A Minor Project Synopsis on

Fruit Maturity Detection

(A CLI for detecting the maturity of a fruit)

Submitted to Manipal University, Jaipur

Towards the partial fulfilment for the Award of the Degree of

BACHELOR OF TECHNOLOGY

In Computer Science and Engineering

2022-2023

By

Saksham Rawat, 209301285

MANIPAL
UNIVERSITY JAIPUR
University under Section 2(f) of the UGC Act

Under the guidance of

Dr. Jay Prakash Singh

Department of Computer Science and Engineering School of Computer Science and Engineering Manipal University Jaipur

Jaipur, Rajasthan

Saksham Rawat

Jay Prakash Singh

Introduction

Fruit maturity detection is an important aspect of the fruit industry as it determines the timing of harvest and the quality of the produce. Inaccurate detection of fruit maturity can lead to economic losses for growers and reduce the overall yield. With advancements in technology, there is a growing need for more sophisticated methods of fruit maturity detection. This college project aims to develop a reliable and efficient system for the detection of fruit maturity. The system will use various techniques such as image processing, machine learning, and data analysis to accurately determine the maturity of fruit. The outcome of this project will provide a valuable tool for the fruit industry and improve the overall yield of fruit production. This system is designed with purpose of providing Inspection of Certain Fruits and assuring their quality. The major objective of this type of applications is to ensure a good product quality to the consumer.

Project Objective

Subject	Description
Literary Study	1. Fruit maturity detection: The process of determining the ripeness and readiness of a fruit for consumption or harvest. 2. Convolutional Neural Networks (CNNs): A deep learning algorithm used for image classification and recognition. 3. Image Processing Techniques: Techniques used to process and analyze digital images, including segmentation, feature extraction, and classification. 4. Fruit Image Datasets Pre-existing datasets of images of fruits used to train and test machine learning models.
Implementation of Proposed Approach	1. Data Collection: Collection of images of fruits at various stages of maturity. 2. Preprocessing of Images Resizing, cropping, and normalization of images.

	3. Building the CNN Model Design and implementation of the CNN model using TensorFlow. 4. Training the Model Training the model using the collected images and the preprocessed images. 5. Evaluation of Model Performance Evaluation of the model's accuracy and precision in detecting the maturity of fruits.
Comparative Analysis with Existing Approach	**Proposed Approach (CNN):** Uses a deep learning algorithm for image classification and recognition. High accuracy and precision in detecting fruit maturity. Time-consuming process of collecting and preprocessing images. **Existing Approach (Manual inspection):** Inspection of fruits by trained personnel. Quick and easily accessible method of determining fruit maturity. High degree of subjectivity and low accuracy in detecting fruit maturity.

Methodology

Week 1	Data Collection: The first step in creating a convolutional neural network is to gather the data. You'll need a large dataset of images of specific fruits, along with labels indicating the level of maturity of each fruit.
Week 2-3	Data Preprocessing: After collecting the data, you'll need to preprocess the images. This may include resizing the images, converting them to grayscale, and normalizing the pixel values.
Week 3-4	Model Design: Once you have preprocessed the data, you can start designing the convolutional neural network. You can start with a simple model with two or three convolutional layers, followed by one or two fully connected layers.
Week 6-7	Model Training: After designing the model, you can train it using the preprocessed data. You'll need to set the hyperparameters such as the learning rate, batch size, and number of epochs. During training, the model will learn the features of the images that are most indicative of the level of maturity of the fruit.
Week 8	Model Evaluation: After the model has been trained, you can evaluate it on a validation dataset. This will give you an idea of how well the model is able to generalize to new, unseen data. Model Deployment: Once you're satisfied with the performance of the model, you can deploy it as a command-line interface (CLI) application. This will allow users to input an image of an apple and receive a prediction of its level of maturity.

Facilities Required

The software and hardware requirements for implementing Fruit Maturity Detection differ depending on the specific use case and application.

Software:

- A Free Cloud Based Jupyter Notebook such as Google Colaboratory
- A programming language such as Python for image processing and analysis.
- Natural language processing libraries such as Keras, NLTK, Scikit-learn or OpenCV for Image pre-processing.
- Deep learning libraries such as TensorFlow and PyTorch for training deep learning-based models.

Hardware:

- A high-performance computer or server with a fast CPU and sufficient memory to handle the data processing and analysis.
- Storage space to store the training data and the results of the analysis.
- GPUs may be necessary for training deep learning-based models.

References

- Raja Hamza and Mohamed Chtourou, "*Apple Ripeness Estimation using Artificial Neural Network*," in *International Conference on High Performance Computing & Simulation, 2005*

- Raymond Erz Saragih and Andi W.R. Emanuel, "*Banana Ripeness Classification Based on Deep Learning using Convolutional Neural Network*," in *3rd east Indonesia Conference on Computer and IT, 2021*

- Kaveri Kangune, Dr. Vrushali Kulkarni, and Prof. Pranali Kosamkar, "*Grapes Ripeness Estimation using Convolutional Neural Network and Support Vector Machine*," in *Global Conference for Advancement in Technology (GCAT) Bangalore, India, 2019*