Jurnal Teknologi Informasi & Komunikasi

Volume 16 Issue 1 Year 2025 | Page 1-11 | e-ISSN: 2477-3255 | ISSN: 2086-4884
Received: 04-11-2024 | Revised: 14-02-2025 | Accepted: 11-03-2025

Deep Learning for Horticulture: Convolutional Neural Network Driven Classification of Banana Types

Ida Astuti[1], Lutfi Nabhan Ibrahim[2], Winda Widya Ariestya[3*], Syamsi Ruhama[4], Diny Wahyuni[5]
[1,2,3,4,5] Faculty of Computer Science and Information Technology, Universitas Gunadarma
[1,2,3,4,5]Jl. Margonda Raya No.100, Depok, Jawa Barat
e-mail: [1]astuti@staff.gunadarma.ac.id, [2]luthfinabhan@gmail.com, [3]winda_widya@staff.gunadarma.ac.id, [4]susi22@staff.gunadarma.ac.id, [5]dwahyuni@staff.gunadarma.ac.id
*Corespondence: winda_widya@staff.gunadarma.ac.id

Abstract: *One of the most widely grown horticulture fruits in Indonesia is the banana. In addition to its various health benefits, bananas are a good source of carbohydrates and vitamins A, C, and E. There are a lot of different kinds of bananas in Indonesia, and occasionally people have trouble telling them apart. This study uses a Convolutional Neural Network (CNN), a Deep Learning technique, to categorize bananas. Four different types of bananas—Cavendish, Kepok, Raja, and Tanduk—were classified. Planning, analysis, creating a banana classification model with CNN, and assessing the outcomes are the four phases of the research process. Data preprocessing, CNN model creation, training, and testing procedures are the next steps in the categorization model design process, which starts with the collection of banana data using a smartphone camera. The optimal model was obtained with the accuracy value of 96%, the average precision and recall values of 97% and 96% respectively. It was found based on test results with multiple tuning parameters, including dataset partition, optimizer use, and epoch. This study offers novelty in terms of the use of a large banana image dataset, extensive exploration of CNN parameters, and the potential application of the model in applications for the horticultural industry. In addition, this study contributes to the development of image-based AI technology in agricultural product classification, which is still relatively underexplored in Indonesia*

Keywords: *Horticulture, Classification, Banana, Deep Learning, Convolutional Neural Network*

1. Introduction

Horticulture is a branch of plant farming that cultivates plants in gardens or yards. Horticultural plants have economic value and are useful as resources for consumption [1]. Bananas are a horticultural plant that offers numerous health benefits due to their high concentration of carbohydrates, vitamins A, C, and E, among other nutrients [2]. The banana (Musa paradisiaca) is a highly favored fruit in Indonesia. Bananas are monocotyledonous plants that lack branches and have moist stems [3].

Planting bananas is quite simple and they may grow anywhere. According to data from the Central Statistics Agency (BPS), 9.60 million tons of bananas were produced in Indonesia in 2022 (https://www.bps.go.id, 2022). East Java, Lampung, and West Java are the three regions in Indonesia that produce the most bananas. In Indonesia, bananas come in a variety of varieties, such as raja, cavendish, susu, ambon, kepok, tanduk, and more [4].

Although there are many varieties of bananas in Indonesia, people find it hard to distinguish based only on their color and shape. With the advancement of technology, especially the

https://doi.org/10.31849/digitalzone.v16i1. 19680

information technology, a number of technologies have emerged and can be applied to address different challenges in object differentiation. Deep Learning is one of technology which can be used to identify and categorize items [5]. One of machine learning technique called "deep learning" imitates the nerve system of the human brain [6]. The Convolutional Neural Network (CNN) approach is one of the Deep Learning techniques that can be employed. CNN is a neural network-based Deep Learning algorithm made to handle arrays of data, including images[7].

Banana classification has become the subject of numerous studies. One of these studies uses the KNN and PCA methods to classify bananas based on RGB and HSV colors (mental, ripe, and too mature bananas), obtained accuracy of 90.9% [8]. An overall accuracy of 89,86% was obtained in other research by categorizing banana types using the Support Vector Machine approach [9]. The five different banana varieties, namely Raja, Kepok, Barangan, Lilin and Awak bananas were classified using the Backpropagation Method resulting in an accuracy level of 80% [10]. Research on the classification of banana types based on color, texture, and image shape features on seven different types of bananas (Ambon, Kepok, Susu, Raja, Mas, Raja Nangka, and Cavendish) was using the SVM and KNN methods. The test results indicate that the accuracy value of the SVM algorithm for classifying banana types based on color, texture, and shape features is 41.67%, 33.3%, and 8.3%, respectively, and the best K value is 2 on the color feature 55.95%, the texture feature 58.33%, and the shape feature 45.24% [11]. Another study classified banana types (Kepok, Emas, Susu, Santen, Ambon, Raja, and Candi) this study proves that the algorithm with Linear Support Vector Machine (Linear SVM) is able to provide the best classification, which is 99.1% [12]. Since the deep learning approach has not being used in the aforementioned research on banana categorization, the image characteristics must be extracted before the learning process begin.

