Crop Disease Prediction and Management

Anand Kunal Mishra, Prithvi Singh Kirar and S Prabakeran

February 14, 2024

CROP DISEASE PREDICTION & MANAGEMENT

Anand Kunal Mishra [1]
CSE Networking and Communication
SRM Institute of Science and Technology
Chennai, India
am9233@srmist.edu.in

Prithvi Singh Kirar [1]
CSE Networking and Communication
SRM Institute of Science and Technology
Chennai, India
pk8631@srmist.edu.in

Dr. S. Prabakeran [2]
Assistant Professor - Networking and Communication
SRM Institute of Science and Technology
Chennai, India
prabakes@srmist.edu.in

*Abstract—**Crop diseases have a significant global impact on agriculture, threatening food security. Detecting and managing these diseases early is crucial for safeguarding crop yields. This research paper introduces a holistic framework for crop disease detection and management, combining advanced image analysis, machine learning, and traditional farming practices. Its primary focus is on early disease identification and effective management. The approach utilizes diverse data sources such as field surveys, remote sensing, and mobile apps to identify disease symptoms, assess severity, and recommend management strategies. Case studies involving various crops showcase the framework's potential to improve crop health, promote agricultural sustainability, and enhance global food security.***

In summary, this research paper presents a comprehensive framework for crop disease detection and management, integrating advanced image analysis, machine learning, and traditional farming practices. It addresses the crucial issue of early disease identification and efficient management by utilizing diverse data sources. Case studies illustrate its potential to enhance crop health, agricultural sustainability, and global food security by offering personalized disease management strategies while reducing reliance on broad-spectrum pesticides. This initiative signifies a significant step towards revolutionizing agriculture for a more sustainable food production future.

INTRODUCTION

Crop Disease Prediction and Management (CDPM), often called "symptom sensing," is an innovative method for swiftly detecting disease symptoms in plants by evaluating visible cues. This approach simplifies disease identification, making it accessible and cost-effective for field operators. CDPM allows on-site assessments, facilitating timely intervention and reducing the economic impact on farmers compared to traditional methods involving lab tests and sample transportation.

CDPM technology is adaptable to various disease characteristics in agriculture, enabling early detection and proactive disease management, and safeguarding crop yields and food security. It replaces subjective tests with systematic, objective digital scanning and software analysis, ensuring more accurate results. Converting scanned data into structured qualified personnel and may require additional human and financial resources. Agricultural institutions may need to mobilize field teams or digital formats like ASCII enhances data analysis, historical record-keeping, and standardized tests in agriculture.

The objectivity of CDPM technology supports data-driven decision-making, benefiting agricultural professionals and researchers. In summary, CDPM has revolutionized disease detection and management in agriculture by providing a user-friendly, efficient, and adaptable approach, ultimately contributing to agricultural sustainability and food security.

MOTIVATION AND PROBLEM STATEMENT

In agricultural institutions worldwide, a common practice involves distributing paper-based forms or tools to farmers or field workers for recording crop disease-related observations, forming the basis of disease tracking and management in modern agriculture. However, these traditional methods rely on manual data collection and come with various challenges.

Field operators, including farmers and agricultural workers, visually assess crop fields, noting disease signs and recording data in these forms. This manual process encompasses a wide range of information, including disease types, severity, and affected crop varieties.

Yet, manual data collection is prone to errors, inconsistencies, and variations among individuals, leading to inaccurate or inconsistent data. Correcting such mistakes can be time-consuming and resource-intensive, necessitating review and correction by knowledgeable personnel.

Data rectification is vital to ensure the reliability and accuracy of disease management information. However, this manual process is time and resource-intensive, involving multiple individuals within the agricultural community. It adds complexity and delays to disease management, which can be problematic when timely intervention is crucial to prevent disease spread and minimize crop losses. Moreover, the manual approach relies on the availability of are crafted through the collection of data from agricultural manuals, academic literature, and field experiments. They provide guidance extension services to rectify data, incurring costs and logistical challenges.

