Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th October 2025. Vol. 5. No. 1; e-ISSN: 2808-4519

Classification of Orange Peel Conditions Using Transfer Learning MobileNetV2 with K-Fold Cross Validation

Mu'az Khalik lubis[1]*, Hermawan SyahPutra[2]

[1,2]*Universitas Negeri Medan*
muazkhalik@gmail.com[1], hsyahputra@unimed.ac.id[2]*

Abstract

The quality of orange peel is one of the main indicators in determining the ripeness level, overall fruit quality, and market value. Manual assessment of orange peel conditions tends to be subjective, inconsistent, and time-consuming, especially when applied to large-scale sorting processes. This study aims to classify orange peel conditions into two categories healthy and damaged using a deep learning approach based on the MobileNetV2 architecture with the transfer learning method. The model was trained and evaluated using the 10-Fold Cross Validation technique to ensure robust and reliable results. The dataset used in this study was obtained from Kaggle, consisting of high-resolution images of orange peels categorized by physical condition. Each image was converted into RGB format, resized to 224×224 pixels, and normalized before training. The experimental results show that the proposed model achieved an average accuracy of 95.74%, with precision, recall, and F1-score of 95.73% each. The results demonstrate that MobileNetV2, when combined with transfer learning, can effectively detect damaged orange peels and can potentially be implemented in automated fruit quality inspection systems.

***Keywords**: MobileNetV2, Transfer Learning, K-Fold Cross Validation, Image Classification, Orange Peel*

1. Introduction

The quality of orange peel is one of the main indicators used to determine the ripeness, overall quality, and market value of the product [1]. A healthy orange peel typically exhibits a bright color, smooth texture, and absence of defects such as black spots, wrinkles, or mold [2]. Conversely, damaged or abnormal peel conditions can be caused by various factors such as disease infection, pest attacks, or unfavorable environmental conditions [3]. Visual evaluation of orange peel quality by humans is often subjective, depends on the observer's experience, and is inefficient when applied on a large industrial scale [4].

In the context of modern agriculture, especially in the era of precision agriculture, the need for automated systems capable of performing rapid, accurate, and consistent fruit quality classification and assessment has become increasingly important [5]. Digital image processing and machine learning technologies have been widely used to identify and classify agricultural products based on visual features such as color, texture, and surface shape [6]. However, traditional methods based on manual feature extraction often face limitations in capturing the natural visual variability of orange peels, particularly under varying lighting conditions or irregular surface shapes [7].

The advancement of deep learning, particularly Convolutional Neural Networks (CNNs), has brought significant progress in image analysis, including agricultural applications. CNNs are capable of automatically and deeply extracting image features without requiring manual preprocessing or feature extraction [8]. Furthermore, the transfer learning approach allows the use of CNN models pre-trained on large-scale datasets such as ImageNet, which can then be fine-tuned for specific tasks such as orange peel condition classification. This approach has proven effective in reducing training time, improving accuracy, and overcoming the limited number of training samples commonly encountered in agricultural research [9].

Previous studies have demonstrated the effectiveness of CNNs in fruit classification. For instance, AM et al [10] successfully classified the ripeness level of mangosteen fruits with an accuracy of 92% using the MobileNet architecture. Another study employed CNN with the EfficientNetB3 architecture to identify diseases in Citrus siamensis leaves and achieved an accuracy of 98%. These findings confirm that the combination of CNN and transfer learning is effective for automatic and accurate plant disease detection [11].

Based on this background, this study aims to develop a classification method for orange peel condition using a CNN-based transfer learning approach with the MobileNetV2 architecture. The primary goal is to classify orange peel conditions into two categories—healthy and damaged—based on surface images of the fruit. The K-Fold Cross Validation technique is employed to comprehensively evaluate model performance and ensure a high level of generalization on new data.

2. Literature

2.1. Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are deep learning architectures specifically designed to process grid-like data, such as images, through convolutional, pooling, and fully connected layers. CNNs have demonstrated significant success in object detection, image classification, and medical or agricultural image processing tasks [12]. Unlike traditional machine learning techniques that rely on handcrafted feature extraction, CNNs automatically learn hierarchical feature representations from low-level details (edges, corners, colors) to high-level abstractions (shapes, textures, and complex patterns) [13]. Common CNN architectures such as VGG16, ResNet50, and EfficientNet have been widely applied across domains. VGG16 [14] introduced deep convolutional layers with a simple structure but high computational cost, whereas ResNet50 [15] used skip connections to prevent vanishing gradients, enabling deeper model training. EfficientNet [16] later introduced a compound scaling method to balance accuracy and efficiency. These architectures have been successfully applied in agricultural contexts—for example, in classifying fruit diseases, detecting plant health conditions, and assessing fruit quality [17]

