Indonesian Journal of Electrical Engineering and Computer Science
Vol. 39, No. 1, July 2025, pp. 322~335
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v39.i1.pp322-335

High-accuracy classification of banana varieties using ResNet-50 and DenseNet-121 architectures

Suastika Yulia Riska, Danang Arbian Sulistyo, Farah Shafiyah Siti Maharani
Departement of Informatic, Faculty of Technology and Design, Institut Teknologi dan Bisnis Asia, Malang, Indonesia

Article Info

Article history:

Received Aug 28, 2024
Revised Mar 7, 2025
Accepted Mar 25, 2025

Keywords:

Banana classification
CNN
DenseNet-121
Image processing
ResNet-50

ABSTRACT

Bananas are a popular fruit in Indonesia due to their affordability, availability, and rich nutritional content. Identifying different banana types is crucial for consumption and processing, yet some types are difficult to distinguish visually. This study aims to classify banana types using convolutional neural network (CNN) architectures, specifically ResNet-50 and DenseNet-121. The dataset consists of five banana classes, which were processed using preprocessing techniques to enhance image quality prior to model training. The results demonstrate that the proposed models can classify banana types with high accuracy. The research methodology includes data collection, preprocessing, CNN model implementation, and performance evaluation using a confusion matrix. The dataset was split into training and testing sets in an 80:20 ratio, with validation data extracted from the training set in a 90:10 ratio. The models were trained on the training data, validated with validation data, and tested on the testing data to assess final performance. The study concludes that the CNN architectures employed are effective in classifying banana types, with the DenseNet-121 model achieving 93.02% accuracy, outperforming the ResNet-50 model, which achieved 92.44%. These results indicate that the models can capture essential features from banana images and produce accurate predictions.

Corresponding Author:

Suastika Yulia Riska
Departement of Informatic, Faculty of Technology and Design, Institut Teknologi dan Bisnis Asia
Jl. Soekarno-Hatta, Rembuksari 1A Malang, Indonesia
Email: riska.suastika@asia.ac.id

1. INTRODUCTION

As the most popular fruits in Indonesia, bananas are frequently consumed due to their affordability and wide availability [1]. They are highly beneficial to health, being rich in vitamins, minerals, and Fiber [2], [3]. Additionally, bananas can be processed into various food products that have a higher market value. However, bananas come in numerous varieties that can be distinguished by their skin colour, texture, shape, size, and taste [4]. Precisely differentiating between certain banana varieties presents a considerable challenge owing to their nuanced visual resemblances [5]. This matter is especially vital for both consumers and companies reliant on accurate classification to guarantee quality and appropriateness. The inability of average consumers to visually distinguish between banana varieties may result in confusion. In industrial applications, such as processed food processing, precise identification is equally critical. The Ambon banana, renowned for its sweetness, is favoured to produce banana chips, underscoring the significance of choosing the appropriate species for applications.

Artificial intelligence (AI) has emerged as a viable approach to streamline the classification of banana varieties. We found that CNN-based models achieved high training accuracy (99.94%) but exhibited a significant drop in testing accuracy (86.56%), suggesting challenges in generalizability. The proposed method in this study tended to have an inordinately higher proportion of misclassifications among visually similar banana varieties, highlighting the need for more advanced architectures [6], [7]. One common example of AI in everyday life is its use in fingerprint recognition or face detection for unlocking smartphones. One method used in such recognition tasks is deep learning (DL). DL is a subfield of AI that focuses on developing complex neural network models with the capability to make decisions based on accurate data [8], [9]. Within DL, convolutional neural network (CNN) architectures are particularly effective in classification tasks, especially those involving images [10], [11].

This study investigated the effects of using CNNs for banana classification. While earlier studies have explored the impact of CNN models in classifying different banana varieties, they have not explicitly addressed its influence on more complex scenarios, such as distinguishing visually similar banana varieties. [12], [13]. A previous study achieved an impressive training accuracy of 99.94% but only attained 86.56% testing accuracy [14], [15]. Although this demonstrates the potential of CNNs, their reliance on fundamental CNN architectures limits their applicability to intricate classification challenges.

The use of more advanced CNN architectures, such as ResNet-50 and DenseNet-121, can enhance accuracy in the identification and classification of banana types. In prior research, ResNet-50 achieved remarkable results, with a 97.7% F1-score accuracy, 97.91% precision, and 97.77% recall in a case study involving the classification of meat images divided into three classes [16], [17]. Additionally, DenseNet-121 outperformed VGG-19 and ResNet-50 in a study on COVID-19 detection using lung X-ray images [18], [19], also divided into three classes. Both architectures have proven effective in image classification, as they can extract relevant features from images and delivering high accuracy. In a study focused on identifying healthy and defective bananas, ResNet-50 was used as the model and provided accurate results, with up to 99% accuracy and minimal error, using a 40:60 ratio of training to testing data [20], [21]. Furthermore, in a fruit recognition study using CNN, DenseNet-121 emerged as the faster model in predicting images, at 45 ms, compared to ResNet-50, which took 74 ms [22].

However, whereas sophisticated CNN architectures like as ResNet-50 and DenseNet-121 have demonstrated exceptional performance across several applications, their specific effectiveness in classifying banana types has yet to be thoroughly examined. Our research indicates that elevated classification accuracy in other fields, such as COVID-19 detection and meat classification, does not inherently result in enhanced performance in banana classification, where visually analogous kinds provide a distinct difficulty. The suggested method might gain from preprocessing techniques like background removal and denoising, without negatively affecting classification accuracy, as noted in prior fruit identification research. The difficulty resides in precisely identifying several banana cultivars, which frequently display nuanced morphological distinctions. This work presents a customized methodology utilizing ResNet-50 and DenseNet-121 architectures for the categorization of bananas to solve existing gaps. This study seeks to assess the efficacy of these architectures by concentrating on three principal contributions: broadening the classification range to five unique banana varieties utilizing a balanced dataset, incorporating preprocessing methods such as background elimination and denoising to enhance image quality, and delivering a comparative analysis to determine the most effective architecture for this task. This work aims to identify the architecture that provides the optimal balance of accuracy and speed through the training and evaluation of these models. The created model can be utilized for autonomous banana classification with high accuracy, significantly contributing to agricultural technology and food processing.

