FruitScan Pro : Fruit Disease and Quality Detection Application

Mr. S. H. Sangale[1], Shiwani Nanaware[2], Rituja Malode[3], Shraddha More[4], Madhura Patil[5]

*1 Professor, Department of Computer Technology, K.K. Wagh Polytechnic, Nashik, Maharashtra, India.

*2,3,4,5 Department of Computer Technology, K.K. Wagh Polytechnic, Nashik, Maharashtra, India.

---***---

Abstract - This research paper proposes a revolutionary approach to agriculture by combining the latest image processing technologies with machine learning. The agricultural industry, which is vital for India's economy, needs efficient methods for monitoring fruit health. Manual monitoring is time consuming and often inadequate. In this paper, we propose a smart farming approach using image processing techniques. This approach improves image quality, segmenting images, extracting features, and classifying diseases and their cures. Fruit diseases are a major economic challenge around the world. Therefore, it is necessary to identify them quickly. Our proposed system offers fast and automatic disease detection. Preprocessing, extracting features, classifying, training/testing. The results prove its effectiveness for accurate disease detection and solutions. As the demand for agricultural productivity increases, this method will be a valuable tool for increasing fruit production with minimum human intervention. Image processing for effective implementation.

***Key Words*:** Fruit Disease, Solution, Image Preprocessing, Feature Extraction, Classification.

1. INTRODUCTION

India is the country that produces the largest number of fruits (44.04 mt) and is the second largest fruit producer in the world (10%). Indian farmers grow a variety of fruits, including apples, bananas, citrus, grapes, mangoes, guavas, papayas, watermelons, etc. The fruit industry contributes approximately 20% to the country's development. In addition to food crops, India also produces and exports fruits. In India, agriculture is the main source of livelihood for 58% of India's population. The ever-changing climatic conditions, as well as certain diseases, have a major impact on crops, resulting in reduced crop yield. India ranks second among the highly populated countries in the world and is still growing. As a result, the demand for food will automatically increase, resulting in an increase in the need for food production. In India, classification of good fruits and bad fruits in most places is barely done by hand, resulting in more errors in fruit grading while exporting.

The production of high-quality fruits has decreased due to poor fruit development, lack of maintenance, and manual evaluation. The quantity and character of the agricultural products have decreased due to disease of fruits. Fruit diseases are mainly caused by viruses and bacteria, but also by bad environment conditions. There are many characteristics and behaviors of these fruit diseases, many of which are not easily distinguishable. It is important to diagnose fruit disease. Application includes identification of diseases in fruit that lead to loss of production and quality showed in reaping. It is also important to distinguish and group diseases of fruit. It is necessary to know what control factors need to be taken one year later to avoid losses in production. To overcome the errors that occur during manual classification, here too, researchers have proposed image detection method to classify the diseased fruits as good fruits to increase the quality of classification during export. In this case, the approach is based on CNN. It detects the quality of fruit layer by layer.

2. Literature Survey:

[1] Fruit Quality Detection and Classification using Computer Vision Techniques. Authors: T Thomas Leonid, Hemamalini, Nanthine T and Krithika For the discovery of fruits and standard conservation, a wide range of approaches have been developed. Deep learning subsets like Tensor Flow Lite have been used to provide fruit identification that is suitable for use by supporters. Micro. Fruit recognition with Arduino and Tensor Flow has been considered in a simple back-to-back tutorial using the Tensor Flow Lite Micro module. Tiny ML is skilled at doing a lot on the device with little cash in small-scale form-factors, minor vigor, and low-cost silicon. When the aim is close, the sampling is done on the RGB sensor, which can be seen as a 1pixel video color camera. The tutorship demonstrates a simple but comprehensive back-to-back. In the absence of a sophisticated foundation in machine learning, little ML can be used quickly. With the use of deep learning, fruit recognition from pictures creates a unique, superior data set of visuals containing fruits.

