IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Convolutional Neural Network-Based Grape Leaf Disease Detection With Regional Language Integration

[1]Jahnavi C, [2]Varsha P, [3]Leena J, [4]Rachana V Murthy

[1]Student, [2]Student, [3]Student, [4]Assistant Professor

[1 2 3 4]Department of CSE(ICB),

[1 2 3 4]K. S. Institute of Technology, Bangalore, India

Abstract: The health of grape plants is crucial for ensuring high-quality vineyard yields and maintaining the economic sustainability of viticulture. Effective disease detection is a pivotal aspect of modern agricultural management, as diseases such as black measles, leaf blight, and black rot can significantly impact crop production. This paper discusses research on some advanced methods in the field of grape plant disease detection by incorporating machine learning algorithms and image processing techniques. In this paper, the use of spectral imaging, neural networks, and field-based monitoring systems for early, precise, and cost-effective diagnosis of diseases is discussed and the user interface is in the regional language Kannada for better usability of farmer. By addressing the limitations of traditional manual inspection methods, this research aims to highlight innovative approaches that enhance efficiency and reduce the environmental impact of disease management practices. The findings underscore the potential of precision agriculture in revolutionizing disease control strategies in viticulture.

Index Terms – Grape Plant Disease Classification, Image Processing, Deep Learning, Feature Extraction, CNN

I. Introduction

The Grape Plant Disease Detection System is an innovative tool designed to help farmers efficiently identify and address grape plant diseases. The project utilizes a specially developed convolutional neural network (CNN) trained on a grape disease dataset to deliver precise and reliable predictions. By using RGB images of grape leaves for detection, the system eliminates the need for costly imaging equipment like hyperspectral cameras, making it both affordable and widely accessible. Deployed as a web-based application using Flask, it can be accessed seamlessly on various devices, such as desktops and mobile phones. Farmers can simply upload photos of grape leaves, and the system swiftly identifies the disease and provides actionable treatment recommendations.

Figure 1: Grape Plant Diseases

Grape farming is a crucial aspect of agriculture, significantly contributing to the economy and food supply. However, grapevines are highly prone to diseases such as Black Measles, Black Rot, and Leaf Blight shown in Figure 1, which can drastically reduce both yield and quality. Detecting these diseases early and accurately is vital for effective management, but traditional diagnostic methods often rely on expert knowledge, making them costly, time-intensive, and out of reach for many farmers. This initiative seeks to address these challenges by introducing an automated, user-friendly system that uses machine learning and computer vision to identify grapevine diseases from leaf images. To further improve accessibility, the system incorporates a distinctive feature of providing outputs in Kannada, ensuring it serves local farmers with limited technical skills. By minimizing the reliance on experts and delivering dependable real-time diagnoses, this project aims to encourage sustainable farming practices, reduce crop losses, and support the agricultural community.

II. Literature Survey

This project on grape plant disease detection involved studying the existing research and technologies in plant health monitoring. With exploring the use of machine learning and deep learning techniques, such as CNNs, which are highly effective for image-based classification tasks. Previous studies mostly utilized generalized datasets like PlantVillage and focused on a wide range of crops but lacked specificity for grape diseases. Furthermore, most of the systems developed were more theoretical than practical. Taking forward this idea, this project attempted to fill in these gaps by designing a dataset for grape diseases, using CNNs for accurate classification, and designing a user- friendly web application in Kannada based on Flask for practical usage.

In [1] authors have used a pre-trained deep convolutional neural network as a feature Extracto 63+9r and a random forest as a classifier. In total, 1003 images of four different classes are used, and an accuracy of 91.66% is obtained. In [2] authors have utilized an improved transfer learning based EfficientNet network for diagnosing grape leaf diseases. The new model is either better or equivalent to the previous results when identifying illness in grape leaves. In [3] authors used the novel image processing algorithm and multi-class SVM. The region of the symptoms of the disease is automatically separated from the healthy parts of the leaf with the help of K-means clustering, and the features are extracted in three color models- RGB, HSV, and l*a*b. In [4] authors have implemented a plant disease detection model along with classification of leaf images in the system and application deep convolutional networks. The results of the experiments run on the newly constructed model have an accuracy of 85% to 95% on individual class assessments, averaging at 96.3%. In [5] authors utilized a new deep learning model with multimodal data and parallel heterogeneous activation functions for the detection of grape diseases. Experiments were performed to demonstrate the model's good performance with 91% accuracy, precision of 93%, recall of 90%, a mean average precision of 91% is obtained. In [6] authors presented a novel framework that uses the model TabPFN to predict blockwise diseases of grapevine from remote sensing imagery from multi-sensor data on climate variables. In [7] authors have focused on fine-tuning cutting-edge pre-trained CNN and vision transformer models to classify grape leaves and diagnose grape leaf diseases through digital images. In [8] authors have used an integrated method-based architecture for convolutional neural network.