The study procedure and outcomes are explained coherently and methodically throughout the publication. It begins with a detailed description of the approach, then presents the experimental findings, and concludes with a discussion of the implications and prospects. Each section helps readers understand idea development and the study’s importance. As described, configuration:

- Section 2 delineates the process, encompassing dataset preparation and CNN implementation.
- Section 3 delineates the experimental findings and performance evaluations.
- Section 4 finishes with the consequences and prospective directions of this research.

2. METHOD

2.1. Deep learning

AI, or better known as artificial intelligence, is a branch of computer science that focuses on the development of systems capable of mimicking human intelligence [23]. AI aims to create systems that are not only capable of mimicking human behavior, but can also execute tasks that traditionally require human intervention in a more efficient and faster manner. Through AI, computers are equipped with the ability to accomplish a variety of complex tasks that typically require thinking, analyzing, decision-making, and problem-solving-activities that previously could only be performed by humans [24], [25]. For example, AI can be used to perform facial recognition, natural language processing, and product recommendations in e-commerce platforms [26], [27], all of which require AI to process data and produce outputs that suit the user's needs.

DL is a fast-growing AI discipline. DL creates sophisticated neural network models [28]. These models use extensive and reliable data analysis to make decisions. The DL method defines "deep" as the number of layers in the neural network used to analyze input. More layers mean a deeper learning process, allowing the model to capture complicated data patterns. These neural networks' multiple processing layers enable computers to learn from vast volumes of data, spot patterns, and make accurate predictions [29]. DL has advanced AI in image recognition, voice processing, and autonomous cars, where speed and accuracy are key. Figure 1 shows research progress.

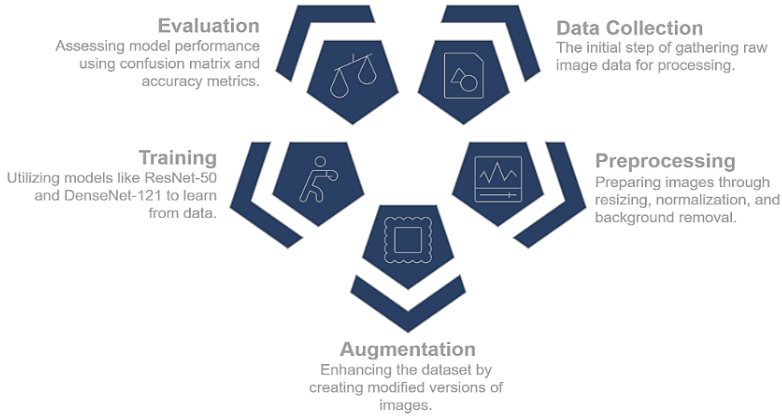


Figure 1. Research stages

2.2. Data collection and pre-processing

Raja, Nangka, Kepok, Barangan, and Ambon bananas are photographed in 860 high-quality images for this study. The dataset includes 537 photos from Kaggle.com, a popular site for datasets, plus 35 separately shot photographs under controlled lighting circumstances to increase banana variety diversity and representation. Table 1 shows that 168 pictures are assigned to Raja banana, 170 to Nangka banana, 176 to Kepok banana, 170 to Barangan banana, and 176 to Ambon banana, ensuring a balanced dataset for analysis.

Table 1. Bananas picture dataset

No	Types of bananas	Sample banana picture	Number of pictures
1.	Ambon banana		176
2.	Barangan banana		170
3.	Kepok banana		176
4.	Nangka banana		170
5.	Raja banana		168
		Total	860

The gathered photos underwent an extensive preprocessing procedure to guarantee uniformity and quality. Initially, all photos were scaled to 224×224 pixels to conform to the input specifications of the CNN designs. Pixel values were standardized to a range of 0-1 to standardize the data and improve model training. Supplementary preprocessing measures involved background removal by segmentation models, utilized to eradicate noise and distractions in the photos, hence maintaining emphasis on the banana features. Data augmentation methods, including random rotations, horizontal flipping, and scaling, were employed to artificially enlarge the dataset, enhancing its diversity and mitigating the risk of overfitting during training. For the pre-processing stages of the dataset are described in the Pseudocode 1.

Pseudocode 1. Preprocessing and classification

Step 1. Import necessary libraries.
- Tensorflow, OpenCV, Numpy and Pandas

Step 2. Load and label the dataset.
- Dataset of 860 images representing five banana types: Raja, Nangka, Kepok, Barangan and Ambon
- Assign label y_i for each image x_i.
- Example: **label_data = [(image, label) for image, label in dataset]**

Step 3. Clean the dataset.
- Validate images based on size and format.
- Remove corrupted or invalid images using the rule $V(x_i)$:

$$\acute{X} = \{x_i |\, V(x_i) = True\}$$

Step 4. Apply pre-processing steps to the dataset.
Step 4.1. Normalize pixel values.

$$\acute{P}_{ij} = \frac{P_{ij} - \min(P)}{\max(P) - \min(P)}$$

Step 4.2. Resize image to 224×224 pixels for CNN compability.
Step 5. Perform data augmentation
- Techniques: random rotations, horizontal flipping, scaling.
- Augmented data:

$$X_{aug} = \cup_{xi \in X} \{T_k(x_i) | T_k \in \tau\}$$

Step 6. Remove background from images.
- Use segmentation models to identify and subtract background:

$$\acute{x}_i = F(x_i) \backslash B(x_i)$$

Step 7. Apply filtering dan denoising.
Step 7.1. Use convolutional filters:

$$\acute{x}_i = x_i * K$$

Step 7.2. Apply median filter for noise reduction:

$$\acute{x}_i = median(x_i, k)$$

Step 8. Split dataset into training, validation, and testing sets
- Training: 80%, validation: 10% (from training), testing: 20%.

Step 9. Define and initialize CNN architectures.
Step 9.1. Use ResNet-50 and DenseNet-121 pre-trained models.
Step 9.2. Configure models with:
- Optimizer: Adam
- Learning rate: 0.001
- Los function: categorical cross entropy

Step 10. Train the models
- Number of Epoch: 50
- Batch size: 32

Step 11. Evaluate model performance.
- Metrics: accuracy, precision, recall, F1-score
- Generate confusion matrics to analyze misclassifications.

Step 12. Use the trained models for classification.
Test the models on unseen data to verify performance.

