## Plant Disease Classification using Convolutional Neural Networks

**Introduction**

The goal of this research is to use leaf pictures and deep learning to accurately and efficiently classify plant diseases. Early plant disease detection can improve crop output and quality by assisting farmers in making timely decisions. A **ResNet18** architecture was modified to meet our classification requirements, and Convolutional Neural Networks (CNNs) were selected due to their excellent performance in image recognition tasks.



**Model Architecture**

The architecture selected for this project is based on **ResNet18**, a deep CNN whose residual connections, which lessen the **vanishing gradient** issue, have shown great accuracy in a variety of image classification tasks. This model was chosen because it strikes a compromise between computing efficiency and depth, making it appropriate for both accuracy and real-time inference.

**Modifications**:

* The final fully connected (FC) layer was adjusted to output predictions for our specific classes, representing various plant diseases.
* The model was trained end-to-end, with weights initialized from scratch to ensure the network could learn features unique to our dataset.

**Training Process and Hyper parameters**

The training process was conducted using the following settings:

* **Number of Epochs**: 2
* **Batch Size**: 32
* **Optimizer**: Adam, with a learning rate of **0.001**
* **Loss Function**: Cross-entropy loss

After preliminary testing revealed that these hyper parameters provided a favourable balance between accuracy and training speed, they were selected. Performance could be enhanced with more tuning, but the resources available were limited.

**Results and Performance Evaluation**

At every epoch, the model's performance was assessed, and progress was tracked by tracking important parameters like accuracy, validation loss, and training loss. Each epoch's results are described in full below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Epoch** | **Training Loss** | **Validation Loss** | **Validation Accuracy** |
| **1/2** | 0.5554 | 0.4119 | 87.68% |
| **2/2** | 0.2491 | 0.1946 | 93.84% |

***Note***: This model was saved as the final best version because it demonstrated a considerable improvement over the training epochs, reaching the greatest **accuracy of 93.84%.**

**Best Model Saved**: The model with the **highest validation accuracy (93.84%)** was saved and is used for final inference on unseen data.

**Evaluation Metrics and Model Justification**

To make sure the model works well across all classes and doesn't disproportionately misclassify some diseases, other measures including precision and recall would be employed in subsequent iterations in addition to accuracy, which reached 93.84% on the validation set. However, the model has enough robustness for real-world use, as evidenced by the high accuracy attained in just two epochs.

### The choice of ResNet18 aligns well with the project’s goals:

* **Accuracy**: To minimize misidentifications and guarantee dependability, high accuracy is essential for disease diagnosis.
* **Efficiency**: Because of the model's efficiency, it can be applied to a variety of deployment scenarios, such as mobile and edge devices, which is useful for farmers and agricultural inspectors to use in the field.

This illustrates how CNNs, more especially **ResNet18**, can be used to classify plant diseases accurately and effectively. With **a 93.84%** validation accuracy, the model exhibits encouraging promise for practical agricultural applications. Future research may include:

1. **Extending Training Epochs**: Accuracy may be further improved to considerable extend by using additional epochs.
2. **Expanding the Dataset**: A wider variety of photos may enhance generalization.
3. **Investigating Other Metrics:** F1-score analysis, precision, and recall would guarantee equitable performance across classes.

The current technology supports quick, on-the-go plant disease identification and offers a solid foundation for field implementation.

---------- END OF THE DOCUMENT ----------