

SUGARCANE DISEASE PREDICTION AND TREATMENT SYSTEM

V. Hemalatha¹, M. Mohamed Farook², M. Raja Pandiyan³, S. Partha Sarathi⁴

¹Associate Professor, Dept. of CSE, NSNCET, Karur

^{2 3 4} Student, Dept. of CSE, NSNCET, Karur

Abstract - Sugarcane disease is a major challenge for the sugar industry in India, often leading to significant crop destruction and financial losses. Early detection and treatment of these diseases are crucial, but farmers may lack the expertise to identify them. This study explores the use of machine learning, specifically image processing and deep learning techniques (CNN), as a potential solution to this problem. By training a deep learning model on a dataset of disease-infected sugarcane images, the study successfully develops a model capable of detecting and classifying sugarcane diseases. This research offers a promising approach to assist farmers in detecting and classifying sugarcane diseases using deep learning algorithms.

Key Words: Sugarcane disease, crop destruction, financial losses, early detection, machine learning, image processing, deep learning, Convolutional Neural Network (CNN), disease classification, agriculture technology, sugar industry in India, disease-infected image dataset, automated disease detection, smart farming, precision agriculture.

1. INTRODUCTION

Sugarcane is a vital crop for the agricultural industry, and India is one of the major producers and exporters of sugarcane. However, sugarcane diseases can cause significant damage to crops, resulting in huge financial losses for farmers. Therefore, it is essential to develop an accurate and efficient system to predict and manage sugarcane diseases. In recent years, there has been a growing interest in using machine learning and image processing techniques for crop disease detection.

This project aims to contribute to the existing literature on using these techniques by developing an automated web application that utilizes machine learning and image processing to predict the presence of disease in sugarcane crops and recommend treatments.

The system will use digital images of sugarcane leaves and stems, and through image processing techniques, the system will extract features that can be used to identify the type of disease present in the crop. Machine learning algorithms will

then be trained on these features to accurately predict the presence of disease.

Once the disease is identified, the system will recommend appropriate treatments to manage its effects and improve crop yield. The recommended treatments will be based on scientific research and will be tailored to the specific type of disease present in the crop.

The proposed system has the potential to revolutionize the way sugarcane diseases are managed by providing farmers with a reliable and efficient tool to identify and manage diseases, reduce crop losses, and improve the sustainability of the sugarcane industry.

1.1 Project Overview:

Sugarcane is a significant cash crop in Pakistan, but it is vulnerable to heavy losses caused by over 50 different diseases, including fungi, bacteria, viruses, and nematodes. These diseases can occur in certain areas, seasons, and specific parts of the plant, making them difficult to identify and manage. Therefore, the development of an automated system for disease prediction and treatment is crucial to mitigate crop losses and improve crop yield.

Furthermore, managing sugarcane diseases can be a challenging task for farmers due to the complex nature of the sugarcane plant and the diverse range of diseases that can affect it. Sugarcane diseases can lead to significant economic losses, impacting the livelihoods of farmers and the overall sustainability of the sugarcane industry. Therefore, the need for an automated system that can accurately predict and manage sugarcane diseases has become increasingly important. By utilizing machine learning and image processing techniques, an automated system can provide farmers with a reliable and efficient tool to detect and manage diseases, reducing crop losses and improving the overall sustainability of the sugarcane industry.

1.2 Objectives:

The proposed project on automated sugarcane disease prediction and treatment is aimed at providing farmers with a valuable tool to help manage sugarcane diseases, reduce crop losses and improve the sustainability of the sugarcane industry. The project objectives include the accurate identification of the type of disease present in sugarcane crops and the provision of recommended treatment options to mitigate its effects and improve crop yield. These

International Research Journal of Education and Technology

Peer Reviewed Journal, ISSN 2581-7795

objectives will be achieved using machine learning and image processing techniques, which will enable the system to accurately identify the disease and provide recommended treatment options.

Disease identification is a critical component of the project, as sugarcane crops can be affected by various types of diseases that can impact their quality and quantity. By utilizing machine learning and image processing techniques, the system will be able to accurately identify the disease present in the sugarcane crops, thus providing farmers with an efficient way to manage the disease.

2. LITERATURE SURVEY :

Over the past few years, the agricultural industry has witnessed a surge of interest in leveraging the power of machine learning and image processing techniques for efficient and accurate crop disease detection. By adopting these cutting-edge technologies, farmers can significantly reduce the time and effort required for manual disease identification and treatment, thus improving the overall productivity and profitability of their farms. In this section, we review the most recent and innovative research in this emerging field, highlighting the benefits, challenges, and limitations of using machine learning algorithms for crop disease detection.

2.1 Research on Sugarcane Disease Detection

The application of machine learning in crop disease detection has been a subject of research in various crops, including sugarcane, apple, grapevine, and cassava. The research has focused on developing algorithms to detect and classify sugarcane diseases using machine learning techniques. Numerous studies have been conducted to explore the potential of machine learning in detecting sugarcane diseases. Researchers have used different machine learning algorithms, including decision tree and support vector machine, as well as deep learning techniques such as convolutional neural networks (CNNs) to classify and diagnose sugarcane diseases accurately. These studies have provided valuable insights into the use of machine learning for sugarcane disease detection and have opened new avenues for research and development in this area.

