Output is the output feature map of the residual block.

Figure 4. Model creation

In ResNet-101, residual blocks are stacked to form the deep architecture. Each residual block typically consists of several convolutional layers followed by batch normalization and ReLU activation functions. The shortcut connection (or skip connection) directly adds the input to the output of the residual function. This allows the model to learn residual mappings, making it easier to optimize and train deeper networks. Additionally, ResNet-101 incorporates other architectural elements such as bottleneck layers, which reduce computational complexity, and global average pooling, which aggregates spatial information before the final classification layer. These elements contribute to the effectiveness and efficiency of the ResNet-101 architecture in various computer vision tasks, including image classification, object detection, and segmentation.

Convolutional layer (CONV): The convolutional layer is a fundamental component in deep neural networks, including architectures like ResNet-101. It processes input feature maps by applying convolution operations using learnable filters, or kernels. These filters extract spatial patterns from the input, enabling the network to learn hierarchical representations of the data. Additionally, the layer incorporates a bias term, providing flexibility to adjust the output. Following the convolution operation, an activation function, often ReLU (Rectified Linear Unit), is applied element-wise to introduce non-linearity, helping the model learn complex relationships within the data. Overall, the convolutional layer plays a crucial role in feature extraction and transformation, facilitating the model's ability to understand and interpret the input data effectively.

Batch normalization (BATCHNORM): Batch normalization is a technique commonly used in deep neural networks to stabilize and accelerate the training process. It operates by normalizing the activations of each layer across the batch dimension during training. This normalization step ensures that the mean activation and variance remain consistent across mini-batches, mitigating issues like internal covariate shift and enabling smoother gradients during backpropagation. As a result, batch normalization helps to stabilize the learning process, leading to faster convergence and improved generalization performance. By reducing the sensitivity of the network to the initialization of parameters and the choice of learning rate, batch normalization facilitates more stable and efficient training of deep neural networks, ultimately enhancing their overall effectiveness in various machine learning tasks.

Relu activation function: The Rectified Linear Unit (ReLU) activation function is a simple yet effective non-linear function used in neural networks. It operates by replacing negative input values with zero while leaving positive values unchanged. Mathematically, the ReLU function can be defined as (***f(x) = max(0, x)***), where (***x***) is the input to the function. This simple thresholding operation introduces non-linearity to the network, allowing it to learn complex patterns and relationships within the data. ReLU activation is computationally efficient and helps mitigate the vanishing gradient problem, making it a popular choice in modern neural network architectures. Its simplicity and effectiveness contribute to its widespread adoption across various deep learning applications, facilitating faster training and improved model performance.

Bottleneck layer: Bottleneck layers are used in deeper ResNet architectures (like ResNet-50, ResNet-101, etc.) to reduce computational complexity. They typically consist of three consecutive convolutional layers: a 1x1 convolution to reduce dimensionality, followed by a 3x3 convolution, and another 1x1 convolution to increase dimensionality.

Global average pooling (GAP):

$$\mathrm{Output}[i] = \frac{1}{h \times w} \sum_{j=1}^{h} \sum_{k=1}^{w} \mathrm{Input}[i, j, k]$$

Global Average Pooling (GAP) is a pooling operation commonly used in convolutional neural networks (CNNs) for spatial dimensionality reduction. Unlike traditional pooling operations like max pooling, which select the maximum value within each pooling window, GAP computes the average value across each feature map's spatial dimensions (height and width). Mathematically, for each feature map, GAP calculates the average of all its values, resulting in a single value for each feature map. This process effectively reduces the spatial dimensions of the feature maps to 1x1, aggregating spatial information and preserving important features. The output of GAP is a vector of values representing the global spatial information of the feature maps. GAP is often employed as the final layer in CNN architectures before the fully connected layers, facilitating translation invariance and enabling the model to focus on the most discriminative features. Additionally, GAP reduces the model's parameter count and computational complexity, making it computationally efficient and less prone to overfitting. Overall, Global Average Pooling plays a crucial role in CNN architectures, contributing to their effectiveness in various computer vision tasks.

