Jurnal Kecerdasan Buatan dan Pembelajaran Mesin(JKBPM)
Vol 1, No 1, 2026, 1- 22 https://doi.org/10.35912/jkbpm.v1i1.6025

Website-Based Convolutional Neural Network for Banana Freshness Detection to Prevent Health Risks

Arya Gunawan[1*], Ketut Artaye[2]
Institut Informatika dan Bisnis Darmajaya, Lampung, Indonesia[1,2]
aryakk008@gmail.com[1] , artajaya@darmajaya.ac.id[2]

Article History:
Received on 12 December 2025
1st Revision on 26 December 2025
2nd Revision on 03 January 2026
3rd Revision on 15 January 2026
Accepted on 28 January 2026

Abstract

Purpose: This study aimed to develop a web-based system to detect the freshness level of Cavendish bananas using image-based classification to help assess fruit quality and reduce losses during distribution.

Methodology/approach: This study used a dataset of 1,030 labeled images of Cavendish bananas categorized into four classes: fresh, ripe, unripe, and rotten. Image preprocessing techniques, such as resizing, normalization, and data augmentation, were applied. A Convolutional Neural Network (CNN) model was trained using Python with machine learning libraries, and the trained model was integrated into a web application built with Streamlit to enable real-time image-based predictions.

Results: The proposed CNN model achieved an accuracy of 99%, with average precision, recall, and F1-score values of 0.99. The web-based system successfully provided real-time freshness predictions through image uploads or camera captures.

Conclusions: This study confirms that the web-based CNN system effectively detects Cavendish banana freshness with high accuracy. The integration into a Streamlit application enables real-time and practical use for quality assessment. The system demonstrates the potential of machine learning to support agricultural quality control, reduce post-harvest losses, and enhance food safety.

Limitations: This study was limited to Cavendish banana images and controlled image conditions; therefore, performance may vary when applied to other banana varieties or different lighting environments.

Contributions: This study contributes to the application of machine learning and computer vision in agricultural quality assessment. The system provides a practical tool for farmers, distributors, and consumers to objectively evaluate banana freshness, reduce economic losses, and support public health.

Keywords: *Banana Freshness Detection, Convolutional Neural Network, Image Classification, Machine Learning, Streamlit*

How to Cite: Gunawan, A., Artaye, K. (2025). Website-Based Convolutional Neural Network for Banana Freshness Detection to Prevent Health Risks. *Jurnal Kecerdasan Buatan dan Pembelajaran Mesin*, 1(1), 1-22.

1. Introduction

Indonesia is one of the world's largest producers of tropical fruits, with bananas as a leading commodity of high economic value(Kalsum, Subandi, & Wiratma, 2023). The Cavendish variety is the most widely cultivated banana variety owing to its uniform size, sweet taste, and high durability during distribution (Khairina et al., 2025). However, maintaining banana freshness after harvest remains a significant challenge, particularly during the storage and transportation stages (Marhaen, Kusmiadi, & Ropalia, 2023). Bananas that exceed their optimal ripeness period undergo changes in texture and peel color, as well as an increased risk of spoilage, which ultimately reduces their quality and market value (Febriansyah, Oktavianus, & Nasrullah, 2023).

Advances in Artificial Intelligence (AI), particularly in the field of computer vision, have opened opportunities for the application of intelligent systems in automated fruit quality analysis (Noordianty, Najma, & Nurlaela, 2024). The Convolutional Neural Network (CNN) method has proven effective in recognizing visual patterns, such as color, texture, and shape, that are difficult to detect through human observation (Raja et al., 2025). This study was conducted at a fruit processing company located in the Lampung region, an area with relatively high banana production (Marhaen et al., 2023). The objective of this study was to develop a CNN-based banana freshness detection system integrated into an interactive web-based platform (Martinus et al., 2025). The proposed system is expected to assist in the fruit quality inspection process by providing fast, accurate, and consistent assessments without requiring additional equipment (Rozan, Rozaki, Wulandari, & Distrianada, 2024).

2. Literature Review and Hypothesis Development

2.1 Cavendish Banana

Cavendish bananas are among the most widely consumed banana varieties in the world because of their uniform shape, soft texture, and distinctive sweet taste (Wahyuni, Noer, & Trisnanto, 2022). In addition to fresh consumption, various parts of bananas, such as peels, have been widely utilized for non-food applications. Previous studies have reported that banana peel powder can be used as a biocoagulant to reduce water turbidity, indicating the broad potential of banana-based materials beyond consumption (Haritha, Raju, Jacob, & Thampi, 2024). This fruit has high economic value and is one of Indonesia’s leading export commodities.

During the distribution process, bananas undergo physical changes, such as the appearance of brown spots, peel darkening, and softening of texture, as the level of ripeness increases (Ikhtiar, Riswan, Asrina, & Puspitasari, 2024). These visual changes serve as primary indicators of freshness that can be observed and analyzed using digital image processing approaches (Wijaya et al., 2024). Through image processing techniques and the application of Convolutional Neural Network (CNN) algorithms, changes in banana peel color and texture can be automatically and objectively analyzed (Syaharani, Budianto, & Adam, 2024).

2.2 Digital Image Processing

Digital Image Processing is the transformation of images into a digital form for analysis using computer algorithms. Its purpose is to extract important information from images through stages such as contrast enhancement, segmentation, edge detection, color transformation, and pattern recognition (Marpaung, Aulia, & Nabila, 2022). In the context of this study, digital image processing plays a crucial role in highlighting changes in color and texture on the peel of Cavendish bananas to detect freshness levels (Ferbangkara et al., 2025). This stage serves as a foundation before the images are further processed using machine learning models. Optimal image processing results can improve the model’s accuracy in distinguishing fruit categories, such as fresh, ripe, unripe, and spoiled.

2.3 Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is an artificial neural network architecture specifically designed to process image data or other forms of spatial data (Fatahna, Sari, Wulanningrum, & Utomo, 2025). CNNs have a high capability of recognizing complex patterns, such as edges, colors, and textures, making them widely used in object recognition, image classification, and visual feature detection tasks (Nugraha & Hermawan, 2023). Unlike traditional neural networks (feedforward neural networks), CNNs employ convolutional layers to capture local features within images while preserving spatial relationships between pixels.

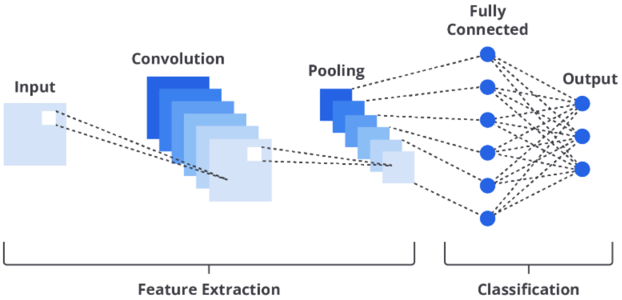


Figure 1. CNN Architecture

The CNN architecture generally consists of several main layers, as follows:

1. Input Layer
   This layer receives the initial data to be processed. Each neuron represents a single feature of the input image, such as the pixel intensity values. The size of the input layer was adjusted according to the dimensions of the image used.
2. Convolutional Layer
   This layer performs a convolution operation between the input image and filters (kernels) to produce feature maps that represent important features, such as edges and texture patterns. The filters slide across the image and progressively extract features from simple to complex levels.
3. Activation Layer
   After convolution, nonlinear activation functions, such as ReLU, are applied to introduce nonlinearity and enable the network to learn complex patterns. This layer does not contain trainable parameters but serves to transform the output to be more adaptable to variations in the image data.
4. Pooling Layer
   Pooling is used to reduce data dimensionality without losing important information by selecting either the maximum (max pooling) or average (average pooling) value within a specific region. This process reduces the number of parameters, accelerates the computation, and improves the model robustness to positional or lighting variations.
5. Fully Connected Layer
   This layer connects all neurons from the previous layer to integrate the extracted features and produce a final decision. At this stage, the features generated by the convolutional layers are flattened into a one-dimensional vector for the classification process.
6. Output Layer
   The final layer is responsible for generating predictions based on processed information. In multi-class classification tasks, the softmax activation function is used so that each neuron represents the probability of the class.

In general, CNNs excel because of their ability to automatically recognize visual patterns without requiring manual feature engineering (Subagiya, 2023). In addition, CNNs are invariant to translation and scale, allowing them to recognize objects even when there are changes in their position or rotation. In this study, CNNs were used to classify the freshness levels of Cavendish bananas based on changes in peel color and texture.

2.4 Streamlit as a Web-Based Platform

Streamlit is an open-source Python-based framework that enables developers to quickly build interactive web applications without requiring front-end development skills (Christian & Al Idrus, 2023). This framework supports direct integration with machine learning models, such as TensorFlow or Keras, allowing the prediction results to be displayed in real time through a web interface. The advantages of Streamlit include ease of implementation, efficient data visualization, and the ability to display the prediction results and evaluation graphs without reloading the page. In this study, Streamlit was used to present banana freshness classification results to users, either through image uploads or direct camera input.