2.3. Convolutional neural networks

CNNs are widely used for image identification, classification, object detection, and face recognition [11], [30]-[32]. CNNs are designed to analyze and understand array data, making them excellent in managing and categorizing images. CNNs classify images into dogs, cats, tigers, and lions after processing them using a series of layers.

Pixel arrays, depending on resolution, are what computers [33]. Resolution determines the dimensions of an image matrix. An RGB image may be represented as a 6×6×3 matrix with 3 representing the three color channels. In contrast, a grayscale picture is a 4×4×1 matrix. CNNs perform a series of operations on each input picture to train and assess deep learning models [34]. Figure 2 shows how the picture is passed via convolutional layers with filters or kernels, pooling layers, fully connected layers, and the SoftMax function to categorize the item with a probability value from 0 to 1. CNNs have three layers: input, hidden, and output. The input layer receives input data, such as images, and converts it into a format the following layers can process [35]-[37]. Hidden layers convolutional and pooling layers perform most CNN computations by extracting important properties from input data and transforming them into more sophisticated representations. Model layers increasingly filter picture input, improving the model's understanding of visual patterns. The output layer, the model's last layer, categorizes the object based on the probabilities derived in the previous phases. CNNs are effective in many applications, notably those that analyze image and video data [38]. This architectural style lets models instantly learn from unprocessed data and precisely recognize complicated patterns, advancing AI technology.

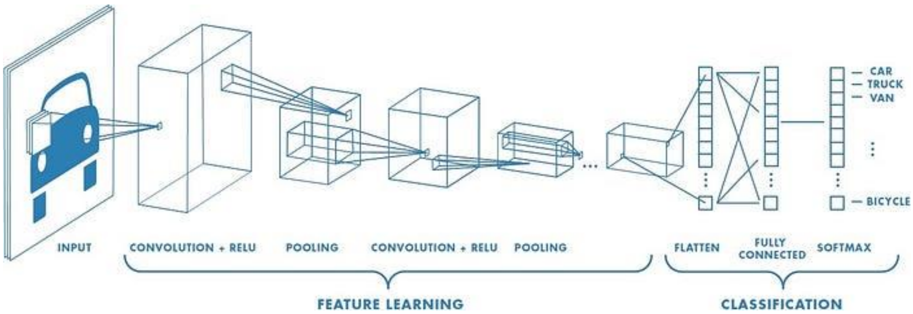


Figure 2. Neural networks with many convolutional layers

2.3.1. ResNet-50 (CNN architecture)

In 2015, Kaiming He and his team created ResNet-50, a ResNet variant [39]. The 50-layer architecture has convolutional, pooling, and fully connected layers. ResNet-50's residual blocks make it superior to other artificial neural networks. A residual block adds the input of the previous layer to the output of a layer after several convolution layers using skip or shortcut connections.

This approach addresses the vanishing gradient problem in deep network training [40]. In normal deep networks, the gradient reduces as it travels to the previous layer, making data learning difficult. Residual connections keep information flowing, thus ResNet-50 can learn with several layers. Residual connections allow the model to learn identity mapping, so a layer with no significant input will not effect the previous layer. This makes ResNet-50 accurate and trainable.

Figure 3 shows that architecture is a common backbone model in AI applications and international image classification, object identification, and segmentation contests like ImageNet. The deep learning community prefers ResNet-50 for its balance between network depth and computational efficiency and its reliable results in many situations.

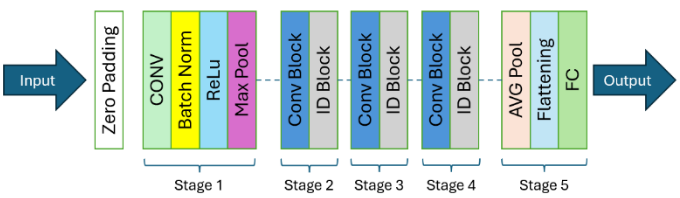


Figure 3. ResNet-50 architecture

2.3.2. DenseNet-121 (CNN architecture)

In 2017, Gao Huang and his colleagues updated DenseNet to create DenseNet-[41]. Convolutional, pooling, and fully linked layers make up the 121-layer model. Its approach is unique compared to ResNet. DenseNet-121's main innovation is dense connections, which link each layer to all previous layers shown in Figure 4.

In DenseNet, each layer's output feeds the next layer and all future levels in a block. Thus, each layer may use all the information from the previous levels, improving learning efficacy and allowing the model to acquire more complex characteristics. DenseNet-121 uses dense connections to improve information and gradient propagation. It also reduces feature duplication, reducing the amount of parameters.

This approach speeds gradient flow and convergence during training while making DenseNet-121 memory and computation economical. Due to feature reuse, DenseNet-121 is known for preventing overfitting [42]. This architecture has performed well in numerous picture classification tasks, including ImageNet, and is extensively used in academia and industry.

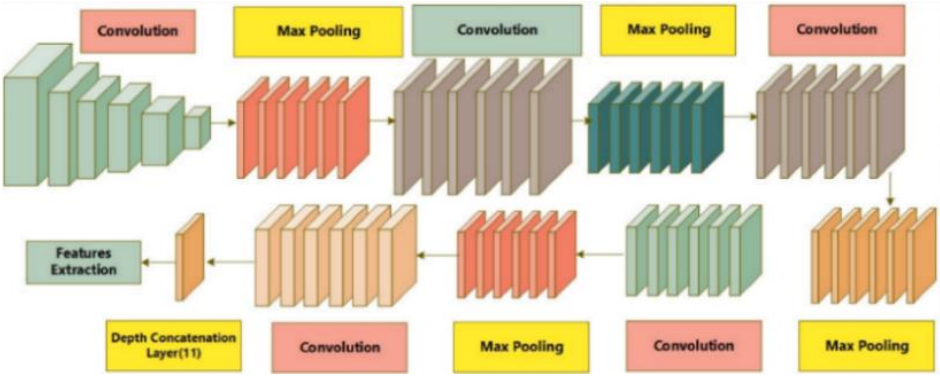


Figure 4. DenseNet architecture

2.4. Evaluation

2.4.1. Confusion matrix

Confusion matrixes are tabular representations used in machine learning and statistical classification to evaluate classification methods [43]. t helps understand how the model distinguishes classes in a classification challenge. The confusion matrix is useful for binary and multiclass classification issues and provides a detailed comparison of model predictions and results [44]. 2×2 confusion matrix for binary classification is typical, as seen in Table 2.