Development of AI-powered sorting and grading system for efficient post-harvest management: Artificial Intelligence (AI) is used in the sorting and grading of banana fruits through the application of deep learning models such as EfficientNetB7 and NasNetLarge. These models are drilled on banana datasets to classify the fruits into various categories based on their quality and ripeness [13].

In the study of M. Senthilarasi, S Md.MansoorRoomi, M Maheesa, R Sivaranajani (2020), proposed a technique to identify banana varieties in India using deep convolutional neural network model to classify eight banana varieties namely Karpuravalli (ABB), Rasthali (AAB), Poovan (AAB), Nendran (AAB), Hill/Malai (AAB), Nali Poovan (AB), Monthan (ABB) and Grand Naine (AAA). The proposed deep learning banana variety classifier has obtained a better accuracy of 88.6% which is higher than the banana variety classification using VGG16 network model [14].

F. Huda and M. P. K. Putra previously carried out a study titled "Klasifikasi Jenis Buah Pisang Menggunakan Algoritma Convolutional Neural Network" which used 3000 banana datasets to be categorized in three different varieties of bananas: Jantan bananas, Kepok bananas and Muli bananas. The study only used learning rate as a criterion and yielded a 78% accuracy score when tested [15].

J. Halim and A. N. Fajar previously conducted a study titled "Klasifikasi Pisang Berbasis Algoritma VGG16 Melalui Metode CNN Deep Learning" which used 550 banana datasets to classify banana ripeness into 5 classes: unripe, slightly ripe, ripe, overripe, and rotten. The best performance of the model shown by the accuracy result of 83.53% with 50 epochs and it simply by just the epochs as the only parameter. [16].

This study aims to contribute to the development of technology in the horticulture sector by providing an automatic classification system that can help the agricultural industry identify and categorize bananas more accurately and efficiently by utilizing CNN as a Deep Learning method that can extract features automatically, so it does not require manual feature extraction as in conventional Machine Learning methods. The CNN algorithm is used because it can automatically extract features without human help. CNN's multi-dimensional object identification skills allow this technology to extract as much information as possible from the data [17]. Another reason for using CNN in this study is based on the findings of research conducted by Kamilaris, A., & Prenafeta-Boldú, FX (2018) showing that CNN is a promising technique with high performance in terms of precision and classification accuracy, outperforming commonly used image transmission techniques. However, the success of each CNN model is highly dependent on the quality of the dataset used [18].

The shortcomings of previous studies used a relatively small dataset of 550-3000 images, this study used 6160 banana images, which are much larger, so it is expected to increase classification accuracy. In addition, there are also shortcomings in previous studies, most previous studies only use a few tuning parameters such as learning rate or number of epochs, so model optimization is still limited. While this study tested several learning scenarios, including dataset splitting, number of epochs, learning rate, and optimizer selection, in order to obtain the best combination of parameters to improve model accuracy.

This study provides novelty in terms of the use of a large banana image dataset, extensive exploration of CNN parameters, and the potential application of the model in applications for the horticultural industry. In addition, this study contributes to the development of image-based AI technology in agricultural product classification, which is still relatively underexplored in Indonesia.

2. Method

The CNN algorithm is used to classify the different kinds of bananas in this study. The following steps make up the research stages used to categorize bananas:

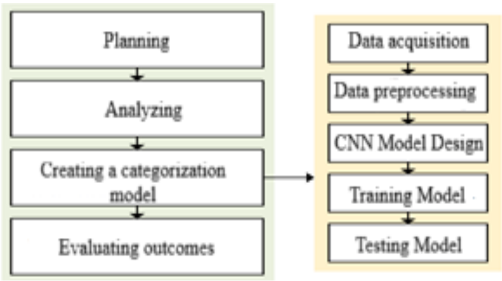


Figure 1: Stages of Research

Planning, analysis, creating a categorization model, and evaluation outcomes are the four primary phases of the research that was conducted. The planning stage is the first step that is completed; it involves organizing the purpose and context of the research that is being conducted. This study was carried out because there are many different kinds of bananas in Indonesia, making it frequently difficult for individuals to tell one banana from another. Cavendish, kepok, raja, and taduk bananas are the several varieties of bananas that are categorized. To execute the training and testing process and implement the classification model, data is required. The information needed consists of pictures of Cavendish, kepok, raja, and tanduk bananas. The Asus Zenfone Max Pro M2 smartphone's camera was used to capture data in the form of pictures under typical lighting conditions using the Fast Burst Camera Mobile app. Training, validation, and test data will all be derived from the acquired images. The CNN algorithm will be used in the creation of the banana classification model that will be developed. In order to classify the test data, the model must be able to determine if the image is of a Cavendish, kepok, raja, and tanduk banana.

