Image Processing And Machine Learning Approach For Tomato Leaf Disease Detection

[1] **Sharanya Prem Kumar,** [2] **Prashant Kumar Soori,**
[1], [2] Heriot – Watt University Dubai
[1] sp2007@hw.ac.uk, [2] p.k.soori@hw.ac.uk

Abstract

Using image processing and machine learning, this research presents a technique for quick plant disease diagnosis in tomatoes. Gray-level co-occurrence matrices (GLCM) are used in the proposed approach to extract textural aspects of leaves and histogram color extraction. The collected characteristics are then utilized to train a support vector machine (SVM) to categorize different communicable and non-communicable plant diseases. The trained model is put on a Raspberry Pi, which captures video using a pi camera and employs the classifying model to diagnose plant disease in real time. On the plant disease SVM model, which diagnoses two common illnesses, bacterial spot and mold, the generated system has an accuracy of 97.29%. It was tested on a few example leaf images to see how accurate it was in real time.

Keywords— camera vision, plant disease detection, support vector machine, texture extraction, color extraction.

I. INTRODUCTION

Plant diseases, according to the United Nations Food and Agriculture Organization (FAO), account for 10-16% of worldwide food production loss each year, with the tomato crop being no exception[1]. Early and precise diagnosis of plant diseases is critical for limiting disease spread and agricultural losses. Various viral and non-infectious diseases can have a substantial influence on productivity and quality in tomato crops. Pathogens such as viruses, bacteria, and fungi cause infectious maladies such as tomato mosaic virus and late blight. These diseases may swiftly and readily spread from plant to plant, resulting in extensive epidemics. Non-infectious maladies, on the other hand, such as nutritional shortages and biotic stress, are caused by environmental variables such as drought, temperature fluctuations, and an unbalanced soil ph[2].

Plant disease detection has typically been accomplished by physical examination, laboratory testing, serological approaches, chemicals, electron microscopy, and nucleic acid-based diagnostics[3]. These procedures, however, can be time-consuming, labour-intensive, and need specific training and equipment. Computer vision techniques have been proved to be useful in identifying plant diseases in recent years, giving an efficient and non-invasive alternative to older approaches.

Machine learning has increased the effectiveness and precision of computer vision-based plant disease detection algorithms. Machine learning algorithms may be trained to recognize patterns and characteristics in photos of damaged plants, allowing them to detect diseases in their early stages. As a result, machine learning is a key weapon in the battle against plant diseases, reducing crop losses and improving food security. Farmers may make better educated decisions about how to distribute resources such as water, manure, and insecticides if they have precise information regarding crop conditions. This culminates in more sustainable and efficient agricultural techniques.

Some of the previous image process techniques used is as follows:

1) Neural networks: To choose the different dimensions of the Faster R-CNN target frame, FCM-KM analysis is performed in conjunction with the R-CNN algorithm for detecting rice infections. According to the concise conclusion covering 3010 images, the precision for detecting three diseases was an average of 97%[4]. In [5] Affected produce leaves are gathered and labelled according to the disease. Image processing is combined with pixel-by-pixel procedures to improve visual information. It is then followed by extraction of features, segmentation, and classification of collected leaf patterns in order to diagnose plant leaf diseases. Several artificial neural network topologies are used, with the greatest accuracy of 98.59% in diagnosing plant illness. This was a huge success, proving the viability of this strategy in the field of Plant Disease Diagnosis and high crop yields.
2) Other image processing techniques: Colour extraction methods such as RGB and HSV are used as descriptors, as are texture extraction methods such as LBP. The Fine KNN classifier, Cubic SVM, Boosted, and Bagged tree classifiers are utilized after feature extraction from all image data in [6] that achieved an accuracy of 99.9%.

II. METHODOLGY

The identification of common tomato diseases that may be identified in leaves, such as bacterial spot and mold, is proposed in this work.

Bacterial spot often appears as tiny, round, moisture blemishes on the leaves, which may eventually turn black or brown. The germs can be transmitted via spraying water, pests, and equipment. Mold is a fungal disease that shows as a furry, greyish, or white development on the fruit and leaves of the tomato crop. These features can be identified using texture extraction.