III. Research Methodology

System architecture: The system architecture for grape disease detection shown in Figure 2 is composed of several interrelated components. The user interface is a web interface built using HTML, CSS, and JavaScript to upload grape leaf images. The Flask backend server processes the upload, communicating with the image preprocessing module, which resizes and normalizes the image using OpenCV and Pillow. The preprocessed image is forwarded to a trained CNN model implemented in TensorFlow/Keras for the classification of diseases. This result, including the prediction and treatment suggestions, is generated and delivered back to the user interface in a friendly format with accessibility features like Kannada language support. A database stores additional information about diseases and treatments, further enriching the feedback provided to users. This architecture, therefore, ensures a smooth, efficient, and accessible process for detecting grape diseases.

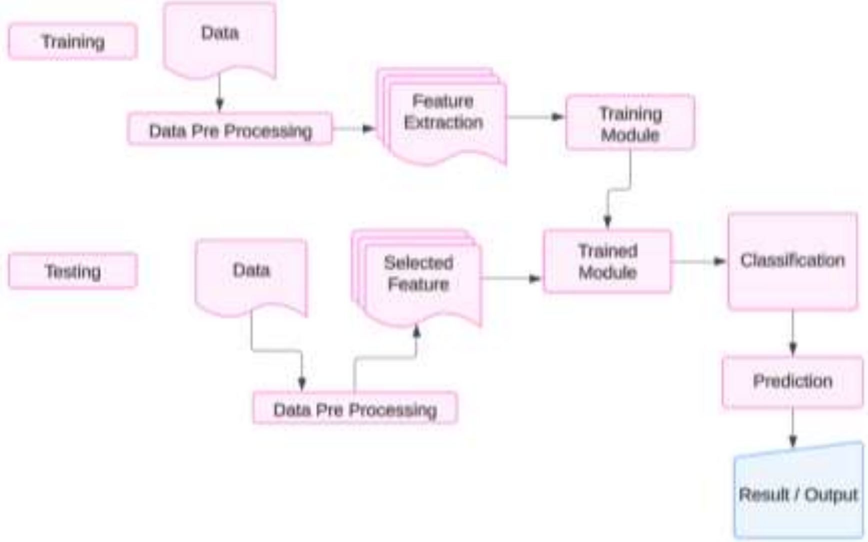

Figure 2: System Architecture

Data collection: Collection of data is an important part of developing grape plant disease detection, and it forms the base from which the deep learning model will be trained and tested. It affects the quality and diversity in the data collected. High-quality, diversifying collections of images of grape leaves, both healthy and infected, are gathered for broad training of the model in this module. Sources of the images of grape leaves include Kaggle database shown in Figure 3. The dataset is classified into various classes, including healthy leaves and different types of diseases, like Black Measles Disease, Black Rot Disease, and Leaf Blight Disease. The gathered images are resized, normalized, and annotated to fit into the model's input requirements; therefore, data consistency can be guaranteed in order to train effectively. The collection has 944 images of Black Rot disease, 1107 images of Black Measles disease, 861 images of Leaf Blight disease and 339 images of Healthy Grape Leaves.

Figure 3: Dataset used to train the CNN model

Data preprocessing: Data preprocessing is preparing the collected images for model training. Images of grape leaves representing both healthy and diseased conditions are resized to a standard 224x224 pixels for uniform input dimensions. The pixel values of these images are normalized by scaling them to a range of [0, 1] to accelerate the training process and improve model performance. Furthermore, the images are converted into NumPy arrays and expanded to include a batch dimension, which aligns them with the input format required by the convolutional neural network. This preprocessing ensures that the data is consistent and ready for effective model training and accurate disease classification. This resizes ensures that regardless of the varying original image dimensions, the dimensions become the same to be able to feed the images to the model without having the problem of dimension mismatch. Then, pixel values for images were normalized by taking pixel values divided by 255 in such a manner that the pixel range could be transformed from [0, 255] to [0, 1]. This normalization step will help make the model train more efficiently and will avoid several other problems from different input ranges. Finally, the images are transformed into NumPy arrays-this is the typical data type used in deep learning models in Python. For compatibility purposes with the model's input requirements, an additional dimension is appended to the array to make it equal to the expected input shape-batch size, height, width, channels. At the end, the images are divided into training and testing sets.