[2] Passion Fruit Disease detection using Image processing Authors: S. B. D. H. Dharmasiri, S. Jayalal Recently, many people have done researches for detecting fruit and vegetable diseases using image processing and deep learning. According to the research paper [4], authors had used image processing technology to identify the pomegranate diseases. Image preprocessing was the first step of the methodology. Image resizing was done under the image preprocessing. Because digital camera had been used to capture the images in this study, the size of those images was very large and takes more time to process. So, all images were resized to 300 x 300 PX. Morphology, color and CCV features were used for feature extraction. K-means clustering technique was used for partitioning the training dataset according to their features. After the clustering, SVM was used for classification to identify the image as infected or non-infected. An intent search technique was provided to find the user intention. The best result was got using morphology feature extraction. Experimental evaluation of this approach was effective and 82% accurate to identify pomegranate disease. In paper [5], the authors presented.

[3] Disease Detection in Pomegranate Plant using Image processing Technique. Author: Sahebgouda, Suamana M Using plant leaf pictures, the authors from [3] suggested a strategy for identifying plant diseases. They also talk about how to identify diseases using feature extraction and segmentation. A strategy for picture division that extricates the range of intrigued based on the RGB values of the image targeted location was proposed in the work [2]. A study by the authors of [4] used various edge detection techniques to identify the red blood cell's borders (RBCs). A neural network and image processing-based pomegranate illness detection system were proposed in [5]. Along with Fruit Spot, Fruit Rot, Leaf Spot, and Bacterial Blight, there are other pomegranate diseases that are discussed. The development of the Grey Level Cooccurrence Matrix (GLCM) provides statistical texture alternatives. A software programmed for image processing was suggested by authors of the work [6] to identify and classify plant leaf disease and proposed a clever irrigation system with a camera integrated into it embedded system that takes an image, processes it, and then displays the results, with this outcome being less precise in identifying the illness [7]. A approach utilizing Artificial Neural Networks (ANN) for the early identification of plant diseases was presented by in the work [8]. With this technology, accuracy can reach 91%.

Different plant diseases are identified utilizing combinations of textures, colors, and features. An OpenCV technique for Canny Edge 5 Detection that relies on various acceptable limits and is helpful for the detection of borders in the picture was proposed in [9]. The authors of [10] suggested a Grab Cut color picture segmentation method based on an area of interest (ROI). Segmentation takes place inside the ROI, which the user selects in the form of a rectangle.

3. PROPOSED SYSTEM

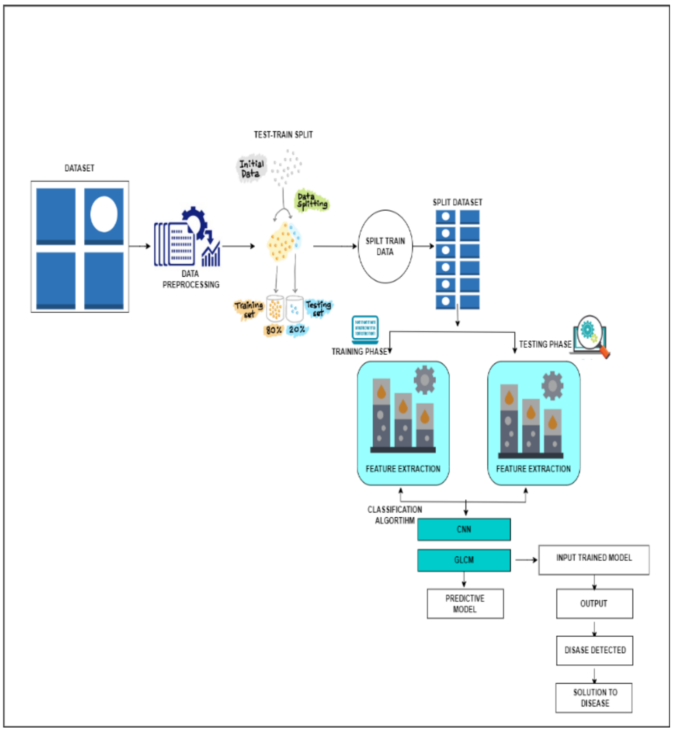

Figure 3.1: System Architecture

System Overview

The android application platform provides a comprehensive and user-centric solution to enhance quality of fruit by detecting its disease and providing cure for the same with the help of image processing. At its core, the system comprises several interconnected modules and functionalities designed to deliver a seamless and effective, low-cost solution for users' identification while purchasing fruits or producing it.