For texture extraction, a gray level co-occurrence matrix is used, and for colour extraction, a histogram is used. The RGB values from the histogram, together with dissimilarity, contrast, homogeneity, energy, and correlation, are concatenated into a vector that is used to train a support vector machine to distinguish between damaged and healthy leaves. The camera, which is placed on a tilting module, scans the whole of the plant in real time, frame by frame, and detects damaged leaves.

As shown in Figure 1, the camera is set to span through the length of the plant and a live feed is fed to the classifying algorithm (SVM).

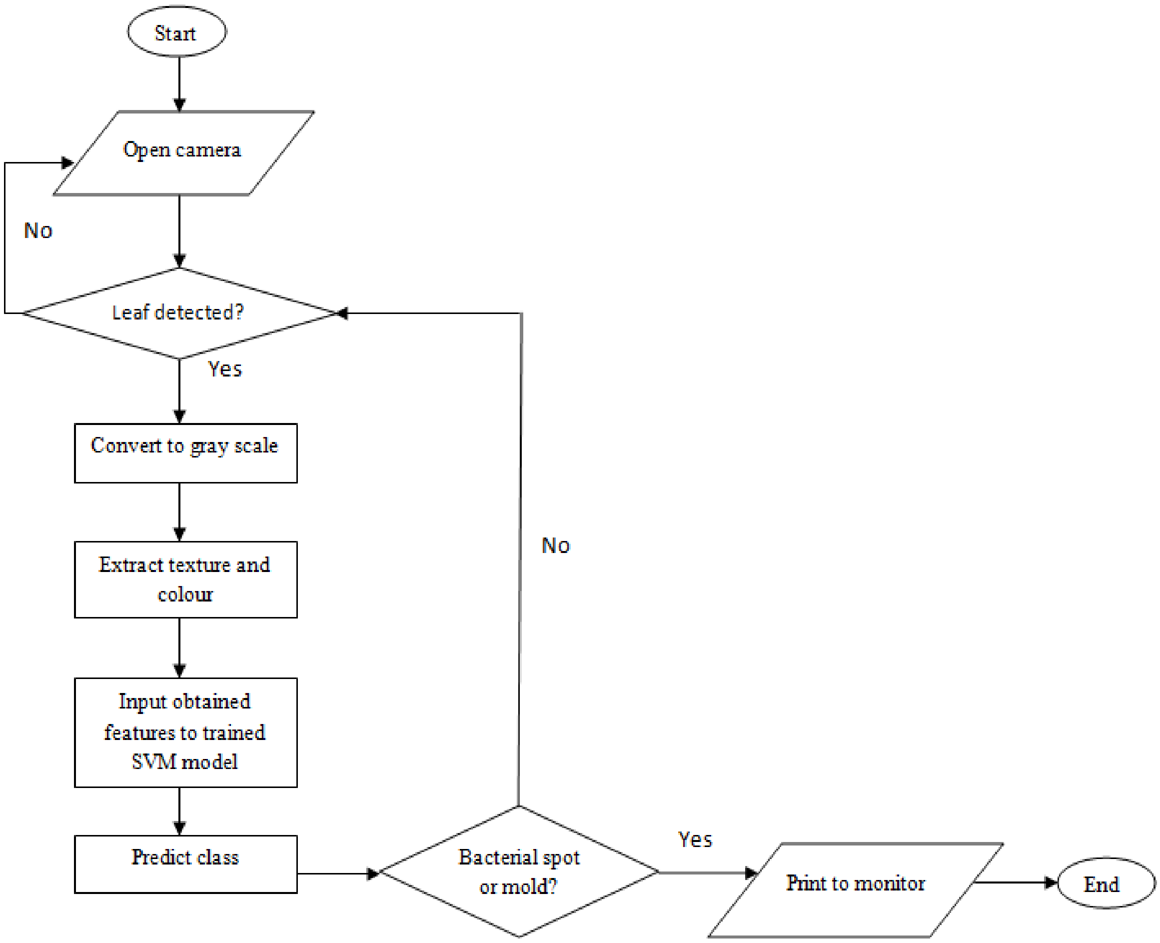

Figure 1: system flowchart

Frame by frame the real time video is converted to gray scale and using GLCM and histogram for colour, the features are extracted. These features are classified into healthy, bacterial spot or mold class using the trained support vector machine. When bacterial spot or Mold is detected, the Raspberry pi-4 prints the alert in the monitor.