Feature extraction: The purpose of feature extraction is to look for meaningful patterns in the preprocessed images, which would then be helpful in correctly classifying the disease. Through the CNN's convolutional layers, the model will automatically learn to extract features like edges, textures, and shapes from the input images, thereby enabling the system to classify different grape diseases and healthy leaves. As the image progresses, the model captures higher-level features progressively with successive convolutional and pooling layers. This auto-extracting feature would eliminate the need to manually engineer the features, meaning the model would learn complicated patterns from the image data for an accurate prediction.

Training and Testing: The training and testing of the model are critical phases in the project of grape plant disease detection. It will ensure that the model is able to classify grape leaves based on their health status. In the training phase, the preprocessed images are fed into the CNN. It learns to associate the extracted features with their corresponding disease classes, including Black Measles Disease, Black Rot Disease, Leaf Blight Disease, and Healthy. It involves the adaptation of the model's weights via back propagation and optimization using a training dataset that is split into training and validation sets. It uses a validation set to measure its performance after each epoch, enabling one to tune hyperparameters toward better accuracy and prevent overfitting.

In the test phase, the model tests images that it has never seen before, known as a test set. The test set helps assess how well the model generalizes to new, unseen data. The accuracy of the model is found by comparing its prediction against the actual labels of test images. Results will then provide a sign of how accurate the model can be at identifying diseases of grape leaf and, consequently, healthy leaves.

IV. RESEARCH METHODOLOGY

Implementation refers to the process of turning a technical specification or algorithm into a functional program, software component, or computer system through programming and deployment. There can be multiple implementations for a single specification or standard. For instance, web browsers implement the specifications recommended by the World Wide Web Consortium, while software development tools implement various programming languages to provide functionality. The following are the libraries used for the implementation of the proposed work: OpenCV, NumPy, TensorFlow, Keras, Sklearn, Pillow (PIL), Flask. Using these libraries a data pre-processing module is implemented. Data preprocessing is a critical step in preparing grape leaf images for the CNN model. This process includes resizing, normalizing, and augmenting the dataset to ensure high-quality inputs for effective training and testing. Also, a training and testing module is implemented. The major steps involved in the training and testing module in the code are basically two: training of the model and evaluation. Finally, a CNN module is implemented. A Convolutional Neural Network (CNN) model was developed using the TensorFlow-Keras framework to classify grape leaf images into four categories: black measles, black rot, leaf blight, and healthy. The model takes RGB images resized to 224×224 pixels as input.

The CNN model includes three convolutional layers, each followed by a max pooling layer, and two fully connected dense layers.

The final layer uses the Softmax activation function for multiclass classification.

1. Input Layer:
   - Input image size: 224 × 224 × 3 (RGB)
2. Convolutional and Pooling Layers:
   - Conv2D Layer 1: It uses 32 filters of 3×3 kernel and applies ReLU activation function, followed by 2×2 MaxPooling.
   - Conv2D Layer 2: It uses 64 filters of 3×3 kernel and applies ReLU activation function, followed by 2×2 MaxPooling.
   - Conv2D Layer 3: It uses 128 filters of 3×3 kernel and ReLU activation, followed by 2×2 MaxPooling.
3. Flatten Layer:
   - After the convolutional blocks. It converts the 3D output to a 1D vector.
4. Fully Connected Layers:
   - 1D vector is fed into a dense layer with 128 neurons and ReLU activation.

To prevent overfitting, a dropout layer with a 50% dropout rate is applied. Finally, a dense output layer with 4 neurons and Softmax activation is used to predict the class probabilities.
Training was conducted with the Adam optimizer and categorical crossentropy as the loss function. The dataset was split 80-20 into training and test sets, and 20% of the training data was further reserved for validation. The model was trained for 10 epochs using a batch size of 32.

Upon evaluation, the CNN achieved approximately 92% accuracy on the test set, indicating effective learning and generalization. The trained model was saved in HDF5 format for future use or deployment.
The training and validation performance of the Convolutional Neural Network (CNN) model was visualized using accuracy and loss graphs across 10 epochs.

In the accuracy graph, the model's training accuracy showed a consistent increase, reaching nearly 98% by the final epoch. Similarly, the validation accuracy also improved steadily and stabilized around 94–95%. This close match between training and validation accuracy indicates that the model is generalizing well and is not overfitting.