Key Components:

1. **Dataset:** A collection of diseased and normal fruit images (fruit are: Apple, Guava and Lemon). The images of damaged/diseased and regular fruits were taken out from plant village database, with a few of the photographs being taken manually with an elevated camera. There are 5460 fruit photos of 11 types prevalent fruit diseases.

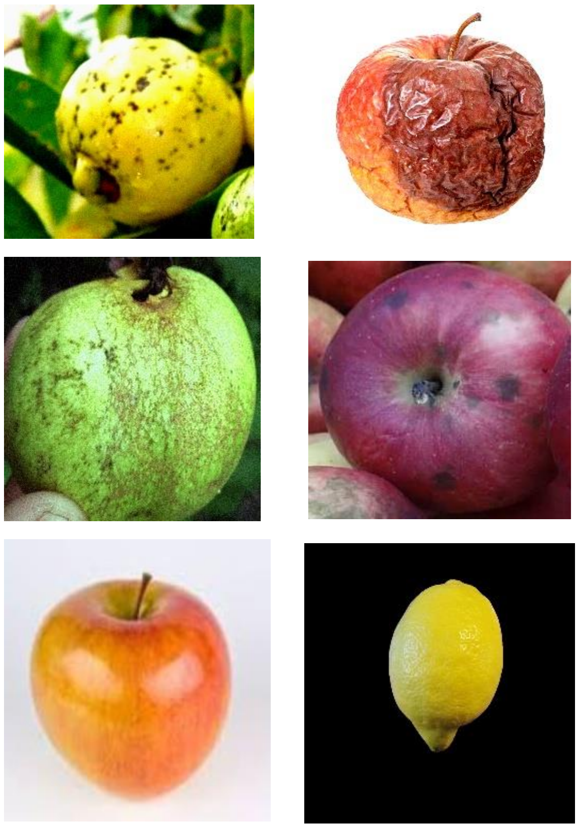

Figure 3.1.1: Dataset of fruits.

2. **Data Preprocessing:** Data preprocessing of images involves a series of operations performed on raw image data before feeding it into a machine learning model. These operations include resizing, normalization, cropping, augmentation (like rotation or flipping), and converting the images to a format suitable for the model input. The goal is to enhance the quality of the data, reduce noise, and make it easier for the model to learn meaningful patterns. It's a set of techniques applied to prepare image data for machine learning algorithms. Some

common libraries and tools used for image preprocessing in Python include OpenCV, PIL (Python Imaging Library), and TensorFlow/Keras preprocessing utilities. These libraries offer functions for tasks like resizing, normalization, cropping, and augmentation.

3. **Train-Test Split:**

Dataset is splinted for training and testing phases.

- **Training:** During the training phase, the model learns from a labelled dataset, which consists of input images along with their corresponding ground truth labels. The training process involves adjusting the model's parameters iteratively to minimize the difference between its predicted outputs and the ground truth labels. For supervised learning tasks, such as image classification or segmentation, the model learns to extract relevant features from the input images and make predictions based on those features. The training phase typically involves feeding batches of images through the model, computing the loss (error) between the predicted outputs and the ground truth labels, and updating the model's parameters using optimization techniques like gradient descent.

Figure 3.3.1: Training Accuracy chart

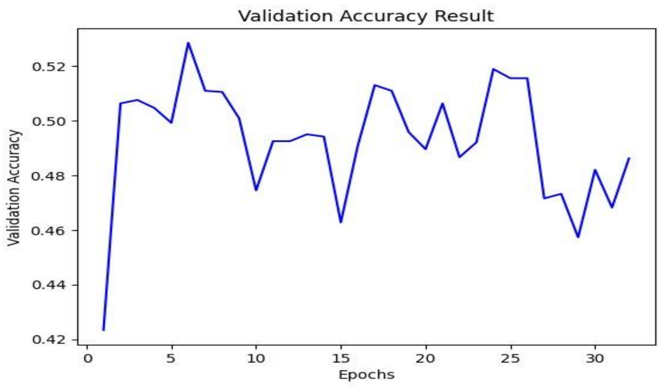

Figure 3.3.2: Validation Accuracy chart

- **Testing:** Once the model has been trained, it is evaluated on a separate dataset called the test set, which the model has not seen during training. The purpose of testing is to assess how well the model generalizes to new, unseen data. The test set allows us to measure the model's performance metrics such as accuracy, precision, recall, or F1 score. By evaluating the model on a separate test set, we can estimate its performance on real-world data and detect any overfitting or underfitting issues.