Following the completion of the planning stage, the research moves on to the second stage, which is the analyzing stage. In order to conduct the research, an analysis of what is required is done at this point. Image data of Cavendish, Kepok, Raja, and Tanduk bananas; 256x256 pixel image data; and image data with the extension .JPG are among the data required for this study.

After the analyzing stage, the research moves on to the third stage, which is the design of a banana classification model. At this point, the data acquiring process, data preprocessing, CNN model creation, training model, and testing model are completed. Data acquisition is the process

https://doi.org/10.31849/digitalzone.v16i1. 19680

of gathering and collecting data that will be utilized as material or sources for the research to be conducted. After collecting data is completed, the current images will go through a number of preprocessing processes to be modified to the needs, including data splitting, image augmentation, and data generator preparation. The dataset grouping in this study was done using multiple splitting ratios, as displayed in table 1.

The ratio for class dataset partition is chosen randomly. The dataset splitting ratio seeks to determine which ratio will produce the best classification results [19]. In this study, various dataset ratio scenarios were tested to find out which one provides the highest accuracy and the lowest loss. The dataset splitting ratio used in this study were 80:10:10, 70:15:15, and 60:20:20, these splitting ratio were chosen because they are often used in Deep Learning because they consider the balance between training, validation, and testing. The main reason for choosing this ratio is to maintain a balance between training and generalization and the possibility of overfitting and underfitting. 80:10:10 is used when the amount of data is quite large, so that most of the data can be used to train the model without reducing validation and testing performance. 70:15:15 or 60:20:20 are more suitable for smaller datasets or when stronger validation is needed to adjust hyperparameters. The model can overfit if too much data is used for training (for example 90% or more), that is, it relies too much on training data and is less able to recognize new data. Conversely, the model can underfit if the training data is too small (e.g. 50%), because it does not learn enough from the data to recognize complex patterns. Following the creation of the dataset, each dataset scenario is inserted into a directory form with the goal of making it easier to understand the directory in the following stage, as shown in Figure 2.

Table 1: Ratio of the Dataset Splitting

Splitting Ratio	Data	Percentage
	Train	60%
60:20:20	Validation	20%
	Test	20%
	Train	70%
70:15:15	Validation	15%
	Test	15%
	Train	80%
80:10:10	Validation	10%
	Test	10%

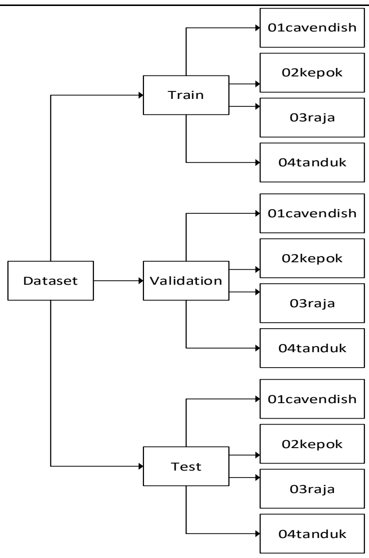


Figure 2: Directory of Dataset Placement

Image augmentation is the second step in the data preprocessing process. Because the CNN algorithm's features such as rescale, rotation range, shear range, zoom range, and horizontal vertical flip are not invariant to the image augmentation, the image augmentation procedure is implemented. Figure 3 is an illustration of the outcomes of this procedure. Furthermore, the purpose of the picture augmentation method is to enable the program to identify additional shapes in the images.

