Fusion : Journal of Research in Engineering, Technolgy and Applied Sciences
E-ISSN 3047-8278
Volume 3 ; Nomor 1 ; April 2026 ; Page 34-42
DOI : https://doi.org/10.66341/fusion.v3i1.359
WEB : *https://ejurnal.faaslibsmedia.com/index.php/fusion*

Integration of the MobileNetV2 Convolutional Neural Network Architecture into a Sorting System for Citrus Fruit Maturity Classification

Fitri Aisah Pohan[1]*, Romy Aulia[2], Syukriadi[3]

[1,2,3] Teknologi Rekayasa Komputer, Politeknik Pertanian Negeri Payakumbuh

[1*]fitriaisyahpohan@gmail.com , [2]romysinggalang@gmail.com , [3]syukriadi@politanipyk.ac.id

Abstract

This research aims to develop an automated sorting system for post-harvest oranges to address subjectivity and the risk of errors caused by human fatigue. The method employed involves developing an automated sorting conveyor prototype based on an ESP32 microcontroller integrated with deep learning technology via a webcam. The continuous ripeness classification process utilizes a Convolutional Neural Network (CNN) with the MobileNetV2 architecture to categorize the fruit into ripe and half-ripe stages. The detection results are then transmitted serially to the ESP32 to drive the sorting servo in real-time. Based on testing with 50 orange samples, the MobileNetV2 CNN model achieved an accuracy of 90% with an F1-Score of 0.90. The implementation of this system is proven to significantly increase post-harvest productivity by reducing processing time by 63.3% compared to manual methods. Additionally, the data counter feature for the Ripe and Half-Ripe categories operates simultaneously to standardize production output monitoring. This study demonstrates that the integration of MobileNetV2 and ESP32 is effectively applied to a prototype-scale sorting system.

Keyword : *Convolutional Neural Network, MobileNetV2, Digital Image, ESP32, Conveyor System.*

1. INTRODUCTION

In Indonesia's agricultural sector, fruit commodities are among the products experiencing relatively high market demand. Among the various types of fruit available, citrus (Citrus spp.) stands out as a highly popular horticultural product that is widely accessible and favored by a broad spectrum of consumers [1]. This crop plays a strategic role as a key horticultural commodity, cultivated extensively across various regions of Indonesia and the world [2]. West Sumatra Province, a major cultivation area in Indonesia, possesses immense production potential. According to data from the West Sumatra Provincial Statistics Agency (BPS), citrus production volume in the region reached 1,218,210 quintals in 2024; Lima Puluh Kota Regency served as the primary production hub, contributing 1,125,400 quintals from 1,310,000 productive trees [3]. This substantial production volume necessitates highly efficient post-harvest handling to ensure quality is maintained. Consequently, the fruit's maturity level serves as a crucial parameter that directly determines product competitiveness and influences market price fluctuations.

Technological advancements in the current era of globalization have profoundly impacted various industrial sectors, including agriculture [4]. However, the adoption of modern technology in horticulture—specifically for sorting commodities like oranges—remains quite limited in practice. Most local farmers still rely heavily on conventional methods, manually sorting oranges based on visual inspection to distinguish between ripe and semi-ripe fruit. Manual sorting methods suffer from several limitations, including subjectivity, time-consuming processes, inconsistency, and susceptibility to errors caused by human fatigue.

With the rapid evolution of modern technology, Artificial Intelligence and digital image processing (Computer Vision) have emerged as solutions to automate sorting processes, making them faster and more accurate [1]. In the realm of object recognition, the Convolutional Neural Network (CNN) stands out as a highly reliable Deep Learning algorithm specifically designed for extracting and processing image data [5]. Leveraging its structural advantages, CNN can extract visual features—such as changes in orange peel color—to non-destructively assess ripeness levels [6]. The effectiveness of CNN architectures has been demonstrated in previous studies; for instance, research on Mandarin oranges achieved 98%–99% accuracy in classifying ripeness levels [7]. Conversely, image classification of avocados achieved a peak accuracy of 90% for the ripe category but showed reduced performance for the unripe category due to low color contrast between classes [8]. However, large-scale CNN models often face challenges regarding high computational demands when deployed on low-cost hardware (a research gap). To address this limitation, applying transfer learning techniques to the lightweight MobileNetV2 CNN architecture offers a precise and efficient solution [9]. This architecture integrates inverted residual structures and linear bottleneck layers that are memory-efficient and offer high inference speeds, making it ideal for resource-constrained devices (edge computing).