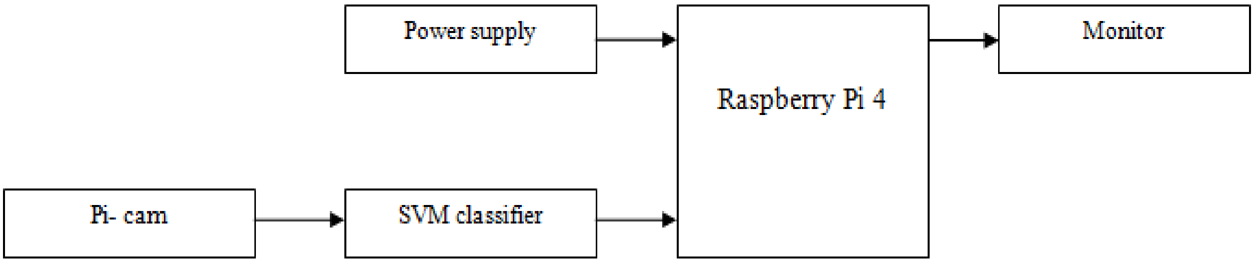

Figure 2: System Block Schematics

A. Data Collection

The plant village dataset was employed, which had multiple photos of healthy and diseased (bacterial spot and mold) leaf images taken under varied lighting circumstances and orientations, which aids in the accurate recognition of leaf on plants under various settings. For this project, 300 pictures from each class were used. The dataset was split into 80% for training and 20% for testing.

B. Preprocessing of Dataset

The photos in the dataset have been scaled to standard sizes (200,200). The photos are then transformed to greyscale using the OpenCV function, such that the GLCM features collected are based on the pixel intensity values in the image.

C. GLCM Texture Extraction

To compute texture, the grey level co occurrence matrix considers the distribution of pixel values. The primary idea behind GLCM is that it generates a matrix that represents the frequency of occurrence of a specific pair of grey level pixel intensities nearby[6]. They may be estimated at different angles and pixel distances to get texture interpretation. The matrix should be a square matrix with the number of rows and columns equal to the number of pixel values present. For this project, the GLCM matrix is produced with a pixel distance of 1 and an angle of 0; that is, the matrix is calculated based on the occurrence of a pair of pixels that are horizontal to each other. Scikit image is a python package that allows GLCM calculation along with OpenCV. *greycomatrix* function of Scikit image is a Python module that works with OpenCV to calculate GLCM. Skimage's greycomatrix method in Python uses numpy to compute the grey level comatrix. The generated matrix has 256 rows and columns since there are 256 intensity levels of greyscale after image conversion. The data is also homogenized. Following the matrix computation, attributes are calculated in order to retrieve texture. In this situation, the distinguishing characteristics are homogeneity, dissimilarity, correlation, contrast, and energy. They provide information on many elements such as how frequently a certain pair occurs, spatial orientation, how dissimilar pixel values are to their neighbor, and how frequently non-zero values occur. They are computed using the *greycoprops* tool. These values are then appended into an array and fed into a classifier.

D. Histogram Colour Extraction

Color is another key indication of disease, and a histogram is used to get color distribution. A histogram uses the three channels red, blue, and green to provide a different distribution for each disease[7]. Each pixel is considered and a quantity is assigned to it based on the 8-bit system, resulting in 256 bins, each of which is incremented depending on the color acquired from the pixel under examination. Because there are three channels, the histogram produced would be three dimensional; however, for computational ease, one channel is suppressed and a histogram is created. This distribution is then supplied into the classifier, which compares the input picture to the current distribution in order to identify the illness. In Python, the *calcHist* function of OpenCV is used to compute this.

E. Support Vector Machine

The support vector machine is a machine learning technique for categorization. Its fundamental premise is to locate the hyper plane that separates the data points of each class. They are changed to higher dimensions areas in order to quickly discover the hyper plane to segregate. SVM uses kernels to transform them into higher dimensional space. There are numerous kernels, however the radial basis function (rbf) kernel is utilized for this project. The rbf kernel can compute complex correlation between data points[8], This enables SVM to find non-linear decision boundaries. The SVM classifier is computed using Sklearn's *SVC* function. The *train_test_split* function divides the data input into the testing and training datasets, using a split of 80% for training and 20% for testing. The images and labels are supplied into several parameters, which are further divided into test and training. The *fit* function is used to fit the images and labels to the model that was trained using the features retrieved from the GLCM and color histogram. The *Sklearn.metrics* function is used to compute the trained model's metrics. This function calculates accuracy, precision, F1-score, and recall, as well as the model's efficiency.

