**CNN Model Utilizing Batch Normalization, Dropout, and Data Augmentation for Tomato Leaf Disease Classification**

**Introduction:**

Leaf disease classification involves the identification and categorization of diseases affecting plant leaves, utilizing techniques from machine learning and computer vision. This process is essential in agriculture to ensure early detection and effective management of plant diseases, ultimately leading to better crop yields and reduced economic losses. By analyzing images of leaves, sophisticated algorithms can detect subtle patterns and symptoms indicative of specific diseases. Advanced models, often trained on large datasets, can classify leaf diseases with high accuracy, enabling farmers and agronomists to take timely actions to mitigate the spread and impact of these diseases.

**Objective:**

The goal of this assignment is to design, train, and evaluate a Convolutional Neural Network (CNN) model on the Rice Lead Disease dataset using Batch Normalization, Dropout, and Data Augmentation techniques. These techniques will help improve the model's performance and generalization. This assignment will help you understand the importance of batch normalization, dropout, and data augmentation in improving the performance and generalization of CNN models.

**Instructions:**

1. Load and Preprocess the Data
   1. Download the dataset and split it into 80 % for training and 20 % for testing. It contains ten main disease classes, as shown in Fig 1.

|  |  |
| --- | --- |
| Bacterial  spot |  |
| Early  blight |  |
| healthy |  |
| Late  blight |  |
| Leaf  Mold |  |
| Septoria  Leaf spot |  |
| Spider  mites Two spotted  spider  mite |  |
| Target  Spot |  |
| Tomato  Mosaic  virus |  |
| Tomato  Yellow  Leaf  Curl  Virus |  |

Fig 1. Some image samples from the dataset

1. Build the CNN Model:
   1. Propose a CNN model and apply it with batch normalization, dropout and data augmentation to prevent overfitting.
2. Compile and Train the Model:
   1. Compile the model with an appropriate optimizer, loss function, and metrics.
3. Evaluate the Model:
   1. Evaluate the model on the test data.
   2. Visualize the accuracy of training and validation and loss over epochs.
4. Report:
   1. Provide a brief report summarizing the model architecture and the impact of batch normalization, dropout, and data augmentation on the model's performance. In your report, include the following sections:
      1. Model Architecture: Describe the CNN architecture you used.
      2. Techniques Used: Explain how you integrated batch normalization, dropout, and data augmentation.
      3. Results: Discuss the training and test performance. Include the plots of accuracy and loss.
      4. Impact of Techniques: Reflect on the impact of batch normalization, dropout, and data augmentation on the model's performance.

Good Luck!