RESULT AND DISCUSSION:

Training accuracy:

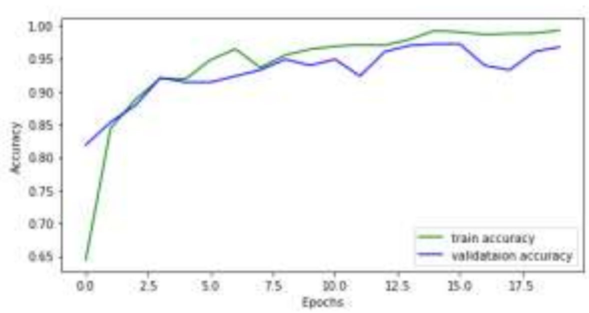

Figure 4.1 Graph for validation accuraccy

Accuracy is a fundamental metric used to evaluate the performance of classification models, measuring the proportion of correctly classified samples out of the total number of samples. In the context of machine learning, accuracy is often visualized over epochs, providing insights into the model's learning progress and performance improvement over time. A typical accuracy graph displays the training accuracy and validation accuracy on the y-axis, while the number of epochs is shown on the x-axis. During model training, both training and validation accuracy are monitored after each epoch. The training accuracy represents how well the model performs on the training dataset, while the validation accuracy reflects its performance on a separate validation dataset, which the model has not seen during training.

1. **Training Accuracy:**

$$\text{Training Accuracy} = \frac{\text{Number of Correct Predictions on Training Data}}{\text{Total Number of Training Samples}} \times 100\%$$

2. **Validation Accuracy:**

$$\text{Validation Accuracy} = \frac{\text{Number of Correct Predictions on Validation Data}}{\text{Total Number of Validation Samples}} \times 100\%$$

As the model trains over multiple epochs, the accuracy values are recorded and plotted on the graph. Initially, both training and validation accuracy may increase as the model learns to extract meaningful patterns from the data. However, as training progresses, the validation accuracy may start to plateau or even decrease, indicating potential overfitting if the model begins to memorize the training data rather than learning generalizable patterns. The accuracy graph provides valuable insights into the model's training dynamics and helps in making decisions regarding model architecture, hyperparameters, and training duration. A well-trained model typically exhibits a convergence of training and validation accuracy, indicating robust learning and good generalization performance (7-8).

TRAINING AND VALIDATION LOSS:

Training loss and validation loss are essential metrics used to assess the performance of machine learning models during training. The training loss quantifies the error between the model's predictions and the actual target values on the training dataset. It reflects how well the model is fitting the training data, with the objective of minimizing this loss over the course of training. However, achieving a training loss of zero may not necessarily indicate a perfect model; it could signify overfitting, where the model memorizes the training data without generalizing well to unseen data.

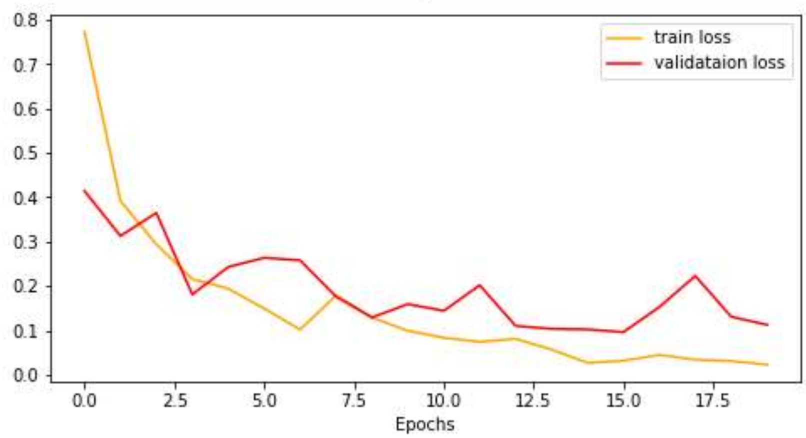

Figure 4.2 Graph for validation loss

On the other hand, validation loss measures the error between the model's predictions and the actual targets on a separate validation dataset, which the model hasn't been trained on. This loss serves as an estimate of how well the model will perform on new, unseen data. Monitoring the validation loss helps detect overfitting; if the validation loss starts to increase while the training loss decreases, it suggests that the model is overfitting the training data and may not generalize well (9).

Achieving low validation loss while maintaining proximity to the training loss indicates that the model is effectively learning without overfitting, leading to better generalization performance on unseen data. Balancing training and validation loss is crucial for building robust models that accurately generalize to new data.

CONFUSION MATRIX:

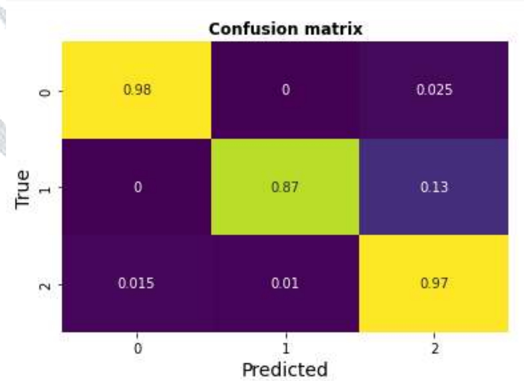

Figure 4.3 Confusion matrix for accuracy

The provided code generates a heatmap visualization of a confusion matrix, a valuable tool for evaluating classification model **performance**. The confusion matrix is computed using the correct and predicted labels, and normalized to highlight class-wise performance. The heatmap visually represents the normalized counts or percentages of true and predicted class labels, with annotations for clarity. This visualization offers insights into the model's classification accuracy and misclassifications, aiding in performance analysis and model refinement.