Table 2. Structure of a confusion matrix

Actual	Prediction	
	Predicted positive	Predicted negative
Actual positive	TP	FN
Actual negative	FP	TN

The confusion matrix consists of four terms that describe the outcomes of the categorization process, specifically:
- True positive (TP): the model correctly predicts the positive class.
- False positive (FP): the model incorrectly predicts the positive class for a negative instance.
- True negative (TN): the model correctly predicts the negative class.
- False negative (FN): the model incorrectly predicts the negative class for a positive instance.

2.4.2. Accuracy

Accuracy measures the proportion of correct predictions (both true positives and true negatives) over the total number of predictions. It provides a general measure of model performance.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Use of accuracy matrix is best for datasets where the classes are balanced (equal positive and negative instances).

2.4.3. Precision

$$Precision = \frac{TP}{TP+FP}$$

Precision is the percentage of positive predictions that were right. It shows the model's false positive avoidance. Useful in spam email detection to reduce false positives.

2.4.4. Recall (sensitivity or true positive rate)

$$Recall = \frac{TP}{TP+FN}$$

Recall is the percentage of positive cases predicted correctly per total positive instances. It shows the model's genuine positive detection. In medical diagnostics, reducing false negatives is crucial.

2.4.5. F1-score

$$F1 - Score = 2.\frac{Precision\,.Recall}{Precision+Recall}$$

The harmonic mean of precision and recall is F1-score. The trade-off between these two measures is balanced, particularly in unbalanced datasets. Effective when erroneous positives and negatives cost a IoT.

3. RESULTS AND DISCUSSION

The dataset is partitioned into 80% for training and 20% for testing to streamline model training and testing, laying the groundwork for model validation. This research examined how picture quality affects model performance using three datasets with distinct attributes. D1 includes photos with natural backdrops that correctly depict real-life events. The images were tagged, cleaned, enhanced, normalized, and resized. D2 adds background removal and filtering to D1, creating a controlled white backdrop that eliminates background noise. Dataset 3 (D3) uses image denoising to improve D2 pictures and highlight banana traits. Most crucially, each dataset maintains the same amount of pictures for each grade, guaranteeing grading consistency. Table 3 shows how this adaptable method may evaluate the model's performance in numerous contexts, boosting the conclusions' applicability.

Table 3. Banana image dataset difference

No.	Proses	D1	D2	D3
1	Class labeling	✓	✓	✓
2	Cleaning the dataset	✓	✓	✓
3	Augmentation	✓	✓	✓
4	Normalization	✓	✓	✓
5	Resize	✓	✓	✓
6	Remove background		✓	✓
7	Filtering		✓	✓
8	Denoising			✓
9	Sample image			

This exhaustive investigation examined 32 hyperparameter combinations. The combinations included 16 and 32 batch sizes, 10 and 50 epochs, Adam and Nadam optimizers, mean squared error (MSE) and cross entropy loss functions, and Sigmoid and SoftMax activation functions. This research evaluates all 32 hyperparameter combinations in a thorough and impartial manner. The goal is to find the best configuration for boosting model performance across datasets with varied attributes. After training, the model is thoroughly tested to see whether it can classify banana pictures. The evaluation process involves testing the test dataset and calculating accuracy, precision, recall, and F1-score. Calculate the probability for each test data class and then choose the class with the highest probability as the projected class. This methodical approach accurately assesses the model's prediction abilities, revealing its strength and reliability in real-world situations.

3.1. Evaluation of classification models with ResNet-50 architecture

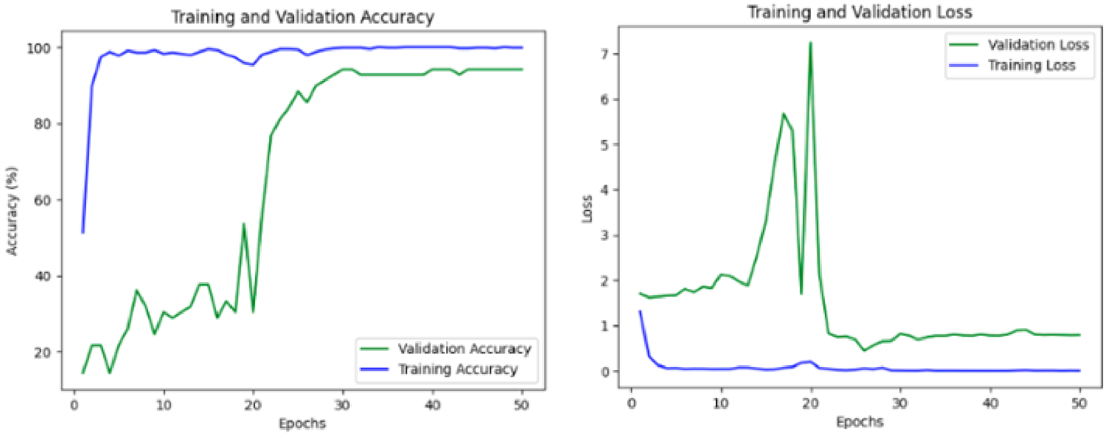
The training results for the ResNet-50 architecture indicate that the model has a total of 28,047,749 According to ResNet-50 architecture training results, the model contains 28,047,749 parameters and 106.99 MB. This data is separated into 27,994,629 trainable parameters (106.79 MB) and 53,120 non-trainable parameters (207.50 KB). The model is adept at gathering and comprehending data features.

The model can recognize complex data patterns because to its many trainable parameters, resulting in excellent classification performance. Table 4 shows the best hyperparameter combinations for accuracy across three datasets. Each dataset has distinct properties that affect model performance. After testing, dataset 2 has the greatest accuracy with batch size of 16, 50 epochs, Nadam optimizer, crossentropy loss function, and sigmoid activation function. This combo has 92.44% accuracy.