2.2 Machine Learning Techniques for Sugarcane Disease Detection

In recent years, machine learning techniques have been extensively used in various fields for classification and prediction tasks. In the field of agriculture, machine learning algorithms have been employed to detect and classify plant diseases. The effectiveness of machine learning techniques in detecting sugarcane diseases has been demonstrated in several studies. In this section, we discuss the different machine learning techniques that have been used for sugarcane disease detection, including decision trees, support

vector machines, and deep learning techniques such as convolutional neural networks.

3. HARDWARE, SOFTWARE ANALYSIS

3.1 Hardware Requirements:

The hardware requirements for our automated sugarcane disease prediction and treatment project are:

1. Camera: The system utilizes images of sugarcane plants for analysis, therefore a camera is needed to capture these images. A mobile camera is recommended as it allows for easy data collection in the field.

2. Computer for Image Processing: The captured images will then need to be processed using image processing techniques, this will include tasks such as image segmentation, feature extraction, and data pre-processing. A computer with enough processing power, memory, and storage is needed to run these operations. The computer should be equipped with software development tools (python, libraries, etc), and enough computational power to run Machine Learning models.

3.2 Software Requirements:

The software requirements for our automated sugarcane disease prediction and treatment project are:

1. Python:

Python is a powerful, open-source programming language that is widely used in data science, machine learning, and image processing. It is the main programming language for this project, as it provides a vast ecosystem of libraries, modules and frameworks. Some of the common used libraries are numpy, pandas, opencv, sklearn, and TensorFlow/Pytorch to build the image processing and machine learning models.

Additional software might be required depending on the specific requirements of the project and complexity, but Python is considered as the backbone of the project.

It is important to note that the specific versions of software and libraries used will depend on the particular implementation of the project and the availability of such versions.

4. SOFTWARE DESIGN AND MODELLING

This System will present a detailed analysis of the system, including the use of various UML diagrams and interaction diagrams to describe how the system works. This will include a description of the system's architecture, classes, objects, and their interactions, as well as timing and sequence diagrams that provide a visual representation of the system's behavior.

1. Use Case Diagram:

In the context of a sugarcane disease prediction system, a use case diagram might be used to show how different actors, such as farmers, interact with the system to predict and manage diseases in sugarcane crops.

2. Class Diagrams:

Class diagrams provide a representation of the system's classes, their attributes, and their interactions with each other. They will be used to describe the objects and data structures that make up the system.

3. Sequence Diagrams:

Sequence diagrams provide a representation of the system's interactions over time. They will be used to describe the system's interactions with the user and other components.

4. Timing Diagrams:

Timing diagrams provide a representation of the system's state and behavior over time. They will be used to describe the system's temporal behavior.

5. High-Fidelity Prototypes:

High-fidelity prototypes will be used to provide a more detailed representation of the system's interface and interactions. They will be used to describe the user interface, input, and output.

5. ALGORITHM ANALYSIS AND COMPLEXITY

5.1 Image Processing

5.1.1 Reading an image

`Imread()`

Syntax: `cv2.imread(path, flag)`

To read the images `cv2.imread()` method is used. This method loads an image from the specified file. If the image cannot be read (because of missing file, improper permissions, unsupported or invalid format) then this method returns an empty matrix.

5.1.2 Display an image

`Imshow()`

Syntax: `cv2.imshow(window_name, image)`

`cv2.imshow()` method is used to display an image in a window. The window automatically fits the image size.

5.1.3 Writing/Saving an image

`Imwrite()`

Syntax: `cv2.imwrite(filename, image)`

`cv2.imwrite()` method is used to save an image to any storage device. This will save the image according to the specified format in current working directory.

5.1.5 Grayscale of Images

Importance of grayscale

I Dimension reduction: For example, In RGB images there are three color channels and three dimensions while grayscale images are single-dimensional.

II Reduces model complexity: Consider training neural articles on RGB images of 10x10x3 pixels. The input layer will have 300 input nodes. On the other hand, the same neural network will need only 100 input nodes for grayscale images.

5.1.6 Histograms Equalization

`equalizeHist()`

Syntax: `cv2.equalizeHist(image)`

Histogram equalization is a method in image processing of contrast adjustment using the image's histogram. This method usually increases the global contrast of many images, especially when the usable data of the image is represented by close contrast values.

5.1.7 Otsu Thresholding

Syntax: `cv2.threshold(image, thresholdValue, maxVal, thresholdingTechnique)`

Thresholding is a technique, which is the assignment of pixel values in relation to the threshold value provided. In thresholding, each pixel value is compared with the threshold value. If the pixel value is smaller than the threshold, it is set to 0, otherwise, it is set to a maximum value (generally 255).

In Otsu Thresholding, a value of the threshold isn't chosen but is determined automatically.