In conclusion, the traditional method of using paper-based forms for manual data collection of crop disease-related observations in agriculture is established but fraught with challenges. These challenges include potential errors and the time-consuming process of data rectification, which involves a collective effort and consumes valuable time and resources. In the digital era of data-driven agriculture, there is a growing need to explore more efficient and accurate methods for gathering, managing, and analyzing crop disease-related data to enhance disease management and promote agricultural sustainability.

OBJECTIVE

Our mission aims to revolutionize agriculture through a comprehensive Crop Disease Management and Prediction system. We're committed to simplifying and optimizing disease identification and management. Unlike time-consuming manual methods, our advanced technology and predictive algorithms significantly reduce the effort and time needed for crop health reports. This empowers quick, informed decisions, minimizing disease spread and financial losses.

While our system excels in early detection and timely management, it can't restore already diseased crops. However, its potential applications go beyond saving time. It provides a digital, standardized repository for crop health data, streamlining management, analysis, and sharing. Moreover, it introduces predictive capabilities, allowing proactive disease management, reducing disease occurrences, and promoting agricultural sustainability. Ultimately, our goal is a transformative system for efficient, data-driven, and predictive crop disease management in agriculture.

LITERATURE SURVEY

Crop disease prediction and control are essential for global food security and agricultural sustainability. Researchers and professionals conduct comprehensive literature surveys to gather insights from diverse sources, focusing on key domains.

Firstly, historical and current crop disease records from various sources serve as the foundation for effective disease prediction and control. These data help uncover trends and patterns, guiding preventive measures and early intervention techniques.

Secondly, modern technology, such as remote sensing and satellite imagery, provides real-time insights into environmental conditions. Researchers use data from satellites, drones, and sensors to monitor variables like temperature and humidity, aiding in early disease detection.

Thirdly, machine learning and data analytics enable the creation of accurate predictive models. These models are nurtured through the analysis of historical data, weather patterns, and crop characteristics gathered from academic sources and online databases. Collaborative global research strengthens our understanding of crop diseases.

Lastly, integrated pest management (IPM) strategies encompass various tactics like biological control and chemical treatments. These strategies
summary, image processing optimizes data quality and enhances the reliability of predictions in crop disease management.

for mitigating crop diseases.

In summary, the literature survey on crop disease prediction and management weaves together insights from historical data, remote sensing technology, machine learning models, and IPM strategies. These sources collectively form a comprehensive roadmap for understanding and addressing crop diseases, ensuring agricultural sustainability and food security.

PROPOSED WORK

Problem statement: 1

The primary aim of creating this system was to detect variations in leaf characteristics, which are integral to our final output. We employed various image recognition techniques to achieve this goal. In the subsequent section, I will outline the three essential steps taken in building the entire system.

Detection of Leaf Disease

System Design

Our Python-based system is platform-independent, offering advantages over traditional methods. It accurately predicts materials from trained plant images, corrects image perspectives, and includes tilt correction and spot detection components.

Image Acquisition

We process plant leaf images at a rate exceeding fifty images per minute. These scanned images are analyzed on a computer following a standard capture process illustrated in the diagram.

Design form

Designing a user-friendly form for farmers to upload leaf images for crop disease prediction is essential. The form should have a clear layout with a "Select Image" button for file upload, instructions on image requirements, a text box for notes, and contact details fields. A prominent "Submit" button allows easy submission. Mobile responsiveness and data security measures should be in place to ensure a smooth user experience while protecting privacy. This design simplifies farmer participation in crop disease prediction, enhancing collaboration in agriculture research and management.

Tilt correction

Tilt correction is crucial to ensure accurate analysis of farmers' submitted images. Image processing techniques automatically rectify any tilts or angles, enhancing dataset consistency and disease recognition precision. Standardized orientation improves image quality, resulting in more accurate prediction models and optimized management strategies.