Figure 5 displays the accuracy and loss graphs for both training and validation. The initial training accuracy is high and remains consistently close to 100%, but the validation accuracy gradually improves and stabilizes at around 90% Figure 5(a). The loss graph demonstrates that the training loss is consistently low from the start, but the validation loss fluctuates before ultimately settling at a low value Figure 5(b). These results demonstrate that the ResNet-50 model does not suffer from overfitting and exhibits strong performance on the validation data. It maintains a steady level of accuracy and has low loss over the course of 50 epochs.

Table 4. Parameter and best accuracy for ResNet-50

Parameter	Dataset 1 (D1)	Dataset 2 (D2)	Dataset 3 (D3)
Batch size	16	16	16
Epoch	50	50	50
Optimizer	Nadam	Nadam	Nadam
Loss function	Cross entropy	Cross entropy	Mse
Activation	Sigmoid	Sigmoid	Sigmoid
Accuration	85.47	92.44	90.12



(a) (b)

Figure 5. Accuracy (a) loss graphs and (b) for training and validation on ResNet-50 architecture

The confusion Matrix representation in Figure 6 displays the highest performing results achieved by ResNet-50 on dataset 2. The precision, recall, accuracy, and F1-score are recorded as 92.81%, 92.52%, 92.44%, and 92.28% respectively. The confusion matrix indicates a high level of agreement between the predicted and actual classes, with only a few misclassifications evident from the off-diagonal values. In general, the model is proficient in differentiating between several categories. The duration required is 4.445 seconds each step, highlighting a substantial computing need for each repetition.

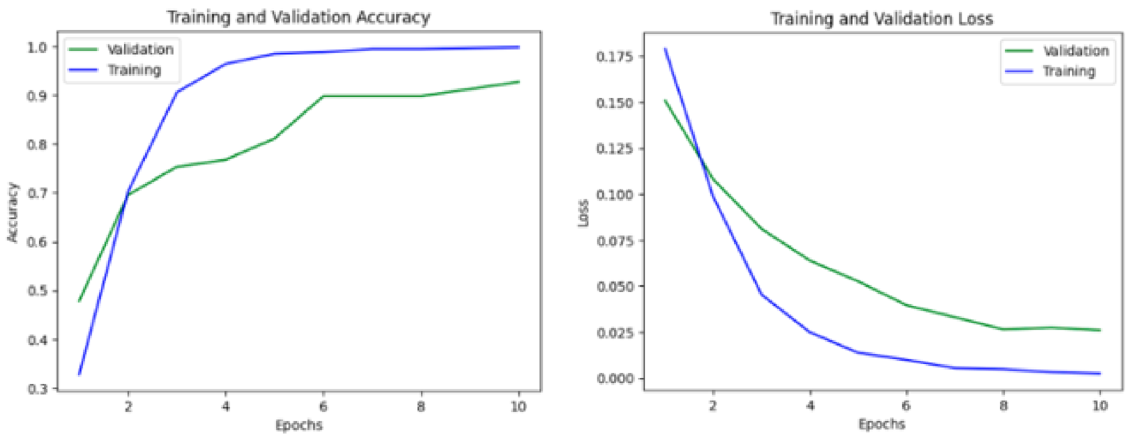
Figure 6. ResNet-50 highest performing results

3.2. Evaluation of classification models with DenseNet-121 architecture

The training results for the DenseNet-121 architecture are displayed in Table 5. The model summary reveals a total of 9,400,389 parameters (35.86 MB), which can be further broken down into 9,316,741 trainable parameters (35.54 MB) and 83,648 non-trainable parameters (326.75 KB). This demonstrates that the model possesses the capability to effectively acquire and understand distinctive characteristics from the given data. Due to its numerous trainable parameters, the model is capable of identifying intricate patterns within the data, resulting in exceptional classification results.

We evaluated the model’s performance using several combinations of hyperparameters, similar to the ResNet-50 design. Figure 7 displays the optimal combination of hyperparameters that achieves the highest accuracy across three distinct datasets, each with unique properties that impact the performance of the model.

According to the test findings, dataset 2 (D2) achieved the highest accuracy of 93.02% utilizing a batch size of 32, 10 epochs, the Adam optimizer, the MSE loss function, and the sigmoid activation function. Figure 7(a) displays the graphs of training and validation accuracy. It is evident that training accuracy experiences a quick increase and then reaches a stable level close to 100% after a few epochs. The validation accuracy exhibits a substantial rise and reaches a stable level of approximately 90%. Furthermore, the loss graph indicates that the training loss is consistently low from the start, whereas Figure 7(b) demonstrates a substantial decline in the validation loss, eventually stabilizing at a low value. The results demonstrate that the DenseNet121 model exhibits no signs of overfitting and achieves high performance on the validation data. It maintains consistent accuracy and experiences minimal loss during multiple training epochs.



(a) (b)

Figure 7. Accuracy (a) loss graphs and (b) for training and validation on DenseNet-121 architecture

Table 5. Parameter and best accuracy for DenseNet-121

Parameter	Dataset 1 (D1)	Dataset 2 (D2)	Dataset 3 (D3)
Batch size	16	32	32
Epoch	10	10	50
Optimizer	Nadam	Adam	Nadam
Loss function	MSE	Mse	Cross entropy
Activation	Sigmoid	Sigmoid	Sigmoid
Accuration	87.21	93.02	91.28

Figure 8 demonstrates the classification model's strong performance, achieving a precision of 93.27%, recall of 93.37%, accuracy of 93.02%, and an F1-score of 92.98%. The confusion matrix indicates a high level of agreement between the predicted and actual classes, with only a few misclassifications evident from the off-diagonal values. In general, the model is proficient in discerning between several categories. The duration required is 5.421 seconds for each step, indicating a substantial processing demand for each iteration.

