Enhanced Plant Disease Detection with Accurate Pesticide Recommendation and Automated mixer for Plant Health

Abha Marathe 1,Pranit Chilbule 2,Aditya Adaki 2,Aabha Lokhande2,Kush Bhakkad 2.

1.Department of Electronics and Telecommunication Engineering.

2.Department of CSE - Artificial Intelligence & Machine Learning.

Vishwakarma Institute of Technology, Pune, 411037, Maharashtra, India.

**Abstract—** Plant diseases significantly affect crop yields and overall agricultural productivity, posing a threat to global food security. This research addresses the critical issue of early disease detection in five economically vital crops: wheat, rice, cotton, sugarcane, and maize. Early detection is essential for sustainable agriculture, and our study proposes a transformative approach leveraging advanced methodologies. By integrating machine learning, remote sensing, and molecular diagnostics, we enhance the precision and effectiveness of disease diagnosis. Our innovative strategy builds on existing literature, which underscores the potential of these technologies to improve agricultural outcomes. Preliminary findings from our data-driven analyses show promising results, with the identification of key biomarkers and the development of robust early detection models. Specifically, our study employs a convolutional neural network (CNN) model connected to a graphical user interface (GUI) for image-based disease identification. This model not only predicts the disease but also recommends appropriate pesticides and maintenance strategies to prevent disease spread. Moreover, the hardware component of our project precisely formulates the necessary pesticide mixtures based on disease severity and affected area, optimizing treatment and reducing environmental impact. This approach, supported by literature on targeted pesticide application, minimizes reliance on broad-spectrum treatments and promotes sustainable agricultural practices. In summary, our research bridges cutting-edge technology and agriculture, paving the way for sustainable crop production and global food security by addressing plant disease management through innovative, data-driven solutions and environmentally conscious practices.

**Keywords** — Crop disease detection, Convolution neural networks, Deep learning, fertilizer, agriculture, pesticide.

**1. INTRODUCTION**

Plant diseases present a formidable obstacle to global agriculture, wreaking havoc on both yield quantity and quality. According to recent estimates, plant diseases are responsible for significant losses, accounting for an average reduction of 42% in the production of the six most important food crops globally​ ([MDPI](https://www.mdpi.com/journal/agriculture/special_issues/plant_disease))​​ ([BioMed Central](https://cabiagbio.biomedcentral.com/articles/10.1186/s43170-021-00042-x))​. This detrimental impact underscores the critical need for timely intervention. Traditional detection methods, which often rely on manual inspection or rudimentary algorithms, lack the accuracy and speed needed for effective intervention. Manual inspection, while thorough, is labor-intensive and prone to human error​ ([BioMed Central](https://onehealthoutlook.biomedcentral.com/articles/10.1186/s42522-021-00038-7))​. Rule-based algorithms, although potentially faster, lack the adaptability to handle the wide range of disease signatures and environmental factors.

Our research addresses these limitations by harnessing the power of deep learning combined with advanced hardware integration to revolutionize crop disease identification and treatment.

Our proposed methodology hinges on two key components:

1. **Deep Learning Models**: The foundation of our system is built upon convolutional neural networks (CNNs) and MobileNetV2. These models are trained on extensive datasets of labeled images, enabling them to learn intricate patterns associated with various disease types. This learning capability allows for accurate classification of plant diseases into multiple categories, providing a detailed understanding of the specific ailments affecting the crops([mdpi](https://www.mdpi.com/2079-9292/11/14/2110)).
2. **Hardware Integration**: Beyond detection, our approach integrates a hardware model that dynamically adjusts pesticide compositions based on the identified disease. This dynamic system eliminates the need for broad-spectrum treatments, reducing reliance on harmful chemicals and promoting targeted interventions. Optimizing treatment in this manner not only enhances efficacy but also minimizes environmental impact, contributing to sustainable agricultural practices​ ([BioMed Central](https://bmcplantbiol.biomedcentral.com/articles/10.1186/s12870-023-04626-9))​.

In summary, our research bridges the gap between cutting-edge technology and agriculture, paving the way for sustainable crop production and global food security. By leveraging the combined strengths of deep learning and hardware integration, we aim to provide a robust solution to the pervasive issue of plant diseases, ensuring both higher yields and better quality of crops, ultimately supporting global food security initiatives​ ([Frontiers](https://www.frontiersin.org/articles/10.3389/fsufs.2020.00001/full))​.