Use image processing for required validation output

Image processing is essential for validating crop disease prediction. It enhances submitted leaf images, identifying critical disease-related features like lesion patterns and discoloration. This step ensures data consistency, is representative of real-world conditions, and aids in quality control by filtering out irrelevant images, improving the system's performance and efficiency. In

Find the spots of interest using the template

Identifying key areas in crop images through predefined templates is a crucial element of crop disease prediction. Templates represent common disease symptoms and guide the system in locating specific regions of interest in the images. By comparing image features to these templates, the system identifies disease-related attributes like lesions and discoloration, aiding in disease severity quantification for accurate prediction and management. This process enhances prediction models, offering insights into disease presence and progression. It empowers farmers to take timely, targeted actions to protect their crops and promote agricultural sustainability.

Spots of interest recognition

Establishing a coordinate system simplified spot recognition in the system, aiming to distinguish between diseased and healthy spots. The critical parameter is the threshold value of 0.75, signifying the minimum acceptable cover rate for a spot to be labeled as diseased. This value was carefully chosen, considering spot characteristics, observed cover rates, and desired classification accuracy. The threshold of 0.75 strikes a balance between sensitivity and specificity, accurately identifying diseased spots while minimizing false positives, enhancing the system's reliability in determining health statuses. This threshold is pivotal in ensuring the accuracy and efficiency of spot recognition.

Algorithm

The Convolutional Neural Network (CNN) is a widely used algorithm for crop disease prediction, specializing in image classification and analysis. CNNs excel in identifying patterns and features in images, making them ideal for diagnosing crop diseases based on leaf images.

This algorithm learns from a large dataset of labeled images, allowing it to recognize distinctive visual characteristics linked to various crop diseases. It segments and analyzes input leaf images, extracting key features like spots, discolorations, and lesions that signify specific diseases. With this deep learning approach, CNNs offer accurate and efficient predictions, enabling timely disease detection and management for improved crop health and higher agricultural yields.

Problem statement: 2

After detecting the spots and identifying the disease in the provided images, the next step was to develop a user interface (UI). To provide stakeholders with easy access, we created a web app using technologies like HTML, CSS, Python, and Flask.

Developing a User Interface

Creating a Structure of User Interface using Flask and Python

We chose Flask for its versatility and simplicity in developing web apps and dynamic UIs for prediction models. Its rich ecosystem and scalability make it ideal for adding functionalities like authentication, database connectivity, and data visualization. Flask's built-in development server streamlines testing and debugging. It offers a responsive and user-friendly experience for our stakeholders, and we easily transitioned our static UI into a dynamic application using Flask.

Problem statement: 3

Uploading the scanned image and utilizing it as an input to Python code

Uploading Image

In image processing tasks using Python, built-in functions are employed, particularly within the OpenCV library, to read and display images. To achieve this, the process involves using the `imread()` method to read an image, creating a GUI window, displaying the image with the `imshow()` method, using the `waitKey(0)` function to keep the window on the screen until user interaction, and finally, utilizing `destroyAllWindows()` to remove the image window from memory once it has been displayed. Reading images in Python with OpenCV is achieved using the `cv2.imread()` method, which loads an image from a specified file. It's important to note that this method may return an empty matrix if the image cannot be read due to various reasons like missing files or unsupported formats. By following these steps, one can effectively read and display images using OpenCV in Python.

How Computer Understands Images

Understanding the working of the project and the algorithm's implementation requires insight into how computers interpret images. The process begins with image upload.

Digital images consist of pixels, with each pixel containing different channels. Computers perceive images as combinations of 0s and 1s, with each pixel being the smallest unit in an image. In a colored image, each pixel channel holds values ranging from 0 to 255, represented in binary for computer comprehension. Merely reading an image is insufficient; the computer must understand its content, meaning, and what it depicts. This is where machine learning comes into play.

Machine learning allows a computer to comprehend and describe the content of an image, similar to how we teach children to identify alphabets or differentiate between an apple and a banana through examples. This is precisely how a computer learns to recognize objects in an image.

Much like humans possess various skills, including recognizing objects in images (e.g., identifying a dog in a picture), computers have machine learning models, which can be viewed as their skills for performing the same task. Just as humans require training to acquire a skill, machine learning models must be trained.