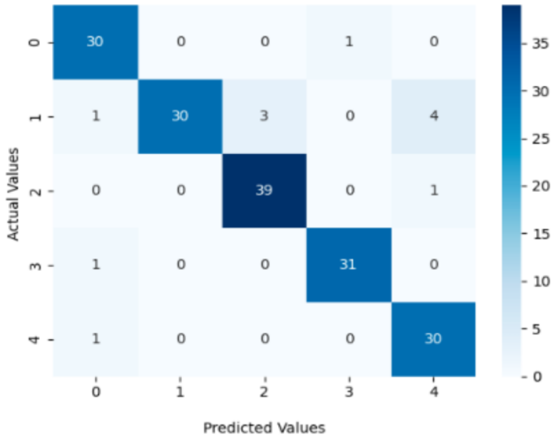


Figure 8. DenseNet-121 highest performing results

This experiment compares ResNet-50 and DenseNet-121 architectures for banana picture classification across datasets and preprocessing methods. For tasks requiring reliable classification, ResNet-50 showed excellent generalization abilities, retaining validation accuracy and low loss. Despite being more computationally intensive, DenseNet-121 outperformed in accuracy, precision, and recall, especially on dataset 2 (93.02% accuracy). These data suggest that DenseNet-121 is preferable for precision-intensive applications, particularly those that need to decrease misclassification. Dataset 2-with background removal and filtering-outperforms others in accuracy and precision, demonstrating the requirement for preprocessing. CNN model feature extraction requires background noise removal, as shown by this finding.

Dataset 3 improved visual quality and classification effectiveness with further denoising. The accuracy gain was minimal, and Dataset 3's computational complexity showed declining benefits compared to dataset 2. Performance improvements must be balanced with processing needs due to the preprocessing complexity-computing efficiency tradeoff. For instance, dataset 3 provided a more comprehensive examination of banana features, yet its performance difference from dataset 2 shows that background reduction and filtering may greatly enhance classification results in various actual circumstances. These findings emphasize the relevance of well-designed preprocessing pipelines for model performance and computational efficiency. Table 6 shows model performance across datasets.

Table 6. Summary of model performance across datasets

Model	Dataset	Accuracy	Precision	Recall	F1-sore	Processing time
ResNet-50	D1	85.47	85.32	85.21	85.10	4.445
	D2	92.44	92.81	92.52	92.28	4.445
	D3	90.12	90.32	90.14	90.01	4.445
DenseNet-121	D1	87.21	87.33	87.14	87.23	5.421
	D2	93.02	93.27	93.37	92.98	5.421
	D3	91.28	91.34	91.21	91.15	5.421

The results of this study have considerable significance for agricultural automation and other fields. By combining sophisticated preprocessing pipelines with advanced CNN architectures, this work demonstrates how automated systems can classify crops with high accuracy and efficiency, significantly reducing the reliance on manual labor in agriculture. The concept extends beyond agriculture, holding significant implications for domains necessitating precise picture categorization, including defect identification in manufacturing and medical diagnostics. These findings create a foundation for formulating resilient, scalable methods that tackle the issue of nuanced visual discrepancies in categorization tasks, thereby connecting research with practical applications. The scalability of this approach is evident in its potential applications across diverse industries, including manufacturing and healthcare. The heightened processing requirements of DenseNet-121 (5.421 seconds per step) underscore the need to investigate lightweight CNN architectures, like MobileNet, for use in resource-limited settings. Future studies should focus on expanding the dataset to include additional banana varieties and exploring more advanced preprocessing techniques, such as color space transformations, to improve classification accuracy. Furthermore, the integration of these models with IoT-based systems might facilitate real-time applications, optimize agricultural operations, and enhance smart farming solutions.

4. CONCLUSION

The study showcases the efficacy of the DenseNet-121 framework in attaining exceptional accuracy in the categorization of banana types, with a precision rate of 93.27%, a recall rate of 93.37%, and an F1-score of 92.98%. Our findings provide conclusive evidence that DenseNet-121 is a highly effective model for classifying banana varieties, particularly in scenarios requiring high precision and recall. DenseNet-121 demonstrated its capability to effectively classify banana varieties, particularly in scenarios requiring high precision and recall. The study emphasizes the significance of preprocessing approaches, such as background removal and noise reduction, in improving the quality of the dataset, which directly enhances model performance. These preprocessing techniques played a critical role in enabling the model to accurately distinguish between visually similar banana types. In addition, DenseNet-121 is highly efficient because of its reduced model size of 35.86 MB makes it suitable for deployment in resource-constrained environments.

However, this study has certain limitations that may impact the generalizability of the results. The relatively slow processing time of DenseNet-121, at 5.421 seconds per step compared to ResNet-50, presents a trade-off between computational efficiency and classification accuracy, potientally limiting its use in real-time or time-sensitive applications. Furthermore, the study was limited to only two CNN architectures, ResNet-50 and DenseNet-121, without exploring other architectures, such as MobileNet or InceptionV3, which may provide better trade-offs between speed and performance. The dataset, while balanced, was limited in scope and may not fully represent the variations in banana morphology across different environmental conditions. As a result, further and more in-depth studies are needed to confirm the model’s effectiveness, particularly in large-scale applications and dynamic environments.

Future research should address these limitations by evaluating more broader range of CNN architectures and experimenting with datasets that are more representative of real-world conditions to further improve classification performance. Expanding the dataset will further improve classification robustness. Additionally, integrating the model into IoT-based smart farming systems and testing it under real-time constraints could enhance its practical applicability in agriculture and industry. These enhancements would contribute to developing a more efficient and widely applicable banana classification system.

FUNDING INFORMATION

This research did not receive any funding from external organizations and was entirely self-funded by the authors

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Suastika Yulia Riska	✓	✓	✓	✓	✓	✓	✓		✓	✓		✓		
Danang Arbian Sulistyo	✓	✓	✓		✓	✓	✓		✓	✓				
Farah Shafiyah Siti Maharani	✓		✓	✓	✓				✓	✓				

C	:	**C**onceptualization	I	:	**I**nvestigation	Vi	:	**Vi**sualization
M	:	**M**ethodology	R	:	**R**esources	Su	:	**Su**pervision
So	:	**So**ftware	D	:	**D**ata Curation	P	:	**P**roject administration
Va	:	**Va**lidation	O	:	Writing - **O**riginal Draft	Fu	:	**Fu**nding acquisition
Fo	:	**Fo**rmal analysis	E	:	Writing - Review & **E**diting			

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no financial or personal relationships that could have influenced the work reported in this paper.

INFORMED CONSENT

Informed consent was not applicable to this study as it did not involve human subjects or personal data.

ETHICAL APPROVAL

Ethical approval was not required for this study as it did not involve human or animal subjects.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, Suastika Yulia Riska, upon reasonable request.

