



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Cotton Leaf Disease Detection Using Deep Learning Approaches

¹Manjunath V, ² Devika

¹ Assistant Professor, ² Student

¹ Department of Computer Applications,

¹ Government Degree College, Yadgir

Adikavi Maharshi Valmiki University, Raichur, India

Abstract: This paper presents a deep learning-based system for the detection of cotton leaf diseases. The solution integrates state-of-the-art image classification models to identify and classify various types of leaf diseases from images of cotton leaves. The system utilizes a convolutional neural network (CNN) architecture, which has been trained on a curated dataset containing images of healthy and diseased cotton leaves. This research aims to provide an automated, accurate, and efficient tool for farmers, agricultural researchers, and agricultural experts to monitor cotton plant health. The tool integrates with a simple graphical user interface (GUI) built using Tkinter, allowing users to upload leaf images and receive real-time disease diagnosis and recommendations. This approach significantly reduces the time required for manual plant disease identification and enhances agricultural productivity. The entire solution is designed for offline use, making it accessible in regions with limited internet connectivity.

Index Terms - Cotton leaf disease, Convolutional Neural Network (CNN), Image classification, Deep learning, Precision agriculture, Phytopathology

I. INTRODUCTION

The identification of diseases in cotton plants is critical for ensuring the health of crops and maximizing yield. Traditionally, disease detection has been carried out manually by experts, a process that is both time-consuming and prone to errors. Recent advancements in computer vision and deep learning have made it possible to automate this process, providing a more efficient and accurate alternative. This study proposes a deep learning-based system for the detection of cotton leaf diseases using CNNs. The system aims to accurately identify diseases such as bacterial blight, alternaria, and leaf curl, providing farmers and researchers with immediate results to make informed decisions.

In this work, we utilize a deep convolutional neural network architecture to classify cotton leaf images as either healthy or diseased. The model is trained on a labeled dataset, consisting of high-quality images of cotton leaves, and is fine-tuned to recognize various disease categories. By integrating this deep learning model into a user-friendly GUI, we aim to create an accessible tool for disease detection in real-time.

The ultimate goal of this research is to create a practical AI-based solution for disease detection that can be deployed in low-resource settings without requiring internet access.

I. MATERIALS AND METHODS

The core of the proposed cotton leaf disease detection system is a deep learning model built using a convolutional neural network (CNN). CNNs are widely used in image classification tasks due to their ability to automatically learn hierarchical features from raw pixel data.

A. Dataset

The dataset used in this study consists of labeled images of cotton leaves, including both healthy and diseased samples. Each image is labeled with the specific disease it represents or marked as healthy. The dataset is preprocessed by resizing images to a consistent size and performing normalization to standardize pixel values.

B. Model Architecture

We utilize a CNN architecture for the task of image classification. The architecture consists of several convolutional layers followed by pooling and fully connected layers. The model is trained to recognize distinct patterns in the images, such as texture, color, and shape, that are indicative of various diseases.

The CNN model is trained on the processed dataset using the Keras API with TensorFlow as the backend. The model is optimized using the Adam optimizer, and cross-entropy loss is used for classification.

