AI-Based Plant Disease Detection Using Drone Imaging and YOLOv8

Adharsh S[1], Shathrugna Dhiru j M[2], R Abhishek[3], Laalu Prasad S K[4], Kasiviswanath N J[5]

[1,2,3,4]Student, School of CSE, Reva University, Bengaluru, India
[5]Assistant Professor, School of CSE, Reva University, Bengaluru, India

Abstract
Agriculture is one of India's key sectors, and the health of the crops directly affects farmers' income. The late identification of the plant disease could have adverse effects on agriculture, resulting in less yield and economic loss. For this project we created a drone based system that splits images of healthy and diseases plant leaves using a deep learning model YOLOv8. In the first stage, we take images of crops using a smartphone camera. The photos are then uploaded to a backend where the model processes them and checks if there is any disease over the crop. This system will also help you in identifying other types of leaf conditions which can be Rust, Virus, Leaf Curl, Shot Hole, Powdery Mildew and Healthy leaves. This data, with an on-line based output using a browser to access the results. An effective solution to such cases for modern agriculture is this method which reduces manual effort and saves time.

Keywords: YOLOv8, Drone, Plant Disease Detection, Deep Learning, Agriculture

INTRODUCTION

Agriculture is a key part of India's economy, and crop health plays an important role in maintaining productivity. One of the major challenges faced by farmers is the early detection of plant diseases, as delayed identification can lead to significant crop loss. Traditionally, disease detection is done through manual observation, which is time-consuming and depends on experience. This method is not efficient for large fields and often leads to late or incorrect diagnosis. With recent advancements in artificial intelligence, especially computer vision, automated disease detection has become possible. Models like YOLOv8 can quickly and accurately identify diseases from images. At the same time, drones make it easier to capture images across large agricultural areas. In this work, a drone-based plant disease detection system is developed by combining UAV technology, an ESP32-based camera setup, and a YOLOv8 model. The system processes images and displays results on a web dashboard, providing a simple and effective solution for real-time crop monitoring.

PROBLEM STATEMEN

Plant diseases are a major factor affecting agricultural productivity, especially in countries like India where farming is a primary source of livelihood. In many cases, diseases are identified only after visible damage appears, by which time a significant portion of the crop may already be affected. The traditional approach to disease detection relies on manual inspection by farmers or agricultural experts. This process is slow, labor-intensive, and often inaccurate due to lack of proper knowledge or experience. For large agricultural fields, it becomes extremely difficult to monitor every plant, increasing the chances of missing early-stage infections. Another major challenge is the limited availability of expert guidance in rural areas. Farmers may not have immediate access to agronomists, leading to delays in diagnosis and treatment. Additionally, there is a lack of real-time monitoring systems that can provide continuous insights into crop health. These challenges highlight the need for an automated, accurate, and scalable solution that can detect plant diseases early, cover large areas efficiently, and provide timely information to support better decision-making in agriculture.

OBJECTIVES

A. Primary Objectives

- To design and build a drone capable of capturing images over agricultural fields
- To develop a motorised camera system using ESP32 for better image coverage
- To train a YOLOv8 model for detecting multiple plant diseases
- To create a backend system for processing images and generating results
- To develop a web-based dashboard for visualizing detection outputs

B. Secondary Objectives

- To reduce the time required for disease detection
- To make the system scalable for different crops and diseases
- To ensure the solution is cost-effective and practical for real-world use
- To provide an easy-to-use interface for farmers and agricultural experts

SYSTEM OVERVIEW

The proposed system is designed to automate plant disease detection by integrating drone technology, embedded systems, and deep learning. It consists of three main components: the drone-based image acquisition system, the processing backend, and the web-based dashboard. The workflow begins with the drone capturing images of crops using a smartphone mounted on a motorised camera system. The camera movement is controlled using an ESP32 microcontroller, allowing better coverage of the field during flight. The captured images are then uploaded to a backend server, where a trained YOLOv8 model processes them to detect and classify plant diseases. The model identifies different disease types and generates outputs such as bounding boxes and confidence scores. These results are sent to a web dashboard, where they are displayed in a clear and user-friendly format. Users can view detected diseases, analyze patterns, and make informed decisions regarding crop treatment. Overall, the system provides a seamless flow from image capture to result visualization, enabling faster and more efficient monitoring of crop health.