**2. LITERATURE REVIEW**

[1] The design and development of a liquid mixing and filling system managed by a programmable logic controller (PLC) is described in detail in this study. The three primary components of the system are the conveyor, filling, and mixing sections. DC motors, solenoid valves, and proximity sensors are used by a PLC to control each of these operations. Ladder logic programming is used in the design of the PLC program, and a Delta PLC is used to oversee the entire procedure. Motor functions are automated using relays, and accuracy is maintained by the use of proximity sensors. The PLC ladder diagram, an explanation of the parts, the experimental configuration, and a study of related systems in the literature are all included in the paper.In order to automatically mix and fill bottles in accordance with user specifications, this system can monitor and control factors including temperature, liquid level, quantity, bottle presence, and system speed. The study also includes reference to pertinent publications in the field, emphasizing how well the system monitors and automates the mixing and filling operations. It is mentioned that the system might not be able to meet more general automation needs because it is restricted to liquid-based applications.[2]The creation of a Smart Water Tank System (SWaTS) that uses a WiFi-based microcontroller unit to automate water level monitoring and control is covered in the article. The urgent demand for efficient water management in the Philippines is met by this method. Relays, LCD I2C displays for real-time tank water level monitoring, contactless water sensors that work with S-24V supplies, and Node MCU V3s for processing and communication are all used in the SWaTS prototype. The Arduino IDE is utilized in the development of the user interface, which is integrated with external applications such as Telegram to facilitate communication through automated message delivery. According to the investigation, the prototype greatly lowers water waste and stops tank overflow. Along with hardware specifications and design concerns, the system's design and implementation are described in detail.The page also offers a comprehensive list of references that address a wide range of pertinent subjects, including home automation, Arduino, software engineering, water supply management, and research methodology. Although the SWaTS reduces waste and improves water management, its usefulness in places with inadequate network infrastructure may be limited due to its reliance on WiFi access. [3]This study investigates the use of deep learning methods for plant disease classification and detection, such as image processing and hyper-spectral imaging. According to research, these methods may successfully diagnose and categorize a wide range of plant diseases with up to 98% accuracy. Hyper-spectral imaging is emphasized as a possible technique for early, asymptomatic illness detection. In order to facilitate the classification of several illnesses on a single leaf, the study focuses on the separation of plant leaves into subparts. It covers around 79 diseases in 14 species. When the GoogleNet model was used to identify diseases, the accuracy was 94%, greater than when the full leaf image was used (82%).The study shows good disease diagnosis accuracy, but it doesn't go into great detail about the difficulties in real-world application or how resource-intensive these techniques might be. [4] various machine learning techniques used for detecting plant diseases, a crucial task for improving crop yields in agriculture-dependent economies. The authors describe a comprehensive system for detecting plant diseases that goes through multiple steps: collecting high-quality plant photos, annotating datasets, processing photos to improve their features, extracting pertinent features like color, shape, and texture, and lastly using machine learning techniques to classify the diseases. Support Vector Machine (SVM), Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), Fuzzy Classifiers, and Convolutional Neural Networks (CNN) are among the classification methods that are compared in the survey. CNNs are praised for their exceptional precision and capacity to identify numerous diseases in a range of crops. In spite of SVM's widespread use, the paper concludes that CNNs perform better in terms of accuracy and disease detection capacity. It implies that additional machine learning methods, like decision trees and Naïve Bayes classifiers, could be explored in future studies to further enhance plant disease detection systems. [5] A method for detecting and classifying grape leaf diseases using Support Vector Machine (SVM) classifiers. Grapes, a major fruit crop in India, suffer significant yield losses (10-30%) due to diseases like Powdery Mildew, Downy Mildew, and Anthracnose. Traditional visual inspection methods are often inaccurate and time-consuming, prompting the need for automated solutions. The proposed system employs digital image processing, starting with image acquisition of grape leaves, followed by preprocessing steps such as resizing, thresholding, and Gaussian filtering to enhance image quality. Segmentation is performed using K-means clustering to identify diseased regions. Both color and texture features are then extracted from these segments, providing 54 feature values per image. The SVM classifier uses these features to categorize the leaf diseases. Tested on a dataset of grape leaf images, the system achieved an overall accuracy of 88.89%, demonstrating its potential as a fast, accurate, and cost-effective tool for aiding farmers in early disease detection and management. Additionally, the system's robustness across different environmental conditions and its scalability for large-scale implementation make it a promising solution for the agricultural sector. The research highlights the importance of integrating technology in agriculture to reduce dependency on manual labor and to increase precision in disease management. [6] The leaf and disease spot regions are segmented using the Sobel operator, HSI color system, and Otsu method. The illness's severity is determined by dividing the number of disease spots by the total area of leaves. The technique is renowned for being quicker, more accurate, and more objective than conventional grading techniques. The paper also offers references for more information on image processing and analysis and covers a variety of threshold selection techniques in image processing, including Otsu's method. It is better at quick and objective disease severity measurement and utilization of advanced image processing techniques, but may require specialized knowledge for implementation and potential sensitivity to image quality. The segmentation process is crucial in isolating the diseased areas, ensuring precise measurement of the affected regions. Moreover, the use of the HSI color system enhances the differentiation between healthy and diseased tissue by leveraging hue, saturation, and intensity values. The Sobel operator is instrumental in edge detection, facilitating the accurate identification of disease boundaries.[7] It covers all the steps involved in the process, including picture pre-processing, segmentation, and feature extraction, and highlights the usage of the K-Nearest Neighbor (KNN) classifier for illness identification. The results of the study demonstrate a high degree of accuracy in disease prediction when compared to other currently used methodologies. The paper also contains recommendations for further research, a thorough methodology, and a review of the literature. It is better with high accuracy in disease prediction, with an emphasis on the K-Nearest Neighbor classifier, but it may face challenges with scalability, particularly for a large dataset. The preprocessing steps involve noise reduction and normalization to improve the quality of the input images, ensuring that the subsequent segmentation accurately isolates the affected areas. Feature extraction focuses on capturing critical attributes such as color, texture, and shape, which are essential for the effective classification of diseases. The use of the KNN classifier is notable for its simplicity and effectiveness, making it a suitable choice for this application. The integration of this system into practical agricultural practices could provide farmers with a reliable and efficient tool for early disease detection, ultimately enhancing crop management and yield. Further exploration could include the use of deep learning models to improve scalability and accuracy, and the development of user-friendly interfaces for broader accessibility.[8]By introducing a deep Convolutional Neural Network (CNN) architecture for image classification, Krizhevsky, Sutskever, and Hinton (Year) surpassed prior state-of-the-art values, reaching top-1 and top-5 error rates of 37.5% and 17.0% on the ILSVRC-2010 dataset. Using 60 million parameters and 650,000 neurons, their eight-layer network was trained on the ImageNet dataset, which included over 15 million tagged images in 22,000 categories. Five of the layers were convolutional, while three were fully-connected. Advanced methods like dropout, overlapping pooling, local response normalization, ReLU nonlinearity, and data augmentation were used in the study to improve performance and prevent overfitting. By showing notable gains in image recognition tasks, the researchers emphasized the significance of big datasets and effective GPU implementation.They also talked about how training such big networks requires a lot of resources, and how unsupervised pre-training could boost productivity. This paper emphasizes how important network depth, sophisticated training techniques, and thorough evaluations are to the advancement of computer vision.[9]A convolutional neural network (CNN) for plant disease identification was built by Shrestha, Deepsikha, Das, and Dey (Year), and it achieved 88.80% accuracy on the test set. Preprocessing the photos, separating the datasets and labels, and saving the labeled data in pickle files for training were all part of the methodology. Convolutional, max-pooling, and fully connected layers were all part of the model architecture. To avoid overfitting, dropout layers were inserted after the first (p=0.25), third (p=0.25), and fifth (p=0.5) layers. Using softmax activation on the output layer, the model generated probabilities for 15 labels and utilized the Adam optimizer to forecast diseases. The study highlighted the model's uses in crop management support for farmers and home plant monitoring, showcasing its versatility and room for enhancement. The suggested strategy demonstrated higher accuracy when compared to current techniques, highlighting the value of CNNs and dropout layers in the identification of plant diseases.[10]The effectiveness of deep learning models, in particular convolutional neural networks (CNNs), for plant disease detection and classification was examined by Wasswa Shafk et al. (Year). Their study used the PlantVillage dataset, which contains color, grayscale, and segmented photos of healthy and diseased plants, to test models such as ResNet50, GoogleNet, AlexNet, and ResNet18. They used strategies such as fine-tuning and transfer learning to overcome issues such as complex feature extraction and overfitting in order to improve performance. When PDDNet models were introduced, they outperformed conventional CNNs in terms of resilience and generalization. The study also included remote sensing and multicriteria decision-making for grape farming, demonstrating the usefulness of computer vision in plant pathology. The study demonstrated these models' excellent rates of accuracy in identifying plant diseases, highlighting its potential for sustained agriculture.