5.1.8 Edge Detection

Syntax: `cv2.Canny(image, T_lower, T_upper, aperture_size, L2Gradient)`

Canny Edge filter in OpenCV. `Canny()` Function in OpenCV is used to detect the edges in an image

6. IMPLEMENTATION

This System is about the background knowledge of our work, involving the basic concepts of deep learning and computer vision. Transfer learning is used for pre-trained initialization in the experiment. Convolutional Neural Network (CNN) Image Classification is applied to the Sugarcane Disease Dataset for our goals. Training Strategy is the mathematical explanation of our experiment details.

6.1 Convolutional Neural Network

CNN is a classic Artificial Neural Network (ANN), specially designed for image tasks and understanding. Nowadays, CNN is the mainstream in the domain of computer vision, commonly used for many tasks such as image classification, object detection, image segmentation, and so on.

Here is an example to depict how CNN recognizes an image:

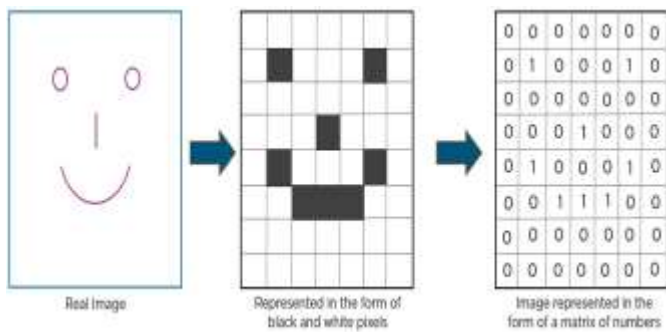


Figure:6.1 Convolutional Neural Network

Architecture

The core component of CNN includes three kinds of layers.

1. convolutional layers
2. pooling layers
3. fully connected layers



Figure:6.2 CNN Architecture

6.2 Django Web Application:

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

The web application is a plant disease detection system that detects diseases in plants. It is designed to help farmers and gardeners identify plant diseases and take necessary measures to prevent their spread. The web application allows users to upload images of plants and automatically detects if the plant is healthy or diseased.

6.2.1 System Architecture:

The web application is built using Django, HTML/CSS, JavaScript, and Bootstrap. The application has a front-end component and a back-end component. The front-end is responsible for rendering the user interface, while the back-end is responsible for processing user requests, running the prediction model, and generating output.

6.2.2 User Interface:

The web application has the following main pages: index, service, about, prediction, and contact. The index page is the homepage of the application and provides a brief introduction to the web application. The service page describes the services provided by the application, and the about page provides information about the developers and the technology used to build the application.

The prediction page is where users can upload images of plants for disease detection. Once an image is uploaded, the application processes the image and returns a diagnosis. The contact page provides a form for users to contact the developers with questions or feedback.

6.2.3 Features:

The web application provides the following features:

Uploading an image: Users can upload an image of a plant to the web application for disease detection.

Removing background: The application uses an algorithm to remove the background of the uploaded image, making it easier to identify the plant.

Converting to grayscale: The application converts the uploaded image to grayscale to simplify image processing.

Equalizing histogram: The application equalizes the histogram of the grayscale image to improve contrast and make it easier to identify diseased regions.

Segmenting regions: The application uses image segmentation to identify regions of the plant that are infected.

International Research Journal of Education and Technology

Peer Reviewed Journal, ISSN 2581-7795

7. CONCLUSION

In conclusion, the sugarcane disease detection website is an essential tool for detecting and predicting diseases in sugarcane plants. The website was designed using Django and includes a user-friendly interface and animations to enhance the user experience. The website provides information about sugarcane diseases and their control and includes a list of over 50 diseases caused by fungi, bacteria, viruses, and nematodes. The website also includes a sugarcane disease detector that provides recommendations for treatment.

Furthermore, we discussed the importance of testing in the software development process. By using both black box testing and white box testing, we can ensure that the sugarcane disease detection website is accurate, efficient, and easy to use for end-users.

Overall, the sugarcane disease detection website has the potential to improve the yield and productivity of sugarcane plants. It is a valuable resource for sugarcane farmers, researchers, and other stakeholders in the sugarcane industry.

REFERENCES

1. Book: Field Guide – Diseases of Australian Sugarcane, Robert Magarey (2022)
2. Iqbal, Muhammad Aamir, and Asif Iqbal. "Sugarcane production, economics and industry in Pakistan." *Am. J. Agric. Environ. Sci* 14.12 (2014): 1470-1477.
3. Hossain, Md Imam, et al. "Current and Prospective Strategies on Detecting and Managing Colletotrichum falcatum Causing Red Rot of Sugarcane." *Agronomy* 10.9 (2020): 1253.
4. Kumar, Arpan, and Anamika Tiwari. "Detection of Sugarcane Disease and Classification using Image Processing." *Int J Res Appl Sci Eng Technol* 7.5 (2019): 2023-30.
5. Report: Sugar Industry in Pakistan – Challenges and Opportunities
6. Power Law (Gamma) Transformations | TheAILearner
7. S. Saha, "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way," Dec. 2018. [Online]. Available: <https://towardsdatascience.com/acomprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>

to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53