RESEARCH ARTICLE OPEN ACCESS

AI-Based Plant Disease Detection

Abdul Mueed Shaikh*, Bashwaraj Sonkawade**, Jatin Jaiswal***,
Anushka Shrivastava****, Prof.B.B.Jadhav
*(Department of CSE AI&ML, G.V. A..I.E.T, Shelu, India
Email: mueed0831@gmail.com)
**(Department of CSE AI&ML, G.V. A..I.E.T, Shelu, India
Email: sonkawade11@gmail.com)
***(Department of CSE AI&ML, G.V. A..I.E.T, Shelu, India
Email: jaiswaljatin44@gmail.com)
****(Department of CSE AI&ML, G.V. A..I.E.T, Shelu, India
Email:shrivastavaanushka007@gmail.com

Abstract:

Farm Assist is an impactful smart farming application built with Python, focusing on AI-powered plant disease detection. The application streamlines agricultural decision-making by offering a least-cost supply list based on user inputs and connecting farmers with experts via chat. Its core innovation lies in AI-driven crop disease identification, providing users with actionable insights on fertilizer types and preventive measures. Farm Assist stands out as a user-friendly, preliminary tool that enhances productivity without complicating the user experience. By leveraging Python's efficiency, this application represents a transformative step towards accessible and effective solutions for farmers, contributing to the sustainable evolution of modern agricultural practices.

Keywords— Smart farming, Python, AI disease detection, Agricultural decision support, Chat interface, CNN-model

I. INTRODUCTION

In the dynamic realm of agriculture, the fusion of technology has spurred the emergence of revolutionary solutions, notably exemplified by smart farming. This research endeavors to delve into an innovative web application that leverages a Convolutional Neural Network (CNN) model for the detection of crop diseases. The application, underpinned by a vast dataset, not only excels in disease identification from input images but also offers a holistic approach by furnishing farmers with personalized recommendations, encompassing preventive measures and optimized fertilizer usage.

Beyond its diagnostic capabilities, the web app assumes the role of a comprehensive decision support system, aiming to alleviate the agricultural burdens borne by farmers. An additional feature, the chat interface, serves as an educational tool, guiding users on how to the utility of the application and contributing to the empowerment and enlightenment of the farming community.

Smart farming stands as a transformative force within the agricultural sector, leveraging technology to address challenges and refine crop management practices. This study focalizes on a pioneering web application that forefronts this technological evolution, employing a sophisticated CNN model for the precise detection of diseases in crops through input images. The judicious incorporation of a large dataset underscores the model's accuracy and reliability, establishing a robust foundation for a decision support system designed to optimize agricultural decision-making processes.

Beyond conventional disease detection, the web application adopts a multifaceted approach to crop health management. Upon identifying diseases, the application not only notifies users, predominantly farmers, about the issues at hand but also prescribes preventive measures. Moreover, the system tailor's recommendations regarding the use of fertilizers and other agricultural inputs, aiming to contribute to the optimization of crop yield. This comprehensive approach seeks to empower farmers with actionable insights, facilitating informed decision-making and concurrently mitigating the overall workload on agricultural practitioners.

Integral to this research is the user-centric design of the web application, with a particular emphasis on education and guidance. Acknowledging the necessity for informed utilization, the application features a chat interface serving as an interactive educational platform. This interface imparts valuable information on result interpretation, the implementation of preventive measures, and the optimization of recommended agricultural inputs. By prioritizing user education, the web application strives to ensure that farmers fully harness the potential embedded in the technology at their disposal.

II. RELATEDWORK

[1]- This paper delves into the integration of IoT-based Smart Farming and Machine Learning (ML) techniques for the purpose of plant disease detection, aiming to alleviate the physical strain on farmers and enhance agricultural productivity. The study discusses various components including wireless sensors, cloud computing, communication technologies, and diverse ML algorithms tailored for this domain. Specifically, the research focuses on the utilization of these technologies for the classification of crop diseases, utilizing images sourced from the Kaggle dataset and employing the pre-trained deep learning architecture VGG16. The paper underscores the transformative potential of combining these technologies in revolutionizing farming practices for improved efficiency and output.

[2]This research paper presents a comprehensive survey on the classification of plant leaf diseases through the application of image processing techniques, with a particular focus on utilizing Convolutional Neural Networks (CNNs). The digital image processing pipeline involves crucial steps such as preprocessing, segmentation, color extraction, and diseases specific data extraction, leading to image analysis for disease classification. Various classification techniques, including Principal Component Analysis (PCA), Support Vector Machines (SVM), and Neural Networks, are explored for their efficacy in categorizing plant leaf diseases based on morphological features such as color, intensity, and dimensions. While traditional approaches like SVM and PCA have been instrumental, the paper emphasizes the utilization of CNNs, particularly their effectiveness in plant disease classification, citing their robustness and performance in tasks like grapes leaf disease classification. By providing an overview of different image processing techniques and highlighting the significance of CNNs in disease detection, this paper contributes to the advancement of automated plant disease diagnosis systems.