**3. METHODOLOGY**

**1.SYSTEM ARCHITECTURE**.

Plant disease identification and curing are the main goals of our study. The application of neural networks (NNs) and computer vision in the trained deep learning model plays a key role in identifying and categorizing images into predefined disease classes, such as bacterial sight, common rust, and blast. The trained model enables us to extract meaningful information from images of the plant causing the disease and automatically recognize various plant diseases which is an integral part .whereas the hardware of the project helps us to generate the pesticide of the particular disease identified by the CV part of the project also it help us to spray pesticide to the farm, the complete working of the project is described in figure 1 of the paper .OpenCV's efficiency in image identification further enhances the capabilities of your toolkit, ensuring quick and effective responses to diverse images. In this specific project, the primary aim is to project plant disease types using a convolutional neural network (CNN)and TensorFlow,NumPy are essential Python libraries that are utilized for feature extraction and data processing of the images . The collaborative integration of these tools forms a robust framework for addressing the challenges of plant disease identification. We had compared various deep learning models for comparison for prediction of disease but we had got an accurate CNN deep learning model.

**2. SOFTWARE**

**2.1.Software tools used in the project**.

* Google Colab -this is used for training the model.
* Python – the language used for coding training     and deploying models  in projects.
* VS Code- IDE used for deploying the project and developing of GUI.