2.2. Transfer Learning

Transfer learning is a powerful approach that allows the use of pre-trained deep neural networks for new but related tasks. Instead of training from scratch, models trained on large datasets such as ImageNet are fine-tuned using smaller, domain-specific datasets [18]. This approach is particularly useful in agriculture, where data scarcity and annotation difficulty are common challenges. By leveraging pre-trained weights, CNNs can effectively generalize and extract relevant visual features from limited datasets. Several studies have confirmed the superiority of transfer learning in agricultural image classification. Ferentinos achieved 99.53% accuracy in plant disease recognition using transfer learning with AlexNet, VGG, and ResNet architectures [19]. Similarly, Li et al. used pre-trained deep CNNs to identify 26 plant diseases across 14 species, achieving an average accuracy above 98% [20]. These studies indicate that transfer learning substantially improves performance and reduces training time compared to traditional methods.

2.3. MobileNetV2 Architecture

The MobileNetV2 model, developed by Sandler et al., is a lightweight CNN architecture optimized for real-time and embedded applications. It introduces inverted residual blocks and linear bottlenecks, which significantly reduce the number of parameters and computational cost while maintaining high accuracy [21]. Compared to heavy architectures such as ResNet or DenseNet, MobileNetV2 offers an ideal balance between model size and predictive performance. MobileNetV2 has been effectively implemented in several agricultural studies. Khandelwal et al applied MobileNetV2 to tomato leaf blight detection with 96% accuracy [22]. Du et al used it for apple bruise detection, achieving 94.3% accuracy [23]. These results confirm that MobileNetV2 is highly suitable for low-power agricultural systems and portable fruit inspection tools.

2.4. K-Fold Cross Validation in Deep Learning

Model validation plays a crucial role in ensuring performance reliability and generalization. The K-Fold Cross Validation method is widely used to assess the stability of deep learning models [24]. By dividing the dataset into K equal subsets and rotating them between training and testing, K-Fold reduces bias caused by random data splits and provides a more comprehensive performance evaluation.

This technique has been applied successfully in agricultural classification problems with limited data [25]. For example, in date fruit classification, using 10-Fold Cross Validation improved consistency and prevented overfitting. In this study, 10-Fold Cross Validation ensures that the proposed MobileNetV2 model is evaluated thoroughly across all data subsets, producing reliable and reproducible results.

3. Research Methodology

This research applied an experimental method using deep learning techniques to classify orange peel conditions into two categories: Healthy and Damaged. The implementation was carried out using the Python programming language with TensorFlow, Keras, NumPy, and Scikit-learn libraries. The workflow in this study consists of several stages including data collection, preprocessing, model development, training & validation, evaluation & result analysis

3.1. Data Acquisition and Preprocessing

The dataset used in this study was obtained from the Kaggle platform, which provides a wide range of open datasets for computer vision research. The dataset consists of orange peel surface images labeled as Healthy and Damaged based on visual characteristics such as color brightness, smoothness, and texture irregularities.

Each image was standardized to ensure consistent input dimensions and quality. The preprocessing steps included:

1. Resizing: Each image was resized to 224 × 224 pixels, which corresponds to the input size required by the MobileNetV2 architecture.
2. Color Conversion: All images were converted into RGB format to ensure compatibility with the pretrained ImageNet weights, which were trained on RGB images.
3. Normalization: Pixel intensity values were scaled to a range of [0,1] to improve gradient stability and training convergence.
4. Label Encoding: Images were separated into directories according to their class labels:
   - D:/Jurnal/Healthy → containing images of healthy orange peels
   - D:/Jurnal/Damaged → containing images of damaged orange peels

After preprocessing, the dataset was transformed into NumPy arrays and stored as training and validation data to facilitate the deep learning process.

Fig. 1: Damaged oranges (left), healthy oranges (right)

3.2. Model Architecture

After preprocessing The proposed model utilized MobileNetV2, a convolutional neural network optimized for efficient computation and high accuracy. The model was initialized with pre-trained ImageNet weights to leverage previously learned visual features, thereby accelerating the training process and improving generalization with limited data.

- The top layers of the original MobileNetV2 were removed and replaced with a new classification head consisting of the following components:
- GlobalAveragePooling2D: Converts the spatial feature maps into a single vector per image to reduce overfitting and preserve essential information.
- Dropout (0.3): Randomly deactivates 30% of neurons during training to improve regularization and reduce model overfitting.
- Dense Layer (Softmax): Produces the final classification output for two categories — Healthy and Damaged.