4. **Algorithm:**

   1. **CNN (Convolutional Neural Network):**

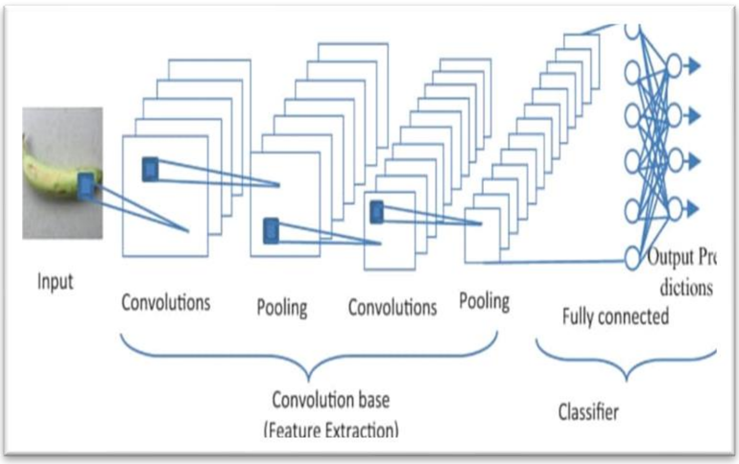

Figure 3.4.1.1: CNN working

Working of CNN algorithm:

- **Convolutional layers:** Convolutional layers are layers that apply filters to the input images in order to detect features such as edges, textures and shapes.

  Mathematical calculation: Let I be the input image, K be the convolutional kernel (also called filter), and S be the stride. The output of the convolutional layer is denoted as O.

  The convolution operation can be mathematically defined as:

$$O_{i,j} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I_{(i\cdot S+m),(j\cdot S+n)} \cdot K_{m,n}$$

Here:

- Oi,j represents the value at the i-th row and j-th column of the output feature map
- $I(i\cdot S+m),(j\cdot S+n)$ represents the pixel value at the $(i\cdot S+m)$-th row and $(j\cdot S+n)$-th column of the input image.
- Km,n represents the weight (or filter coefficient) at the m-th row and n-th column of the convolutional kernel.
- M and N are the dimensions of the convolutional kernel.
- S is the stride, which represents the step size at which the kernel is moved across the input image.

The summation is performed over all elements of the convolutional kernel, and the result is stored in the corresponding location in the output feature map.

- **Pooling layers:** These are layers that reduce the dimensionality of the feature maps generated by a convolutional layer while preserving important information.

  Max Pooling: Max pooling retains the maximum value from a set of values within a pooling window. If M is the size of the pooling window, the max pooling operation is defined as:

$$O_{i,j} = \max_{m=0}^{M-1} \max_{n=0}^{N-1} I_{(i\cdot M+m),(j\cdot M+n)}$$

Here:

- Oi,j is the value at the i-th row and j-th column of the output after max pooling.
- $I(i\cdot M+m),(j\cdot M+n)$ represents the pixel value at the $(i\cdot M+m)$-th row and $(j\cdot M+n)$-th column of the input feature map.
- M is the size of the pooling window.

Average Pooling: Average pooling calculates the average value within a pooling window. If M is the size of the pooling window, the average pooling operation is defined as:

$$O_{i,j} = \frac{1}{M\cdot N} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I_{(i\cdot M+m),(j\cdot M+n)}$$

Here:

- Oi,j is the value at the i-th row and j-th column of the output after average pooling.
- $I(i\cdot M+m),(j\cdot M+n)$ represents the pixel value at the $(i\cdot M+m)$-th row and

$(j \cdot M+n)$-th column of the input feature map.

