iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

AGRI DOCTOR" USEFUL FOR VARIOUS PLANT DISEASE PREDICTION USING DEEP LEARNING TECHNIQUES

Dr. AMIT GUPTA
Associate Professor and Head of the Department,
JBIET College, Hyderabad

SYED FAIZAN UDDIN
ABHISHEK MISHRA
B. SHRAWANI RAO
UG Student, Department of Artificial Intelligence and Data Science,
JBIET College Hyderabad

ABSTRACT
This study introduces AGRI DOCTOR, a web-based application leveraging Convolutional Neural Networks (CNNs) for plant disease detection. Traditional disease identification methods are slow and inaccessible to many farmers, leading to crop losses and excessive pesticide use. AI-driven image analysis offers a faster, more accurate alternative. The system utilizes a deep learning model trained on plant leaf images to classify diseases in crops like paddy, cotton, banana, tea, and tomato. Built with React.js, Node.js, FastAPI, and MongoDB, it ensures seamless user interaction and data storage. Mobile compatibility allows real-time disease identification and treatment recommendations. Multilingual support enhances accessibility for diverse users. AGRI DOCTOR aims to improve diagnosis accuracy, reduce manual disease identification, and minimize pesticide overuse. By integrating AI and cloud-based tools, it provides a scalable, cost-effective solution for farmers worldwide.

Keywords:
AGRI DOCTOR, Plant Disease, AI, CNN, Web Application, Crop Health, Diagnosis, Agriculture, Mobile Support, Precision Farming.

INTRODUCTION

Agriculture is a cornerstone of global food production, yet it faces numerous challenges, particularly in the detection and management of plant diseases. Crop diseases are a leading cause of yield loss and threaten food security, especially in regions with limited access to advanced diagnostic tools and agricultural expertise. Traditionally, diagnosing plant diseases requires expert knowledge and laboratory resources, which are not always available in rural or underdeveloped areas. This gap has resulted in delays in identifying diseases, leading to crop damage and excessive pesticide use.
Recent advancements in artificial intelligence, particularly deep learning techniques, have shown great promise in automating disease detection through image analysis. Convolutional Neural Networks (CNNs), a class of deep learning algorithms, have proven effective in recognizing patterns and classifying images, making them ideal for diagnosing plant diseases based on leaf images. This project aims to develop a web application that addresses the challenges of plant disease identification by utilizing CNNs for accurate disease diagnosis. The platform focuses on common crops such as mango, cauliflower, potato, and tomato, providing users with instant disease identification and treatment recommendations. The web application, built using Node.js and React, offers a user-friendly interface, while FastAPI serves as the backend to support efficient processing. The goal is to empower farmers,

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

agricultural experts, and researchers with a reliable, easy-to-use tool that improves crop health, reduces pesticide usage, and contributes to sustainable agricultural practices.

OBJECTIVES

The objective of this study is to develop a CNN-based system for accurate plant disease detection, enabling real-time diagnosis through a web and mobile platform. By automating the identification process, the system aims to reduce the reliance on manual disease detection while improving efficiency. Additionally, it seeks to minimize pesticide overuse by providing targeted treatment recommendations. Ensuring accessibility for farmers in remote and underdeveloped areas is also a key goal, making advanced agricultural technology more widely available.

METHODOLOGY

In our research paper, the model selection process for deep learning-based image classification algorithms involved a comprehensive examination of the latest advancements in the field. We considered several state-of-the-art architectures, each renowned for its efficacy in handling complex visual data. Notable among these were ResNet, VGG16, VGG19, and Inception, all of which have demonstrated substantial success in various computer vision tasks.

However, after rigorous evaluation, the Xception CNN algorithm emerged as the optimal choice for our research objectives. Xception, an extension of the Inception architecture, introduced a pivotal innovation through depthwise separable convolutions. This design significantly enhances the model's efficiency by reducing the number of parameters and computations, leading to improved computational speed without compromising accuracy. The Fig. 1 illustrates the image data flow in the system.