2.5 Framework

Frameworks are used to support system development, model training, and integration between the machine learning model and application interface. The three main frameworks used in this study were TensorFlow, Keras, and Streamlit.

1. TensorFlow
   TensorFlow is an open-source framework developed by Google that supports machine learning and deep learning processes. It was used to build the CNN architecture, train the banana image classification model, and manage the training parameters. Its main advantages include parallel computation capabilities and high scalability (Pramukti & Setiawan, 2025).
2. Keras
   Keras is a high-level API that operates on TensorFlow. This framework is used to define neural network structures, specify activation functions, and perform model parameter tuning in an easy and efficient manner (Siswanto, Dewi, Kholifah, Widhiati, & Aryani, 2023).
3. Streamlit
   Streamlit was used to develop an interactive web application that displays banana freshness detection results in real time. This framework enables direct integration between the CNN model and user interface without the need to write HTML, CSS, or JavaScript code.

2.6 Programming Language

Python libraries play an essential role in supporting all stages of the research process, ranging from image data processing and model training to result visualization. The combination of these libraries enables the system development process to be efficient, flexible, and accurate. The Python libraries used in this study included the following:

2.7 Python Libraries

Python libraries play an important role in supporting all stages of research, from image data processing and model training to result visualization. The combination of these various libraries enables the system development process to be more efficient, flexible, and accurate. The libraries used in this study include:

1. OpenCV (Open-Source Computer Vision Library)
   OpenCV is an open-source library that is widely used in computer vision and digital image processing. In this study, OpenCV was utilized for image preprocessing tasks, such as resizing images to a uniform size, performing image augmentation to increase training data diversity, and converting color formats from RGB to other formats, such as grayscale or HSV. In addition, this library supports various operations, including filtering, blurring, and edge detection, which help improve the data quality before being fed into the CNN model.
2. Pillow (PIL)
   Pillow is a fork of the Python Imaging Library (PIL) that is designed for digital image processing. In this study, Pillow was used to open, display, crop, rotate, and save the processed images during the data preprocessing stage. This library is particularly useful for adjusting image formats and ensuring consistent image sizes and orientations before training with CNN models.
3. NumPy (Numerical Python)
   NumPy is a fundamental library for scientific computing in Python that supports mathematical operations, multidimensional array manipulation, and various linear algebra functions. In this study, NumPy was used to represent images as numerical arrays, normalize pixel values, and support matrix computations during convolution operations in CNN architectures. Its high processing speed also optimizes the model training time.
4. Pandas
   Pandas is a library that focuses on data processing and analysis using data frames. In this study, Pandas was employed to manage image metadata, such as file names, class labels, and model evaluation statistics. Furthermore, it assists in analyzing the model performance by presenting evaluation metrics such as precision, recall, and F1-score in a structured and easily interpretable format.

5. Matplotlib
   Matplotlib was used as the primary library for two-dimensional data visualization. In this study, it was used to display graphs of accuracy and loss progression during the CNN model training process, as well as to visualize the classification results using bar charts or line graphs. These visualizations help researchers understand the learning dynamics of the model and evaluate the effectiveness of the applied training parameters.
6. Seaborn
   Seaborn is a statistical data visualization library built on top of Matplotlib that provides a modern and informative graphical style. In this study, Seaborn was used to visualize the confusion matrix of the CNN prediction results, enabling the evaluation of the model's ability to distinguish between banana image classes (fresh, ripe, unripe, and spoiled). Additionally, it supports the presentation of data distribution plots, heatmaps, and comparative performance metric visualizations, making the analysis results easier to understand.

2.8 Tools

Tools play an important role in supporting the entire research process, from system development and model training to the storage of the experimental results. The use of appropriate tools can improve work efficiency, accelerate computational processes, and facilitate documentation and collaborations. The tools utilized in this study are described below.

1. Google Colab

Google Colab is a cloud-based computing platform built on Jupyter Notebook and is provided free of charge by Google. This platform allows users to execute Python code directly on cloud servers with support for Graphics Processing Units (GPU) and Tensor Processing Units (TPU), making it highly suitable for training deep learning models such as CNN. In this study, Google Colab was used as the primary environment for CNN model training because of its capability to handle intensive computations without requiring additional hardware. Furthermore, Colab supports direct integration with Google Drive, enabling easy storage and access to the model files and datasets. Its collaborative notebook feature also allows researchers to document and share their experimental results interactively.

2. Visual Studio Code (VS Code)

Visual Studio Code is a lightweight and flexible code editor that is widely used in modern software development. VS Code supports multiple programming languages and provides extensions that facilitate machine learning and web application development, such as Python, TensorFlow, and Streamlit. In this study, VS Code was used as the main development environment for building and managing a Streamlit-based web application that integrates the CNN model. Features such as the integrated terminal, debugging tools, and version control support enhance productivity and simplify application testing and maintenance.

3. Google Drive

Google Drive was used as an online storage medium for all research files, including image datasets, trained CNN models, evaluation results, and experimental documentation. The integration between Google Drive and Google Colab simplifies file synchronization and automatic updates. In addition to storage, Google Drive serves as a backup repository to ensure that all research outputs remain secure and accessible whenever needed. The use of Google Drive supports efficiency and portability in cloud-based research work flows.

2.9 Software Development Life Cycle (SDLC)

The Software Development Life Cycle (SDLC) is a systematic framework that defines the structured stages of software development, ranging from requirements analysis to maintenance. The SDLC is used to ensure that software is developed in a planned and efficient manner and in accordance with predefined objectives. The implementation of the SDLC aims to produce high-quality software that meets user requirements while minimizing the risk of errors during the development process. Each stage within the SDLC is interconnected and can be carried out sequentially or iteratively, allowing continuous evaluation and improvement.

In this study, the SDLC was employed as a methodological reference in the development of a web-based application for detecting banana freshness levels using the Convolutional Neural Network (CNN) method. By applying the SDLC, the application development process can be conducted in a structured and well-documented manner, making it easier to replicate in future research. This approach also assists researchers in managing system requirements, model implementation processes, functional testing, and application maintenance to ensure optimal performance (Surbakti, 2025).

3. Research Methodology

3.1 Research Flow

This study was conducted through several systematic and structured stages arranged sequentially to ensure that the development process of a banana freshness detection system based on Convolutional Neural Networks (CNN) runs effectively and in a well-directed manner. Each stage has a logical relationship with the others, starting from the conceptual phase to system implementation and evaluation. The research flow consists of seven main stages, as described below.

1. Literature Review
   In the initial stage, a literature review was conducted to obtain theoretical foundations relevant to this research topic. The reference sources included scientific journals, books, research reports, and articles discussing digital image processing, Convolutional Neural Network (CNN) algorithms, and web-based application development using the Streamlit framework. The purpose of this stage was to understand the concepts, methods, and previous studies that support the design of the system to be developed.
2. Field Observation
   After understanding the theoretical background, direct observations were performed on the research object, namely Cavendish bananas, at various ripeness levels. Observations were conducted in storage and processing environments to identify the visual characteristics at each stage, such as changes in peel color, texture, and spotting patterns. The results of these observations served as references for the image data collection process.
3. Data Collection
   In this stage, we collected a dataset consisting of Cavendish banana images representing four freshness categories: fresh, ripe, unripe, and spoiled. Images were acquired using a digital camera under various angles and lighting conditions to ensure data diversity and representativeness. All images were stored in separate folder structures according to their respective classes to facilitate labeling and model training.
4. Data Preprocessing
   The collected image data were further processed to prepare them for CNN model training. The preprocessing steps included resizing the images to a uniform size (150 × 150 pixels) and normalizing the pixel values using a rescale of 1. /255 to map them into a 0–1 range, and applied data augmentation techniques such as rotation, zooming, and horizontal flipping to increase data variation and prevent overfitting. This stage is essential to ensure optimal data quality during the learning process.
5. CNN Model Development
   At this stage, a Convolutional Neural Network (CNN) architecture for banana freshness classification was designed and developed. The model architecture consists of several layers, including convolutional, pooling, flatten, dense, and output layers with ReLU and softmax activation functions. The TensorFlow and Keras frameworks were used to facilitate the model design and training parameter configuration.
6. Model Training and Evaluation
   The designed CNN model was trained using the dataset divided into training, validation, and testing sets with a ratio of 70:15:15. The training process was conducted over several epochs to obtain the optimal performance. Model evaluation was performed by calculating performance metrics such as accuracy, precision, recall, and F1-score, and generating a confusion matrix to assess the classification performance for each class. The evaluation results served as the basis for determining whether the model achieved sufficient accuracy and stability of the model.

7. Model Integration into a Web Application
   After the CNN model achieved optimal accuracy, it was integrated into a web-based application developed using the Streamlit framework. The application allows users to upload banana images or capture images directly using a camera and receive automatic, real-time predictions of freshness levels. This stage also included user interface (UI) testing to ensure that the system was user-friendly and responsive across various devices.