Implementing CNN-based fruit sorting requires integration with physical systems so that model decisions can be physically executed. Regarding this integration, previous research has successfully developed a smart conveyor system based on deep learning and IoT control [10]. However, large-scale industrial systems typically entail very high component investment costs. Therefore, the novelty of this research lies in the development of a cost-effective, automated sorting system prototype based on the ESP32 microcontroller, integrated with a MobileNetV2 model for edge computing testing. This prototype is specifically designed to simultaneously classify oranges into "Ripe" and "Semi-Ripe" categories using a mini conveyor system. This prototype-scale approach is expected to serve as a starting point for the adoption of affordable, appropriate technology by small-scale horticultural businesses.

2. METHODS

System Design Process and System Flowchart

The architectural design of the orange sorting system integrates the mechanical conveyor structure and electronic hardware configuration into a unified physical system. The mechanical specifications of the conveyor feature a track width of 15 cm and a total length of 55 cm, while electrical interconnections and power distribution pathways between components are defined through the system circuit schematic. The conveyor design and system circuit schematic are illustrated in Figure 1.

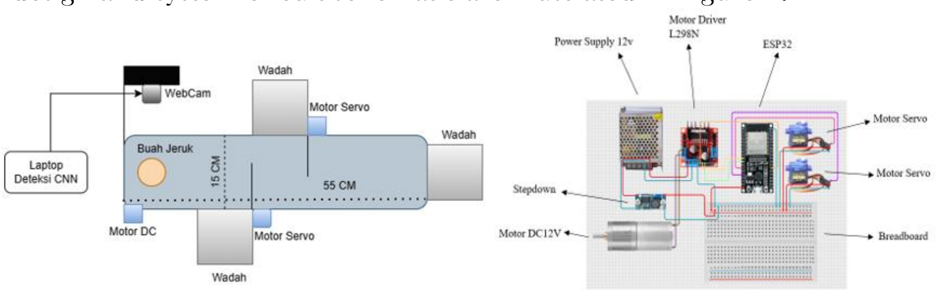


Figure 1. Device design and system circuit schematic.

Main power distribution originates from a 12V power supply, routed directly to the L298N motor driver to drive the 12V DC motor that powers the conveyor at a constant speed. The voltage from this power supply is also stably stepped down to 5V using an LM2596 module to meet the power requirements of two servo motors (Left Servo and Right Servo), which act as mechanical sorting arms for objects along the track.

Fusion : Journal of Research in Engineering, Technolgy and Applied Sciences
E-ISSN 3047-8278
Volume 3 ; Nomor 1 ; April 2026 ; Page 34-42
DOI : https://doi.org/10.66341/fusion.v3i1.359
WEB : *https://ejurnal.faaslibsmedia.com/index.php/fusion*

Dataset Characteristics and Image Pre-processing

The dataset used in this study consists of [1,000] digital images of standard oranges, captured using a smartphone camera in a controlled lighting environment. The dataset focuses on two primary ripeness classification categories: Ripe: Oranges with skin color predominantly and evenly yellow or orange, Semi-ripe: Oranges exhibiting a color gradient blend of green and yellow.

The image pre-processing stages performed before feeding the images into the learning model include:

- Resizing: Adjusting image dimensions to 224 × 224 pixels to match the input size required by the MobileNetV2 architecture.
- Normalization: Processing images using the MobileNetV2 `preprocess_input` function to adjust pixel values according to the ImageNet pre-trained model's normalization standards, ensuring input data compatibility with the format used during training.
- Data Augmentation: Applying techniques such as brightness adjustment (range: 0.8–1.2), 10% image zooming, and horizontal flipping to increase training data diversity; this enhances model robustness against variations in lighting conditions, object position, and object size during real-time orange sorting.