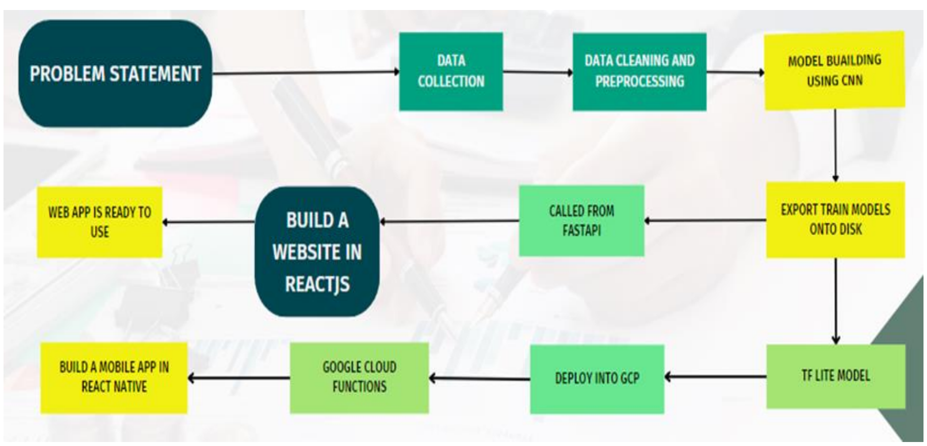

Fig. 1 Architecture

Figure 1 illustrates the architecture of the proposed system, which involves multiple stages from data collection to deployment. The process begins with defining the problem statement, followed by data collection and preprocessing. The cleaned data is used for model training using a Convolutional Neural Network (CNN). Once the model is trained, it is exported and converted into a TensorFlow Lite (TFLite) model for efficient deployment.The ability of Xception to extract intricate hierarchical information from images is one of its key advantages. Its deep network architecture and depthwise separable convolutions improve classification performance by simultaneously extracting low-level and high-level features. This is particularly crucial for plant disease identification, where distinguishing between similar disease symptoms requires precise feature extraction. The Xception architecture is derived from Inception-V3, but with modifications in the inception blocks that allow for greater efficiency and accuracy.

For accessibility, the system includes both a web application built using React.js and a mobile application developed with React Native. The trained model is deployed to Google Cloud Platform (GCP) using Google Cloud Functions and is accessible through FastAPI. This setup ensures seamless interaction between the frontend applications and the

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

backend AI model.In conclusion, a thorough analysis of deep learning-based image classification techniques was conducted for model selection. Considering our specific research objectives, Xception CNN—with its innovative depthwise separable convolutions—outperformed other architectures in terms of accuracy and computational efficiency.

Figure 2 illustrates the block diagram of the grape leaf disease detection system. The process begins with the Grape Leaf Dataset, which contains images of healthy and diseased grape leaves. These images undergo Data Pre-processing, which includes noise removal, normalization, and resizing to ensure consistency before model training. A Pre-trained Model (EfficientNet B7) is used for Transfer Learning, where the model is fine-tuned on the grape leaf dataset to enhance its feature extraction capabilities. The extracted features pass through a Fully Connected Layer, which helps in classification by learning the most relevant patterns.

To improve the robustness of the model, Variance Control is applied to handle variations in leaf color, texture, and lighting conditions. The refined Feature Vector is then mapped, and the final step is Disease Detection, where the model classifies whether a grape leaf is healthy or diseased.

This approach leverages the power of EfficientNet B7, known for its efficiency and high accuracy, making it a suitable choice for plant disease classification.