Through this research flow, the developed system is expected to detect banana freshness levels rapidly, accurately, and efficiently. These stages are illustrated in Figure 3.1, which presents the overall research workflow.

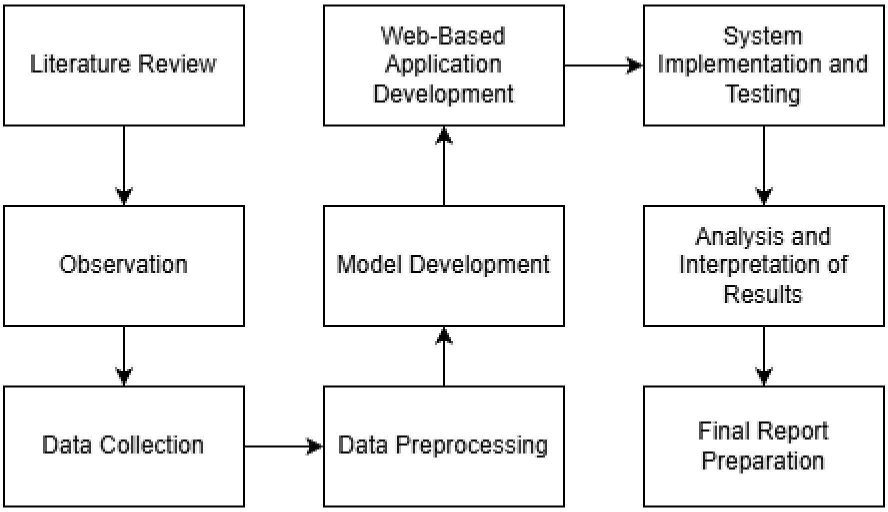

Literature Review
Web-Based Application Development
System Implementation and Testing
Observation
Model Development
Analysis and Interpretation of Results
Data Collection
Data Preprocessing
Final Report Preparation

Figure 2. Research Flow Diagram

3.2 Type of Research

This study employed a quantitative experimental approach aimed at analyzing banana image data through the application of Convolutional Neural Network (CNN) algorithms and evaluating model performance based on numerical metrics. This approach was selected because it provided objective and measurable results regarding the performance of the banana freshness detection system.

3.3 Data Sources

The data source used in this research consisted of primary data obtained directly through photographs of Cavendish bananas in plantation areas, warehouses, and sorting facilities at a fruit processing company located in the Lampung region. Each image was captured from various angles and under different lighting conditions to ensure the diversity of the dataset. A total of 1,030 images were used in this study, with a balanced distribution across the four classes: fresh, ripe, unripe, and spoiled.

3.4 Research Flow Diagram

The research flow diagram illustrates the systematic sequence of processes conducted, starting from data collection to model integration into a web-based application (Ramdany, Kaidar, Aguchino, Putri, & Anggie, 2024). This diagram provides a visual overview of the research workflow and the interrelationships among the processes at each stage. In general, the research workflow consists of eight main stages, as shown in Figure 3.2, which presents the Research Flow Diagram.

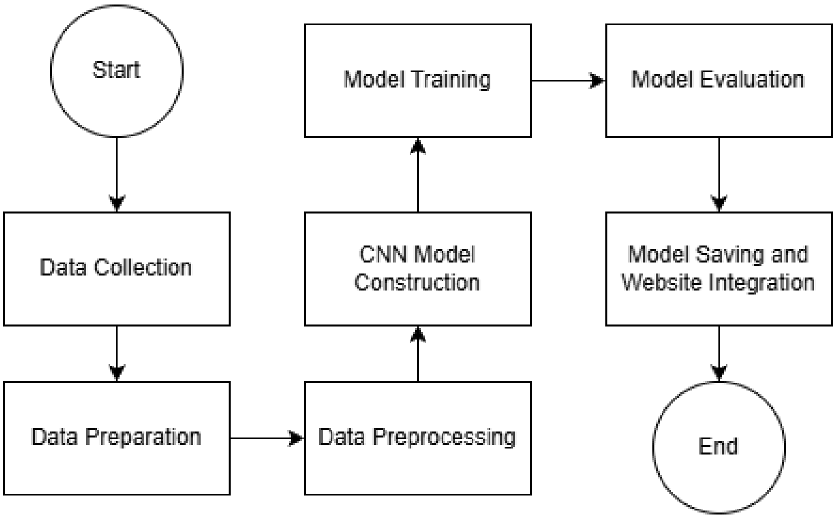


Figure 3. Research Flow Diagram

Explanation of the Diagram Stages

1. Data Collection
   At this stage, image data of Cavendish bananas were collected directly through image acquisition under various lighting conditions and viewing angles. Each image was labeled according to its freshness level (fresh, ripe, unripe, or spoiled). The collected data served as the basis for forming the dataset used for model training.
2. Data Preparation
   All image data were collected and organized into directories based on their respective classes, followed by verification of data completeness and class balance. The validated dataset was then divided into three subsets: 70% training data, 15% validation data, and 15% testing data.
3. Data Preprocessing
   The collected images were resized to 150 × 150 pixels and normalized using a rescale of 1. /255, and augmented using techniques such as rotation, zooming, and horizontal flipping. The purpose of this stage was to increase data variability and enhance the model's generalization capability.
4. CNN Model Development
   This stage involves designing the Convolutional Neural Network architecture, which consists of several main layers, including convolutional, pooling, flatten, dense, and output layers with softmax activation. The TensorFlow and Keras frameworks were used to build the model and configure the network parameters.
5. CNN Model Training
   The model was trained using the training dataset with a batch size of 32 and 30 epochs. During training, two main callbacks were applied: EarlyStopping, which automatically stopped training when performance improvement stagnated, and ModelCheckpoint, which saved the best-performing model based on the validation results.
6. Model Evaluation
   The trained model was evaluated using the testing dataset. The evaluation process involved calculating performance metrics, such as accuracy, precision, recall, and F1-score, as well as visualizing the confusion matrix to assess how well the model distinguished between different banana freshness classes. Model performance evaluation using the confusion matrix was conducted to calculate the following values:
   Calculation Formula:
   a. Accuracy

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{1}$$

   Purpose: To measure the percentage of correct predictions (including fresh, ripe, unripe, and spoiled bananas) from the total dataset.

b. Precision

$$Precision\ = \frac{TP}{TP+FP}\times 100\% \tag{2}$$

Purpose: To measure the model's consistency in predicting a specific class (e.g., the "fresh" banana class).

c. Recall *(Sensitivity)*

$$Recall\ = \frac{TP}{TP+FN}\times\ 100\% \tag{3}$$

Purpose: To measure the model's ability to correctly identify all relevant samples belonging to a specific class.

d. F1-Score

$$F1 - Score = 2\ \times \frac{Precision\ \times Recall}{Precision\ \times Recall} \tag{4}$$

The harmonic mean of precision and recall was computed to provide a balanced evaluation of the model performance, particularly on imbalanced datasets.

7. Model Storage and Implementation
   The best-performing model was saved in. h5 format to enable reuse without retraining. This stage also ensured the model's compatibility with the Streamlit-based web application framework.

8. Website Integration
   The CNN model was integrated into a Streamlit-based web application that enabled users to directly detect banana freshness levels. Users can upload images or use a camera, after which the system displays the classification results along with the model confidence score.

3.5 Application Development Method

The application development in this study adopts the System Development Life Cycle (SDLC) approach using the Waterfall model (Kurniyanti & Murdiani, 2022). The Waterfall model is a systematic, linear, and sequential approach in which each stage must be completed before proceeding to the next stage. This model is considered suitable because the development workflow of the banana freshness detection application follows clear and well-structured stages.

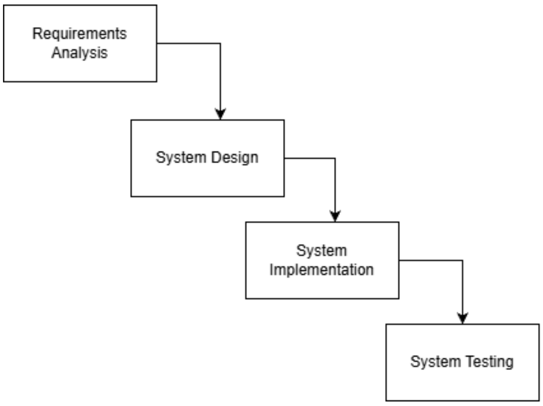


Figure 4. SDLC Waterfall Design Flow

Explanation of the SDLC Stages:

1. Requirements Analysis
   The requirements analysis phase aims to identify the functional and non-functional requirements of a web-based banana freshness detection system. This stage is crucial to ensure that the developed system meets user needs and operates according to the defined objectives of the study.

a. Functional Requirements

Table 1. Functional Requirements

No	Requirement	Description
1	Image Upload	Users can upload banana images from their gallery or capture them using a camera.
2	Category Prediction	The system classifies banana freshness levels as spoiled, ripe, unripe, or fresh.
3	Confidence Score Display	The system displays the model's confidence level as a progress bar and percentage value.
4	Category Explanation	The system provides educational information and recommendations based on classification results.

b. Non-Functional Requirements

Table 2. Non-Functional Requirements

No	Requirement	Description
1	User Interface	A simple and user-friendly interface was implemented using Streamlit.
2	Accessibility	The application can be accessed through a web browser without requiring additional installations.
3	Performance	Predictions were generated in less than 5 s.
4	Compatibility	The application can run on PCs or laptops with any operating system, as long as a web browser is available.

