Automated Grading of Fruits Using Image Processing and Machine Learning Techniques

USER'S GUIDE

Nehemiah Joel Balozi

Table of Contents

Part A: Introduction

Welcome to the Automated Fruit Classification and Sorting System documentation. This system aims to revolutionize agricultural practices in Tanzania by providing an efficient and accurate solution for grading and sorting fruits. With a focus on crops such as oranges, tomatoes, and apples, this system leverages advanced image processing and deep learning techniques to automate the quality assessment and sorting process.

Purpose and Goals

The primary goal of this system is to address the challenges faced by farmers in achieving high productivity due to outdated farming techniques, inadequate labor skills, and market price fluctuations. By automating the grading and sorting of fruits, we aim to enhance efficiency, reduce labor costs, and improve the overall quality of agricultural produce in Tanzania.

Target Audience

This documentation is intended for farmers, agricultural workers, and anyone interested in leveraging technology to improve agricultural practices. Whether you are a novice or an experienced user, this guide will provide you with the necessary information to understand, install, and utilize the Automated Fruit Classification and Sorting System effectively.

Part B: System Requirements

Supported Operating Systems

- Windows 10 (recommended) or newer

Python Version

- Python 3.11.4 (or higher)

Hardware Requirements

- Processor: Intel Core i5 or equivalent
- RAM: 8GB (recommended)
- Storage: Sufficient disk space for storing images and model files

Part C: Installation Guide

Installing Python

1. Download the latest version of Python from the official website (https://www.python.org/).
2. Run the installer and follow the on-screen instructions to complete the installation.
3. Ensure that Python is added to the system PATH during installation for easy access from the command line.

Setting Up the Development Environment

1. Install Visual Studio Code (or any preferred code editor).
2. Open Visual Studio Code and install the necessary extensions for Python development.
3. Clone or download the project repository from GitHub (if applicable).
4. Open the project folder in Visual Studio Code.

Install Dependencies

1. Open the command prompt or terminal.
2. Navigate to the project directory.
3. Use pip to install the required libraries, which are

    a. Numpy, A fundamental package for scientific computing with Python, providing support for multi-dimensional arrays and matrices.

    b. Opencv-python, OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. The "opencv-python" package provides Python bindings for OpenCV

    c. Pillow, Pillow is a Python Imaging Library (PIL) fork. It adds image processing capabilities to Python interpreter

    d. Tensorflow, TensorFlow is an open-source machine learning framework for high-performance numerical computations. The "tensorflow" package provides the TensorFlow Python API.

    e. Scikit-learn, Scikit-learn is a machine learning library for Python. It features various classification, regression, and clustering algorithms, including support vector machines, random forests, gradient boosting, k-means, and DBSCAN

Part D: Functionality Overview

Functionality Overview

Our system employs deep learning techniques, specifically convolutional neural networks (CNNs), to automate the process of fruit sorting. It utilizes image processing and analysis to categorize fruits (oranges, tomatoes, and apples) based on their visual features such as color, texture, shape, and size.

The system consists of the following components;

Image Preprocessing: Input images are resized and normalized for consistency in analysis.

Model Training: CNN models are trained on a diverse dataset of multifruit images to learn the patterns and characteristics associated with different fruit types and qualities.

Model Loading: Trained models are loaded into memory for inference during the classification process.

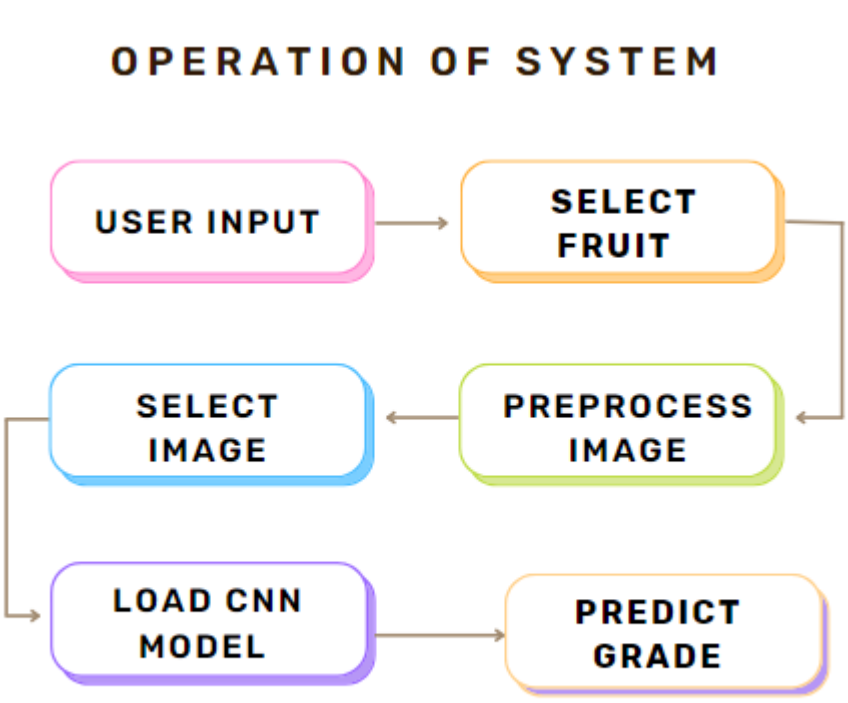

Figure 1: Operation mechanism of the system

Part E: Usage Instructions

Launch the Application

Open the Python script (provided) in Visual Studio Code.

Run the script to launch the graphical user interface (GUI).

Input Fruit Type

Enter the type of fruit (oranges, apples, or tomatoes) in the designated field.

Classify Image

Click on the "Classify Image" button to initiate the image classification process.

Select a sample image of the corresponding fruit type when prompted.

View Results

The predicted class and grade (one or two) of the fruit will be displayed.

The predicted image will also be displayed for visual inspection.

Part F: References

[1] Putri, I. "Design of orange grading system based on real time image processing." *IOP Conference Series. Earth and Environmental Science*. Vol. 644. No. 1. IOP Publishing, 2021.

[2] Alhashi, Osama A., et al. "Grading of Apples and Oranges by Image Processing." (2016).

[3] Kamble Apeksha B., et al. "Grading and Sorting of Apple by Using Image Processing" *International Research Journal of Engineering and Technology (IRJET)*. Volume. 06 Issue. 03. Mar 2019

[4] Li, Yanfei, et al. "Apple quality identification and classification by image processing based on convolutional neural networks." *Scientific Reports* 11.1 (2021). 16618

[5] Jhawar, Jyoti. "Orange sorting by applying pattern recognition on color image." *Procedia Computer Science* 78 (2016). 691-697.

[6] Nehemiah, B. Grading of Oranges Quality Based on Image Processing. Final Project Report, 2023. Available at the United African University of Tanzania Library