Model Architecture and Training Parameters

This study employs a transfer learning method using the MobileNetV2 architecture with pre-trained weights from ImageNet. All layers of the base model were frozen (trainable = False), so training was performed only on the added classification layer. The model was compiled using the Adam optimizer with a learning rate of 0.0005, Categorical Cross-Entropy loss function, a batch size of 16, and 15 epochs. To enhance training stability and reduce the risk of overfitting, EarlyStopping and ReduceLROnPlateau callbacks were employed.**Prosedur Pengujian dan Logika *Confidence Threshold***

The data processing sequence begins when the webcam captures a visual image of an orange in the initial compartment. This digital image is transmitted directly to a Python-based processing unit, where the fruit's ripeness level is detected using the MobileNetV2 model. The resulting prediction probability is then filtered against a predefined confidence threshold.

Confidence Threshold Selection Logic and Mechanical Response

The system establishes a confidence threshold of 0.75 (75%). This threshold is selected to maintain classification precision, ensuring the system does not erroneously execute mechanical sorting actions (thereby minimizing the risk of false positives). The system's response mechanism operates as follows:

- Detected as Ripe (Confidence ≥ 75% Threshold): The system sends an instruction code via USB Serial to the ESP32 to actuate the "Ripe" sorting servo.
- Detected as Semi-Ripe (Confidence ≥ 75% Threshold): The system sends an instruction code via USB Serial to the ESP32 to actuate the "Semi-Ripe" sorting servo.
- Below Threshold (Confidence < 75% Threshold): The prediction result is disregarded, and the system does not trigger either servo motor. The orange continues along the conveyor belt until it naturally falls into the collection bin at the end of the line.

System Physical Testing Procedure

Testing is conducted in real-time using 50 orange samples representing various stages of ripeness. Each sample is passed along the conveyor belt individually. The ESP32 microcontroller calculates movement timing intervals using the `millis()` function to precisely actuate the sorting servos while simultaneously executing state-locking logic to ensure accurate counter increments without duplicate

Fusion : Journal of Research in Engineering, Technolgy and Applied Sciences
E-ISSN 3047-8278
Volume 3 ; Nomor 1 ; April 2026 ; Page 34-42
DOI : https://doi.org/10.66341/fusion.v3i1.359
WEB : *https://ejurnal.faaslibsmedia.com/index.php/fusion*

counts. At the end of the cycle, sorting data is automatically saved to a local file named `Log.csv`. The overall system workflow is illustrated in Figure 2.

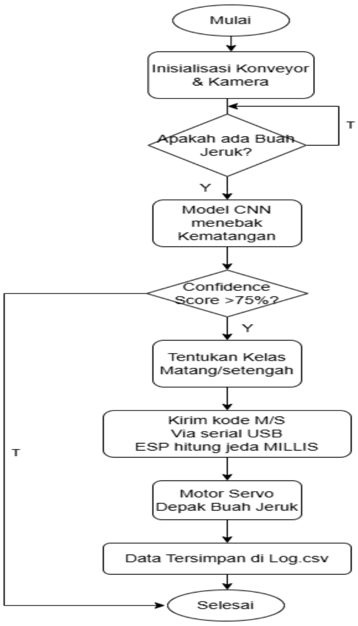


Figure 2. System Workflow Chart

Data Analysis Techniques

The validity of system performance testing is measured using a confusion matrix, which is a standard evaluation technique in machine learning-based classification for calculating accuracy, precision, recall and F1-score classification metrics [11].

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \text{ x } 100\% \tag{1}$$

Legend:
TP = True Positive (Correctly classified as the positive class).
TN = True Negative (Correctly classified as the negative class).
FP = False Positive (Incorrectly classified as the positive class).
FN = False Negative (Incorrectly classified as the negative class).