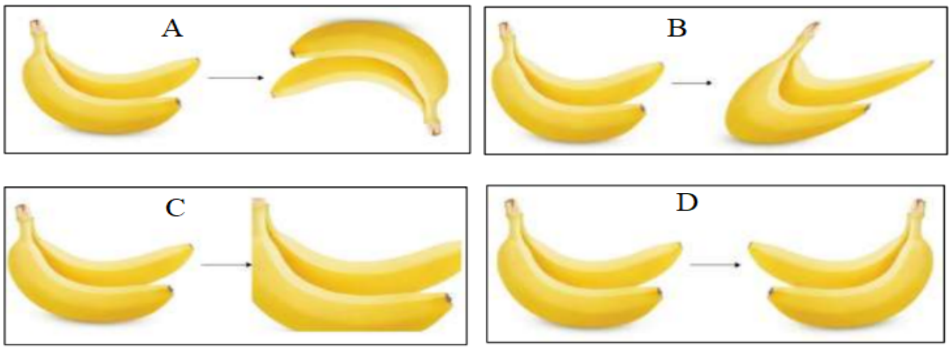


Figure 3. Example of rescale results (A), rotation range (B), shear range (C), zoom range and horizontal vertical flip (D)

The third step in the preprocessing stage is creating the data generator, which defines the data source, following the completion of the image augmentation procedure. Several picture parameters are also defined during this procedure, such as the target size, batch size, class mode, and shuffle. The image dimension that will be used for training is called the target size. This study uses a target size of 256x256. Target size will automatically resize the image from 512x512 pixels to 256x256 pixels. The selection of the 256×256 pixel size is based on considerations of computational efficiency, input consistency, and optimization of the CNN architecture. This size is large enough to retain important details in the texture, color, and shape of the banana, but also small enough to allow fast and accurate processing in model classification. In addition, many Deep Learning studies in image processing use 256×256 because it has been proven to provide optimal results with various CNN architectures such as VGG16, ResNet, and MobileNet. Each training step's batch size is the quantity of photos entered. One way of classification selection is class mode. Class modes can be either category or binary. To change the order of the images in a folder so that it differs from the original order, use the shuffle command.

Images of different banana varieties are then classified using the CNN model design. Tensorflow and Keras are used in the model design. The CNN model is divided into two phases: feature extraction and classification. In Figure 4, the CNN model design flow is depicted.

Feature extraction is the initial step in CNN design. Through this process, the image will be encoded into a feature that is represented by a number. The Convolutional Layer and the Pooling Layer are the two components that make up the feature extraction process. The preceding layer will supply input to the convolutional layer. If the convolutional layer is an input layer, on the other hand, the layer is not computed and its input is the image's original size (256x256).

Four convolutional layers were used in this research. There are 32 filters in the first and second layers, and 64 filters in the third and fourth layers. Every layer has a ReLu activation function and a 3x3 kernel size. The negative element value is changed to 0 using the ReLu activation function. By using down-sampling techniques, the pooling layer section seeks to decrease input (the amount of parameters). Max Pooling is the pooling type utilized in this investigation, and it will extract the maximum value from the kernel [20]. This study uses four Max Pooling layers, each with a 2x2 kernel size. The method moves on to the classification stage once the image has passed the feature extraction stage. This procedure, which includes the Flatten Layer and Fully-Connected Layer, will examine the outcomes of the earlier feature extraction. The final max pooling results will be fed into the Flatten Layer stage as a 2-dimensional array. In order to classify the image, a flatten function is required since the Fully-Connected Layer, the

https://doi.org/10.31849/digitalzone.v16i1. 19680

subsequent stage, requires input in the form of a 1-dimensional array. The Flatten Layer findings will be fed into the Fully-Connected Layer portion as a 1-dimensional array. The final layer has four neurons, the number of which varies depending on the type of banana fruit to be classified. The final layer employs the Softmax activation function. The Softmax function will calculate the probability for each existing class. The Backpropagation algorithm is used in the part with completely linked layers. This algorithm works by completing two steps of calculation: forward calculation, which calculates the error value between the system output value and the value that should be, and backward calculation, which corrects the weight using the error value [21].

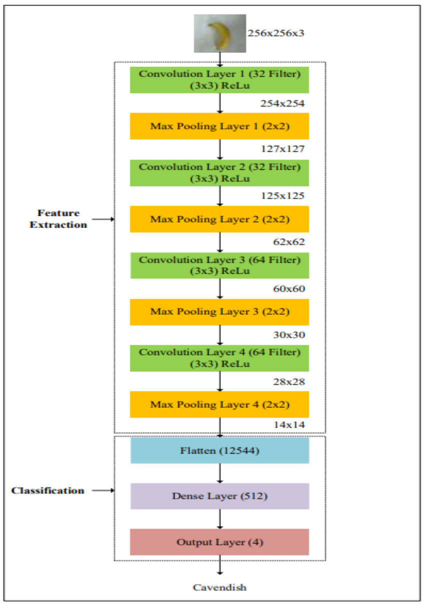


Figure 4: CNN Model Illustration

Model training is part of the ongoing process. This procedure is supposed to yield a model that has been trained to recall visual patterns accurately from each learned class. The training process is carried out with previously preprocessed training and validation data. Figure 4 illustrates the model training process flow. In training, the testing process is carried out utilizing three epoch scenarios: 15, 20, and 25. With the addition of epoch scenarios, the testing is performed using a variety of data splitting ratios from Table 1 epoch scenarios and two optimization scenarios.