2. System Design

The design phase describes how the system operates in terms of workflow logic, user interface, and interaction with the CNN model. This stage serves as a blueprint for system implementation to ensure that all components function as intended by the user.

a. Use Case Diagram

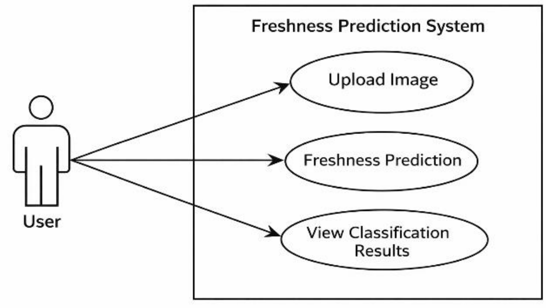


Figure 5. Use Case Diagram

Explanation:

The use case diagram illustrates the interaction between the users and the web-based fruit freshness detection system. In this system, users can perform three main activities: upload fruit images, perform freshness predictions, and view classification results. The process begins when the user uploads a fruit image to the system. Subsequently, the system executes the classification process using the CNN model to predict the freshness level of the fruit, such as unripe, ripe, or spoiled. The prediction results are then displayed to the user as a classification output. This diagram helps visualize the user’s role and interaction flow with the main features provided by the system.

b. Activity Diagram
The activity diagram illustrates the workflow of user interaction with the system for banana freshness detection using a web-based application. This diagram shows the main steps performed by the users, starting from the initial interaction with the system until the detection results are obtained.

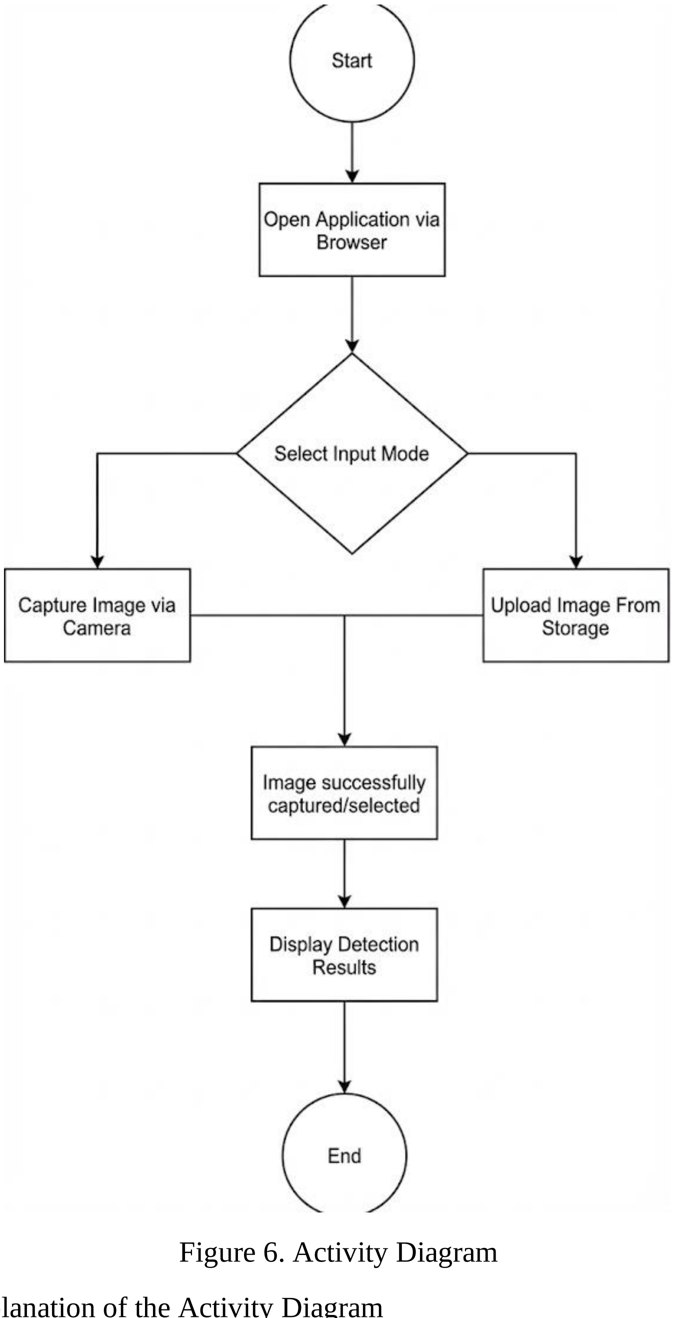

Start
Open Application via Browser
Select Input Mode
Capture Image via Camera
Upload Image From Storage
Image successfully captured/selected
Display Detection Results
End

Figure 6. Activity Diagram

Table 3. Explanation of the Activity Diagram

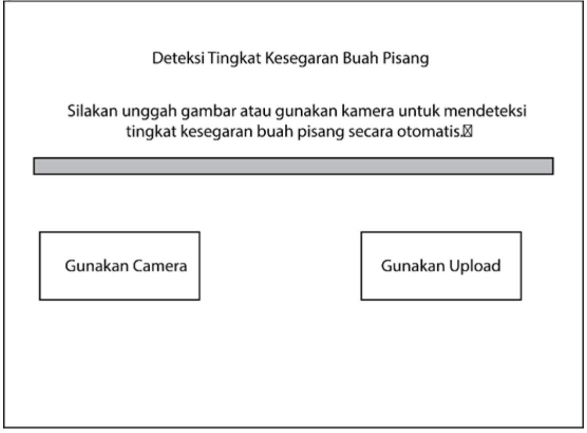
No	Activity	Description
1	Start	The process begins when a user accesses the application.
2	Open Application via Browser	The user opens the application using a web browser.

3	Select Input Mode	The user selects an image input method: camera or file upload.
4a	Capture Image via Camera	If the camera option is selected, the user can capture an image directly.
4b	Upload Image from Storage	If the upload option is selected, the user selects an image file from their device.
5	Image Successfully Captured/Selected	The image was successfully received by the system for processing.
6	Display Detection Result	The system displayed the banana freshness classification results.
7	End	The process ends, and the user can view the results and the system recommendations.

c. Application Interface Design

The application interface was designed to be simple and user-friendly. The interface design of the application is described below.

1) Main Page



Deteksi Tingkat Kesegaran Buah Pisang

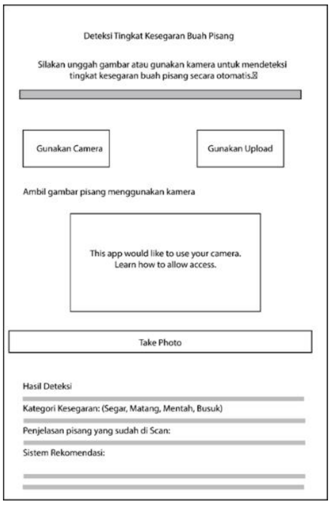
Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.⊠

Figure 7. Main Page

Explanation:

Figure 7 shows the main page interface of the “Banana Freshness Detection” application, which consists of the application title, usage instructions, and two action buttons labeled “Use Camera” and “Use Upload.” This interface allows users to select their preferred image input method before proceeding with the freshness detection process.

2) Use Camera Page



Deteksi Tingkat Kesegaran Buah Pisang

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.⊠

Ambil gambar pisang menggunakan kamera

This app would like to use your camera.
Learn how to allow access.

Hasil Deteksi

Kategori Kesegaran: (Segar, Matang, Mentah, Busuk)

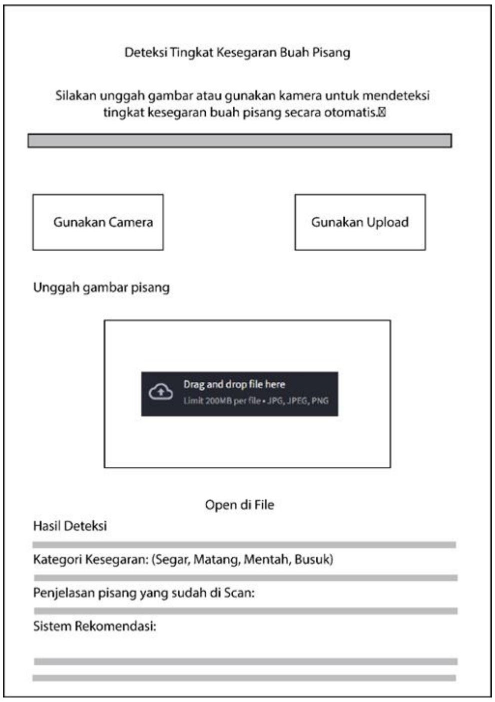
Penjelasan pisang yang sudah di Scan:

Sistem Rekomendasi:

Figure 8. Use Camera Page

Explanation:
Figure 8 presents the “Use Camera” input mode of the application. This interface provides instructions for activating the camera along with a notification regarding the required camera access permission so that the system can utilize the user’s device’s camera. Once the camera is activated, users can scan or capture an image of a banana and then press the “Take Photo” button to take a photo. The system subsequently displays the detection results in the form of banana freshness categories (Fresh, Ripe, Unripe, or Spoiled), accompanied by recommendations indicating whether the fruit is suitable for consumption based on the classification results.