This formula is used to determine the proportion of correct predictions across the entire test dataset. In addition to accuracy, two other important metrics are precision and recall; precision describes the accuracy of positive predictions—specifically, how many of the predicted positives are truly positive. This process is carried out to minimize false positives—instances where a model predicts a positive outcome when the actual outcome is negative. The formula for Precision is as follows:

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)} \tag{2}$$

Recall measures a model's ability to capture all actual positive cases—indicating how much of that class's data was successfully recognized—thereby helping to minimize false negatives, which occur when the model fails to identify positive data. Both formulas are adopted from classification evaluation concepts widely used in deep learning and machine learning research [12]. The formula for Recall is as follows:

Fusion : Journal of Research in Engineering, Technolgy and Applied Sciences
E-ISSN 3047-8278
Volume 3 ; Nomor 1 ; April 2026 ; Page 34-42
DOI : https://doi.org/10.66341/fusion.v3i1.359
WEB : *https://ejurnal.faaslibsmedia.com/index.php/fusion*

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)} \quad (3)$$

The F1-score is the average of precision and recall. It provides an indication of the balance between precision and recall [11]. The F1-score yields a lower value when both precision and recall are very low, thereby offering a fairer representation of model performance, particularly in cases of class imbalance. The F1-score is calculated using the following formula:

$$F1 - score = 2x \frac{Precision\ x\ Recall}{Precision + Recall} \quad (4)$$

Time efficiency is calculated based on T_manual—the time required when sorting is performed manually by humans—and T_system—the time spent by the automated system. The efficiency analysis also considers the stability of the classification and sorting processes during repeated operations, including an assessment of output consistency in relation to input variations [13].

$$Efisiensi\ Waktu\ = \frac{T_{\text{manual}} - T_{\text{sistem}}}{T_{\text{manual}}}\ x\ 100\% \quad (5)$$

3. RESULTS AND DISCUSSION

Implementation Results

The mechanical system features a physical track measuring 55 cm in length and 15 cm in width, ergonomically designed to ensure a stable flow of oranges during the scanning process. All electronic components—including the ESP32 microcontroller, webcam, DC motor for the conveyor belt, and MG996R servo actuator arm—have been fully integrated into the device's mechanical structure and are ready for field operation. The assembled conveyor system, complete with all electronic components, is shown in Figure 3.

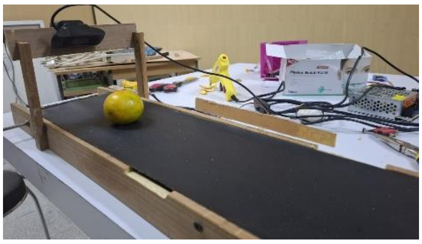
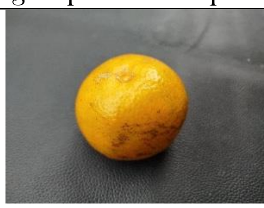


Figure 3. Physical prototype of the automatic orange sorting device.

This study utilizes a dataset comprising 1,000 orange images—consisting of 500 semi-ripe and 500 ripe oranges—collected manually. The dataset is split using an 80:20 ratio, resulting in 800 images for training and 200 for testing. Each ripeness level comprises 400 training samples and 100 testing samples. The system employs a Convolutional Neural Network based on the MobileNetV2 architecture, trained to detect and classify orange ripeness quality. The classification categorizes oranges into two types: semi-ripe and ripe; Table 1 displays images representing these two classifications.

Table 1. Classification of orange ripeness samples.

 Half-Ripe	 Ripe

Fusion : Journal of Research in Engineering, Technolgy and Applied Sciences E-ISSN 3047-8278
Volume 3 ; Nomor 1 ; April 2026 ; Page 34-42
DOI : https://doi.org/10.66341/fusion.v3i1.359
WEB : *https://ejurnal.faaslibsmedia.com/index.php/fusion*

The dataset was trained to classify images of oranges. Once the training process concluded without signs of overfitting, the CNN model was saved as a binary file named `model_jeruk.keras`. This model file was integrated into the main Python program to perform automated classification as the fruit passed through the conveyor's camera scanning area. The stability of the training results was visually monitored using the accuracy curves and loss values shown in Figure 4.