Fig. 1 End-to-End System Workflow for Drone-Based Crop Disease Detection

METHODOLOGY

The proposed system follows a structured approach that combines hardware, software, and deep learning techniques to detect plant leaf diseases efficiently. The methodology can be divided into four main stages: image acquisition, data processing, model inference, and result visualization. In the first stage, images of crops are captured using a drone equipped with a smartphone camera.

The camera is mounted on a motorised setup controlled by an ESP32 microcontroller, allowing it to adjust its position and capture a wider area during flight. In the second stage, the captured images are uploaded to the backend system through a web interface. The backend processes the images by resizing and preparing them for analysis. In the third stage, a trained YOLOv8 model is used to detect and classify plant diseases from the images. The model identifies different disease categories and generates outputs such as bounding boxes and confidence scores. In the final stage, the results are displayed on a web-based dashboard. The dashboard provides visual outputs, including annotated images and summary information, which helps users easily understand the condition of crops and take necessary actions. This step-by-step methodology ensures efficient data flow from image capture to final decision-making, making the system practical for real-world agricultural use.

Fig. 2: Proposed System Methodology Flow

RESULTS AND DISCUSSION

The proposed system was evaluated based on detection accuracy, processing time, and overall performance under practical, real-world conditions. The YOLOv8 model was tested using a dataset containing multiple plant disease classes, including Rust, Virus, Curl, Shot Hole, Powdery Mildew, and Healthy leaves. Experimental results indicate that the model performs consistently well across all categories, achieving high precision and recall values. Among the evaluated classes, the Healthy category showed the highest accuracy, suggesting that the model can clearly distinguish between healthy and diseased leaves. However, slight variations in performance were observed for diseases with smaller lesions or visually similar patterns, such as Shot Hole, which can be more challenging to detect accurately. In terms of computational performance, the system demonstrated efficient processing capabilities. When executed on a CPU-based environment, the average processing time was approximately 3–4 seconds per image, which is acceptable for offline analysis. With GPU acceleration, the processing time was significantly reduced, enabling faster inference and making the system more suitable for near real-time applications. This improvement highlights the scalability of the system for deployment in larger agricultural settings. Additionally, the integration of drone-based image acquisition enhanced the system's ability to monitor crops over large areas with minimal manual effort. The captured images were seamlessly transmitted to the processing module, and the results were displayed through a user-friendly web dashboard. The dashboard provided clear visualization of detected diseases along with classification results, making it easier for users to interpret the output. The entire workflow, from image capture to result display, operated smoothly and reliably during testing. Overall, the system demonstrates strong potential for practical use in precision agriculture. Its ability to combine accurate disease detection, efficient processing, and real-time monitoring makes it a valuable tool for early disease identification and crop management. This can ultimately assist farmers in taking timely actions, reducing crop losses, and improving agricultural productivity for use in real agricultural environments.

Disease Class	Precision	Recall	F1 Score	mAP@0.5
Rust	0.91	0.88	0.89	0.91
Virus	0.87	0.85	0.86	0.87
Curl	0.89	0.87	0.88	0.90
Shot Hole	0.85	0.83	0.84	0.85
Powdery Mildew	0.90	0.89	0.89	0.90
Healthy	0.96	0.95	0.95	0.96
Overall Mean	0.89	0.87	0.88	0.90

Table 1: Performance Metrics of YOLOv8 Model

FUTURE WORKS

Although the proposed system shows promising results, several improvements can be made to enhance its performance and usability in real-world scenarios. One important improvement is the implementation of automatic image transfer from the smartphone to the backend during drone flight. This would eliminate the need for manual uploads and enable fully real-time monitoring. The current system supports only horizontal camera movement. In future work, a two-axis gimbal system can be developed to allow both pan and tilt control, improving image coverage and flexibility. Another enhancement is the integration of GPS-based mapping.

By using location data from captured images, the system can generate field maps highlighting disease-affected areas, helping farmers take targeted action. The model can also be extended to detect additional conditions such as pest damage and nutrient deficiencies. Training the system on multi-crop datasets will further increase its applicability across different agricultural environments. Finally, deploying the model directly on mobile devices using lightweight frameworks like TensorFlow Lite can reduce dependency on backend servers and improve system efficiency in remote areas. These improvements will make the system more autonomous, scalable, and suitable for large-scale agricultural deployment.