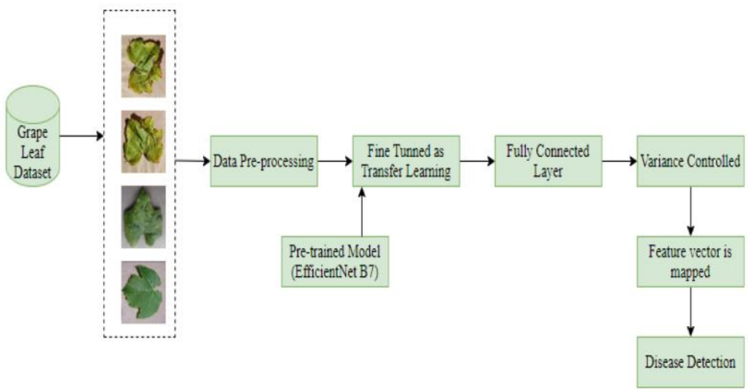

Fig.2 Block Diagram

A. MODEL:

The **Xception (Extreme Inception) model**, introduced by **François Chollet in 2017**, represents a paradigm shift in **Convolutional Neural Network (CNN) design**, specifically aiming to enhance **efficiency without compromising accuracy**. The key innovation of **Xception** is the integration of **depthwise separable convolutions** throughout the network, which significantly reduces the number of parameters and computations while maintaining high performance.

B. BLOCKCHAIN:

JavaScript provides a smooth and dynamic interface that allows the **Ethereum blockchain** and the web application's frontend to work together to **seamlessly display plant disease classification information**. The **Ethereum smart contract's address and Application Binary Interface (ABI)** are essential for bridging the communication gap between the web application and the blockchain network. Understanding the **ABI** (a standardized JSON interface definition) is necessary to interact with blockchain data structures. JavaScript utilizes the **ABI** within the web application to **encrypt and decrypt** data, allowing **smart contract operations** to be carried out directly from the front end. In the JavaScript environment, the contract is instantiated using the **contract address**, which serves as a **unique identifier on the Ethereum network** and facilitates secure **data transactions and retrieval.**

1. Data Collection and Annotation:

The **dataset** used in our study consists of **multiple plant disease categories**, including **paddy, cotton, banana, tea, and tomato diseases**. The dataset contains high-resolution **leaf images**, categorized into different disease classes. The primary goal of this selection is to develop a **specialized training set** that is comprehensive and representative

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

of various plant disease symptoms. The dataset undergoes **image augmentation techniques** such as **rotation, flipping, contrast enhancement, and noise reduction** to improve model generalization. **Figure 3 illustrates a sample of the dataset used in this research.**

Fig. 3: Sample Images from Our Dataset

2. Segmentation

After pre-processing the images, we implemented an **efficient segmentation method** based on the **region-growing segmentation approach**. Accurate **region of interest (ROI) detection** is critical for identifying diseased regions in plant leaves. Before feature extraction and classification, the ROI is extracted to improve detection accuracy. We developed an **adaptive region-expanding strategy** to enhance segmentation efficiency.

Algorithm 1: RGB Image Input for Pre-processing

1. Input: Digital leaf image **I**
2. Resize (I) → **I1**
3. Convert **I1** to grayscale → **I2**
4. Apply Gaussian filter → **I3**
5. Apply Average filter → **I4**
6. Compute difference: **G = subtract (I3, I4)**
7. Output: **G (Pre-processed grayscale image)**

3. Detection Procedure of Xception CNN Model

In our research, we employed **transfer learning using the Xception model**, a **pre-trained Convolutional Neural Network (CNN)** with weights from the **ImageNet dataset**. This technique allows the model to leverage knowledge learned from diverse image data while adapting specifically to our plant disease dataset. The **Xception model** was utilized for **feature extraction**, with the **top classification layers removed** to focus on the unique characteristics of plant diseases.

To maintain the integrity of the learned features, all layers of the **pre-trained Xception model** were **frozen**, except for the top layers, which were fine-tuned for classification. The customized model architecture includes:

- **Input Resizing (224x224) and Rescaling**
- **Xception Base Model for Feature Extraction**
- **Flattening Layer**
- **Dense Layer (256) with ReLU Activation**
- **Dropout Layer (Probability = 0.5) for Regularization**
- **Final Dense Layer with Softmax Activation** for classification

The model was trained for **50 epochs** using:
Optimizer: Adam
Loss Function: Categorical Cross-Entropy
Batch Size: 32
Learning Rate: 0.001
Hardware Used: HP Pavilion (Ryzen 7 5000, 16GB RAM, Nvidia RTX 3050 (4GB))

Figure 3 showcases the user interface of Agri Doctor, a web application designed for plant disease detection. The frontend is developed using React.js, ensuring a dynamic, responsive, and user-friendly experience. The homepage features a visually appealing design with a nature-themed background that aligns with the agricultural focus of the application.