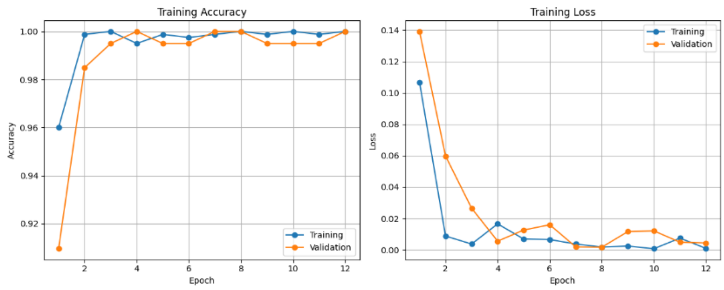


Figure 4. Accuracy and loss graphs from the MobileNetV2 CNN model training.

Training the MobileNetV2 architecture on a dataset of 1,000 images demonstrated a very rapid and stable convergence rate. In the first epoch, the model immediately achieved a training accuracy of 96.0% and a validation accuracy of 90.9%. Model performance improved rapidly, reaching optimal stability in the 99.5%–100% range from the third to the twelfth epoch. No significant gap was observed between the training and validation curves, indicating that the model experienced neither overfitting nor underfitting and possessed excellent generalization capabilities for classifying orange ripeness levels.

Functional testing was conducted to validate the CNN model's performance in real-time fruit detection on a conveyor belt. Selected camera captures and their corresponding prediction results are summarized in Table 2.

Table 2. Real-time citrus maturity detection results.

Detection Image	*Confidence Score* (%)	Prediction Result	Sorting Status
	100%	Half-Ripe	Successful
	100%	Ripe	Successful
	99.9 %	Half-Ripe	Successful
	99.3 %	Ripe	Successful
	91.8 %	Half-Ripe	Successful

To assess system reliability, a validity test was conducted using 50 actual orange samples. The raw test data (confusion matrix) and the corresponding evaluation metrics are presented in an integrated format in Table 3.

Table 3. Confusion Matrix for Orange Classification Testing

Actual Class	Predicted: Half-Ripe	Predicted: Ripe	Total Samples
Half-Ripe	22 (TN)	3 (FP)	25
Ripe	2 (FN)	23 (TP)	25
Total Predictions	24	26	50

Based on the raw fruit count data in Table 3 above, a Python program automatically performed calculations using standard Deep Learning evaluation metrics (such as accuracy, precision, sensitivity, and F1-score). The converted data, representing model performance percentages, is summarized in Table 4; the MobileNetV2 CNN model achieved an overall accuracy of 90.0% with a balanced F1-score of 0.90 for both classes. Overall, this 90.0% validity performance demonstrates the model's high reliability in recognizing the characteristics of the citrus fruit.

Table 4. Citrus Fruit Ripeness Evaluation Metric Test Results

Evaluation Parameter	Half-Ripe Orange Class	Ripe Orange Class
Precision	91.7%	88.5%
Recall	88.0%	92.0%
F1-Score	0.90	0.90
Accuracy (Overall)	90.0%	

In addition to prediction validity, aspects of mechanical performance and system operational efficiency were evaluated through a comparative study between manual sorting and automated sorting using a conveyor system. Data regarding work-time efficiency are presented in Table 5.

Table 5. Comparative Analysis of Sorting Time Efficiency: Manual vs. System

Number of Samples	Manual Sorting Time	System Sorting Time	Time Reduction	Work Time Efficiency (%)
10 Oranges	40 seconds	15 seconds	25 seconds	62.5%
20 Oranges	82 seconds	30 seconds	52 seconds	63.4%
30 Oranges	125 seconds	45 seconds	80 seconds	64.0%
Average	82.3 seconds	30.0 seconds	52.3 seconds	63.3%

Based on the data in Table 5, calculations using the time efficiency formula demonstrate that the use of this automated conveyor system reduces average processing time by 63.3% compared to manual labor.