This architecture was selected due to its ability to balance accuracy and computational efficiency, making it suitable for agricultural and industrial deployment scenarios.

3.3. Model Training and Validation

The training phase involved optimizing the model parameters using the Adam optimizer with a learning rate of 1×10^{-5}. The Categorical Cross-Entropy function was employed as the loss function, as it is widely used for multi-class classification tasks. Training was conducted with the following hyperparameter configuration:

- Optimizer: Adam ($lr = 1\times10^{-5}$)
- Loss Function: Categorical Cross-Entropy
- Batch Size: 8
- Epochs: 30
- Validation Strategy: 10-Fold Cross Validation

The 10-Fold Cross Validation technique was selected to ensure robust model generalization and minimize overfitting. The dataset was divided into ten subsets; in each iteration, nine folds were used for training and one for testing. This process was repeated ten times so that every image was used in both training and validation phases.

Performance metrics such as accuracy, precision, recall, F1-score, and the confusion matrix were computed for each fold and averaged to evaluate the overall performance of the classification model.

3.4. Research Stage

Fig. 1 illustrates a series of stages carried out in this research. The following describes each step in the research process:

1. Problem Identification:
   Identify the challenges in manual orange peel quality inspection and the need for an automated classification approach using deep learning.
2. Data Collection:
   Collect orange peel images from the Kaggle platform and organize them into Healthy and Damaged categories.
3. Preprocessing:
   Resize, convert to RGB, and normalize the images to ensure consistent input dimensions and quality.
4. Model Development:
   Construct and compile the MobileNetV2 model architecture for orange peel classification using transfer learning.
5. Model Training and Validation:
   Train the model using 10-Fold Cross Validation to ensure fair and consistent performance evaluation.
6. Evaluation:
   Measure model performance using accuracy, precision, recall, F1-score, and a confusion matrix.
7. Result Analysis:
   Analyze and interpret the experimental outcomes to assess the model's robustness, generalization, and potential industrial applications.

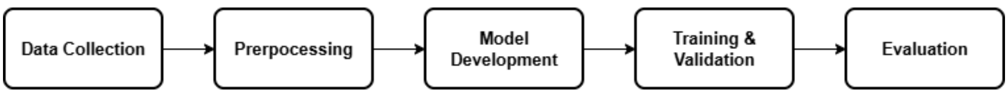

Fig. 2: The research workflow of orange peel condition classification using MobileNetV2 and K-Fold Cross Validation

4. Results and Discussion

This section presents the experimental results and an in-depth discussion of the model's performance in classifying orange peel conditions using MobileNetV2 with transfer learning and 10-Fold Cross Validation. The evaluation focuses on the model's classification accuracy, generalization capability, and robustness across all folds.

4.1. Model Training Process

The model training was conducted on a system equipped with an Intel Core i5 processor, 8 GB RAM, and Windows 11 operating system using Python 3.12 with TensorFlow 2.x and Keras frameworks. The training process utilized the Adam optimizer with a learning rate of 1×10^{-5}, a batch size of 8, and ran for 30 epochs on each fold. During training, the model demonstrated smooth convergence as the loss function consistently decreased over epochs, while accuracy increased steadily. No significant gap between training and validation accuracy was observed, indicating that overfitting was successfully minimized through dropout regularization and cross-validation.In each fold, 90% of the data were used for training, and 10% were reserved for testing. This procedure was repeated ten times, ensuring that every image in the dataset was used for both training and validation. The 10-Fold Cross Validation method allowed for a comprehensive assessment of the model's generalization ability and reliability on unseen data.

4.2. Performance Evaluation

The model performance in each fold is summarized in Table 1. The obtained accuracy values range from 87.72% to 100%, demonstrating consistent and reliable classification results.

Table 1: Model accuracy for each fold

Fold	Accuracy (%)
1	87.72
2	96.49
3	98.21
4	92.86
5	94.64
6	92.86
7	100.00
8	100.00
9	96.43
10	98.21

The average accuracy achieved across all folds was 95.74%, with very low variance between folds, confirming that the model was able to generalize well. This performance indicates that MobileNetV2, even with frozen base layers, effectively captured relevant visual patterns on the orange peel surface such as color distribution, texture uniformity, and visible damage marks. The model benefited from pre-trained ImageNet weights, which accelerated feature learning and improved convergence stability.

4.3. Performance Evaluation

The confusion matrix shown in Table 2 provides a detailed breakdown of the model's classification outcomes for both categories.