III. RESULTS AND DISCUSSIONS

The proposed system was put through accuracy tests. The experiment was carried out under a variety of situations, and the findings were documented and evaluated.

The developed machine learning algorithm for tomato leaf disease detection of three categories namely bacterial spot (class: [0]), mold (class: [1]) and healthy (class: [2]) were tested on an unseen dataset and produced the following results as shown in Table 1.

Table 1: Trained Model Metrics

Metrics	percentage
Accuracy	97.29%
Precision	97.23%
F1- score	97.23%
Recall	97.25%

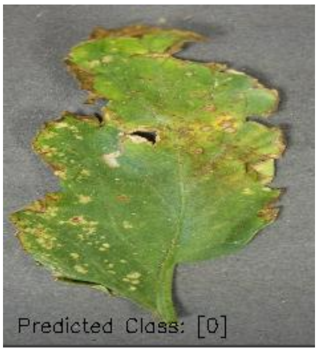

Figure 3: Bacterial spot disease detection

Figure 4: Mold disease detection

Figure 5: Mold detected as healthy leaf

Figure 3, 4, 5 is the displayed output on the monitor screen the classified the in-frame leaf into their respective classes.

Table 2: obtained results on unseen dataset

Sample number	Predicted class	Actual class
1	Bacterial spot	Bacterial spot
2	Bacterial spot	Bacterial spot
3	Bacterial spot	Bacterial spot
4	healthy	healthy
5	healthy	healthy
6	healthy	healthy
7	healthy	Mold
8	mold	mold
9	mold	mold

CONCLUSION

This study proposes a fully automated tomato disease diagnosis system based on image processing and machine learning, which would be run on a Raspberry Pi computer and the pi camera. The SVM was trained with only the retrieved texture features from 100 photos using GLCM, and the accuracy was 79%. When tested on certain unknown data, some were mistakenly detected. Histogram characteristics from colour extraction were added to the SVM with an additional 200 photos of varying orientation and illumination to improve accuracy and precision. This boosted the precision to 97.29%. When tested on nine unseen photos, eight of them were successfully recognised.

REFERENCES

[1] S. Chakraborty and A. C. Newton, "Climate change, plant diseases and food security: an overview," *Plant pathology,* vol. 60, no. 1, pp. 2-14, 2011.

[2] N. Gobalakrishnan, K. Pradeep, C. Raman, L. J. Ali, and M. Gopinath, "A systematic review on image processing and machine learning techniques for detecting plant diseases," in *2020 international conference on communication and signal processing (ICCSP)*, 2020: IEEE, pp. 0465-0468.

[3] M. U. Ahmed, M. M. Hossain, and E. Tamiya, "Electrochemical biosensors for medical and food applications," *Electroanalysis: An International Journal Devoted to Fundamental and Practical Aspects of Electroanalysis,* vol. 20, no. 6, pp. 616-626, 2008.

[4] G. Zhou, W. Zhang, A. Chen, M. He, and X. Ma, "Rapid detection of rice disease based on FCM-KM and faster R-CNN fusion," *IEEE access,* vol. 7, pp. 143190-143206, 2019.

[5] A. Dhakal and S. Shakya, "Image-based plant disease detection with deep learning," *International Journal of Computer Trends and Technology,* vol. 61, no. 1, pp. 26-29, 2018.

[6] H. Ali, M. Lali, M. Z. Nawaz, M. Sharif, and B. Saleem, "Symptom based automated detection of citrus diseases using color histogram and textural descriptors," *Computers and Electronics in agriculture,* vol. 138, pp. 92-104, 2017.

[7] M. J. Swain and D. H. Ballard, "Indexing via color histograms," in *Active perception and robot vision*, 1992: Springer, pp. 261-273.

[8] K. M. Tao, "A closer look at the radial basis function (RBF) networks," in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, 1993: IEEE, pp. 401-405.