3) Use Upload Image Page



Deteksi Tingkat Kesegaran Buah Pisang

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.

Unggah gambar pisang

Hasil Deteksi

Kategori Kesegaran: (Segar, Matang, Mentah, Busuk)

Penjelasan pisang yang sudah di Scan:

Sistem Rekomendasi:

Figure 9. Use Upload Image Page

Explanation:
Figure 9 shows the “Use Upload Image” input page of this application. On this page, users are provided with instructions to upload banana images from their device storage. After an image is selected and uploaded, the system processes the image using the banana freshness detection model. The displayed results include the freshness classification (Fresh, Ripe, Unripe, or Spoiled) and recommendations regarding the suitability of the fruit for consumption.

3. Implementation
The implementation stage involves applying the design results to a web-based application developed using the Streamlit framework. The trained CNN model was saved in. h5 format and integrated into the application to classify the freshness of bananas. The application provides two input modes, namely “Use Camera” and “Upload Image,” and displays the classification results in the form of freshness categories (Fresh, Ripe, Unripe, Spoiled), along with the confidence score and consumption suitability recommendations. The implementation was carried out using Google Colab and Anaconda, and the application was deployed via Streamlit Cloud so that it could be accessed directly through a web browser.

4. Testing
   The testing stage was conducted to ensure that the system functioned properly and that the prediction results were accurate. Functional testing was performed using the black-box testing method on the main application features, and model performance testing was conducted using accuracy, precision, recall, and F1-score metrics. The results indicate that the CNN model achieved an accuracy of 97–99% with a prediction time of less than 5 s, demonstrating that the system is responsive, accurate, and suitable for real-time banana freshness detection.

4. Results and Discussions

4.1 Research Results

This study resulted in the development of a Convolutional Neural Network (CNN) model capable of automatically detecting the freshness level of Cavendish bananas based on digital images. The dataset used in this study consisted of 1,030 banana images divided into four classes: fresh, ripe, unripe, and spoiled. Each class contained a balanced number of images to maintain an optimal model training performance.

Table 4. Dataset Distribution

Class	Number of Images
Spoiled	261
Ripe	249
Unripe	256
Fresh	264
Total	1030

Prior to training, data preprocessing was performed, including resizing images to 150 × 150 pixels, rescaling pixel values to 1/255, and applying data augmentation techniques, such as rotation, zooming, and horizontal flipping, to increase image variability. This process enhanced the model’s generalization capability and prevented overfitting. The CNN model was developed using the TensorFlow and Keras frameworks with a multi-layer architecture consisting of convolutional, max pooling, flatten, dense, and output layers with a softmax activation function. The training process was conducted for 30 epochs with a batch size of 32 and utilized the EarlyStopping callback to automatically stop training when the validation accuracy no longer improved. The training results showed that the model achieved an accuracy of 99% on the training dataset, 98% on the validation dataset, and 97% on the testing dataset, indicating very good and stable performance across all datasets. The average precision, recall, and F1-score values also reached 0.99, demonstrating the model’s consistency in recognizing each class in a balanced manner.

4.2 Confusion Matrix Analysis

Model evaluation using a confusion matrix indicated that the CNN model was capable of classifying all four banana freshness classes with a very low error rate.

1. The fresh banana class achieved the highest classification accuracy of 99%.

2. The ripe and unripe banana classes achieved accuracy levels between 98% and 99%, whereas the spoiled banana class showed slightly lower accuracy owing to greater variations in texture and image lighting conditions.

$$\frac{TPtotal}{Total} = \frac{40+38+39+39}{40+38+39+2} = \frac{156}{158} = 98.73\% \tag{5}$$

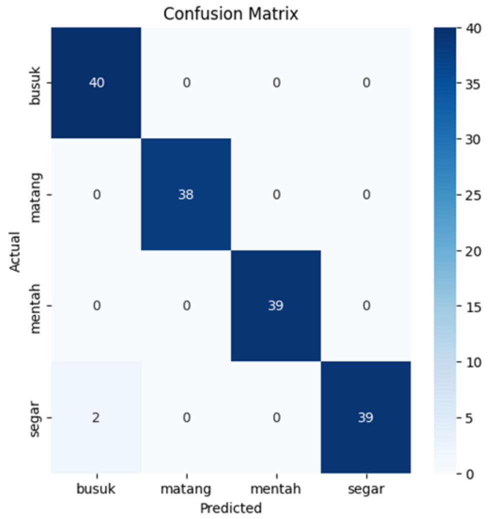


Figure 10. Confusion Matrix

Overall, the model demonstrated strong generalization capability and was able to effectively distinguish visual features at each stage of banana ripeness. Several factors contributed to these results, including the application of data augmentation, use of the ReLU activation function, and Adam optimizer, which accelerated training convergence and improved model stability.

3. Visualization of Training and Validation Accuracy

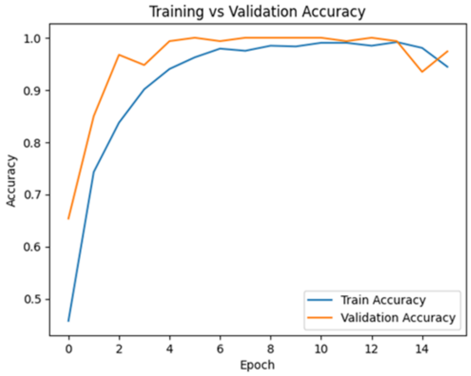


Figure 11. Trend Visualization

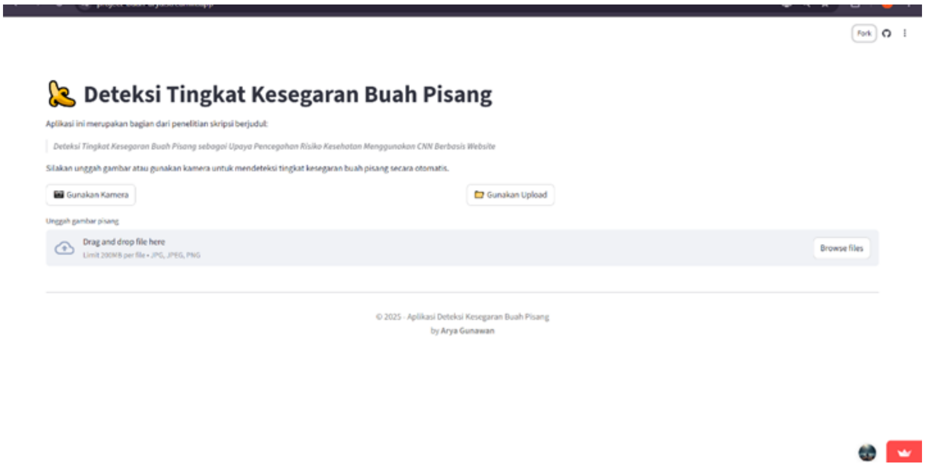
Based on the displayed graph, the training accuracy increases as the number of epochs increases and then stabilizes at a high value exceeding 98%. In addition, the validation accuracy reached 100% in several epochs, indicating that the model has excellent generalization capability for unseen data. The very small difference between the training and validation accuracy curves also suggests that the developed model does not suffer from overfitting or underfitting, indicating that its performance is optimal and stable

4.3 Website Implementation

To facilitate user access to the CNN model, a system was developed as an interactive web-based application using the Streamlit framework. The website provides two image input methods: direct camera capture and manual image upload methods.

1. Main Page

   The main page serves as the initial interface for users accessing the application. On this page, users are presented with a brief explanation of the application’s function and purpose, which is to detect the freshness of bananas. Two main buttons are provided to allow users to select the image input method, namely “Use Camera” and “Upload Image.” The interface was designed to be intuitive and accessible to users from various backgrounds.