SHOWS THE PARAMETERS:

	precision	recall	f1-score	support
0	0.99	0.98	0.98	204
1	0.93	0.87	0.90	31
2	0.95	0.97	0.96	195
accuracy			0.97	430
macro avg	0.96	0.94	0.95	430
weighted avg	0.97	0.97	0.97	430

PRECISION: Precision measures the accuracy of positive predictions made by the model. It is calculated as the ratio of true positive predictions to the total number of positive predictions made by the model.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives}+\text{False Positives}}$$

Precision indicates the proportion of correctly predicted positive instances out of all instances predicted as positive. A high precision value suggests that the model has fewer false positives.

RECALL: Recall, also known as sensitivity or true positive rate, measures the ability of the model to correctly identify positive instances from all actual positive instances. It is calculated as the ratio of true positive predictions to the total number of actual positive instances.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives}+\text{False Negatives}}$$

Recall indicates the proportion of correctly predicted positive instances out of all actual positive instances. A high recall value suggests that the model has fewer false negatives.

F1 SCORE: The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both precision and recall.

$$\text{F1 Score} = 2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$$

The F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0. It provides a way to compare models with different precision and recall values.

In summary, precision measures the accuracy of positive predictions, recall measures the ability to correctly identify positive instances, and the F1 score balances both precision and recall, providing a comprehensive evaluation of classification model performance.

CONCLUSION

In conclusion, the adoption of ResNet101 algorithms for disease prediction in potato crops reflects a commitment to both accuracy and efficiency in disease identification within the proposed system. Renowned for their proficiency in real-time object detection, these deep learning architectures facilitate swift and precise identification of diseases affecting potato crops, offering farmers timely insights into their plant health. However, the system goes beyond mere disease identification by providing actionable remedies for the detected diseases. This integrated approach not only diagnoses the problem but also equips farmers with practical guidance on addressing and mitigating identified issues. By empowering farmers with actionable solutions, the system significantly contributes to enhancing agricultural productivity, reducing crop losses, and supporting the economic well-being of farmers. Ultimately, the system's comprehensive and innovative approach aligns with broader goals of achieving global food security and promoting sustainable agricultural practices, underscoring the pivotal role of technology in advancing the agriculture sector and ensuring a more resilient and sustainable food supply for future generations.

REFERENCE:

[1] Bakay, Melahat Sevg ̈ul, and ̇Umit A ̆gbulut. ”Electricity production based forecasting of greenhouse gas emissions in Turkey with deep learning, support vector machine and artificial neural network algorithms.” Journal of Cleaner Production 285 (2021): 125324.

[2] Salim, Alaa Sabeeh, et al. ”Novel Image Caption System Using Deep Convolutional Neural Networks (VGG16).” 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applica- tions (HORA). IEEE, 2022.

[3] Cruz, A.Ampatzidis, Y. , De Bellis, L. , Pierro, ,Panattoni, A. , Materazzi, A. , and Luvisi, A. . Automatic diagnosis of olive quick decline syndrome and grapevine yellows for the agriculture industry. In 2019 Second International Conference on Artificial Intelligence for Industries. (AI4I), pages 9–12. IEEE, 2019.

[4] Dhakal, A. and Shakya, S. . Image-based plant disease detection with deep learning. International Journal of Computer Trends and Technology, 61(1): 26–29, 2018.

[5] Ferentinos, K. P. . Deep learning models for plant disease detection and diagnosis. Computers and electronics in agriculture, 145:311–318, 2018.

[6] Islam, M. , Dinh, A. , Wahid, K. , and Bhowmik, P. . Detection of potato diseases using image segmentation and multiclass support vector machine. In 2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE), pages 1–4. IEEE, 2017.

[7] Ji, M. , Zhang, L. , and Wu, Q. . Automatic grape leaf diseases identification via unitedmodel based on multiple convolutional neural networks. Information Processing in Agriculture, 7(3):418–426, 2020.

[8] Ji, M. , Zhang, L. , and Wu, Q. . Automatic grape leaf diseases identification via unitedmodel based on multiple convolutional neural networks. Information Processing in Agriculture, 7(3):418–426, 2020.

[9] Khan, A. , Nawaz, U. , Ulhaq, A. , and Robinson, R. W. . Real-time plant health assessment via implementing cloud-based scalable transfer learning on aws deeplens. Plos one, 15(12):e0243243, 2020.