In both cases, training occurs through examples. Much like a child learns to identify an apple, a machine learning model can be taught to recognize an apple in an image by providing numerous example images containing apples. From these examples, the model learns the features of an apple, such as its shape and color. Consequently, when a new image containing an apple is presented to the computer with this trained model, it can apply its knowledge of apples and identify the presence of apples in the new image.

Walkthrough of the algorithm

Mask R-CNN (Region-based Convolutional Neural Network) is a A potent deep learning model, such as the fine-tuned Mask R-CNN, revolutionizes crop disease prediction and management. By leveraging a labeled dataset of crop images, the model performs robust data preprocessing, including resizing, normalization, and augmentation, and excels in both bounding box and pixel-wise mask predictions. During inference, it accurately identifies disease locations and boundary masks, enabling comprehensive disease type recognition and severity assessment based on area coverage and visual factors. This invaluable tool empowers timely, data-driven decision support, including alert systems, historical data trend analysis, and seamless integration with precision agriculture technologies. Coupled with a user-friendly interface and a commitment to continuous improvement through regular updates, Mask R-CNN elevates crop disease management by ensuring early detection, precise diagnosis, and effective control strategies.

Comparison of the Algorithm with the pre-existing algorithms

Mask R-CNN is a deep learning model specialized in instance segmentation, capable of not only detecting objects but also providing pixel-level masks to distinguish distinct instances of the same object class. This sets it apart from algorithms like YOLO and Faster R-CNN, which primarily focus on object detection, offering bounding boxes but not pixel-wise segmentation masks. While Mask R-CNN excels in accuracy, especially for object localization and instance segmentation, it can be computationally intensive and slower compared to the real-time processing speed of YOLO, which sacrifices some accuracy in favor of speed.

Another key difference lies in the architectural approach. Mask R-CNN employs a multi-stage architecture, featuring different branches for object detection and instance segmentation. It leverages multiple feature maps at various resolutions to enhance accuracy. In contrast, YOLO relies on a single-stage architecture, dividing the image into a grid to predict bounding boxes, potentially not capturing objects at different scales as effectively as Mask R-CNN.

Data annotation requirements also set them apart. Mask R-CNN demands pixel-level segmentation annotations, a more time-consuming process. This means datasets used for Mask R-CNN often require more detailed labeling. On the other hand, YOLO and Faster R-CNN only need bounding box annotations, typically quicker to create than pixel-wise masks.

In terms of use cases, Mask R-CNN is well-suited for applications where detailed object instance segmentation is critical, such as medical image analysis, autonomous vehicles, and tasks requiring fine-grained localization and segmentation. YOLO shines in real-time object detection scenarios like video analysis and robotics, where quick decision-making is paramount.

The complexity of their architectures differs significantly. Mask R-CNN features a relatively complex structure with multiple subnetworks for object detection and segmentation, while YOLO is known for its simplicity, employing a single-stage architecture that directly predicts bounding boxes and class probabilities.

In summary, the choice between Mask R-CNN and YOLO depends on specific application requirements. Mask R-CNN's strength lies in its pixel-level instance segmentation accuracy but with higher computational demands, whereas YOLO prioritizes real-time object detection speed at the cost of some segmentation detail. Understanding these distinctions is crucial for selecting the most appropriate model for a given task.

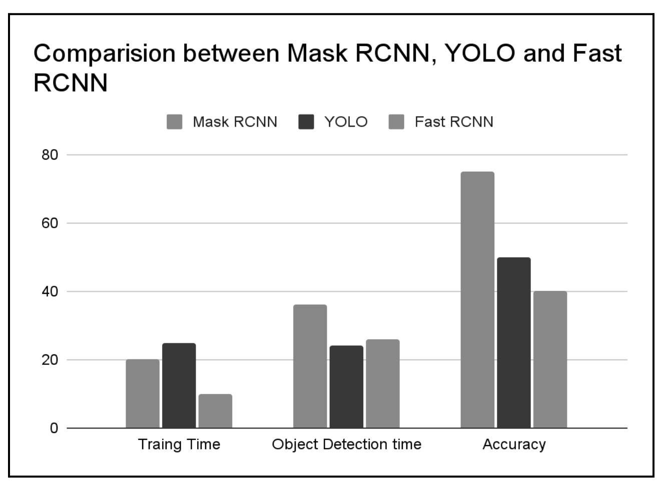
Dataset Used

The dataset comprises 87K RGB images of 38 different classes, including Tomato, Strawberry, Soybean, Squash, Raspberry, and more. It's split into an 80/20 ratio for training and validation, with an additional directory containing 33 test images for predictions.