**2.2 .Convolutional neural network (cnn)**

It is a type of neural network commonly used for image recognition and classification tasks. A deep learning model is a machine learning model that uses multiple layers of artificial neural networks for learning and prediction. CNNs are deep learning models designed specifically for computer vision and have made significant progress in this field. They use convolution techniques, pooling, and full layers to extract important features from the image. CNNs are widely used in computer operations such as object recognition, image classification and image recognition.

**2.3. Mobilenet-v2**

It is a lightweight, open-source neural network architecture designed for drawing and manual vision. It is a sequel to the original mobile phone but improved and works better. MNv2 reduces the work required in the convolution process by using a different level of variation (separation filter and channel filter). MNv2 also adds additional layers, including batch normalization and point-wise convolution, to improve accuracy and performance. MNv2 is widely used in many different applications that use powerful and important functions such as classification, object detection and image classification.

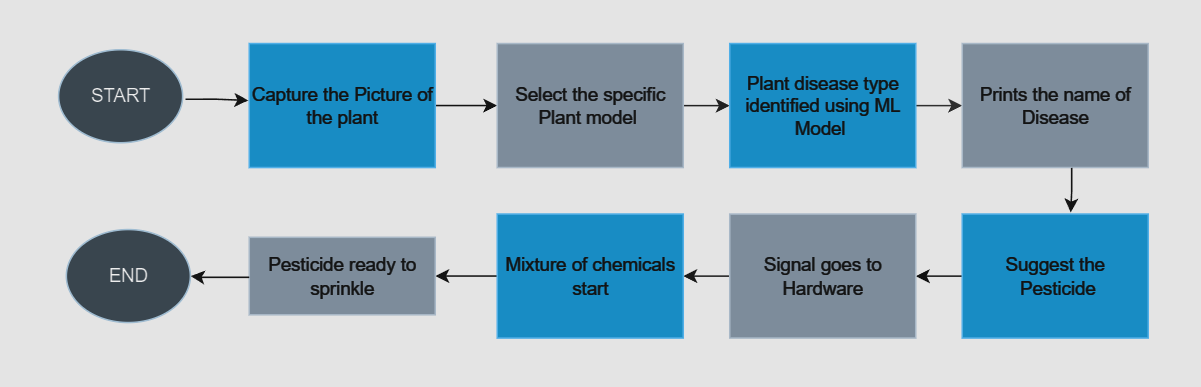


Figure 1:Flow Chart showing project general functioning.

**3)Hardware**

Our project's hardware is essential to the real-world application of our machine learning-based pesticide spraying and plant disease prediction system. It consists of a number of essential elements aimed to automate and enhance the process of identifying diseases and applying pesticides in agricultural areas.Figure 2 describes the complete hardware setup of the project.

1. Mixture system: Our hardware setup depends upon a novel valve system that regulates the flow and application of the pesticide mixture to the plants. This system is made up of carefully positioned electronically operated valves spaced across a series of pipes or tubes. By using these remotely operated valves, the pesticide combination may be sprayed precisely and intentionally into the field's impacted areas.

2. Pesticide Mixing Unit: A pesticide mixing equipment that precisely mixes different chemicals and substances to make the necessary pesticide mixture is located next to the valve system. The sensors in this device detect and manage the concentration of each ingredient, ensuring accuracy and effectiveness in the treatment of pests.

3. Mechanism of Spraying: A spraying device that is attached to the valve system is in the role of spraying the pesticide mixture on the plants. Depending on the particular needs of the crop and disease being targeted, this mechanism can be set up to distribute the pesticide in different forms, such as mist or droplets.

4. Control Panel: We have created a simple control interface to enable seamless use and tracking of the hardware components. Agricultural workers or farmers may enter criteria including the kind of crop, disease signs, and desired pesticide concentration using this interface. Additionally, it gives instantaneous feedback on the condition of the hardware parts and the spraying process's advancement, allowing for any necessary modifications and interventions in a timely manner.