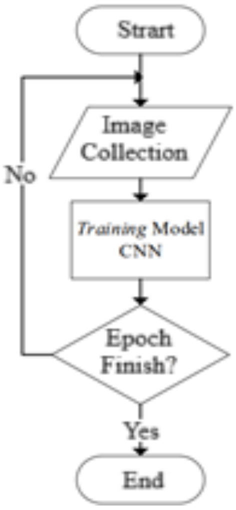


Figure 5. Training Model Flow

After the banana classification model design stage was completed, the research ended with the fourth stage, namely the results and evaluation stage. At this stage, the results were evaluated against the results of the banana classification model design that had been made.

3. Results and Discussion

The quantity of images taken of Cavendish, Kepok, Raja, and Tanduk bananas is shown in Table 2, and a total of 6160 images of these banana types were obtained. Additionally, the data was separated into three dataset scenarios, as shown in Table 3: 80:10:10, 60:20:20, and 70:15:15.

Table 2: Data Quantity by Banana Type

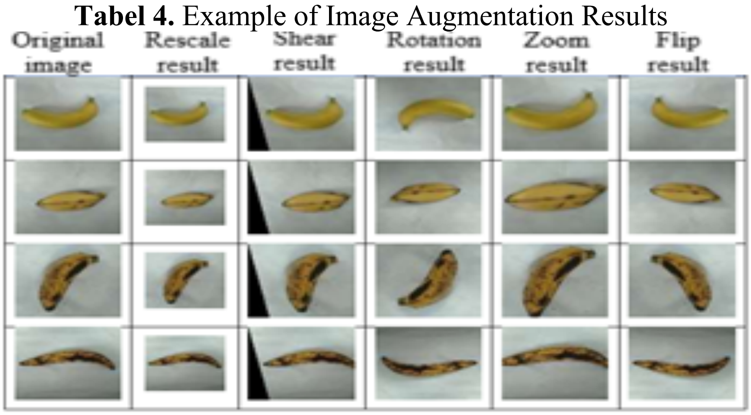
Number	Banana Type	Quantity of images
1	Cavendish	1540
2	Kepok	1540
3	Raja	1540
4	Tanduk	1540

Table 3. Results of Data Splitting Scenario

Scenario	Data	Banana Type			
		Cavendish	Kepok	Raja	Tanduk
60:20:20	Train	924	924	924	924
	Validation	308	308	308	308
	Test	308	308	308	308
70:15:15	Train	1078	1078	1078	1078
	Validation	231	231	231	231
	Test	231	231	231	231
80:10:10	Train	1232	1232	1232	1232
	Validation	154	154	154	154
	Test	154	154	154	154

The training_datagen and validation_datagen augmentation process has multiple parameters, including rescale of 1/255 for pixel conversion, rotation_range of 180 for 180-degree rotation, shear_range of 0.2 for angle shift, zoom_range of 0.2 for image enlargement, and horizontal_flip and vertical_flip. However, only the rescale parameter is used for the augmentation procedure on test_datagen. Table 4 shows the illustration of the outcomes of the image enhancement.

Tabel 4. Example of Image Augmentation Results



Creating a CNN model is the next step in the process after the data preparation is finished. A CNN model is constructed using a sequential model, which consists of four layers. It is evident from Figure 6, the model's output, that each layer generates distinct output parameters and shapes. It is possible to verify the model's output by doing computations by hand.

```
Model: "sequential"

Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 254, 254, 32)      896

max_pooling2d (MaxPooling2D) (None, 127, 127, 32)      0

conv2d_1 (Conv2D)            (None, 125, 125, 32)      9248

max_pooling2d_1 (MaxPooling2 (None, 62, 62, 32)        0

conv2d_2 (Conv2D)            (None, 60, 60, 64)        18496

max_pooling2d_2 (MaxPooling2 (None, 30, 30, 64)        0

conv2d_3 (Conv2D)            (None, 28, 28, 64)        36928

max_pooling2d_3 (MaxPooling2 (None, 14, 14, 64)        0

flatten (Flatten)            (None, 12544)             0

dropout (Dropout)            (None, 12544)             0

dense (Dense)                (None, 512)               6423040

dense_1 (Dense)              (None, 4)                 2052
=================================================================
Total params: 6,490,660
Trainable params: 6,490,660
Non-trainable params: 0
```

Figure 6: Summary of Model Results

Following the completion of the CNN architecture, research testing scenarios and the training process architecture are prepared. Several research testing scenarios are conducted during the training process by varying the number of epochs, optimization, and dataset partition.