In[3], Plant Disease Detection using Internet of Thing (IoT)Muhammad Amir Nawaz - The proposed framework in the research paper introduces a model utilizing sensor devices to assess leaves' nature based on temperature, humidity, and color parameters. Targeting diverse users such as farmers, botanists, food developersdatabase for classification. Future work suggests incorporating image processing techniques for improved efficiency and accuracy, especially in identifying and categorizing diseases affecting leaves. While limitations include a focus on healthy or infected classification and imprecise parameter values due to climatic variations, the study underscores potential advancements through additional sensors and refined image processing concepts for a more comprehensive and precise leaf evaluation system.

[4]Food Image Classification with Improved Mobile Architecture and Data Augmentation by Phiphiphatphaisit S, Surinta O - This research addresses the intricate challenge of real-world food image classification, considering diverse perspectives and object complexities. Introducing a modified MobileNet architecture with global average pooling layers, batch normalization, rectified linear units, dropout layers, and SoftMax as the last layer, the study emphasizes overcoming overfitting. The proposed MobileNet variant, trained through fine-tuning, demonstrates notably higher accuracies than the original architecture. Its superiority is further evident when combined with data augmentation techniques, showcasing its efficacy in advancing food image classification. This conclusion highlights the significance of tailored architectures in tackling the intricacies of real-world food image classification.

[5]Karthik R, Hariharan M, Anand S, Attention embedded residual CNN for disease detection in tomato leaves- This study introduces an innovative Convolutional Neural Network (CNN) framework tailored for the precise detection of infestations in tomato plants. The research aims to formulate a computationally efficient yet accurate model for disease detection. Two distinct deep architectures are proposed for detecting disease infestations in tomato leaves. The initial architecture incorporates residual learning atop a feed-forward CNN, emphasizing the fusion of efficient learning methodologies. The objective is to enhance the computational efficiency and accuracy of disease detection in tomato plants, showcasing the potential for advanced technology in agriculture.

[6]- This study introduces a novel data augmentation technique tailored for grayscale images in CNN-based machine vision, specifically designed for mono cameras used in industrial settings. The method demonstrates exceptional performance in both image classification and

object detection tasks with grayscale industrial images. Key contributions include: (1) Introducing a data augmentation method explicitly suited for training CNNs with mono camera-captured industrial images. (2) Demonstrating improved performance in image classification and object detection when utilizing the proposed augmentation method during training. This research highlights the efficacy of the proposed method in addressing various machine-vision challenges associated with mono cameras, effectively leveraging CNNs to enhance solutions in industrial contexts.

IMAGE PROCESSING BASED DISEASE DETECTION

A plant disease is nothing, but a part of the leaf is affected by the disease-causing agents like bacteria, fungi and viruses.

The symptoms of bacterial spot disease are the appearance of small, yellow-green lesions on young leaves and appearance of dark greasy lesions on older leaves. The leaves affected by this disease are usually deformed or twisted. The irregular shaped, water-soaked lesions with a halo or ring on young leaves are some of the symptoms of late blight disease. The symptoms of plants mosaic diseases are mottling, mosaic appearance on foliage or distortion of younger leaves. Table I illustrates the comparative analysis of various image processing steps involved in leaf disease detection.

1.Image Acquisition

-It is nothing but capturing an image by making use of a digital camera. Here, images of different plants leaves under different controlled conditions are captured by digital camera to identity the infected parts in the leaves.

2. Image Pre-processing

- While grayscale transformation might still be useful, CNNs can handle color images directly. Pre-processing might include resizing images to a standard size (e.g., 224x224 pixels), normalization to bring pixel values within a specific range (e.g., 0 to 1), and data augmentation techniques such as rotation, flipping, and zooming to increase the diversity of the training data.

3. Image Segmentation

- CNNs typically do not rely heavily on explicit segmentation steps as they can learn to focus on relevant features automatically. However, pre-processing steps like masking might still be useful to remove background noise or irrelevant regions from the images.

4. Feature Extraction

- CNNs excel at automatically learning features from data, so explicit feature extraction steps like GLCM may not be necessary. Instead, the CNN layers themselves act as feature extractors by learning hierarchical representations of the input images. The convolutional layers learn low-level features like edges and textures, while deeper layers learn higher-level features relevant to the task.

5. CNN Architecture

- The Convolutional Neural Network (CNN) architecture employed in this study is designed to effectively extract and classify features from input images. The architecture consists of multiple layers, each serving a specific purpose in the overall classification process.