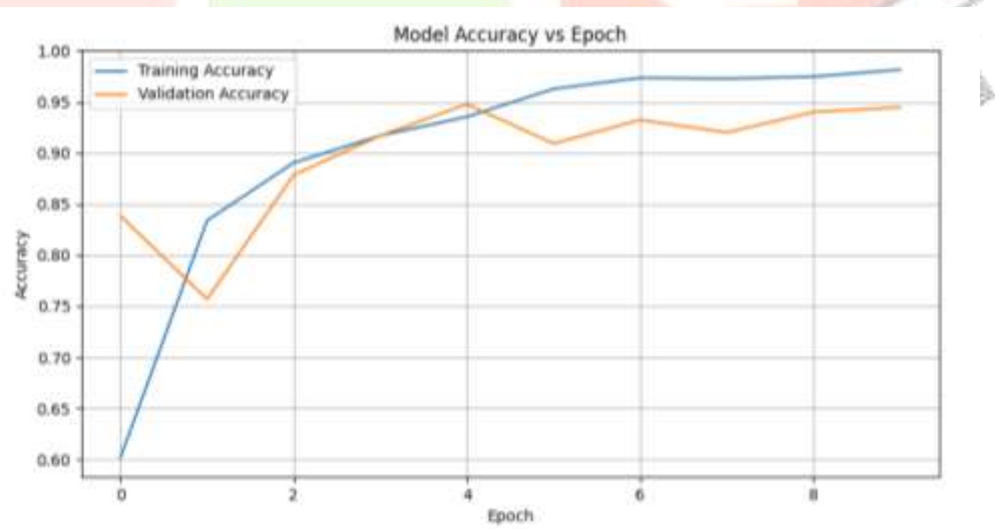

Figure 4: Accuracy vs Epoch graph

In the loss graph, the training loss decreased significantly and remained below 0.1, while the validation loss also dropped and stabilized around 0.15. The minimal gap between training and validation loss reflects that the model is learning effectively without memorizing the training data.

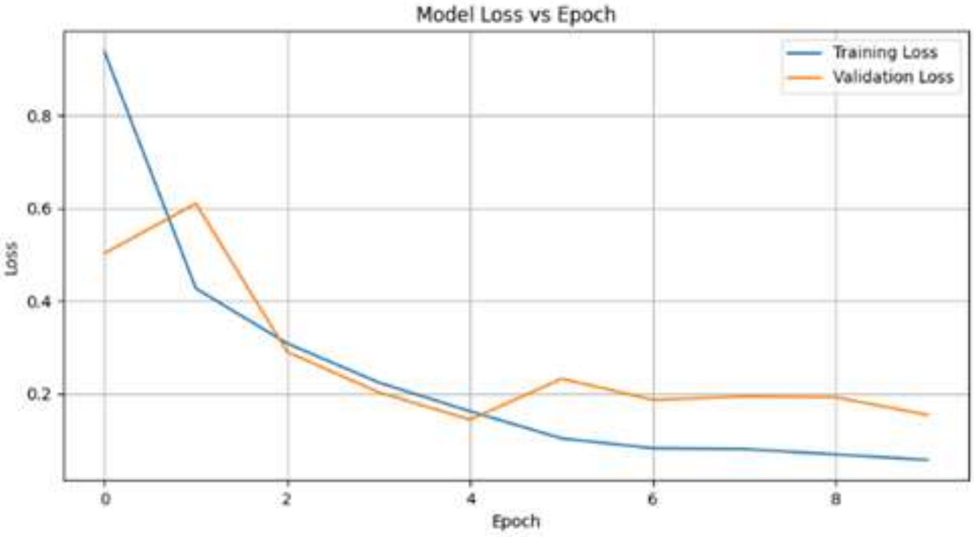

Fig 5: Loss vs Epoch graph

Overall, both graphs demonstrate strong model performance with good learning progression and minimal signs of overfitting, confirming the reliability of the trained CNN for grape leaf disease classification.

The implementation of the grape leaf disease detection system using a Convolutional Neural Network (CNN) has proven to be effective in accurately classifying grape leaf images into four distinct categories. Through the use of essential libraries such as OpenCV, NumPy, TensorFlow, and Keras, robust data preprocessing and model training modules were developed. The model architecture was designed to efficiently extract features and perform accurate classification. The performance evaluation through graphs and test accuracy confirmed that the model achieved high accuracy with minimal overfitting. Overall, the CNN model demonstrates strong potential for real-world application in agricultural settings, aiding farmers in early detection and management of grape plant diseases.

V. Regional Language Interface Development And Integration

To make the system accessible to Kannada-speaking users, a web-based user interface was developed using HTML, CSS, and JavaScript. The interface allows farmers to interact with the system in their native language (Kannada) without needing technical knowledge.

The frontend consists of Title, buttons, and instructions displayed in Kannada, Input field to upload grape leaf images, A Scan button labeled in Kannada, and Prediction results (such as disease name) and treatment recommendations shown completely in Kannada.