The model training procedure is carried out right after the CNN model's ready to use. The train_generator data will be used to train the model, and the validation_generator data that was previously generated will be used for validation. Each epoch's history is shown in its entirely when the verbose parameter is set to 1. 15, 20, and 25 are the epochs used. Following the compilation of the dataset, 18 research testing scenarios were used for the model training procedure, as shown in Table 5.

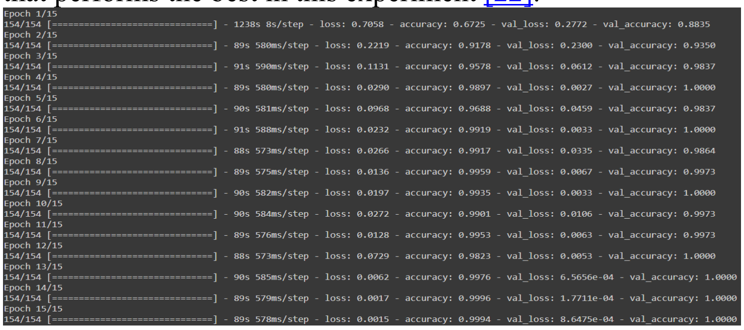
Table 5. Research Testing Scenario Training Outcomes

Research Scenario	Dataset Sharing (Data Train:Test: Validation)	Scenario Optimization	Scenario Epoch	*Scenario Learning*	Rate Accuracy	Validation Accuracy	Loss	Validation Loss
Skenario 1	60:20:20	Adam	15	0.001	99.95%	100.00%	0.20%	0.00%
Skenario 2	60:20:20	SGD	15	0.01	84.39%	93.02%	40.75%	17.79%
Skenario 3	60:20:20	Adam	20	0.001	99.92%	100.00%	0.32%	0.00%
Skenario 4	60:20:20	SGD	20	0.01	96.00%	98.46%	11.06%	6.37%
Skenario 5	60:20:20	Adam	25	0.001	99.57%	99.92%	1.35%	2.79%
Skenario 6	60:20:20	SGD	25	0.01	96.92%	98.70%	7.72%	3.89%
Skenario 7	70:15:15	Adam	15	0.001	99.23%	99.89%	2.39%	0.33%
Skenario 8	70:15:15	SGD	15	0.01	94.50%	96.75%	13.91%	8.45%
Skenario 9	70:15:15	Adam	20	0.001	99.54%	100.00%	1.30%	0.00%
Skenario 10	70:15:15	SGD	20	0.01	96.08%	98.48%	11.34%	5.60%
Skenario 11	70:15:15	Adam	25	0.001	99.70%	100.00%	0.83%	0.18%
Skenario 12	70:15:15	SGD	25	0.01	95.50%	99.24%	11.64%	2.65%

Research Scenario	**Dataset Sharing (Data Train:Test: Validation)**	**Scenario Optimization**	**Scenario Epoch**	***Scenario Learning***	**Rate Accuracy**	**Validation Accuracy**	**Loss**	**Validation Loss**
Skenario 13	80:10:10	Adam	15	0.001	99.94%	100.00%	0.15%	0.00%
Skenario 14	80:10:10	SGD	15	0.01	93.26%	98.05%	16.83 %	6.09%
Skenario 15	80:10:10	Adam	20	0.001	99.57%	100.00%	1.00%	0.00%
Skenario 16	80:10:10	SGD	20	0.01	97.04%	98.54%	8.94%	4.16%
Skenario 17	80:10:10	Adam	25	0.001	99.92%	100.00%	0.21%	0.00%
Skenario 18	80:10:10	SGD	25	0.01	87.09%	94.16%	34.51 %	13.52%

Scenario 13 had the highest accuracy, validation accuracy, loss, and validation loss values out of the 18 scenarios that were run, according to the training comparison data in Table 6. Scenario 13 is composed with 15 epochs, the Adam optimizer, and a dataset comparison of 80:10:10. In Figure 6, the training procedure for scenario 13 is displayed.