System Performance Discussion and Analysis

Model Performance Analysis and Influencing Factors

An overall accuracy of 90.0% and an F1-Score of 0.90 across 50 actual samples demonstrate the reliability of the MobileNetV2 architecture in extracting orange peel color features. The five prediction errors (3 False Positives and 2 False Negatives) were influenced by three primary factors in the operational environment:

- Lighting: Light reflection on the smooth orange peel caused a green hue in the "Semi-Ripe" class to be detected as the bright color characteristic of the "Ripe" class.
- Camera Viewpoint (Single-View): Green patches on the underside of the fruit facing away from the camera were not scanned by the system.
- Conveyor Movement (Motion Blur): Occasional object movement caused slight image blurring, thereby reducing the sharpness of color gradients.

Comparison with Previous Research

Compared to previous studies utilizing large-scale CNNs on server PCs (98%–99% accuracy) or MobileNetV2 on avocados (90% accuracy), this study achieved a comparable accuracy level (90.0%). The study's advantage lies in its full physical integration with an ESP32 microcontroller and an automated conveyor system, which was proven to reduce processing time by 63.3%.

System Limitations

The system's primary limitations include image acquisition from a single viewpoint (single-camera setup), an evaluation scope restricted to visual peel color features (excluding texture or sugar content/Brix levels), and conveyor dimensions that remain at a laboratory-prototype scale. 3.1.4

Research Contributions

This research makes a scientific contribution through the implementation of the lightweight MobileNetV2 CNN architecture using transfer learning, achieving 90.0% accuracy and an F1-Score of 0.90 in an edge computing environment. Additionally, it offers a practical contribution in the form of an ESP32-based sorting conveyor prototype capable of accelerating the orange sorting process by 63.3%, while also featuring an automated system that records sorting results into a CSV-formatted log file (Log.csv).

4. CONCLUSION

This study successfully implemented a prototype system for sorting orange ripeness based on the MobileNetV2 CNN model and an ESP32 microcontroller. Real-time functional testing demonstrated that the prototype classifies ripeness levels with high reliability, achieving an overall accuracy of 90.0% and a balanced F1-Score of 0.90 for both the "Ripe" and "Semi-ripe" categories. Beyond its validity in computer vision, the physical implementation of this smart conveyor prototype significantly improved operational efficiency in the field. The automated system performs sorting consistently with an average processing time of 30.0 seconds, reducing the average work duration by 63.3% compared to manual labor. These results demonstrate that lightweight AI technology can be applied to low-cost agricultural equipment to help farmers consistently maintain harvest quality. However, the device has limitations, such as a camera that views only from a single angle (top-down) and susceptibility to external light reflections. Therefore, future development should incorporate multi-angle cameras, utilize an enclosed lighting box, and transmit data records to an online database (cloud) to enable remote monitoring of harvest quantities.

5. REFERENCES

[1] A. Hafiz, A. Angel Tumanggor, M. Delvin Ibo, M. Yazid Noor, K. Palendeo Sitepu, And H. Syahputra, “Deteksi Jeruk Nipis Dan Jeruk Jungga Menggunakan Algoritma Cnn,” *Jurnal Mahasiswa Teknik Informatika)*, Vol. 9, No. 4, Pp. 6245–6251, Aug. 2025.

[2] M. Al Khalidi, Rd. S. Handayani, I. Ismadi, L. Nazirah, And N. Nilahayati, “Analisis Kemiripan Morfologi Bagian Vegetatif Tanaman Jeruk Purut Manis Lokal Aceh Dan Jeruk Purut (Citrus Hystrix),” *Jurnal Agrium*, Vol. 22, No. 1, Pp. 18–26, Mar. 2025, Doi: 10.29103/Agrium.V22i1.21194.