Deteksi Tingkat Kesegaran Buah Pisang

Aplikasi ini merupakan bagian dari penelitian skripsi berjudul:

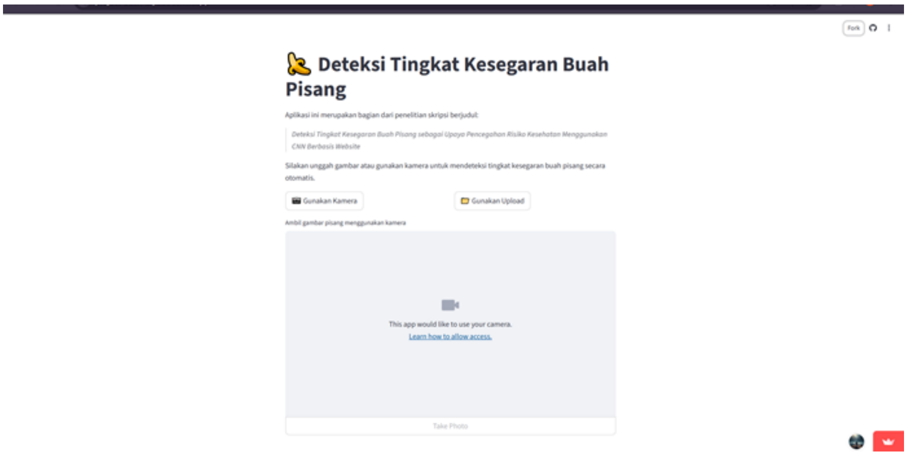
> *Deteksi Tingkat Kesegaran Buah Pisang sebagai Upaya Pencegahan Risiko Kesehatan Menggunakan CNN Berbasis Website*

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.

© 2025 · Aplikasi Deteksi Kesegaran Buah Pisang
by Arya Gunawan

Figure 12. Main Page

2. Use Camera Page
   On this page, users can directly activate the camera on their devices. This feature facilitates real-time detection of banana freshness without requiring users to save images beforehand. After an image is captured, the system automatically sends it to the CNN model for processing and displays the classification results.



Deteksi Tingkat Kesegaran Buah Pisang

Aplikasi ini merupakan bagian dari penelitian skripsi berjudul:

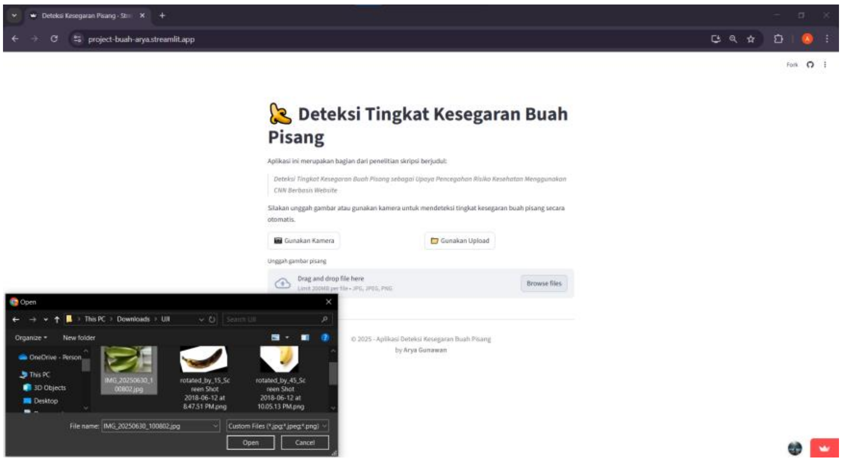
> *Deteksi Tingkat Kesegaran Buah Pisang sebagai Upaya Pencegahan Risiko Kesehatan Menggunakan CNN Berbasis Website*

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.

Fork
Gunakan Kamera
Gunakan Upload
Ambil gambar pisang menggunakan kamera
This app would like to use your camera.
Learn how to allow access.
Take Photo

Figure 13. Use Camera Page

3. Upload Image Page
   As an alternative, users can upload banana images from their device galleries. This feature is useful for users who already have banana images and wish to determine their freshness. After an image is uploaded, the system performs image preprocessing and displays the classification results based on the trained CNN model.



Deteksi Kesegaran Pisang · Str
project-buah-arya.streamlit.app
Fork

Deteksi Tingkat Kesegaran Buah Pisang

Aplikasi ini merupakan bagian dari penelitian skripsi berjudul:

> *Deteksi Tingkat Kesegaran Buah Pisang sebagai Upaya Pencegahan Risiko Kesehatan Menggunakan CNN Berbasis Website*

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.

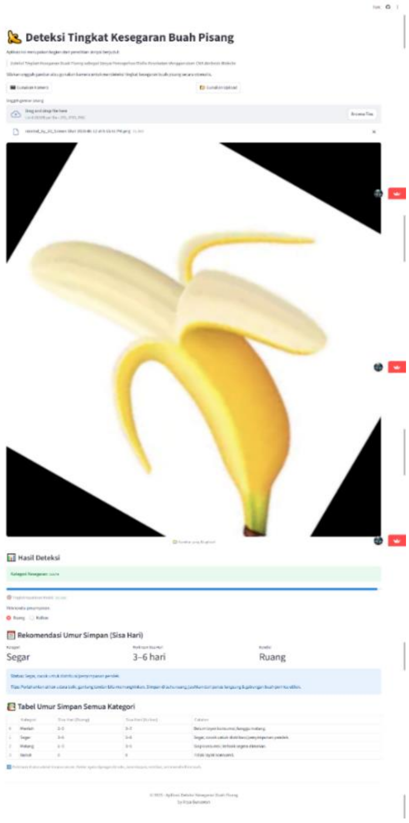
Gunakan Kamera
Gunakan Upload
Unggah gambar pisang
Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG
Browse files

© 2025 - Aplikasi Deteksi Kesegaran Buah Pisang
by Arya Gunawan

Open
This PC › Downloads › UII
Search UII
Organize
New folder
OneDrive - Person
This PC
3D Objects
Desktop
IMG_20250630_100802.jpg
rotated_by_15_Screen Shot 2018-06-12 at 8.47.51 PM.png
rotated_by_45_Screen Shot 2018-06-12 at 10.05.13 PM.png
File name: IMG_20250630_100802.jpg
Custom Files (*.jpg;*.jpeg;*.png)
Open
Cancel

Figure 14. Image Upload Page

4. Detection Results Page
   After the image is successfully processed, the classification results are displayed on this page. The information presented includes the predicted class labels Spoiled, Ripe, Unripe, or Fresh, as well as the model's confidence score. The results interface was designed to be clear and informative, so that users could easily understand the freshness status of the tested banana.
   a. Detection Page – Fresh Banana Result
      Based on the detection results, the banana was classified as fresh with a confidence level of 99.88%. Under room temperature conditions, fresh bananas can last for approximately 3–6 d and are still suitable for consumption. To maintain quality, bananas should be hung to allow proper air circulation, stored in a shaded area, and kept away from ethylene-producing fruits that can accelerate ripening.



Deteksi Tingkat Kesegaran Buah Pisang

[illegible]

Hasil Deteksi

[illegible]

Rekomendasi Umur Simpan (Sisa Hari)

Kategori	[illegible]	Kondisi
Segar	3–6 hari	Ruang

[illegible]

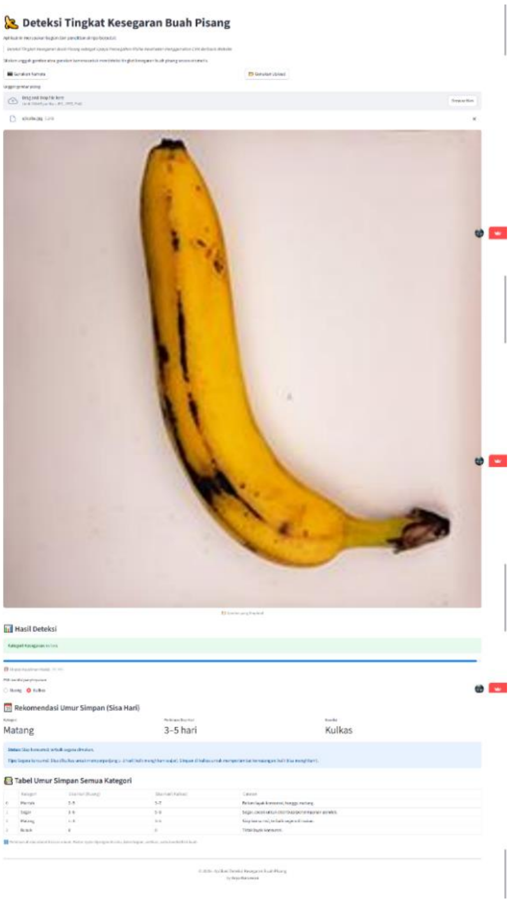
Tabel Umur Simpan Semua Kategori

	Kategori	[illegible]	[illegible]	Catatan
0	Mentah	[illegible]	[illegible]	[illegible]
1	Segar	3-6	3-6	[illegible]
2	Matang	[illegible]	[illegible]	[illegible]
3	Busuk	0	0	[illegible]

Figure 15. Fresh Banana Detection Result

b. Detection Page – Ripe Banana Result

Based on the detection results shown in the image, the banana was classified as ripe, with a model confidence level of 99.99%. This condition indicates that the banana is ready for consumption, as its flavor and texture are of peak quality. When stored in a refrigerator, ripe bananas can last for approximately 3–5 d and may remain usable for an additional 1–2 d, although the peel may darken. A ripe status suggests that the banana should be consumed promptly to prevent it from becoming overly soft or spoiled. Refrigeration is highly recommended to slow down further ripening, even though the peel color may be affected. Therefore, bananas in the ripe category should ideally be consumed immediately to achieve the best taste, or temporarily stored in the refrigerator if they are not to be consumed right away.