- M is the size of the pooling window.

In both cases, the pooling window is moved across the input feature map with a certain stride, similar to the convolutional layer. The pooling layer helps reduce the spatial dimensions of the feature map, reducing the computational load and capturing the most essential information.

- **Fully connected layers:** These are layers that take the output of the previous layer and use it as a classification of the input image. The CNN learns to recognize objects and patterns in images by adjusting its internal parameters, such as weights and biases, based on the information it receives from the input and the output it wants (labels). Backpropagation, gradient descent, and other techniques are used to reduce the difference between the predicted and the actual outputs. Mathematical calculation: The fully connected layer is often represented by a matrix multiplication followed by a bias addition. Let's denote the input to the fully connected layer as X, the weight matrix as W, the bias vector as b, and the output as Y. The mathematical formula for a fully connected layer is given by:

$$Y = \sigma(W \cdot X + b)$$

Here:

- X is a column vector representing the flattened output of the preceding layers (e.g., the output of convolutional and pooling layers).
- W is the weight matrix connecting the neurons in the current layer to the neurons in the next layer. The matrix W has dimensions $(N\text{current}, N\text{next})$, where $N\text{current}$ is the number of neurons in the current layer, and $N\text{next}$ is the number of neurons in the next layer.
- b is the bias vector for the next layer, with dimensions $(N\text{next}, 1)$.
- $\cdot$ denotes the matrix multiplication.
- $\sigma(\cdot)$ is an activation function applied element-wise to the result of the matrix multiplication. Common activation functions include sigmoid (σ), hyperbolic tangent ($\tanh$), or rectified linear unit (ReLU).

The fully connected layer connects each neuron in the current layer to every neuron in the next layer, which allows for learning complex relationships in the data. The activation function introduces non-linearity to the model, enabling it to learn and approximate more complex functions.

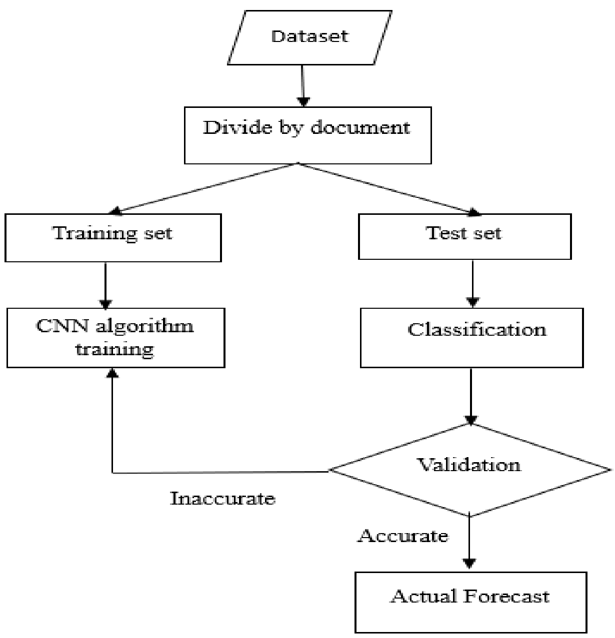

Figure 3.4.1.2: CNN Flowchart

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 03 | March - 2024 SJIF Rating: 8.176 ISSN: 2582-3930

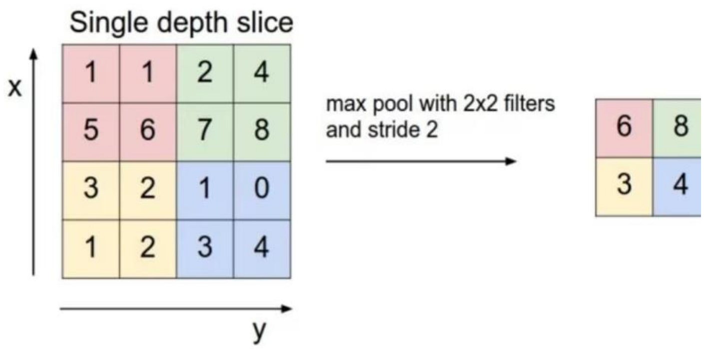

Figure 3.4.1.3: Example

4. RESULT

Figure 4.1: Actual image to the application

The above figure shows that the image of diseased/normal fruit is being uploaded as an input to the application/model by the user by click on the "Select Image" button, and further the result is shown by clicking on the "Detect" button.