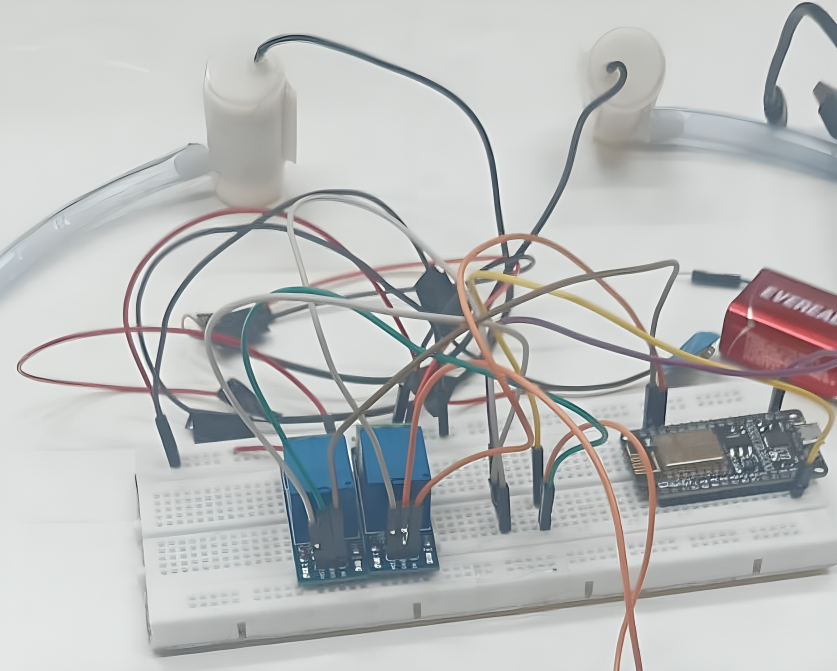


Figure 2: Section of the project involving work with hardware.

**3)Work flow of deployment of model**

1. Importing Required Libraries: imported the necessary libraries, such as Keras, NumPy, and TensorFlow on Google Collab which helped in the pre-processing of images and training of CNN models.
2. Data Loading and Pre-processing: Pre-processing involves setting parameters such as batch size, image height, and width. Using TensorFlow, To evaluate the model's performance, the dataset was divided into 80% for training and 20% for validation. An additional test set, also comprising 20% of the original dataset, was used to assess the model's final performance. This approach ensures that the model is evaluated on unseen data, providing a reliable measure of its generalization capability.
3. The pre-trained MobileNetV2 model is loaded. Importing of the CNN model MobileNetV2(pre-trained) from the tension flow hub website. Key features have already been learned by training it on a large dataset, which is why we imported it as a base model.
4. Freezing Pre-trained Layers: To freeze the pre-trained layers during training and prevent the layers from being changed. This is done to guarantee that the MobileNetV2 layers remain fixed during training and that only the weights of the final dense layer are changed.
5. Compiling of the Model:Before training specified the model's accuracy metrics, optimizer, and loss function after compiling the model and the training of the model can be started.
6. Training of the model: fit the model for training of data to begin training. In this stage, the model gains the ability to identify features and trends. We have a batch size of 33 and 5 epochs configured. To train the model, five training epochs are utilized. An epoch is a complete cycle of the training dataset. Keep track of the accuracy and loss in training statistics for every batch.
7. Model Evaluation: A separate validation dataset, prepared in the same directory as the training dataset and separated during data pre-processing, was used to assess the model's performance after it had been trained. This helps in understanding how models respond to unknown data. Loss and accuracy are calculated on the validation data set.
8. Testing and Inference: Finally we used a different test dataset to see how well our trained model performs in real-world scenarios and classify the waste.
9. Deployment: After training the model on Google Colab, the next step was to deploy these models for using it in real time. In this study pycharm is used as an IDE for deployment of work.
10. Integrating microcontroller and sensors: after the deployment of the model we used microcontroller and valve to make a mixture of the pesticide..Setting up hardware:In order to regulate the operation of the valves, the
11. The Arduino Uno microcontroller was connected to the hardware and parts of the project were coded. Relays were controlled by relay modules, which made it easier to integrate and operate the valves in the system.
12. Spraying mechanism :Spray nozzles were mounted to a motorized arm and connected to an Arduino Uno microcontroller to create the spraying mechanism. The pesticide mixture was able to pass through the nozzles for targeted plant spraying because the Arduino Uno actuated the relay module, which in turn activated the valves. Based on forecasts of plant diseases, the system was calibrated to modify the direction and intensity of spraying.

**4. RESULT AND ANALYSIS**

**1.Comparative Analysis of Algorithms**

Innovative approaches and integrated technologies have advanced plant disease detection in recent years. With one image per class, Few-Shot Learning (FSL) achieved a median accuracy of 55.5%, demonstrating its superior accuracy over classical fine-tuning. When depth transfer and the Cayley-Klein metric were combined, 100% of tea diseases could be recognized. When it comes to detecting soybean mosaic virus, convolutional neural networks (CNNs), such as the CNN-SVM model, have demonstrated exceptional accuracy. Accurate detection has been achieved through model improvements such as Faster R-CNN and Feature Pyramid Networks (FPN) integration. By using image recognition on smartphones to identify crop diseases and pests, mobile applications help farmers even more.

Research on parallel developments suggests utilizing deep learning and hardware integration to detect diseases and emphasizes hue and saturation features for precise classification. With better metrics than current approaches, CNNs and MobileNetV2 provide precise multi-category illness classification. Hardware models that change pesticide formulations dynamically in response to known diseases facilitate focused interventions and lessen their negative effects on the environment. The GUI Tinker UI for disease prediction and pesticide recommendations is one example of a practical implementation, as is targeted spraying mechanisms driven by Arduino. Water management prototypes demonstrate the environmental advantages of these systems by reducing water waste and preventing tank overflow.