🍌 Deteksi Tingkat Kesegaran Buah Pisang

📊 Hasil Deteksi

○ [illegible] ● Kulkas

📅 Rekomendasi Umur Simpan (Sisa Hari)

Matang

3–5 hari

Kulkas

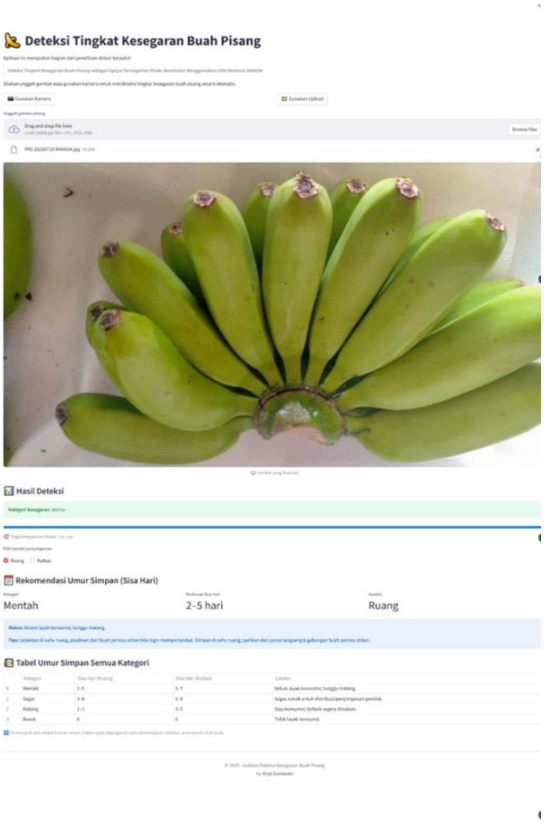
Tabel Umur Simpan Semua Kategori

	Kategori	Sisa Hari (Ruang)	Sisa Hari (Kulkas)	Catatan
0	Mentah	2–5	5–7	[illegible]
1	Segar	[illegible]	[illegible]	[illegible]
2	Matang	[illegible]	[illegible]	[illegible]
3	Busuk	0	0	[illegible]

Figure 16. Ripe Banana Detection Result

c. Detection Page – Unripe Banana Result

In this detection result display, the image shows that the banana is classified as unripe with a model confidence level of 100%. This indicates that the banana is not yet suitable for consumption, as it is still in the early stages of ripening. When stored at room temperature, unripe bananas have an estimated shelf life of approximately 2–5 days before transitioning to the fresh or ripe stages. Unripe status means that the bananas require time to ripen naturally. Recommended storage tips include keeping bananas at room temperature, avoiding direct heat exposure, and separating them from ethylene-producing fruits (such as apples or papayas) if the ripening process is intended to be slowed. With proper handling, bananas gradually ripen and become ready for consumption after several days.



🍌 Deteksi Tingkat Kesegaran Buah Pisang

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.

📊 Hasil Deteksi

Kategori Kesegaran: [illegible]

● Ruang ○ Kulkas

📅 Rekomendasi Umur Simpan (Sisa Hari)

Kategori	Perkiraan Sisa Hari	Kondisi
Mentah	2–5 hari	Ruang

Status: Belum layak konsumsi, tunggu matang.

Tips: Letakkan di suhu ruang, pisahkan dari buah pemicu etilen bila ingin memperlambat. Simpan di suhu ruang; jauhkan dari panas langsung & gabungan buah pemicu etilen.

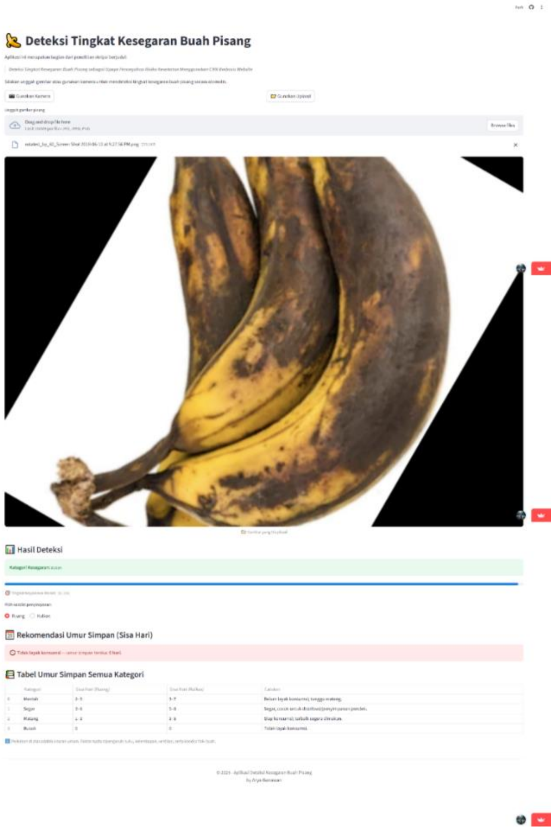
📗 Tabel Umur Simpan Semua Kategori

	Kategori	Sisa Hari (Ruang)	Sisa Hari (Kulkas)	Catatan
0	Mentah	2-5	5-7	Belum layak konsumsi; tunggu matang.
1	Segar	3-6	5-8	Segar, cocok untuk distribusi/penyimpanan pendek.
2	Matang	1-3	3-5	Siap konsumsi; terbaik segera dimakan.
3	Busuk	0	0	Tidak layak konsumsi.

© 2025 - Aplikasi Deteksi Kesegaran Buah Pisang
by Arya Gunawan

Figure 17. Unripe Banana Detection Result

d. Detection Page – Spoiled Banana Result
Based on the detection results shown in the image, the banana was classified as spoiled with a model confidence level of 99.26%. This indicates that the banana has passed its consumable stage, has a shelf life of 0 d, and is no longer suitable for consumption. At this stage, the quality of the banana significantly deteriorates in terms of taste, texture, and nutritional content, potentially posing health risks if consumed. A spoiled status also means that the banana cannot be stored any longer, either at room temperature or in a refrigerator, because the spoilage process can no longer be halted. Therefore, bananas in this condition should be immediately separated or discarded to prevent them from affecting nearby fruits.



🍌 Deteksi Tingkat Kesegaran Buah Pisang

Aplikasi ini merupakan bagian dari penelitian skripsi berjudul:

Deteksi Tingkat Kesegaran Buah Pisang sebagai Upaya Pencegahan Risiko Kesehatan Menggunakan CNN Berbasis Website

Silakan unggah gambar atau gunakan kamera untuk mendeteksi tingkat kesegaran buah pisang secara otomatis.

📊 Hasil Deteksi

Kategori Kesegaran: Busuk

Tingkat keyakinan model: [illegible]

Pilih kondisi penyimpanan:

● Ruang ○ Kulkas

📅 Rekomendasi Umur Simpan (Sisa Hari)

Tidak layak konsumsi — umur simpan tersisa: 0 hari.

📗 Tabel Umur Simpan Semua Kategori

	Kategori	Sisa Hari (Ruang)	Sisa Hari (Kulkas)	Catatan
0	Mentah	2-3	5-7	Belum layak konsumsi, tunggu matang.
1	Segar	3-5	5-8	Segar, cocok untuk distribusi/penyimpanan pendek.
2	Matang	1-2	2-3	Siap konsumsi, sebaiknya segera dimakan.
3	Busuk	0	0	Tidak layak konsumsi.

Prediksi ini merupakan estimasi umum. Faktor suhu, kelembapan, [illegible], serta kondisi fisik buah.

© 2025 · Aplikasi Deteksi Kesegaran Buah Pisang
by Arya [illegible]

Figure 18. Spoiled Banana Detection Result

5. Conclusions

5.1 Conclusion

Based on the research findings and the implementation of the web-based banana freshness detection system using Convolutional Neural Networks (CNN), several conclusions can be drawn. The developed system successfully classifies banana images into four freshness categories—spoiled, ripe, unripe, and fresh—with very high accuracy. The CNN model achieved a testing accuracy of 99%, with average precision, recall, and F1-score values of 0.99, indicating consistent and reliable performance. The confusion matrix results demonstrate that almost all images were correctly classified, with only two misclassifications occurring in the "fresh" category. The integration of the trained model into a Streamlit-based web application enables users to perform real-time predictions through image uploads or camera input. Additionally, the user interface enhances usability by displaying freshness categories, confidence scores, consumption suitability status, and informative recommendations. Overall, the system provides a practical and accurate solution for assessing banana freshness digitally.

5.2 Research Limitations

This study has several limitations. First, the dataset was limited to Cavendish banana images collected under relatively controlled lighting conditions, which may affect generalization in diverse real-world environments. Second, the model has not yet been extensively validated across different banana varieties or varying storage and distribution conditions. Third, field testing in actual plantation or warehouse environments has not been fully conducted, limiting the evaluation of system performance under operational conditions.