Table 2: Confusion matrix of classification results

Actual / Predicted	Healthy	Damaged
Healthy	271	10
Damaged	14	267

From Table 2, it can be observed that out of all Healthy samples, 271 were correctly classified, while 10 were misclassified as Damaged. Similarly, 267 Damaged samples were correctly identified, with 14 misclassified as Healthy. The small number of misclassifications indicates that the model had strong discriminative power in recognizing subtle visual cues between healthy and damaged orange peels. Most errors occurred in images where the damaged regions were minor or appeared similar in color to the surrounding healthy skin, making it visually ambiguous even for human observers. Overall, the confusion matrix demonstrates the high sensitivity and specificity of the MobileNetV2 model for both classes, with only a small fraction of borderline cases being misclassified.

4.4. Comparative Discussion

When compared with previous studies in similar domains, the results achieved in this research are competitive and, in some cases, superior. For instance, AM et al. (2024) reported 92% accuracy in classifying mangosteen ripeness using MobileNet, while Khandelwal et al. (2024) achieved 96% accuracy for tomato leaf disease detection using MobileNetV2 [12][22]. In contrast, this study reached 95.74% average accuracy, proving that the same architecture is effective for fruit surface classification tasks with similar levels of complexity.

The strong performance of MobileNetV2 in this research can be attributed to:
- Efficient feature extraction: The inverted residual and linear bottleneck design allow for compact yet rich feature representations.
- Transfer learning benefits: Utilizing pre-trained ImageNet weights accelerated training and improved model generalization with limited dataset size.
- K-Fold Cross Validation: This validation approach minimized bias and variance, ensuring stable and unbiased accuracy estimates.

The findings confirm that lightweight CNN architectures such as MobileNetV2 can provide a good balance between computational efficiency and predictive accuracy, making them suitable for real-time agricultural inspection systems and mobile-based applications.

5. Conclusion

The dataset used in this research was obtained from the Kaggle platform, containing diverse orange peel images that represent various real-world conditions, such as differences in color, texture, and lighting. Through systematic preprocessing, including resizing, normalization, and RGB conversion, the data were prepared for optimal model performance. The application of MobileNetV2 as a feature extractor allowed the model to automatically learn visual characteristics such as color variations, peel smoothness, and surface defects, without requiring manual feature engineering. The results showed that the proposed model achieved an average accuracy of 95.74%, with precision, recall, and F1-score all reaching 95.73%. These consistent results across all folds indicate that the model generalizes well and performs robustly on unseen data. The confusion matrix further revealed that misclassifications were minimal and primarily occurred in images with subtle damage regions that closely resembled healthy peel textures. Overall, this research demonstrates that MobileNetV2, when fine-tuned using transfer learning, is capable of effectively recognizing the visual differences between healthy and damaged orange peels. The findings validate the potential of deep learning-based methods as reliable tools for automating quality inspection processes in the agricultural sector. In addition, the use of 10-Fold Cross Validation proved crucial in reducing overfitting and ensuring unbiased evaluation, which is essential when working with limited datasets. The combination of accuracy, computational efficiency, and stability makes this approach suitable for integration into industrial fruit sorting systems or portable inspection tools for field-based agricultural operations.

References

[1] N. Haryanti, D. Agungtio, R. S. Alsada, M. S. Suryani, and C. N. Harun, "Quality of orange (Cirtus sp) fruit from five districts in South Sumatera," 2024.

[2] A. W. Dari, A. C. Narsa, and N. M. Zamruddin, "Literature Review: Aktivitas Kulit Jeruk dalam Bidang Farmasi," *Proceeding of Mulawarman Pharmaceuticals Conferences*, vol. 12, pp. 125–151, Dec. 2020, doi: 10.25026/mpc.v12i1.417.

[3] E. M. Damayanti *et al.*, "INFEKSI CPSV (Citrus Psorosis Virus) TERHADAP TANAMAN CITRUS DI KABUPATEN GORONTALO," *Seminar Nasional Teknologi*, 2022.

[4] E. B. Ferreira, C. C. de Oliveira, K. B. Ferreira, J. L. da Silva, and A. M. S. Vieira, "Industrial reuse of orange peel: A review of current applications," in *Frontiers of Knowledge: Multidisciplinary Approaches in Academic Research*, Seven Editora, 2024. doi: 10.56238/sevened2024.026-060.

[5] A. Aghababaei, F. Aghababaei, M. Pignitter, and M. Hadidi, "Artificial Intelligence in Agro-Food Systems: From Farm to Fork," Feb. 01, 2025, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/foods14030411.

[6] H. Pathak, C. Igathinathane, K. Howatt, and Z. Zhang, "Machine learning and handcrafted image processing methods for classifying common weeds in corn field," *Smart Agricultural Technology*, vol. 5, Oct. 2023, doi: 10.1016/j.atech.2023.100249.