CONCLUSIONS

The proposed conversational image recognition chatbot demonstrates the effective integration of computer vision and conversational artificial intelligence to enable interactive and accessible image understanding. Unlike conventional image recognition systems that generate static outputs, the proposed system supports context-aware dialogue, thereby significantly enhancing user engagement and interaction. This work validates the feasibility of multimodal systems that combine visual perception with natural language understanding, highlighting their relevance in domains such as accessibility, education, and assistive technologies. Experimental results indicate that the system allows users to explore visual content in an intuitive manner by asking follow-up questions and receiving meaningful, context-driven responses. This interactive approach improves usability and provides a more engaging experience compared to traditional methods. Additionally, the system shows consistent performance in interpreting diverse image inputs, demonstrating its robustness and adaptability across different use cases. From a practical perspective, the chatbot design emphasizes simplicity and accessibility, making it suitable for users with varying levels of technical expertise. The ability to convert visual data into conversational insights can support learning, assist visually impaired individuals, and enhance decision-making in real-world scenarios. Furthermore, the modular architecture of the system allows for easy integration with other applications and platforms, increasing its potential for scalability and future enhancements. The results also highlight the importance of combining vision and language technologies to create more human-centered AI systems. By enabling natural and meaningful interactions, the proposed system bridges the gap between complex machine outputs and user understanding. In conclusion, the developed system presents a promising step toward intelligent, interactive visual assistants. With further advancements in model accuracy, contextual understanding, and real-time processing, the system can be extended to handle more complex queries and diverse datasets. This opens up opportunities for broader adoption and real-world deployment, contributing to the evolution of smart and user-friendly AI-driven solutions.

ACKNOWLEDGMENT

The authors would like to sincerely thank the faculty of the **School of Computing and Artificial Intelligence and Machine Learning, REVA University, Bengaluru**, for their continuous support and guidance throughout the development of this project. Their suggestions, feedback, and encouragement helped us stay focused and motivated at every stage. The knowledge and direction provided by them played an important role in shaping our understanding and successfully completing this work.

The authors also express their gratitude to their mentors and colleagues for their valuable ideas, discussions, and constructive feedback during the design and implementation of the Conversational Image Recognition Chatbot system. Their support helped us overcome challenges, improve our approach, and refine the overall system. The exchange of ideas and teamwork greatly contributed to enhancing both the technical and practical aspects of the project.

We would also like to acknowledge the support of the laboratory staff and technical team for providing access to the necessary tools, equipment, and resources required during the development and testing phases. Their assistance ensured that our work could be carried out smoothly without interruptions.

In addition, we are thankful to our friends and classmates for their encouragement, support, and helpful suggestions throughout this journey. Their involvement made the process more collaborative and engaging.

Finally, the authors would like to thank **REVA University** for providing a positive academic environment, along with the required infrastructure and resources, which made it possible to carry out this project effectively. The support provided by the institution played a key role in enabling us to complete this work successfully.

REFERENCES

1. S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using deep learning for image-based plant disease detection,” Frontiers in Plant Science, vol. 7, 2016.
2. D. Hughes and M. Salathé, “An open access repository of images on plant health to enable the development of mobile disease diagnostics,” arXiv preprint arXiv:1511.08060, 2015.

3. G. Jocher et al., "Ultralytics YOLOv8," 2023. [Online]. Available: https://github.com/ultralytics/ultralytics
4. C. Zhang and J. M. Kovacs, "The application of small unmanned aerial systems for precision agriculture: a review," Precision Agriculture, vol. 13, no. 6, pp. 693–712, 2012.
5. M. D. Bah, A. Hafiane, and R. Canals, "Deep learning with unsupervised data augmentation for weed detection," arXiv preprint arXiv:1812.10259, 2018.
6. K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," Computers and Electronics in Agriculture, vol. 145, pp. 311–318, 2018.
7. Redmon, J., and A. Farhadi, "YOLOv3: An incremental improvement," arXiv:1804.02767, 2018.
8. Bochkovskiy, A., C. Wang, and H. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv:2004.10934, 2020.
9. Tan, M., and Q. Le, "EfficientDet: Scalable and efficient object detection," CVPR, 2020.
10. Espressif Systems, "ESP32 Technical Reference Manual," 2023. [Online]. Available: https://docs.espressif.com
11. Betaflight Development Team, "Betaflight Firmware Documentation," 2023. [Online]. Available: https://betaflight.com
12. Ultralytics, "YOLOv8 Training and Usage Guide," 2023. [Online]. Available: https://docs.ultralytics.com
13. Facebook Open Source, "React.js Documentation," 2023. [Online]. Available: https://react.dev
14. R. Girshick et al., "Region-based convolutional networks for accurate object detection," IEEE TPAMI, 2016.
15. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," CVPR, 2015.