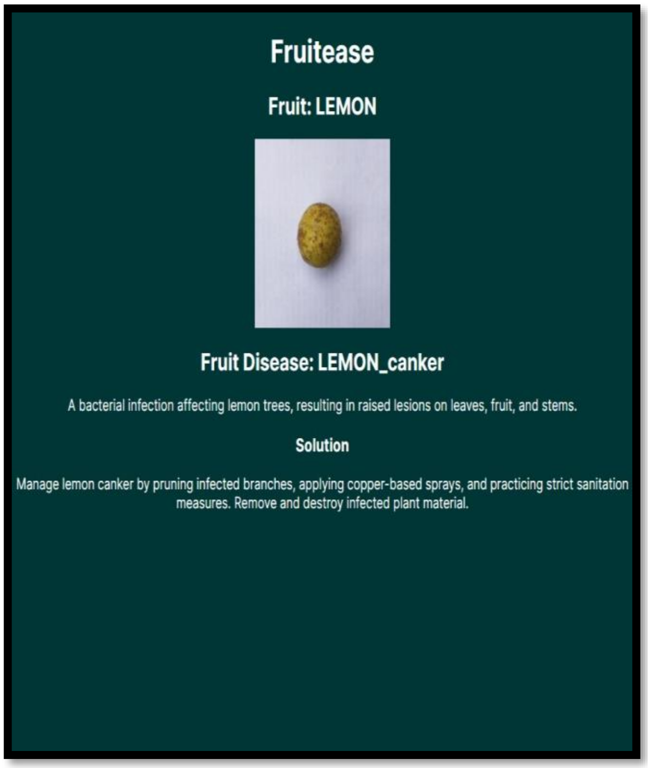

Figure 4.2: Output for the uploaded image

The above figure shows the actual output of the application. The inputted fruit image with the detected disease on it (disease name), the description of the disease and the possible solution for the disease recovery.

5. CONCLUSIONS

In this system, Image processing-based approach is proposed for fruit disease detection. This proposed system describes different techniques of image processing for several fruit species that have been used for detecting fruit diseases. The disease of the fruit is known and the cure is suggested. Agricultural field is a base of Indian economy, most of the population is dependent on income from agribusiness. So, for improvement in this field it is important to provide new technologies to increase profit ratio. We are proposing this system for better performance to agricultural area.

ACKNOWLEDGEMENT

The implementation and documentation of this project would not be succeeded without the kind support from individuals. First of all, we would like to express our special gratitude to Mr. S. H. Sangale Project Guide, who always gives us valuable advice and kind assistance to complete this project. Last but not least, we would like to thank the Faculty of Computer Technology, Maharashtra State Board of Technical Education for giving us the great knowledge. Finally, we would like to give our appreciation to our parents who support us since the beginning till the end of this project

REFERENCES

[1] T Thomas Leonid1 , Hemamalini2 , Nanthine T2 and Krithika2 1Assistant Professor(SG), Overall Description "Fruit quality detection and classification using Computer Vision Techniques", 2023

[2] S.B.D.H. Dharmasiri, S. Jayalal "Passion Fruit Disease Detection using Image Processing"

[3] Dr.M. Senthamil Selvi, Mrs.S. Jansi Rani, A. Nisha, K. Sneka, "Tomato Leaf Disease Detection using Image Processing".

[4] R.Ramya1 , Dr.P.Kumar2 , K.Sivanandam3 , M.Babykala3, "Detection And Classification Of Fruit Diseases Using Image Processing & Cloud Computing"

[5] Sudharshan Duth P, "Tomato Fruits Disease Detection Using Image Processing"

[6] Sahebgouda, Sumana M, "Disease Detection In Pomegranate Plant Using Image Processing Techniques"

[7] Xian Pan, Rajamohan Parthasarathy, Ruiyun Chao, "Application of Image Recognition Technology in Lychee Downy Mildew"