[7] Z. Huang *et al.*, "ORD-YOLO: A Ripeness Recognition Method for Citrus Fruits in Complex Environments," *Agriculture (Switzerland)*, vol. 15, no. 15, Aug. 2025, doi: 10.3390/agriculture15151711.

[8] S. Castillo-Girones, S. Munera, M. Martínez-Sober, J. Blasco, S. Cubero, and J. Gómez-Sanchis, "Artificial Neural Networks in Agriculture, the core of artificial intelligence: What, When, and Why," Mar. 01, 2025, *Elsevier B.V.* doi: 10.1016/j.compag.2025.109938.

[9] H. Uyar, I. Karvelas, S. Rizou, and S. Fountas, "Data value creation in agriculture: A review," *Comput Electron Agric*, vol. 227, Dec. 2024, doi: 10.1016/j.compag.2024.109602.

[10] A. AM, M. I. Mardin, and H. Saleh, "DISEASE DETECTION SYSTEM IN COFFEE PRUNS USING MOBILENET ALGORITHM," *Jurnal Kesehatan dan Teknologi*, vol. 1, no. 1, pp. 30–35, Sep. 2024, doi: 10.64879/jkestek.v1i1.33.

[11] B. S. Acarya, A. Muhaimin, and K. M. Hindrayani, "Identifikasi Penyakit Daun Jeruk Siam Menggunakan Convolutional Neural Network (CNN) dengan Arsitektur EfficientNet," *G-Tech: Jurnal Teknologi Terapan*, vol. 8, no. 2, pp. 1040–1048, Apr. 2024, doi: 10.33379/gtech.v8i2.4120.

[12] A. Dwi Rizkika, A. Yudo Husodo, and R. Dwiyansaputra, "IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK DALAM KLASIFIKASI TAHAPAN KEMATANGAN BUAH MANGGIS (Implementation Of Convolutional Neural Network Method In Classification Of Maturity Stages Of Mangosteen Fruit)," 2024.

[13] M. Mesárošová, O. Mihálik, and M. Jirgl, "CNN Architecture for Posture Classification on Small Data," in *IFAC-PapersOnLine*, Elsevier B.V., Jun. 2024, pp. 299–304. doi: 10.1016/j.ifacol.2024.07.413.

[14] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Apr. 2015, [Online]. Available: http://arxiv.org/abs/1409.1556

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016. [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[16] M. Tan and Q. V Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," 2019.

[17] V. Bischoff, K. Farias, J. P. Menzen, and G. Pessin, "Technological support for detection and prediction of plant diseases: A systematic mapping study," *Comput Electron Agric*, vol. 181, p. 105922, Feb. 2021, doi: 10.1016/J.COMPAG.2020.105922.

[18] K. Bykov, K.-R. Müller, and M. M.-C. Höhne, "Mark My Words: Dangers of Watermarked Images in ImageNet," Mar. 2023, [Online]. Available: http://arxiv.org/abs/2303.05498

[19] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput Electron Agric*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/J.COMPAG.2018.01.009.

[20] Y. Li, J. Nie, and X. Chao, "Do we really need deep CNN for plant diseases identification?," *Comput Electron Agric*, vol. 178, p. 105803, Nov. 2020, doi: 10.1016/J.COMPAG.2020.105803.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," 2018.

[22] S. Khandelwal, A. Raut, H. Vyawahare, D. Theng, and S. Dhande, "Optimizing Performance in Mango Plant Leaf Disease Classification through Advanced Machine Learning Techniques," *Engineering, Technology and Applied Science Research*, vol. 14, no. 6, pp. 18476–18480, Dec. 2024, doi: 10.48084/etasr.8220.

[23] Z. Du, X. Zeng, X. Li, X. Ding, J. Cao, and W. Jiang, "Recent advances in imaging techniques for bruise detection in fruits and vegetables," *Trends Food Sci Technol*, vol. 99, pp. 133–141, May 2020, doi: 10.1016/J.TIFS.2020.02.024.

[24] D. Abriha, P. K. Srivastava, and S. Szabó, "Smaller is better? Unduly nice accuracy assessments in roof detection using remote sensing data with machine learning and k-fold cross-validation," *Heliyon*, vol. 9, no. 3, p. e14045, Mar. 2023, doi: 10.1016/J.HELIYON.2023.E14045.

[25] C. Mahlich, T. Vente, and J. Beel, "From Theory to Practice: Implementing and Evaluating e-Fold Cross-Validation," 2024.