5.3 Suggestions and Directions for Future Research

Future research should focus on expanding the dataset by incorporating various banana varieties and diverse lighting conditions to improve model generalization. The implementation of more advanced CNN architectures, such as MobileNet or EfficientNet, may further enhance computational efficiency and scalability. Integration with mobile platforms or IoT-based systems, such as smart cameras in storage and distribution facilities, could enable automated real-time monitoring. Additionally, the development of reporting features, including exportable detection results and integration with warehouse or logistics management systems, would enhance operational applicability. Finally, conducting comprehensive field trials in plantations and storage facilities is essential to validate system robustness and practical effectiveness in real-world scenarios.

Author Contributions

AG contributed to conceptualization, study design, data collection, data analysis, manuscript drafting, and final approval of the manuscript. KA contributed to methodological supervision, critical revision of the manuscript, validation of results, and final approval of the manuscript. Both authors have read and agreed to the published version of the manuscript.

References

Christian, J., & Al Idrus, S. I. (2023). Introduction to Citrus Fruit Ripens Using the Deep Learning Convolutional Neural Network (CNN) Learning Method. *Asian Journal of Applied Education (AJAE), 2*(3), 459-470. doi:https://doi.org/10.55927/ajae.v2i3.5003

Fatahna, I., Sari, P. D. K., Wulanningrum, R., & Utomo, W. C. (2025). *Implementasi Computer Vision Terhadap Jenis Kualitas Pisang Susu Menggunakan Metode YOLOv8n Berbasis WebApps.* Paper presented at the Seminar Nasional Teknologi & Sains. doi:https://doi.org/10.29407/9ezqf773

Febriansyah, F., Oktavianus, D., & Nasrullah, A. (2023). Pengembangan Produk Olahan Hasil Pertanian Tidak Layak Jual Pepaya APeS dan Pisang KeMPeS. *Yumary: Jurnal Pengabdian kepada Masyarakat, 4*(2), 165-174. doi:10.35912/yumary.v4i2.2445

Ferbangkara, S., Mulyani, Y., Mardiana, M., Pratama, R. W. A., Putri, R. A. M., & Rafi'syaiim, M. A. (2025). Analisis Akurasi dan Optimalisasi Dataset untuk Klasifikasi Tanaman Aristolochia acuminata dengan Algoritma CNN. *Jurnal Teknologi Riset Terapan, 3*(1), 13-20. doi:10.35912/jatra.v3i1.5014 doi:https://doi.org/10.35912/jatra.v3i1.5014

Haritha, M., Raju, G., Jacob, A. M., & Thampi, G. (2024). *Remediation of waste water through natural coagulants such as lemon and banana peel.* Paper presented at the E3S Web of Conferences. doi:https://doi.org/10.1051/e3sconf/202452903004

Ikhtiar, M., Riswan, K. A., Asrina, A., & Puspitasari, A. (2024). Hubungan Perceived Severity Dengan Perilaku BABS Pada Masyarakat Pesisir kab. Takalar Tahun 2024. *Media Kesehatan Politeknik Kesehatan Makassar, 19*(1), 77-83. doi:https://doi.org/10.32382/medkes.v19i1.654

Kalsum, U., Subandi, Y., & Wiratma, H. D. (2023). Petani Tanggamus Mitra PT. Great Giant Pineapple Mengekspor Pisang Mas ke Singapura Tahun 2021. *Primer: Jurnal Ilmiah Multidisiplin, 1*(2), 152-164. doi:https://doi.org/10.55681/primer.v1i2.63

Khairina, S. Q., Hidayatulloh, F. S., & Triyonggo, Y. (2025). Pengaruh Kompetensi Dan Employee Engagement Terhadap Kinerja Karyawan Dengan Iklim Organisasi Sebagai Mediasi. *Jurnal Manajemen Dan Organisasi, 16*(4), 421-432. doi:https://doi.org/10.29244/jmo.v16i4.66574

Kurniyanti, V. A., & Murdiani, D. (2022). Perbandingan Model Waterfall Dengan Prototype Pada Pengembangan System Informasi Berbasis Website. *Jurnal Syntax Fusion, 2*(08), 669-675. doi:https://doi.org/10.54543/fusion.v2i08.210

Marhaen, M., Kusmiadi, R., & Ropalia, R. (2023). Kajian Penggunaan Daun Pisang Kering dalam Pematangan Buah Pisang (Musa Paradisiaca L CV. Kepok) dengan Metode Pemeraman di Lubang Tanah. *Jurnal Ilmiah Pertanian dan Peternakan, 1*(1), 35-46. doi:https://doi.org/10.35912/jipper.v1i1.2602

Marpaung, F., Aulia, F., & Nabila, R. C. (2022). Computer Vision Dan Pengolahan Citra Digital: Pustaka Aksara. (ISBN: 978-623-8230-27-3)

Martinus, M., Ferbangkara, S., Annisa, R., Hidayatullah, V., Pratama, R. W. A., & Makarim, A. R. (2025). Pemodelan AI dengan CNN Untuk Klasifikasi Tanaman Uvaria Grandiflora di Hutan Tropis Indonesia. *Jurnal Teknologi Riset Terapan, 3*(1), 1-11. doi:10.35912/jatra.v3i1.5012 doi:https://doi.org/10.35912/jatra.v3i1.5012

Noordianty, A. S., Najma, S., & Nurlaela, R. S. (2024). Kajian Literatur: Penerapan Aspek Sanitasi Terhadap Mutu dan Produk Pangan. *Karimah Tauhid, 3*(7), 7308-7317. doi:https://doi.org/10.30997/karimahtauhid.v3i7.14024

Nugraha, R. S., & Hermawan, A. (2023). Optimasi Akurasi Metode Convolutional Neural Network Untuk Klasifikasi Kualitas Buah Apel Hijau. *Jurnal Mnemonic*, *6*(2), 149-156. doi:https://doi.org/10.36040/mnemonic.v6i2.6730

Pramukti, P., & Setiawan, I. R. (2025). Implementasi Algoritma Yolov8 (You Only Look Once) Untuk Deteksi Jenis Buah Pisang Secara Real-Time. doi:https://doi.org/10.37150/x3pp0849

Raja, H. F. M., Muhammad, M. A., Martinus, M., Pandu, W., Muhkito, A., & Muhammad, A. (2025). Classification of Rare Mussaenda Species in Indonesia's Tropical Forests Using the CNN Algorithm. *Jurnal Teknologi Riset Terapan, 2*(2), 115-122. doi:10.35912/jatra.v2i2.5011

Ramdany, S., Kaidar, S. A., Aguchino, B., Putri, C., & Anggie, R. (2024). Penerapan UML class diagram dalam perancangan sistem informasi perpustakaan berbasis web. *Journal of Industrial and Engineering System*, *5*(1). doi:https://doi.org/10.31599/2e9afp31

Rozan, K., Rozaki, Z., Wulandari, R., & Distrianada, R. I. (2024). *Pemanfaatan Teknologi oleh Petani Milenial.* Paper presented at the Seminar Nasional Agribisnis.

Siswanto, S., Dewi, M. U., Kholifah, S., Widhiati, G., & Aryani, W. (2023). Penggunaan Model Deep Learning Untuk Meningkatkan Efisiensi Dalam Aplikasi Machine Learning. *Jurnal Penelitian Sistem Informasi (JPSI), 1*(4), 215-238. doi:https://doi.org/10.54066/jpsi.v1i4.1619

Subagiya, B. (2023). Eksplorasi penelitian Pendidikan Agama Islam melalui kajian literatur: Pemahaman konseptual dan aplikasi praktis. *Ta'dibuna: Jurnal Pendidikan Islam*, *12*(3), 304-318. doi:https://doi.org/10.32832/tadibuna.v12i3.14113

Surbakti, Y. S. B. (2025). Metode Waterfall Dalam System Development Life Cycle (SDLC): March.

Syaharani, M. A., Budianto, T. A. C., & Adam, R. I. (2024). Klasifikasi Buah Segar Dan Busuk Menggunakan Algoritma Convolutional Neural Network (CNN). *JATI (Jurnal Mahasiswa Teknik Informatika), 8*(5), 10823-10827. doi:https://doi.org/10.36040/jati.v8i5.11132

Wahyuni, N. P. O., Noer, I., & Trisnanto, T. B. (2022). Sikap konsumen dalam pembelian buah pisang Cavendish di pasar modern Kota Bandar Lampung. *Journal of Food System and Agribusiness*, 201-207. doi:https://doi.org/10.25181/jofsa.v6i2.2455

Wijaya, P., Makarim, A. R., Muhammad, M. A., Febriyani, C., Hidayatullah, V., & Annisa, R. (2024). Technology-Based Classification of Clerodendrum Paniculatum Using CNN and Confusion Matrix. *Jurnal Teknologi Riset Terapan, 2*(1), 27-36. doi:10.35912/jatra.v2i1.4598