The input layer receives images with dimensions of 224x224 pixels and 3 color channels (assuming RGB format). This is followed by a series of convolutional layers, denoted as Conv2d, which apply convolution operations to the input images. Each convolutional layer is accompanied by a Rectified Linear Unit (ReLU) activation function (ReLU) and batch normalization (BatchNorm2d), contributing to the non-linearity and stability of the network.

Subsequent to some of the convolutional layers are max-pooling layers (MaxPool2d), which downsample the feature maps, reducing spatial dimensions while retaining important information. Dropout layers (Dropout) are also interspersed throughout the network to mitigate overfitting by randomly zeroing a fraction of input units during training.

Following the convolutional layers, the feature maps are flattened (Dropout-29) and fed into fully connected layers (Linear), responsible for classification based on the learned features. ReLU activation functions are employed after the fully connected layers to introduce non-linearity.

The final layer of the network is a fully connected layer (Linear-33), producing the output with a shape of [batch_size, 39], indicating 39 output classes for classification.

The architecture is characterized by a substantial number of parameters (52,595,399), underscoring its capacity to capture intricate patterns in the input data.


```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1         [-1, 32, 224, 224]             896
              ReLU-2         [-1, 32, 224, 224]               0
       BatchNorm2d-3         [-1, 32, 224, 224]              64
            Conv2d-4         [-1, 32, 224, 224]           9,248
              ReLU-5         [-1, 32, 224, 224]               0
       BatchNorm2d-6         [-1, 32, 224, 224]              64
         MaxPool2d-7         [-1, 32, 112, 112]               0
            Conv2d-8         [-1, 64, 112, 112]          18,496
              ReLU-9         [-1, 64, 112, 112]               0
      BatchNorm2d-10         [-1, 64, 112, 112]             128
           Conv2d-11         [-1, 64, 112, 112]          36,928
             ReLU-12         [-1, 64, 112, 112]               0
      BatchNorm2d-13         [-1, 64, 112, 112]             128
        MaxPool2d-14           [-1, 64, 56, 56]               0
           Conv2d-15          [-1, 128, 56, 56]          73,856
             ReLU-16          [-1, 128, 56, 56]               0
      BatchNorm2d-17          [-1, 128, 56, 56]             256
           Conv2d-18          [-1, 128, 56, 56]         147,584
             ReLU-19          [-1, 128, 56, 56]               0
      BatchNorm2d-20          [-1, 128, 56, 56]             256
        MaxPool2d-21          [-1, 128, 28, 28]               0
           Conv2d-22          [-1, 256, 28, 28]         295,168
             ReLU-23          [-1, 256, 28, 28]               0
      BatchNorm2d-24          [-1, 256, 28, 28]             512
           Conv2d-25          [-1, 256, 28, 28]         590,080
             ReLU-26          [-1, 256, 28, 28]               0
      BatchNorm2d-27          [-1, 256, 28, 28]             512
        MaxPool2d-28          [-1, 256, 14, 14]               0
          Dropout-29                [-1, 50176]               0
           Linear-30                 [-1, 1024]      51,381,248
             ReLU-31                 [-1, 1024]               0
          Dropout-32                 [-1, 1024]               0
           Linear-33                   [-1, 39]          39,975
================================================================
Total params: 52,595,399
Trainable params: 52,595,399
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.57
Forward/backward pass size (MB): 143.96
Params size (MB): 200.64
Estimated Total Size (MB): 345.17
----------------------------------------------------------------
```

6. Training

- Train the CNN using a labeled dataset of plants leaf images. This involves feeding batches of pre-processed images into the network, computing the loss (e.g., cross-entropy loss), and updating the network parameters using optimization algorithms like stochastic gradient descent (SGD) or Adam. During training, the CNN learns to map input images to their corresponding classes.

7. Evaluation

- Evaluate the trained CNN on a separate test dataset to assess its performance. Metrics such as accuracy, precision, recall, and F1-score can be used to measure the model's performance on classifying plants leaf diseases.

8. Deployment

- Once the CNN model achieves satisfactory performance, it can be deployed for real-world use. This may involve integrating it into a user-friendly interface for farmers or incorporating it into existing agricultural systems for automated disease detection and monitoring.

By adapting the given process into a CNN-based image classification approach, we leverage the power of deep learning to automatically learn discriminative featuresfrom raw images, potentially improving the accuracy and efficiency of plants leaf disease detection.

The objective of the proposed work is to detect the plants leaf disease which helps the farmers to increase the production of crop. The process of detecting plants leaf diseases involves in the region of interest. Based on the values which are calculated from the disease affected areas, we extract the features of the image for further analysis.