How Algorithm Understands Which Part Of Image Is Leaf

Mask R-CNN, which stands for Region-based Convolutional Neural Network, adeptly identifies and outlines leaves within images by combining object detection and instance segmentation. Initially, it employs object detection to locate potential regions containing objects, facilitated by a region proposal network (RPN). The RPN generates bounding box proposals, suggesting areas where objects like leaves might be present. Then, Mask R-CNN extracts features from these proposed regions using a convolutional neural network (CNN) to capture essential object information. The distinctive feature of Mask R-CNN lies in its instance segmentation step, where it surpasses conventional object detection. This phase not only identifies objects but also supplies pixel-level segmentation masks for each object instance.

The network creates binary masks for detected objects, marking pixels within the object as '1' and the background as '0'. In the case of leaves, this results in precise delineation of leaf boundaries, allowing for accurate localization and distinction from other elements in the image. Notably, Mask R-CNN can manage multiple instances of objects within the same image, generating separate masks for each instance, ensuring accurate segmentation and identification. In essence, Mask R-CNN's dual approach to object detection and instance segmentation enables precise leaf recognition, delivering valuable insights for applications ranging from crop disease detection to plant health assessment and botanical research.



PROJECT DESIGN

Requirement Analysis

The requirement analysis for our crop disease detection and management web application, developed with Flask, prioritizes an intuitive, user-friendly interface accessible across various devices. The system must employ advanced image processing and machine learning algorithms to accurately identify crop diseases, offering real-time predictions and tailored management advice. To enhance user engagement and collaboration, a messaging system should facilitate communication between users and agricultural experts. Scalability, sustainability, and regular updates are vital, ensuring the system adapts to new outbreaks and remains current. Data privacy and security measures are imperative to safeguard user information. The application should also feature data analytics capabilities for insights into disease trends, geographical patterns, and best practices for disease management. In summary, this analysis underscores the need for a robust, user-centric crop disease detection and management web application, empowering farmers, supporting informed decisions, and promoting sustainable disease management in agriculture.

Stakeholder Identifications

Identifying stakeholders for a Flask-based crop disease detection and management web app is essential for project success. Key stakeholders include farmers, agricultural extension services, researchers, app developers, government regulatory bodies, data providers, and end users. Additionally, funding agencies, local communities, environmental organizations, and educational institutions play vital roles. Engaging these stakeholders ensures the app meets diverse needs. Farmers rely on it for disease management, while government agencies and researchers use it for data-driven decisions. App developers ensure functionality, and regulatory bodies oversee compliance. Data providers and decision-makers contribute to data accuracy. Local communities' feedback tailors the app, and environmental organizations appreciate its sustainability impact. Educational institutions can use it for teaching. Identifying and involving these stakeholders is crucial for the app's success and its positive influence on agriculture.

Investigation

Our investigation into the Flask-based crop disease detection and management project is comprehensive and multifaceted. We prioritize its success by conducting in-depth research to identify prevalent crop diseases and understand farmers' specific challenges. This information helps us tailor our disease detection algorithms effectively. We continually explore the latest advancements in image processing and machine learning to enhance prediction accuracy and speed. The design and user experience of the web application are under scrutiny to ensure it is intuitive and accessible to both farmers and agricultural experts.

Moreover, we have a strong focus on collaboration. We work closely with local agricultural authorities and farming communities to gather real-world insights and feedback, aligning our project's development with actual agricultural needs and expectations. In addressing data security and privacy concerns, we have implemented robust measures to handle farmer-submitted data with the utmost care, emphasizing data confidentiality and integrity. This comprehensive approach underscores our commitment to technological excellence and the well-being of the farming community.

