

# Computer Vision-Based Fruit Grading System Using Machine Learning Techniques

Arpitha Sumesh

*School Of Computing And Information Technology, REVA University  
Bangalore, India*

Reshma Panibhate

*School Of Computing And Information Technology, REVA University  
Bangalore, India*

Sowmya Sundari L K

*Assistant Professor*

*School Of Computing And Information Technology, REVA University  
Bangalore, India*

---

## **Abstract**

The agriculture sector plays a key role in the economic development of India. The task of fruit grading is vital in the agricultural industry because there is a great demand for high-quality fruits in the market. However, fruit grading by humans is inefficient, labor-intensive, and prone to error. The automated grading system not only speeds up the time of processing but also minimizes errors. The fruit grading system is a method used to categorize fruits based on various factors such as size, shape, color, texture, and overall quality. This system ensures consistency and standardization in the fruit market, facilitating fair trade practices and consumer satisfaction. Grading criteria may vary depending on the type of fruit and regional regulations, but generally, fruits are sorted into different grades such as premium, standard, and utility based on their appearance and taste. Premium-grade fruits typically exhibit uniformity in size, shape, and color, with minimal blemishes or defects, making them more desirable and commanding higher prices. Standard-grade fruits may have slight imperfections but still meet acceptable quality standards for consumption. Utility-grade fruits often have noticeable defects or irregularities and are typically used for processing or juicing rather than being sold fresh. Fruit grading not only helps buyers make informed decisions but also incentivizes growers to maintain high standards of cultivation and post-harvest handling practices to meet market demands.

**Index Terms**—Fruit Grading, Computer vision, Frontend and Backend, Machine Learning, CSS, HTML, Python, Fruit category, Database, Grading, Agriculture, Distributors, Packaging.

---

Date Of Submission: 24-04-2024

Date Of Acceptance: 04-05-2024

---

## **I. Introduction**

The introduction of the Fruit Grading System project sets the stage for understanding its purpose and scope. In this project, we aim to develop a comprehensive system for grading fruits based on various parameters such as size, color, weight, and quality. The system will be designed to cater to the needs of fruit producers, distributors, and retailers who require an efficient and accurate method of assessing the quality of fruits for sale or distribution. To achieve this goal, we have chosen to utilize a combination of frontend and backend technologies. For the frontend, we will be using HTML, CSS, and JavaScript to create a user-friendly interface that allows users to interact with the system seamlessly. The frontend will be responsible for displaying information, collecting user inputs, and providing feedback on the grading process. On the backend, we will be employing Python Flask as the web framework to handle server-side logic and communication with the frontend. Flask provides a lightweight and flexible framework for building web applications, making it an ideal choice for our project. Additionally, we will be utilizing MySQL as the database management system to store and manage data related to fruits, grading criteria, and user information. MySQL offers robust features for data storage and retrieval, ensuring the scalability and reliability of our system. Overall, the Fruit Grading System project aims to streamline the process of grading fruits by leveraging modern web technologies and database management systems. By providing a user-friendly interface and robust backend functionality, we seek to

enhance efficiency and accuracy in fruit grading operations, ultimately benefiting producers, distributors, and consumers alike.

## **II. Literature Survey**

The literature survey conducted for the Fruit Grading System project involved a comprehensive review of existing research, studies, and technologies related to fruit grading, image processing, and database management. Numerous scholarly articles, academic papers, and industry reports were examined to gain insights into the current state-of-the-art techniques and methodologies employed in fruit grading systems. These sources provided valuable information on various aspects such as the challenges faced in manual grading processes, the benefits of automation through image processing algorithms, and the importance of database management in storing and managing grading data efficiently. Additionally, the survey focused on exploring advanced image processing algorithms like ResNet50 and InceptionV3, which have demonstrated exceptional performance in fruit recognition and classification tasks. These algorithms leverage deep learning techniques to extract features from fruit images and accurately assess their quality based on visual cues. Furthermore, the literature survey highlighted the significance of frontend technologies such as HTML, CSS, and JavaScript in creating user-friendly interfaces for interacting with grading systems, as well as the role of backend frameworks like Python Flask in handling server-side logic and communication with databases. Overall, the literature survey provided valuable insights and knowledge that informed the design and development of the proposed Fruit Grading System, ensuring that it incorporates the latest advancements and best practices in the field.

## **III. Problem Statement**

The Fruit Grading System project addresses the inefficiencies and inconsistencies inherent in traditional fruit grading methods. Currently, the process relies heavily on manual inspection, leading to subjective assessments and time-consuming operations. Moreover, the lack of standardized criteria often results in discrepancies and inaccuracies in grading outcomes. This project seeks to overcome these challenges by developing a comprehensive software solution that automates the grading process using advanced machine learning algorithms. By integrating frontend, backend, and database technologies, the system aims to streamline fruit grading operations, improve accuracy, and enhance efficiency for stakeholders involved in fruit production and distribution.

## **IV. Objectives**

The main objectives of this system are: The system should accurately assess the quality and attributes of the fruits being graded. This includes factors such as size, color, ripeness, and external defects. Enhance accessibility and scalability of fruit grading operations through modern web technologies and database management systems. The system aims to automate the process of grading fruits, reducing the time and labor required compared to manual grading methods. This can lead to increased efficiency in fruit processing facilities. Implementing a fruit grading system can help reduce costs associated with manual labor and potential losses due to inconsistent grading. Create a user-friendly interface using HTML, CSS, and JavaScript to facilitate interaction with the system.