Fusion : Journal of Research in Engineering, Technolgy and Applied Sciences **E-ISSN 3047-8278**
Volume 3 ; Nomor 1 ; April 2026 ; Page 34-42
DOI : https://doi.org/10.66341/fusion.v3i1.359
WEB : *https://ejurnal.faaslibsmedia.com/index.php/fusion*

[3] Badan Pusat Statistik Provinsi Sumatera Barat, "Produksi Tanaman Hortikultura Provinsi Sumatera Barat 2024," Padang, 2025.
[4] M. Riziq Sirfatullah Alfarizi, M. Zidan Al-Farish, M. Taufiqurrahman, G. Ardiansah, And M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman Untuk Machine Learning Dan Deep Learning," *Karimah Tauhid*, Vol. 2, No. 1, Pp. 1–6, 2023.
[5] B. Yanto, L. Fimawahib, A. Supriyanto, B. Herawan Hayadi, And R. Rizki Pratama, "Klasifikasi Tekstur Kematangan Buah Jeruk Manis Berdasarkan Tingkat Kecerahan Warna Dengan Metode Deep Learning Convolutional Neural Network," *Inovtek Polbeng - Seri Informatika*, Vol. 6, No. 2, Pp. 259–268, 2021.
[6] S. Siagian, K. Ibnutama, And R. Mahyuni, "Implementasi Metode Ekstraksi Ciri Warna Untuk Mendeteksi Kematangan Buah Jeruk," *Jurnal Sistem Informasi Triguna Dharma (Jursi Tgd)*, Vol. 1, No. 6, P. 898, Nov. 2022, Doi: 10.53513/Jursi.V1i6.5861.
[7] R. Singh, C. Nickhil, K. Upendar, P. Mishra, And S. Chandra Deka, "Deep Learning-Based Approach For Classifying Mandarin Orange Maturity," *Sustainable Food Technology*, Nov. 2025, Doi: 10.1039/D5fb00408j.
[8] M. Mukhofifah And E. Nurraharjo, "Sistem Deteksi Kematangan Buah Alpukat Menggunakan Metode Pengolahan Citra," *Jurnal Dinamika Informatika*, Vol. 11, No. 1, Pp. 12–23, Mar. 2019, Doi: 10.35315/Informatika.V11i1.8144.
[9] A. C. Hartono And A. R. Muslikh, "Penerapan Transfer Learning Mobilenetv2 Pada Klasifikasi Citra Jenis Buah-Buahan," *Journal Of Information System And Application Development*, Vol. 3, No. 2, Pp. 103–111, 2025, Doi: 10.26905/Jisad.V3i2.16187.
[10] J. Vitasari, R. R. Nugroho, Muhammad Andra Kusuma Ramadhan, Owen Pratama Endramawan, And Mochammad Rifki Ulil Albaab, "Smart Conveyor Real-Time Sort Rotten Tomatoes With Deep Learning Method Integrated Iot Control," *Jurnal Ilmiah Research And Development Student*, Vol. 3, No. 1, Pp. 242–255, Jan. 2025, Doi: 10.59024/Jis.V3i1.1135.
[11] Septian Dwi Chandra, Hardian Oktavianto, And Ari Eko Wardoyo, "Klasifikasi Malware Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Website," *Jurnal Penelitian Teknologi Informasi Dan Sains*, Vol. 2, No. 2, Pp. 84–99, May 2024, Doi: 10.54066/Jptis.V2i2.1931.
[12] R. F. Christianti, "Estimasi Jarak Berbasis Convolutional Neural Network Dan Regresi Dalam Sistem Deteksi Kendaraan Udara Nirawak Berbasis Suara," *Jurnal Sinta: Sistem Informasi Dan Teknologi Komputasi*, Vol. 2, No. 3, Pp. 153–162, Aug. 2025, Doi: 10.61124/Sinta.V2i3.92.
[13] S. Sathyanarayanan, "Confusion Matrix-Based Performance Evaluation Metrics," *African Journal Of Biomedical Research*, Pp. 4023–4031, Nov. 2024, Doi: 10.53555/Ajbr.V27i4s.4345.