REFERENCES

[1] I. Sulistyowarni, S. Sundari, and S. Halim, "The potential of banana trading commodity to fulfill market demand and support food security in defense economic perspective (study in Bogor district)," *Jurnal Pertahanan & Bela Negara*, vol. 10, no. 3, p. 307, Dec. 2020, doi: 10.33172/jpbh.v10i3.1114.

[2] M. F. Afzal *et al.*, "Bioactive profile and functional food applications of banana in food sectors and health: a review," *International Journal of Food Properties*, vol. 25, no. 1, pp. 2286–2300, Dec. 2022, doi: 10.1080/10942912.2022.2130940.

[3] S. Lopes, C. V. Borges, S. M. de Sousa Cardoso, M. F. de Almeida Pereira da Rocha, and M. Maraschin, "Banana (Musa spp .) as a source of bioactive compounds for health promotion," in *Handbook of Banana Production, Postharvest Science, Processing Technology, and Nutrition*, Wiley, 2020, pp. 227–244.

[4] F. M. A. Mazen and A. A. Nashat, "Ripeness classification of bananas using an artificial neural network," *Arabian Journal for Science and Engineering*, vol. 44, no. 8, pp. 6901–6910, Aug. 2019, doi: 10.1007/s13369-018-03695-5.

[5] P. S. Kumar *et al.*, "Ripening dependent changes in skin color, physicochemical attributes, in-vitro glycemic response and volatile profiling of banana varieties," *Food Bioscience*, vol. 56, p. 103274, Dec. 2023, doi: 10.1016/j.fbio.2023.103274.

[6] Y. Xu *et al.*, "Artificial intelligence: A powerful paradigm for scientific research," *Innovation*, vol. 2, no. 4, p. 100179, Nov. 2021, doi: 10.1016/j.xinn.2021.100179.

[7] Y. K. Dwivedi *et al.*, "Artificial intelligence (AI): multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy," *International Journal of Information Management*, vol. 57, p. 101994, Apr. 2021, doi: 10.1016/j.ijinfomgt.2019.08.002.

[8] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K. R. Müller, "Explaining deep neural networks and beyond: a review of methods and applications," *Proceedings of the IEEE*, vol. 109, no. 3, pp. 247–278, Mar. 2021, doi: 10.1109/JPROC.2021.3060483.

[9] Y. R. Shrestha, V. Krishna, and G. von Krogh, "Augmenting organizational decision-making with deep learning algorithms: Principles, promises, and challenges," *Journal of Business Research*, vol. 123, pp. 588–603, Feb. 2021, doi: 10.1016/j.jbusres.2020.09.068.

[10] S. Sakib, N. Ahmed, A. J. Kabir, and H. Ahmed, "An overview of convolutional neural network: its architecture and applications," *Preprints 2018*, no. February. Nov. 20, 2018, doi: 10.20944/preprints201811.0546.v1.

[11] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[12] A. S. M. F. Al Haque, M. A. Hakim, and R. Hafiz, "CNN based automatic computer vision system for strain detection and quality identification of banana," in *2021 International Conference on Automation, Control and Mechatronics for Industry 4.0, ACMI 2021*, Jul. 2021, pp. 1–6, doi: 10.1109/ACMI53878.2021.9528269.

[13] T. T. Le, C. Y. Lin, and E. J. Piedad, "Deep learning for noninvasive classification of clustered horticultural crops – A case for banana fruit tiers," *Postharvest Biology and Technology*, vol. 156, p. 110922, Oct. 2019, doi: 10.1016/j.postharvbio.2019.05.023.

[14] R. S *et al.*, "Deep learning based dual channel banana grading system using convolution neural network," *Journal of Food Quality*, vol. 2022, pp. 1–9, May 2022, doi: 10.1155/2022/6050284.

[15] E. Almeyda and W. Ipanaqué, "Recent developments of artificial intelligence for banana: application areas, learning algorithms, and future challenges," *Engenharia Agricola*, vol. 42, no. SpecialIssue, 2022, doi: 10.1590/1809-4430-Eng.Agric.v42nepe20210144/2022.

[16] O. Jarkas *et al.*, "ResNet and Yolov5-enabled non-invasive meat identification for high-accuracy box label verification," *Engineering Applications of Artificial Intelligence*, vol. 125, p. 106679, Oct. 2023, doi: 10.1016/j.engappai.2023.106679.

[17] B. Xu *et al.*, "Evaluation of deep learning for automatic multi-view face detection in cattle," *Agriculture (Switzerland)*, vol. 11, no. 11, p. 1062, Oct. 2021, doi: 10.3390/agriculture11111062.

[18] K. Hussain, S. A. Khan, M. Muzaffar, Z. R. Khan, and S. U. Farooq, "Performance evaluation of CNN architectures for COVID-19 detection from X-ray images," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 11, no. 1, pp. 80–93, Jan. 2023, doi: 10.1080/21681163.2022.2052750.

[19] L. Da Quach, N. Pham-Quoc, D. C. Tran, and M. F. Hassan, "Identification of chicken diseases using VGGNet and ResNet models," in *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 334, 2020, pp. 259–269.

[20] K. Yan, M. K. C. Shisher, and Y. Sun, "A transfer learning-based deep convolutional neural network for detection of fusarium wilt in banana crops," *AgriEngineering*, vol. 5, no. 4, pp. 2381–2394, Dec. 2023, doi: 10.3390/agriengineering5040146.

[21] J. D. Thiagarajan *et al.*, "Analysis of banana plant health using machine learning techniques," *Scientific Reports*, vol. 14, no. 1, p. 15041, Jul. 2024, doi: 10.1038/s41598-024-63930-y.

[22] F. Salim, F. Saeed, S. Basurra, S. N. Qasem, and T. Al-Hadhrami, "DenseNet-201 and Xception pre-trained deep learning models for fruit recognition," *Electronics (Switzerland)*, vol. 12, no. 14, p. 3132, Jul. 2023, doi: 10.3390/electronics12143132.

[23] S. Khanam, S. Tanweer, and S. Khalid, "Artificial intelligence surpassing human intelligence: factual or hoax," *Computer Journal*, vol. 64, no. 12, pp. 1832–1839, Dec. 2021, doi: 10.1093/comjnl/bxz156.

[24] D. M. Townsend and R. A. Hunt, "Entrepreneurial action, creativity, & judgment in the age of artificial intelligence," *Journal of Business Venturing Insights*, vol. 11, p. e00126, Jun. 2019, doi: 10.1016/j.jbvi.2019.e00126.