## **V. Scope**

The scope of the Fruit Grading System project encompasses the development of a comprehensive software solution designed to revolutionize the fruit grading process. Utilizing frontend technologies such as HTML, CSS, and JavaScript, alongside Python Flask for backend logic and MySQL for database management, the system aims to provide a user-friendly interface for inputting data and receiving real-time feedback on fruit grading. Additionally, the integration of advanced machine learning algorithms like ResNet50 and InceptionV3 enhances grading accuracy. Ultimately, the project seeks to streamline fruit grading operations, improve efficiency, and reduce manual effort across the supply chain, ultimately benefiting producers and healthcare professionals alike to create an intuitive user interface for interacting with the system. Python Flask is then employed as the backend framework to handle server-side logic and facilitate communication between the frontend and the underlying algorithms. Machine learning algorithm InceptionV3 is integrated to automate the grading process based on predefined criteria. Finally, MySQL is used as the database management system to store and manage data related to fruits, grading criteria, and user information. Throughout the development process, testing and validation are conducted to ensure the system's accuracy, efficiency, and reliability. At the beginning of our project, we formed groups and modularized the project. Important points of consideration were to define and visualize all the objectives clearly, gather requirements and evaluate them, consider the technical requirements needed and then collect technical specifications of various peripheral components (Hardware) required, analyze the coding languages needed for the project, define coding strategies, analyze future risks / problems, define strategies to avoid these risks else define alternate solutions to these risks,

check financial feasibility.

## **VI. Methodology**

The methodology of the Fruit Grading System project involves a multi-stage approach to develop a comprehensive software solution for grading fruits. Initially, requirements gathering and analysis are conducted to understand the needs of stakeholders and define grading criteria. Next, frontend technologies like HTML, CSS, and JavaScript are utilized

## **VII. Hardware And Software Requirements**

Hardware: Processor: Intel Core i3 or more. RAM: 4GB or more. Hard disk: 250 GB or more.

Software: Operating System : Windows 10, 7, 8. Ana-conda Navigator, Spyder, Python, flask. MYSQL. Anaconda Navigator: Anaconda Navigator is a graphical user interface(GUI) included with the Anaconda distribution, which is a popular open-source platform for data science and machine learning tasks. Navigator simplifies the management of Python packages, environments, and tools used in data analysis and scientific computing. It provides an intuitive interface to create and manage virtual environments, allowing users to isolate projects and their dependencies. Navigator also offers a package manager that makes it easy to install, update, and uninstall various Python packages and libraries. Through Navigator, users can launch applications like Jupyter Notebook, JupyterLab, Spyder, and more, enhancing their productivity and facilitating streamlined workflows in data-related projects. Python: Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed. Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very

effective.

MySQL: MySQL is well known as the world's most widely used open-source database (back-end). It is the most supportive database for PHP as PHP-MYSQL is the most frequently used open-source scripting database pair. The user-interface which WAMP, LAMP and XAMPP servers provide for MySQL is easiest and reduces our work to a large extent.

FLASK: A Flask is a Web Application Framework that is built with Flexibility and Speed In the Mind. Flask is Built in Python, which many data Scientists are familiar with. Flask takes care of the Environment and Project setup involved in web Applications Allowing the Developer to focus on their application rather than thinking about HTTP, routing, dataset etc. Flask allows Data Scientists to create simple Single page Applications and one should Help or look into if they want to create Products for Consumers Flask is a micro web framework written in Python. It is classified as a microframework because it doesn't require particular tools or libraries. its no database abstraction layer, form validation, or the other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions which will add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and a number of other common framework related tools Flask was created by Armin Ronacher of Pocco, a world group of Python enthusiasts formed in 2004. According to Ronacher, the thought was originally an April Fools joke that was popular enough to form into significant application. When Ronacher and Georg Brandl created a bulletin board system written in Python, the Pocco projects Werkzeug and Jinja were developed. Flask has become popular among Python enthusiasts. As of October 2020, its second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the foremost popular web framework within the Python Developers Survey 2018.

## **VIII. Conclusions**

In conclusion, the Fruit Grading System project represents a significant advancement in the field of fruit grading, offering a comprehensive solution that leverages modern technologies to enhance accuracy, efficiency, and user experience. The user-friendly interface and centralized data management ensure ease of use and data integrity, while the scalability and flexibility of the system cater to the evolving needs of the fruit industry. Overall, the Fruit Grading System project offers a promising solution to improve fruit grading operations for producers, distributors, and retailers, ultimately benefiting the entire supply chain and contributing to the growth and efficiency of the industry.

### **IX. Future Enhancements**

In the future, the Fruit Grading System project can be further enhanced through various modifications and updates. One potential area for improvement is the integration of machine learning models trained on larger and more diverse datasets to improve the accuracy of fruit grading. Additionally, incorporating feedback mechanisms from users and stakeholders can help refine the grading criteria and algorithms over time. Furthermore, expanding the system to support additional features such as automated sorting and packaging based on grading results can further streamline fruit processing operations.