The Adam optimizer performs better than the SGD optimizer across all cases. According to the findings of the analysis in Afis Julianto's research [22], the longer the training procedure and the higher the accuracy value, the larger the epoch value employed. According to Afis Julianto's research, the learning rate values of 0.01 and 0.001 are employed in learning; 0.001 is the learning rate that performs the best in this experiment [22].



```
Epoch 1/15
154/154 [==============================] - 1238s 8s/step - loss: 0.7058 - accuracy: 0.6725 - val_loss: 0.2772 - val_accuracy: 0.8835
Epoch 2/15
154/154 [==============================] - 89s 580ms/step - loss: 0.2219 - accuracy: 0.9178 - val_loss: 0.2300 - val_accuracy: 0.9350
Epoch 3/15
154/154 [==============================] - 91s 590ms/step - loss: 0.1131 - accuracy: 0.9578 - val_loss: 0.0612 - val_accuracy: 0.9837
Epoch 4/15
154/154 [==============================] - 89s 580ms/step - loss: 0.0290 - accuracy: 0.9897 - val_loss: 0.0027 - val_accuracy: 1.0000
Epoch 5/15
154/154 [==============================] - 90s 581ms/step - loss: 0.0968 - accuracy: 0.9688 - val_loss: 0.0459 - val_accuracy: 0.9837
Epoch 6/15
154/154 [==============================] - 91s 588ms/step - loss: 0.0232 - accuracy: 0.9919 - val_loss: 0.0033 - val_accuracy: 1.0000
Epoch 7/15
154/154 [==============================] - 88s 573ms/step - loss: 0.0266 - accuracy: 0.9917 - val_loss: 0.0335 - val_accuracy: 0.9864
Epoch 8/15
154/154 [==============================] - 89s 575ms/step - loss: 0.0136 - accuracy: 0.9959 - val_loss: 0.0067 - val_accuracy: 0.9973
Epoch 9/15
154/154 [==============================] - 90s 582ms/step - loss: 0.0197 - accuracy: 0.9935 - val_loss: 0.0033 - val_accuracy: 1.0000
Epoch 10/15
154/154 [==============================] - 90s 584ms/step - loss: 0.0272 - accuracy: 0.9901 - val_loss: 0.0106 - val_accuracy: 0.9973
Epoch 11/15
154/154 [==============================] - 89s 576ms/step - loss: 0.0128 - accuracy: 0.9953 - val_loss: 0.0063 - val_accuracy: 0.9973
Epoch 12/15
154/154 [==============================] - 88s 573ms/step - loss: 0.0729 - accuracy: 0.9823 - val_loss: 0.0053 - val_accuracy: 1.0000
Epoch 13/15
154/154 [==============================] - 90s 585ms/step - loss: 0.0062 - accuracy: 0.9976 - val_loss: 6.5656e-04 - val_accuracy: 1.0000
Epoch 14/15
154/154 [==============================] - 89s 579ms/step - loss: 0.0017 - accuracy: 0.9996 - val_loss: 1.7711e-04 - val_accuracy: 1.0000
Epoch 15/15
154/154 [==============================] - 89s 578ms/step - loss: 0.0015 - accuracy: 0.9994 - val_loss: 8.6475e-04 - val_accuracy: 1.0000
```

Figure 7. Scenario 13 Training Process

The training procedure was conducted for 15 epochs, as shown by the outcomes of scenario 13's training process in Figure 7. The 41-minute training process was conducted via Google Collaboratory. In scenario 13, 4928 training data and 616 validation data were used for the training procedure.

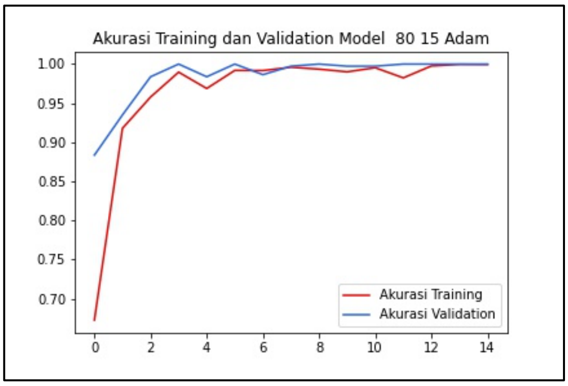


Figure 8. Scenario 13 Accuracy and Accuracy Validation Graph

The accuracy and accuracy validation graph for scenario 13 is shown in Figure 8. It is evident from Figure 8 that accuracy and accuracy validation are increasing with each epoch. The first to second epochs see the fastest increase. Following the second epoch, there is typically no discernible rise or reduction in accuracy or accuracy validation.