We use many algorithms including CNN, INCEPTION and VGG19 to train the model, but CNN and VGG19 are very good because their accuracies and losses are 20.12 and 68.09 respectively for SVM model and 20.109 and 68 for VGG19 model. We chose to use the CNN algorithm after realizing that all other algorithms lost accuracy and ability to analyze images. This study analyzed various models based on f1 score, ground accuracy, recall, precision, and accuracy in Table 1.

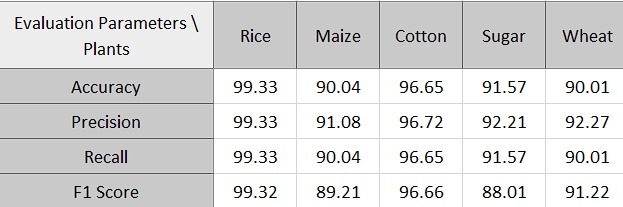


Table 1: comparison of different algorithms.

Since each data set contains many variables and many different data sets, five separate data sets were used for training and classification purposes in this study. The following data were used in this study: four groups of rice disease, five groups of cotton disease, three groups of sugarcane disease, four groups of maize disease, and three groups of wheat disease.CNN model is used in this study since CNN is the most effective model in training, validation and actual testing of the project.

Several crucial parameters were employed by the CNN model created for the task of plant disease detection in order to guarantee successful and efficient training. Using pre-trained features, the base model was MobileNetV2, with its layers frozen. Three color channels (RGB) were used to scale each input image to 224 by 224 pixels. By employing a rescaling layer to shift pixel values from the range [0, 255] to [0, 1], normalization was accomplished. During training, batches of 32 photos each were employed as the batch size. To accurately assess the performance of the model, the dataset was divided into 80% for training and 20% for validation. To effectively update the model weights, the Adam optimizer was used. In order to balance training time and performance, the model underwent 32 epochs of training. Based on recommended procedures for transfer learning and optimizing pre-trained models, these parameters were chosen.

Various tests were used in this study, including the following.

1. Accuracy: Make accurate predictions of all samples. (Total number of guesses / number of correct guesses) is 100% standard.

2.F1-score: This index includes recall and precision, the calculation formula is 2\* (recall\*precision) / (recall

3. It is the pressure-motion ratio defined in machine learning. It is the ratio of correct quality. The prediction and calculation formula for each good prediction is Precision = TP / (TP + FP) where TP represents positive and incorrect FP.

4. Recall rate - In machine learning, the recall rate is expressed as is a metric: it represents the ability of the model to find all relevant events. In short, the return is calculated as: return = true positive / (true positive + negative).

5. Loss: The difference between the actual value and the expected value is shown as loss. A function called loss, usually represented as L(y\_pred, y\_true), calculates the difference between the expected value and the actual value.

Measurements such as TP, TN, FP, and FN are used to evaluate classification problems, especially when dealing with heterogeneous data. These parameters are used to determine the F1 score, accuracy, precision, and return of classification.

True Positive (TP) is the number of true positive predictions.

True Negatives (TN): The number of true negatives correctly predicted for a given event.

False Positive (FP): The number of positive events that were not detected.

False Negatives (FN): The number of negative events that are not as expected.

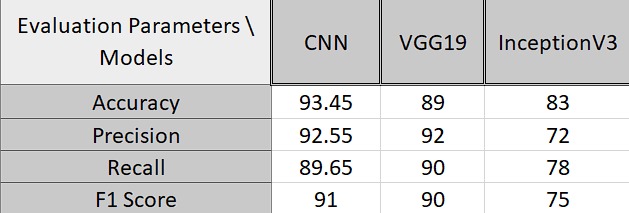
The research prepares accuracy, F1 score, accuracy, recovery and loss results (all these results are shown in Table 2 for comparison of various site data) using TP, TN, FP and FN values ​​.. 

Table 2: Comparing Various Datasets Using CNN Algorithm

The evaluation parameters for a machine learning model trained to detect illnesses in various plant species, including rice, maize, cotton, sugar, and wheat, are shown in the table above. F1-score, recall, accuracy, and precision are among the evaluation measures. We can see from the data that different plant types perform differently. With values of 99.33%, 99.33%, 99.33%, and 99.32%, respectively, the model obtains the greatest accuracy, precision, recall, and F1-score on rice plants. Conversely, sugar plants had the lowest results of any plant species in terms of accuracy, precision, recall, and F1-score, with 91.75%, 92.21%, and 88.01%, respectively.The accuracy of the model ranges from 90.04% to 99.33%, suggesting that it performs well overall across all plant types.Precision, recall, and F1-score also show strong performance, albeit with some variations among plant types. Analytically speaking, the dataset that was utilized to train the model appears to be sufficiently varied to capture variances among various plant species. However, additional research might look into possible biases in the dataset or variables affecting how different plant kinds perform differently in the model.Assessing the model's usefulness in actual situations and its capacity to generalize to new data would be beneficial as well. The machine learning model trained on the rice dataset outperformed other plant species in terms of assessment criteria such as accuracy, precision, recall, and F1-score. Rice got the highest ratings when compared to maize, cotton, sugar, and wheat, indicating that the model was effective in identifying diseases in rice plants.