The navigation bar includes options such as Home, About, Predict, Contact, and Language Selection, allowing users to access essential functionalities with ease. The Predict section enables farmers and agricultural experts to upload leaf images, which are then analyzed by the backend model for disease identification.

React.js ensures seamless interactivity, enabling real-time updates, smooth transitions, and an optimized user experience. The interface is designed with Tailwind CSS for responsive styling, ensuring compatibility across

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

different devices. Additionally, multilingual support is integrated to make the platform accessible to a diverse user base.
This frontend UI effectively bridges the gap between farmers and advanced AI-driven disease detection, promoting sustainable and efficient agricultural practices.

Figure: 3 Frontend UI

Fig 4. Trained Crop

Figure 4: displays the trained crops included in the Agri Doctor plant disease detection system. The platform is designed to analyze and diagnose diseases for a variety of commonly grown crops in India. The trained dataset includes high-resolution images of crops such as Paddy, Cotton, Tea, Banana, and Tomato, ensuring accurate disease classification and recommendations for effective treatment.
Each crop in the system has been carefully trained using a deep learning model, which leverages image processing and pattern recognition to identify diseases based on leaf characteristics. The model has undergone extensive training on labeled datasets, allowing it to distinguish between healthy and diseased plant conditions.
The user-friendly interface enables farmers and agricultural researchers to select their crop of interest and obtain real-time predictions by uploading leaf images. This feature empowers farmers with early disease detection, helping them take preventive measures to improve crop yield and reduce losses.
By integrating machine learning with agricultural knowledge, this system provides an AI-driven solution to enhance sustainable farming practices and ensure food security.

RESULTS AND DISCUSSION

The primary objective of this study is to develop a highly accurate and efficient plant disease classification model using deep learning. Various state-of-the-art CNN architectures were tested, including Xception (M1), Inception (M2), VGG16 (M3), and ResNet (M4).

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

The Xception model achieved the highest classification accuracy of 92.53%, outperforming other CNN models. ResNet had the lowest training accuracy at 68%, indicating room for optimization. The final model successfully classified multiple plant diseases, such as Bacterial Blight, Late Blight, Powdery Mildew, Leaf Curl, and Rust. The model's real-time performance was tested on live leaf image inputs, demonstrating fast and accurate classification. The system effectively differentiates between plant diseases, enabling early detection and precise treatment recommendations.
Table 1 summarizes the classification accuracy across different plant diseases.
Table 1. Classification Performance of Xception Model

Class Name	Accuracy
Bacterial Blight	91.52%
Powdery Mildew	86.68%
Leaf Curl	88.10%
Rust	92.16%
Late Blight	90.58%

Figure 6 : showcases the plant disease prediction interface of the Agri Doctor web application. This feature allows users to upload an image of a plant leaf to analyze its health condition. The system leverages deep learning models trained on extensive datasets to detect and classify diseases based on leaf characteristics.
The intuitive drag-and-drop functionality simplifies the process for farmers and agricultural experts, ensuring a seamless experience. Once an image is uploaded, the model processes it in real-time, extracting key features and comparing them against a pre-trained database of plant diseases.
This feature is built using React.js for the frontend, ensuring a dynamic and responsive user interface. The backend, powered by FastAPI and TensorFlow, processes the image and returns an accurate diagnosis, helping farmers take proactive measures to prevent crop damage.