IMPLEMENTATION AND TECHNOLOGIES

Technologies Used

<u>Software Used</u>

Firebase: Firebase Realtime Database is a cloud-hosted NoSQL database that stores and syncs data in real-time, making it easily accessible across web and mobile devices. It provides real-time collaboration features, offline data access, and seamless synchronization.

macOS: macOS, developed by Apple Inc., is a Unix-based operating system known for its user-friendly interface, stability, and performance. It features elements like the Dock, Finder, and Menu Bar, along with capabilities such as Time Machine and Spotlight for efficient file management and searching.

VS Code: Visual Studio Code (VS Code) is an open-source, cross-platform source code editor developed by Microsoft. It is highly customizable and supports various programming languages. VS Code offers features like integrated terminal, Git support, code completion, debugging tools, and an extensive extension marketplace, making it a popular choice for developers across different domains.

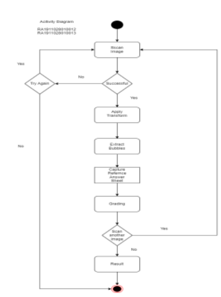
<u>Programming Language Used</u>

Python: Python is a versatile, high-level programming language known for its readability and extensive standard library. It is used in web development, data analysis, artificial intelligence, and scientific computing. Python's simplicity and cross-platform compatibility make it a popular choice for a wide range of applications.

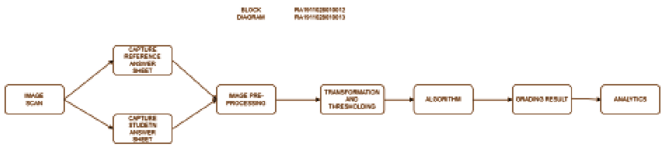
FLASK: Flask is a micro web framework for Python, offering lightweight and efficient web application development. It follows the WSGI standard and supports routing, request processing, and templating. Flask's modularity allows developers to choose components as needed, offering flexibility but requiring more architectural decisions. It is ideal for small to medium-sized web applications and APIs, providing simplicity and versatility for developers.