[25] G. Zhao, Y. Li, and Q. Xu, "From emotion AI to cognitive AI," *International Journal of Network Dynamics and Intelligence*, pp. 65–72, Dec. 2022, doi: 10.53941/ijndi0101006.

[26] P. A. Olujimi and A. Ade-Ibijola, "NLP techniques for automating responses to customer queries: a systematic review," *Discover Artificial Intelligence*, vol. 3, no. 1, p. 20, May 2023, doi: 10.1007/s44163-023-00065-5.

[27] H. Huang, A. A. Zavareh, and M. B. Mustafa, "Sentiment analysis in e-commerce platforms: a review of current techniques and future directions," *IEEE Access*, vol. 11, pp. 90367–90382, 2023, doi: 10.1109/ACCESS.2023.3307308.

[28] R. Miikkulainen *et al.*, "Evolving deep neural networks," in *Artificial Intelligence in the Age of Neural Networks and Brain Computing*, Elsevier, 2019, pp. 293–312.

[29] R. Y. Choi, A. S. Coyner, J. Kalpathy-Cramer, M. F. Chiang, and J. P. Campbell, "Introduction to machine learning, neural networks, and deep learning," *Translational Vision Science and Technology*, vol. 9, no. 2, 2020, doi: 10.1167/tvst.9.2.14.

[30] M. M. Taye, "Theoretical understanding of convolutional neural network: concepts, architectures, applications, future directions," *Computation*, vol. 11, no. 3, p. 52, Mar. 2023, doi: 10.3390/computation11030052.

[31] J. Naranjo-Torres, M. Mora, R. Hernández-García, R. J. Barrientos, C. Fredes, and A. Valenzuela, "A review of convolutional neural network applied to fruit image processing," *Applied Sciences (Switzerland)*, vol. 10, no. 10, p. 3443, May 2020, doi: 10.3390/app10103443.

[32] M. Krichen, "Convolutional neural networks: a survey," *Computers*, vol. 12, no. 8, p. 151, Jul. 2023, doi: 10.3390/computers12080151.

[33] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1993*, Sep. 1993, pp. 279–288, doi: 10.1145/166117.166153.

[34] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach," in *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, New York, NY, USA: ACM, 1996, pp. 11–20.

[35] A. Ahmad *et al.*, "Artificial neural networks classification of s-band absorption performance in eco-friendly microwave absorbers," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 15, no. 1, pp. 1007–1014, Feb. 2025, doi: 10.11591/ijece.v15i1.pp1007-1014.

[36] K. O. Bachri, B. Santoso, D. K. Y. Hutapea, C. O. Sereati, and L. W. Pandjaitan, "Tomato leaf disease recognition system using Faster R-CNN," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 22, no. 6, pp. 1414–1421, Dec. 2024, doi: 10.12928/TELKOMNIKA.v22i6.25519.

[37] S. U. M. Rao, K. Sreekala, P. S. Rao, N. Shirisha, G. Srinivas, and E. Sreedevi, "Plant disease classification using novel integration of deep learning CNN and graph convolutional networks," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 36, no. 3, pp. 1721–1730, Dec. 2024, doi: 10.11591/ijeecs.v36.i3.pp1721-1730.

[38] M. M. Islam, S. Nooruddin, F. Karray, and G. Muhammad, "Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects," *Computers in Biology and Medicine*, vol. 149, p. 106060, Oct. 2022, doi: 10.1016/j.compbiomed.2022.106060.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[40] I. U. W. Mulyono, E. H. Rachmawanto, C. A. Sari, and M. K. Sarker, "A high accuracy of deep learning based CNN architecture: classic, VGGNet, and RestNet50 for Covid-19 image classification," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 22, no. 5, pp. 1187–1195, Jul. 2024, doi: 10.12928/TELKOMNIKA.v22i5.26017.

[41] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Q. Weinberger, "Convolutional networks with dense connectivity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 8704–8716, Dec. 2022, doi: 10.1109/TPAMI.2019.2918284.

[42] S. K. Noon, M. Amjad, M. A. Qureshi, and A. Mannan, "Overfitting mitigation analysis in deep learning models for plant leaf disease recognition," in *Proceedings - 2020 23rd IEEE International Multi-Topic Conference, INMIC 2020*, Nov. 2020, pp. 1–5, doi: 10.1109/INMIC50486.2020.9318044.

[43] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognition*, vol. 91, pp. 216–231, Jul. 2019, doi: 10.1016/j.patcog.2019.02.023.

[44] M. Hasnain, M. F. Pasha, I. Ghani, M. Imran, M. Y. Alzahrani, and R. Budiarto, "Evaluating trust prediction and confusion matrix measures for web services ranking," *IEEE Access*, vol. 8, pp. 90847–90861, 2020, doi: 10.1109/ACCESS.2020.2994222.

BIOGRAPHIES OF AUTHORS

Suastika Yulia Riska, S.Pd., M.Kom., Suastika Yulia Riska, S.Pd., M.Kom., currently working at the Informatics Engineering Study Program, Faculty of Technology and Design, Asia Institute of Technology and Business Malang. He graduated from S1 Informatics Engineering Education, State University of Malang and he graduated from S2 Informatics Engineering, ITS. The research fields carried out are machine learning, data mining, expert system, and computer vision. She can be contacted at email: riska.suastika@asia.ac.id.

Danang Arbian Sulistyo, S.ST, M.Kom., currently working at the Informatics Engineering Study Program, Faculty of Technology and Design, Asia Institute of Technology and Business Malang. He graduated from the D4 - Pens ITS Program and S2 from STTS Surabaya, and is now pursuing a doctoral program in the Electrical and Informatics Engineering Study Program, State University of Malang. His research area is in the field of natural language processing, network and expert system. He can be contacted at email: danangarbian@gmail.com.

Farah Shafiyah Siti Maharani, S.Kom., currently studying at the Informatics Engineering Study Program, Faculty of Technology and Design, Asia Institute of Technology and Business Malang. She started her undergraduate studies in 2020 with a concentration in intelligent systems. Farah was born in Batam and is passionate about exploring various areas of technology. She can be contacted at email: farahs2maharani.23@gmail.com.