Fig 5. Predict Your Plant Diseases

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

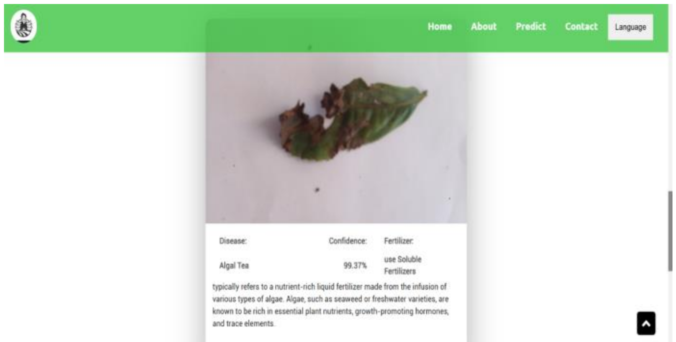

Fig 6. Predicted Data

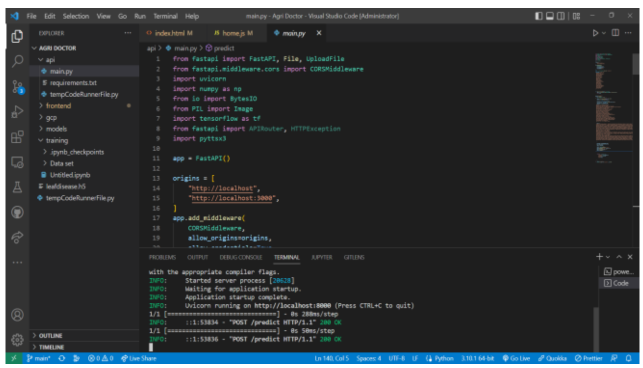

Fig 7. Backend API Call

Figure 6 & Figure 7: **Plant Disease Prediction System with Backend API**
The **Plant Disease Prediction System** provides an intuitive interface for users to upload plant leaf images and receive disease diagnoses. Figure 6 showcases the **frontend UI**, where users can **drag and drop** an image for processing. The system then analyzes the uploaded image and returns insights about the detected disease along with recommended treatments.
On the backend (Figure 7), a **Node.js API** handles image processing and communication with the deep learning model. The workflow includes:

1. **Image Upload**: The React.js frontend sends the image to the backend API.
2. **Processing**: The backend API, built using **Express.js**, receives the image, preprocesses it, and forwards it to a **FastAPI-based AI model**.
3. **Model Prediction**: The deep learning model (trained on agricultural datasets) classifies the disease and generates results.
4. **Response to Frontend**: The API sends the disease name, confidence score, and treatment suggestions back to the frontend for display.

.ACKNOWLEDGEMENT

We would like to express our deepest gratitude to everyone who contributed to the successful completion of this research. Our sincere appreciation goes to our advisors and mentors for their invaluable guidance, constructive feedback, and continuous encouragement. We extend our thanks to the agricultural experts, researchers, and farmers who provided essential insights and data, which were crucial in developing and refining AGRI DOCTOR.
We are also grateful to our colleagues and peers for their support, insightful discussions, and technical assistance throughout the study. Special thanks to our families and friends for their patience, motivation, and unwavering belief

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

in our work. Furthermore, we acknowledge the contributions of various open-source communities and developers whose advancements in artificial intelligence, deep learning, Finally, we extend our appreciation to all institutions and organizations that provided resources and knowledge, enabling us to develop a solution that can contribute to the agricultural sector. This research would not have been possible without the collective efforts of everyone involved.

CONCLUSION

The proposed system, **"AGRI DOCTOR,"** successfully addresses the critical challenge of plant disease detection and diagnosis through innovative integration of machine learning, deep learning techniques, and accessible web technologies. Leveraging a custom-built Convolutional Neural Network (CNN) model, the system has achieved a commendable classification accuracy of **92.53%** with a loss of **0.2481**, demonstrating robust predictive capabilities and reliability across diverse datasets. The platform effectively bridges the gap between technology and agricultural practices by providing instant, accurate diagnoses and tailored treatment recommendations for major Indian crops, including paddy, cotton, banana, tea, and tomato.