<u>MODELS</u>

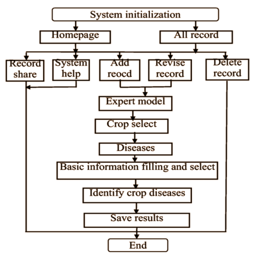
Activity Diagram



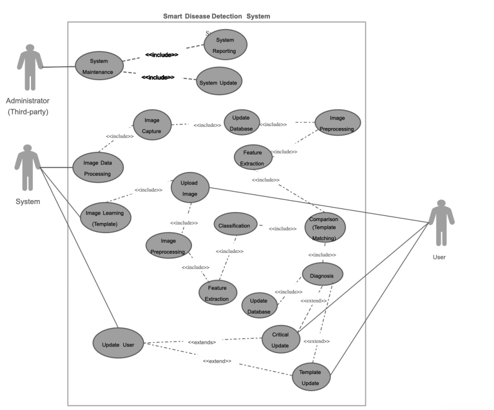
Data Flow Diagram



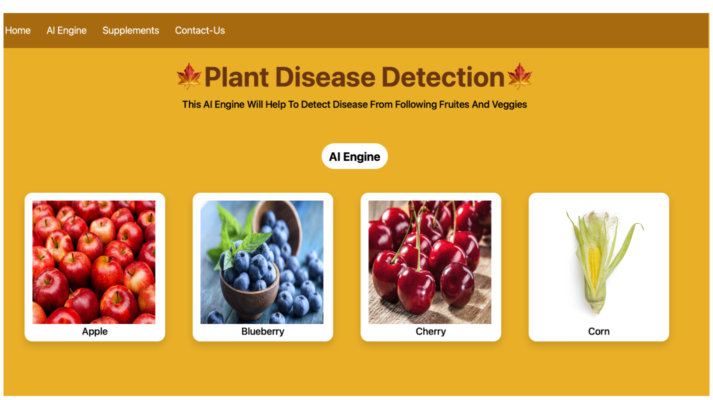
Architecture Diagram



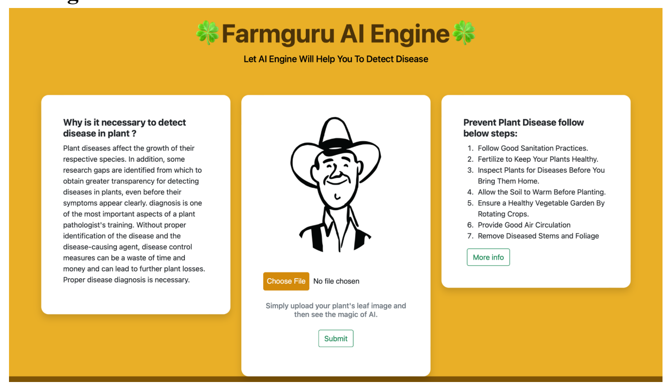
Use Case Diagram



HOME SCREEN



AI Engine



CONCLUSION AND FUTURE SCOPE

Future Scope

The future scope of our college-level crop disease detection and management project is promising. We can enhance it by expanding the dataset to cover a broader range of crops and diseases, improving the user interface for accessibility, and exploring integration with IoT and remote sensing technologies. Collaboration with agricultural experts and local communities for feedback is crucial. Optimizing the algorithm for efficiency and resource conservation is also a key aspect. These steps aim to bridge the gap between academia and real-world agricultural applications, benefiting local farmers and sustainability.

Conclusion

In conclusion, our crop disease detection and management project is a significant step forward in leveraging technology to address agricultural challenges. We've demonstrated its capability in detecting and managing crop diseases with a well-trained dataset, showing promise for further expansion and real-world applications. This project can significantly impact agriculture by enhancing crop health and ensuring food security through timely disease management. However, it's a stepping stone to more extensive implementations. Future development, like dataset expansion and algorithm optimization, is crucial for real-world effectiveness. Our project exemplifies technology's potential in improving crop health, sustainability, and global food security.

REFERENCES

[1] Krishna, S., & Singh, D. (2020). "Deep Learning-based Tomato Plar Diseases Detection and Classification Using Image Processing and Machin Learning Algorithms." Computers and Electronics in Agriculture, 17 105338.

[2]Chollet, F., et al. (2015). "Keras." GitHub Repositor https://github.com/keras-team/keras

[3] Abadi, M., et al. (2016). "TensorFlow: Large-Scale Machine Learnin on Heterogeneous Systems." arXiv preprint arXiv:1603.04467.

[4] Python Software Foundation. (2020). "Python Programming Language. https://www.python.org/

[5] Grinberg, M. (2018). "Flask Web Development: Developing We Applications with Python." O'Reilly Media.

[6]Kaggle. "Plant Pathology 2020 - FGVC7 https://www.kaggle.com/c/plant-pathology-2020-fgvc7

[7] LeCun, Y., Bengio, Y., & Hinton, G. (2015). "Deep Learning." Natur 521(7553), 436-444.

[8] Russakovsky, O., et al. (2015). "ImageNet Large Scale Visua Recognition Challenge." International Journal of Computer Vision 115(3), 211-252.

[9] Pedregosa, F., et al. (2011). "Scikit-learn: Machine Learning i Python." Journal of Machine Learning Research, 12, 2825-2830.

[10] Brownlee, J. (2019). "Deep Learning for Computer Vision. Machine Learning Mastery.

[11] Szegedy, C., et al. (2016). "Rethinking the Inception Architectur for Computer Vision." In Proceedings of the IEEE Conference o Computer Vision and Pattern Recognition.

[12] Lyra, Maria & Ploussi, Agapi & Georgantzoglou, Antonios. (2011). MATLAB as a Tool in Nuclear Medicine Image Processing. 10.5772/19999.

[13] Chandran, Jaideep. (2019). An Image-Based Colorimetric Technique for PortableBlood Gas Analysis.