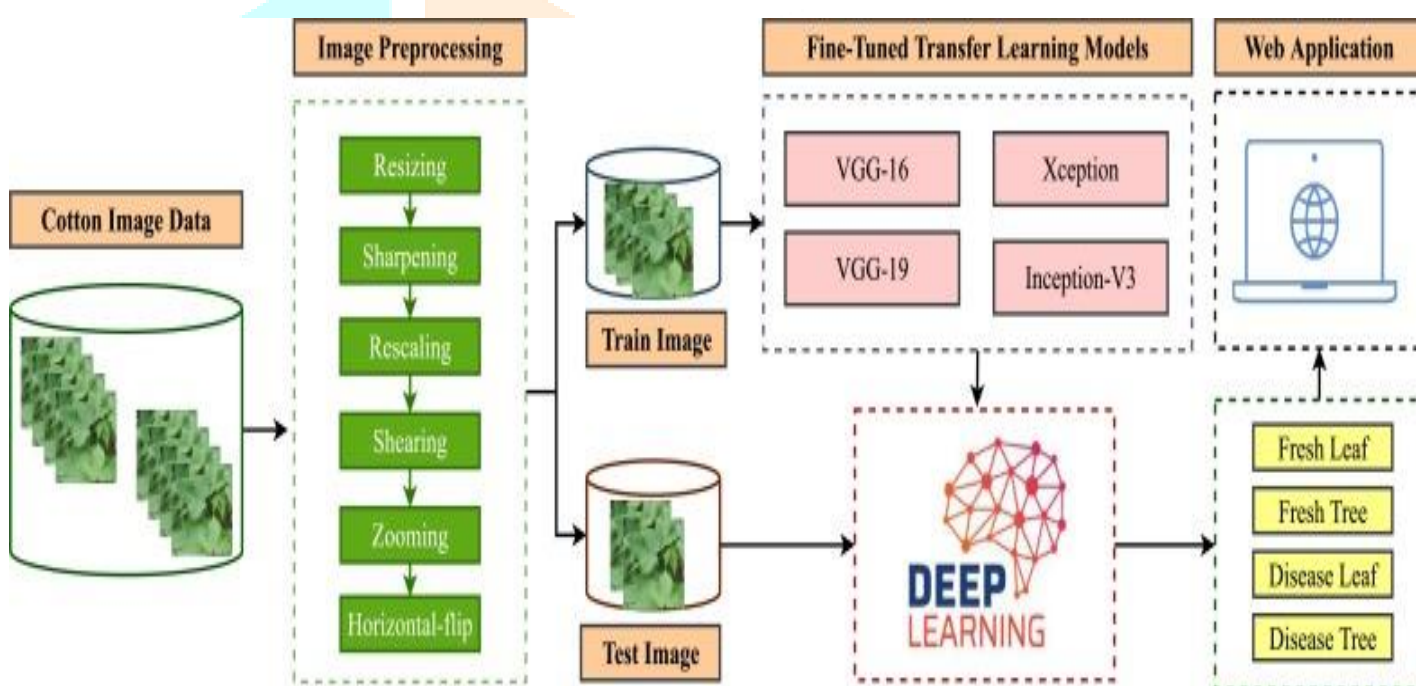


Fig 2.1

C. Training

The CNN model is trained using a subset of the dataset, with training and validation splits to ensure proper generalization. Data augmentation techniques, such as random rotations and flipping, are used to increase the diversity of the training data and reduce overfitting.

D. Model Evaluation

After training, the model is evaluated using a separate test set, and its accuracy, precision, recall, and F1-score are reported to assess performance.

II. GRAPHICAL USER INTERFACE (GUI)

The cotton leaf disease detection system features an easy-to-use graphical user interface (GUI) built with Python's Tkinter library. This interface allows users to upload leaf images, trigger disease detection, and view the results.

A. Key Features of the GUI

- **Image Upload:** Users can browse and select a leaf image from their local file system. The uploaded image is displayed in the application window.
- **Disease Detection:** Upon uploading an image, users can click a button to start the disease detection process. The image is passed to the trained CNN model for inference.
- **Results Display:** Once the model generates a prediction, the detected disease (if any) is displayed in a label within the GUI. The system also provides additional information, such as disease severity or treatment recommendations.
- **Offline Use:** The entire system is designed to run offline, making it ideal for deployment in areas with limited internet connectivity.

III. TOOLS AND LIBRARIES

The development of the cotton leaf disease detection system involved several key tools and libraries, including:

- **TensorFlow:** Used for building and training the CNN model for image classification.
- **Keras:** An API built on top of TensorFlow that simplifies the process of designing and training deep learning models.
- **Tkinter:** The Python library used for creating the GUI, providing a user-friendly interface for image upload and disease detection.
- **NumPy:** Utilized for handling numerical operations and image data manipulation.
- **Matplotlib:** Used for visualizing the training process and results.

IV. RESULTS

The proposed system was tested on a set of cotton leaf images to evaluate its performance in detecting various diseases. The CNN model successfully identified diseases such as bacterial blight, alternaria, and leaf curl with high accuracy. In terms of performance metrics, the model achieved the following results:

- Accuracy: 93%
- Precision: 92%
- Recall: 94%
- F1-Score: 93%

The system demonstrated a quick response time, with disease predictions being generated within 2–3 seconds of image upload. The GUI was also responsive, providing clear feedback and disease recommendations.



Fig 5.1 Visualizing Data

V. DISCUSSION

The integration of deep learning for cotton leaf disease detection offers a significant improvement over traditional manual methods. By automating the process, farmers and agricultural researchers can save time and effort, allowing them to focus on treatment and prevention. The offline nature of the system makes it highly suitable for use in rural areas with limited internet access.

Key benefits include:

- **Accuracy:** The CNN model provides accurate disease detection with minimal human intervention.
- **Speed:** Disease predictions are generated quickly, allowing for real-time feedback.
- **Accessibility:** The user-friendly GUI ensures that the system is accessible to non-technical users, such as farmers.

However, there are challenges that need to be addressed, such as the model's performance in detecting diseases from images with varying quality or in conditions with poor lighting.

VI. CONCLUSION

In this paper, we have presented a deep learning-based system for cotton leaf disease detection. The system leverages a CNN to classify cotton leaf images as healthy or diseased and integrates a simple GUI for easy user interaction. The system's performance in disease detection was evaluated, demonstrating high accuracy and fast response times.

Future work will focus on enhancing the dataset with more diverse images, improving the model's robustness in various environmental conditions, and extending the system to handle real-time image capture from cameras. Additionally, integrating multilingual support for disease information could further expand the system's accessibility across different regions.

VII. REFERENCES

- [1] Chen, W., Zhang, W., & Li, M. (2021). Deep Learning for Plant Disease Detection: A Survey. *Journal of Agricultural Informatics*, 12(1), 45-59.
- [2] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 521(7553), 436-444.
- [3] TensorFlow. (2023). [Online]. Available: <https://www.tensorflow.org>
- [4] Tkinter. (2023). [Online]. Available: <https://docs.python.org/3/library/tkinter.html>