Styling was done using **CSS** to create a simple, clean, and mobile-responsive layout, improving readability and ease of use.

Backend Integration: The interface communicates with the CNN model hosted on a Flask server. When a user uploads an image and clicks Scan, the image is sent to the backend via a POST API call. The model processes the image, predicts the disease category, and sends the result back to the frontend. Based on the prediction, appropriate treatment recommendations in Kannada are displayed on the page. Organic and chemical treatment options were separately detailed for each disease type. Additionally, YouTube link for further treatment guidance were embedded for ease of understanding.

VI. Result

The result page displays the predicted disease or health status of the uploaded grape leaf. It provides detailed information about the identified condition, including its name and relevant treatments in Kannada. The content is presented in a clear and organized manner, often in the local language for better accessibility. Additional resources, such as external links for further guidance, are included to assist users in addressing the identified issue effectively. Figure 6 shows the home page to insert the photo.

Figure 6: Home Page

Figure 7: Output after classification

VII. Future Enhancement

1. The system can further be extended to detect even a much wider range of grape diseases, like powdery mildew or downy mildew by adding other disease categories into the dataset.
2. On-the-go disease detection through real time: The farmers can upload the images on the spot in the mobile app or through integrating the smartphones with the system. Through this, disease detection happens much quicker.
3. Make the System Multilingual: Adding language support for different regions will make the tool more accessible to a global audience, helping farmers from different countries use it easily.
4. Using IoT based solution: By integrating IoT-enabled cameras and sensors in vineyards, real-time monitoring of grape plants can be automated. These devices can capture leaf images continuously and send them to the trained CNN model deployed on a cloud server or edge device.

VIII. Conclusion

The grape plant disease detection project provides a glimpse into the effectiveness of machine learning, through Convolutional Neural Networks (CNNs), in overcoming bottlenecks in modern farming. With the aid of a large dataset of labelled images of grape leaves, the model was able to work well in identifying and differentiating common grape diseases among them being Black Measles Disease, Black Rot Disease, Leaf Blight Disease, and Healthy leaves. This system offers a quick, automated means of diagnosing health issues in plants, which is important to prevent the spread of disease and reduce crop loss in vineyards. The implementation of this technology will provide grape farmers with an efficient monitoring tool to determine the health of their grape plants to allow early detection of disease and intervention. With continuous improvement, including extension of the dataset with addition of more diseases, optimal performance of the model, further improvement of the system towards the accuracy and applicability range can be achieved. More importantly, real-time collection and analysis of data from mobile applications or IoT devices will improve the usability of the system, making it reach farmers in rural areas. The project also acts as an entry point to artificial intelligence in agriculture, thus establishing its potential to transform farming. A system that reduces reliance on manual inspections and offers data-driven decisions quicker will aid in promoting more sustainable methods of farming and resource optimization. The future might see the model scaled up to accommodate various crops and expand its functionalities to make it a critical component in the agricultural sector, contributing to global efforts to improve food security and sustainable farming methods.

REFERENCES

[1] Bhavya Jain et. al, “Grapes Disease Detection Using Transfer Learning”, arXiv 2022.

[2] Atul Kumar Uttam et. al, “Grape Leaf Disease Prediction Using Deep Learning”, IEEE 2022.

[3] Seyed Mohamad Javidan et. al, “Diagnosis of grape leaf diseases using automatic K-means clustering and machine learning”, ScienceDirect 2023.

[4] Kavithamani V et. al, “Advanced Grape Leaf Disease Detection using Neural Network”, IEEE 2023.

[5] Ruiheng Li et. al, “High-Performance Grape Disease Detection Method Using Multimodal Data and Parallel Activation Functions”, journal Plants 2024.

[6] Weiying Zhao et. al, “Grapevine Disease Prediction Using Climate Variables from Multi-Sensor Remote Sensing Imagery via a Transformer Model”, ICLR 2024.

[7] Ismail Kunduracioglu et. al, “Advancements in deep learning for ac- curate classification of grape leaves and diagnosis of grape diseases”, Springer 2024.

[8] Ajinkya Ghuge et. al, “Grape leaf disease detection using image processing and CNN”, IJARCCE 2024.

[9] Bin Liu et. al, “Grape Leaf Disease Identification Using Improved Deep Convolutional Neural Networks”, Frontiers 2024.

[10] Kunduracioglu et. al, “Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases”, Springer 2024.

[11] Kunduracioglu et. al, “Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases”, Springer 2024.

[12] Moh. Arie Hasan et. al, “Identification of Grape Leaf Diseases Using Convolutional Neural Network”, IOPscience 2020.