**2.GUI**

We created the Tinker UI GUI to serve as an intuitive user interface for our machine learning model deployment and to enable interaction between users. We developed a frontend interface that let users enter factors for disease prediction, such as plant symptoms and field conditions, using Python modules like Flask and PyQT. The GUI improved user involvement and streamlined the farm management decision-making process by offering real-time feedback on the prediction findings and recommended pesticide mixes for spraying.

GUI (graphical user interface) that would allow users to interact with the model for prediction and display the output on display. The website/ GUI(tinker UI) has been created in our study to classify real-time diseases. The output produced by the GUI is shown in Figure 3, and 4. It demonstrates how the disease is categorized into different categories and how this is done by properly streaming video and utilizing deep learning libraries like Keras and TensorFlow in conjunction with techniques like OpenCV. The GUI displays the diseases that had impacted the plant, along with the precautions, treatment options, and pesticide concentrations that should be used. Figure 3 shows that the plant that has caused rice blast disease had necessary precautions for it. Figure 4 shows that the plant that has caused red rot disease had necessary precautions for it.

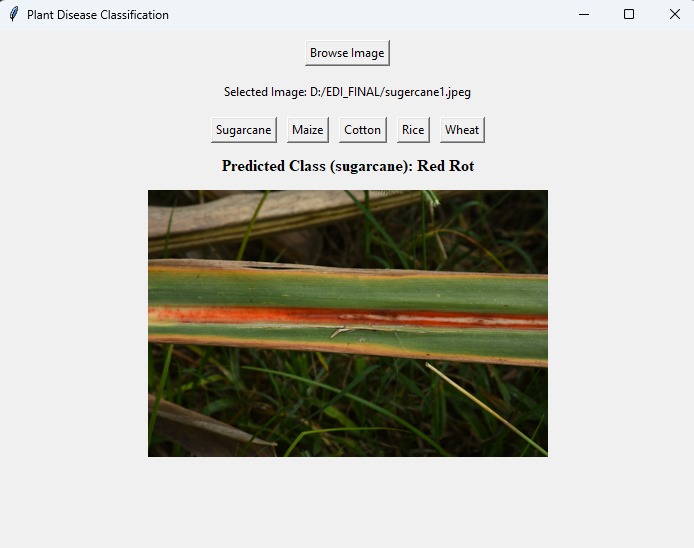
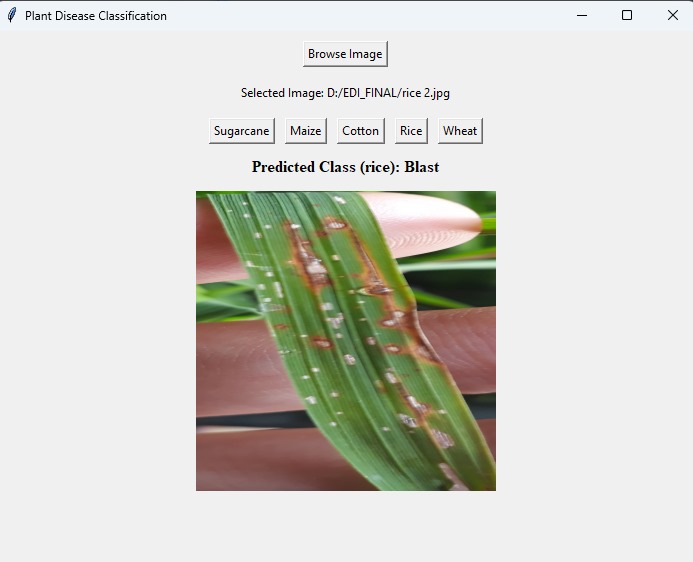


Figure 3: Rice Disease is Classified Figure 4: Sugarcane Disease is Classified.

**5. CONCLUSION & FUTURE SCOPE**

The paper proposes an effective method for crop disease detection and identification using deep learning models and hardware integration. It employs convolutional neural networks (CNNs) and MobileNetV2 to classify images of diseases in five different plants. The hardware model mixes pesticides based on detected diseases and software-provided measurements. Performance evaluation on a test dataset demonstrates superiority in accuracy, precision, recall, and F1-score compared to existing methods. Challenges include data scarcity, image variability, hardware complexity, and ethical/environmental concerns. Future directions suggest advanced models, additional features, improved hardware, and exploring agricultural impacts. The study contributes valuable insights for crop disease detection, addressing urgent issues in maize, cotton, rice, sugarcane, and wheat farming, with potential benefits for food security and sustainable agriculture. Implementation of the proposed solution promises significant advancements in farming practices, benefiting farmers, academics, and policymakers alike.The spraying mechanism is equipped with sensors to monitor environmental conditions such as wind speed and direction, ensuring optimal spraying efficiency and minimizing potential environmental impact.