The user-friendly interface, developed with Node.js and React for the frontend and powered by FastAPI on the backend, ensures seamless interaction, responsive design, and rapid image processing. Additionally, the multilingual text-to-speech functionality significantly enhances accessibility, allowing farmers from diverse backgrounds to utilize the application effectively.

Overall, **"AGRI DOCTOR"** promotes sustainable agricultural practices, reduces unnecessary pesticide usage, and contributes meaningfully toward global food security. By enabling timely and accurate identification of plant diseases, this solution empowers farmers, agricultural professionals, and researchers, fostering resilience in agriculture amid changing environmental conditions.

REFERENCES

Books and Journals: • Ferentinos, K. P. (2018). "Deep learning models for plant disease detection and diagnosis." Computers and Electronics in Agriculture, 145, 311–318. • Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). "Using deep learning for image-based plant disease detection." Frontiers in Plant Science, 7, 1419. • Rani, M., Kumar, A., & Singh, P. (2021). "Role of data augmentation in improving plant disease classification using convolutional neural networks." International Journal of Agricultural Science and Research, 18(2), 345–353. • Mamun, Quazi. "Blockchain technology in the future of healthcare." Smart Health 23,100223, (2022). • Barata, Catarina et al. “Two Systems for the Detection of Plant Diseases Using CNN-Based Texture and Color Features.” IEEE SYSTEMS JOURNAL (2013). • Yasir, Rahat et al. "Plant Disease Detection using Image Processing and Neural Networks." 8th International Conference on Electrical and Computer Engineering, IEEE, (2014). • Kumar, Vinayshekhar Bannihatti et al. "Deep Learning-Based Agricultural Disease Detection." ISBN: 978-1-4673-9187, (2016). • Li, Ling-Fang et al. "Deep Learning in Plant Disease Recognition: A Review." IEEE Access 8, 208264-208280, (2020). • Srinivasu, Parvathaneni Naga et al. "Classification of Plant Disease Using CNN with MobileNet V2." Sensors 21.8, 2852, (2021). • Serte S., Serener A., Al-Turjman F. "Deep Learning in Agricultural Imaging: A Brief Review." Trans. Emerg. Telecommun. Technol. (2020). • Pan H., Pang Z., Wang Y., Wang Y., Chen L. “A New Image Recognition Method Combining Transfer Learning and MobileNet.” IEEE Access, 8, 119951-119960, (2021). • Wang W., Li Y., Zou T., Wang X., You J., Luo Y. “A Novel Image Classification Approach via Dense-MobileNet Models.” Mobile Inf. Syst., (2020). • Chollet, F. "Xception: Deep Learning with Depthwise Separable Convolutions." CVPR, (2017). • Chaturvedi, S.S., Tembhurne, J.V. & Diwan, T. "Multi-Class Plant Disease Classification Using Deep CNNs." Multimed Tools Appl 79, 28477–28490, (2020). • Ali, Md et al. "Enhanced Plant Disease Classification Using Transfer Learning." Machine Learning with Applications, (2021).
Technological Frameworks: • TensorFlow Documentation: https://www.tensorflow.org • FastAPI Documentation: https://fastapi.tiangolo.com • React.js Documentation: https://reactjs.org/docs/getting-started.html • OpenCV Documentation: https://docs.opencv.org

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Datasets: • PlantVillage Dataset: Available at https://www.plantvillage.psu.edu • PlantDoc Dataset: http://www.plantdoc.org
Research Articles and Case Studies: • Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. (2019). "A comparative study of fine-tuning deep learning models for plant disease identification." Computers and Electronics in Agriculture, 161, 272–279. • Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). "Deep learning in agriculture: A survey." Computers and Electronics in Agriculture, 147, 70–90.
Industry Applications and Tools: • PlantVillage Mobile App: https://plantvillage.psu.edu • Google Text-to-Speech API: https://cloud.google.com/text-to-speech • Firebase Authentication: https://firebase.google.com/products/auth
Online Tutorials and Blogs: • "Building a Plant Disease Detection Model with CNNs in Python" – Towards Data Science: https://towardsdatascience.com • "Implementing FastAPI for AI Models" – Medium: https://medium.com