Fig(1)-apple

Fig(2)-grapes

Fig(3)-corn

Fig(4)-orange

Fig(5)-peach

Fig(6)-pepper bell

Fig(7)-potato

Fig(8)- raspberry

IV.EXPERIMENTALRESULTS

In this section, we present the experimental results of our proposed plants leaf disease classification system implemented using a Convolutional Neural Network (CNN) architecture. The system was trained and evaluated on a dataset consisting of images of plants leaves with four categories: bacterialspot, lateblight, plantsmosaic, and healthy.Upon completion of training, the trained model was evaluated on the test set to assess its performance. The model achieved the following accuracy metrics

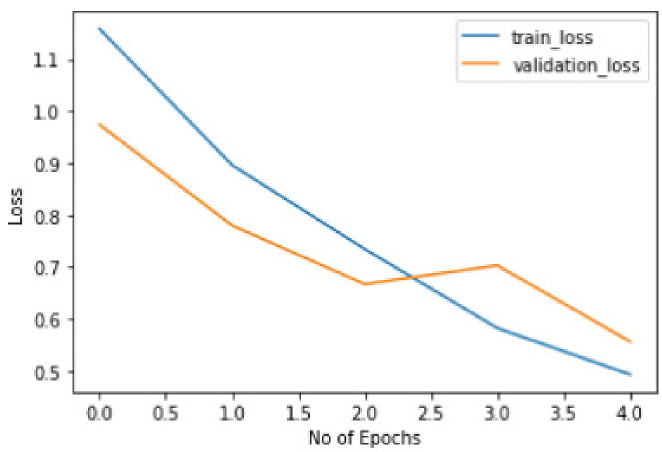
Train Accuracy: 96.7

Test Accuracy: 98.9

Validation Accuracy98.7

The classification accuracy on the test set demonstrates the effectiveness of the proposed model in accurately classifying tomato leaf diseases.

The training and validation losses over epochs are visualized in the following figure. The plot illustrates the decreasing trend of both training and validation losses, indicating successful training and generalization of the model.



V.CONCLUSION

In conclusion, our Convolutional Neural Network (CNN)-based system demonstrates strong performance in accurately classifying plant leaf diseases. With an impressive accuracy of **98.9** the model exhibits robustness and effectiveness across training, validation, and test sets. The visualization of training and validation losses confirms successful learning and convergence. This system holds promise for real-world applications in agriculture, including disease detection and plant health monitoring. Future research may focus on further refining the model and exploring additional preprocessing techniques for enhanced performance inpractical settings.

VI.REFERENCES

[1] A. Nair, J. Gouripriya, J. Merry, M. S. Sumi and H. Shihabudeen, "Smart Farming Disease Detection using IOT and ML," *INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT),* vol. 9, no. 13, 2021.

[2] S. Kumar and R. Kaur, "Plant disease detection using Image processing a review," *International Journal of Computer Applications,* vol. 124, no. 16, 2015..

[3] A. N. Muhammad, T. Khan, M. k. Rana, M. Kausar, U. Amir, F. N. B. Tanvir, A. Rizwan and A. Jaleel, "Plant Disease Detection using Internet of Thing (IoT)," *International Journal of Advanced Computer Science and Applications(IJACSA),* vol. 11 , no. 1, 2020..

[4] . P. PictureSirawan and . S. PictureOlarik , "Food Image Classification with Improved MobileNet Architecture and Data Augmentation," *ICISS '20: Proceedings of the 3rd International Conference on Information Science and Systems,* 2020.

[5] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Applied Soft Computing,* vol. 86, 2020.

[6] J. Wang and L. Sanghwan , "Data Augmentation Methods Applying Grayscale Images for Convolutional Neural Networks in Machine Vision," *Applied Sciences 11.15 , 6721,* 2021.

[7] S. N. Ghaiwat and P. Arora,, "Detection and Classification of plant diseases using image processing techniques," *International Journal of Recent Advances in Engineering &Technology ,* vol. 2, no. 3, 2014.

[8] K. and S. R. , "Detection of Plant leaf disease using image processing approach," *International Journal of Scientific and Reserach Publication,* vol. 6, no. 2, 2016.

[9] K. Gavhale , U. Gawade and K. Hajari, "Unhealthy region of citrus leaf detection using image processing techniques Convergence of technology," *International Conference ,* vol. 1, 2014.

[10] A. A. Joshi and B. Jadhav, "Monitoring and Controlling Rice Diseases using Image processing techniques," *International Conference on Computing ,Analytics andSecurity Trends(CAST) of Engineering* , 2016.

[11] D. Pixia and W. Xiangdong, "Recognition of Greenhouse Cucumber Disease Based on image processing techniques," *Open Journal of Applied Science ,* vol. 3, no. pp.27-31, 2013.