**6. REFERENCES**

[1] S. Eswar, L. Jaiganesh, N. Hariprasad, M. Mohamedimthiyas and A. Gopikrishnan, "Automatic Liquid Mixing and Filling Using PLC," 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), Coimbatore, India, 2018, pp. 1-6, doi: 10.1109/ICCTCT.2018.8551099.

[2]Raagas, D. P., Melgazo, D. E., Ergina, D. C., & Verecio, R. L. (2022). Smart Water Tank system using WIFI-based microcontroller unit. *South Florida Journal of Development*, *3*(2), 2670–2685. <https://doi.org/10.46932/sfjdv3n2-086>

[3] Li, L., Zhang, S., & Wang, B. (2021). Plant disease detection and classification by Deep Learning—A Review. *IEEE Access*, *9*, 56683–56698. <https://doi.org/10.1109/access.2021.3069646>

[4] U. Shruthi, V. Nagaveni and B. K. Raghavendra, "A Review on Machine Learning Classification Techniques for Plant Disease Detection," 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 2019, pp. 281-284, doi: 10.1109/ICACCS.2019.8728415.

[5] P. B. Padol and A. A. Yadav, "SVM classifier based grape leaf disease detection," 2016 Conference on Advances in Signal Processing (CASP), Pune, India, 2016, pp. 175-179, doi: 10.1109/CASP.2016.7746160.

[6] S. Weizheng, W. Yachun, C. Zhanliang and W. Hongda, "Grading Method of Leaf Spot Disease Based on Image Processing," 2008 International Conference on Computer Science and Software Engineering, Wuhan, China, 2008, pp. 491-494, doi: 10.1109/CSSE.2008.1649.

[7] Shruti et al., "Plant Leaf Disease Detection Using Machine Learning," 2023 1st International Conference on Circuits, Power and Intelligent Systems (CCPS), Bhubaneswar, India, 2023, pp. 1-6, doi: 10.1109/CCPIS59145.2023.10291669.

[8] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017, May 24). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. <https://doi.org/10.1145/3065386>

[9]Dey, Naiwrita. (2020). Plant Disease Detection Using CNN. [10]1109/ASPCON49795.2020.9276722. [10]Shafik, W., Tufail, A., De Silva Liyanage, C. et al. Using transfer learning-based plant disease classification and detection for sustainable agriculture. BMC Plant Biol 24, 136 (2024). <https://doi.org/10.1186/s12870-024-04825-y>

[11] Kamilaris, Andreas & Prenafeta Boldú, Francesc. (2018). A review of the use of convolutional neural networks in agriculture. The Journal of Agricultural Science. 156. 1-11. 10.1017/S0021859618000436.

[12] Qiang, Zhenping & He, Libo & Dai, Fei. (2019). Identification of Plant Leaf Diseases Based on Inception V3 Transfer Learning and Fine-Tuning. 10.1007/978-981-15-1301-5\_10.

[13] Mohanty, Sharada & Hughes, David & Salathe, Marcel. (2016). Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science. 7. 10.3389/fpls.2016.01419.

[14] Bothra, Mushkan & P, Nishitha & Km, Madhu & Ks, Pavithra & Neharaj, & Bs, Divya. (2023). Leaf Disease Detection and Pesticide Recommendation using Deep Learning Algorithm.

[15] D. Al Bashish, M. Braik and S. Bani-Ahmad, "A framework for detection and classification of plant leaf and stem diseases," 2010 International Conference on Signal and Image Processing, Chennai, India, 2010, pp. 113-118, doi: 10.1109/ICSIP.2010.5697452.

[16] Ramesh Maniyath, Shima & P V, Vinod & M, Niveditha & R, Pooja & N, Prasad & N, Shashank & Ram, Hebbar. (2018). Plant Disease Detection Using Machine Learning. 41-45. 10.1109/ICDI3C.2018.00017.

[17] Mitra, Mr.Dipra & Gupta, Shikha. (2022). Plant Disease Identification and its Solution using Machine Learning. 152-157. 10.1109/ICIEM54221.2022.9853136.

[18] Hassan, Sk & Maji, Arnab & Jasiński, Michał & Leonowicz, Zbigniew & Jasińska, Elżbieta. (2021). Identification of Plant-Leaf Diseases Using CNN and Transfer-Learning Approach. Electronics. 10. 1388. 10.3390/electronics10121388.

*An image-processing based algorithm to automatically identify plant disease visual symptoms*. [19](2008, November 18). An Image-processing Based Algorithm to Automatically Identify Plant Disease Visual Symptoms - ScienceDirect. <https://doi.org/10.1016/j.biosystemseng.2008.09.030>