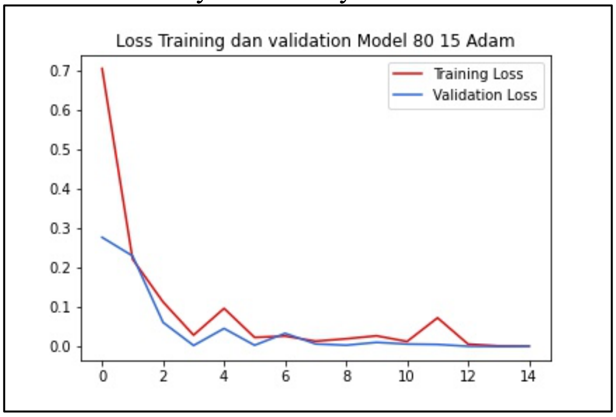


Figure 9. Scenario 13 for Loss Validation and Loss Graph

The loss and validation loss graph for scenario 13 is shown in Figure 9. It is clearly shown from Figure 9 that loss and validation loss reduce with each epoch. The first three epochs see the fastest decline. Testing is the next step in the process after model training is finished. The training result model from scenario 13 is the model that is utilized for testing. The data being tested comes from the 80:10:10 dataset, which contains up to 616 images.

Each image's prediction result is shown in Figure 10. Numbers 0 through 3 appear in the Image Label and Prediction Label columns. The Cavendish banana class is represented by the number 0, the Kepok banana class by the number 1, the Raja banana class by the number 2, and the Tanduk banana class by the number 4. The bar chart can be used to create a graph that shows the prediction results from the test data. As shown in Figure 11, 154 of the photos are projected to be Cavendish bananas, 154 to be Kepok bananas, 130 to be Raja bananas, and 178 to be Tanduk bananas.

	Nama File	Label Gambar	Label Prediksi
0	cavendish (9).jpg	0	0
1	cavendish (14).jpg	0	0
2	cavendish (29).jpg	0	0
3	cavendish (70).jpg	0	0
4	cavendish (62).jpg	0	0
5	cavendish (104).jpg	0	0
6	cavendish (101).jpg	0	0
7	cavendish (100).jpg	0	0
8	cavendish (103).jpg	0	0
9	cavendish (102).jpg	0	0
10	cavendish (105).jpg	0	0
11	cavendish (125).jpg	0	0
12	cavendish (149).jpg	0	0
13	cavendish (157).jpg	0	0
14	cavendish (155).jpg	0	0
15	cavendish (151).jpg	0	0
16	cavendish (153).jpg	0	0
17	cavendish (208).jpg	0	0
18	cavendish (197).jpg	0	0
19	cavendish (210).jpg	0	0
20	cavendish (219).jpg	0	0
21	cavendish (218).jpg	0	0
22	cavendish (221).jpg	0	0
23	cavendish (231).jpg	0	0
24	cavendish (261).jpg	0	0
25	cavendish (248).jpg	0	0
26	cavendish (263).jpg	0	0
27	cavendish (293).jpg	0	0

Figure 10. Results of the Prediction

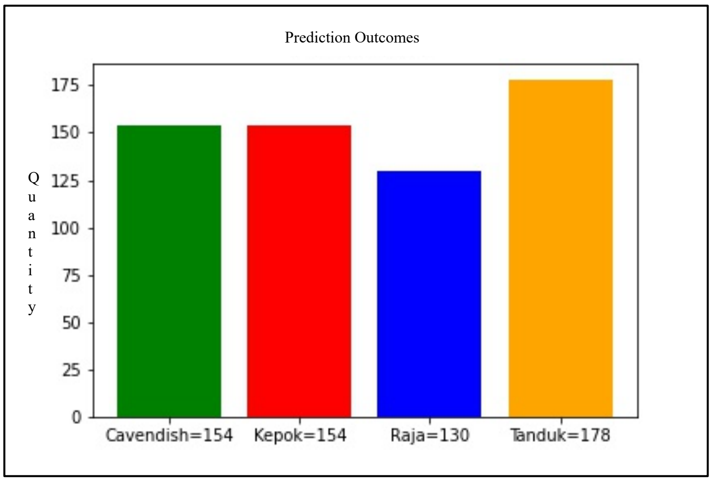


Figure 11. Graph of Prediction Outcomes

Evaluation of the outcomes is the next step in the process once the model testing is finished. This procedure seeks to ascertain whether or not the developed model performs well. All of the forecasts from the Cavendish, Kepok, and Tanduk banana test data match the labels, as seen in Figure 12. In the meantime, 24 of the 154 Raja banana images were mispredicted.

Accuracy, precision, recall, and f1-score are then determined using the equation following the completion of the confusion matrix:

Accuracy = (TP + TN) / (TP+FP+FN+TN)

Precision = (TP) / (TP+FP)

Recall = (TP) / (TP + FN)

F1=2. Precision.Recall / Recall+Recall

After analyzing 616 images of banana types, the TP value was 592, FN was 24, TN was 592, and FP was 24.

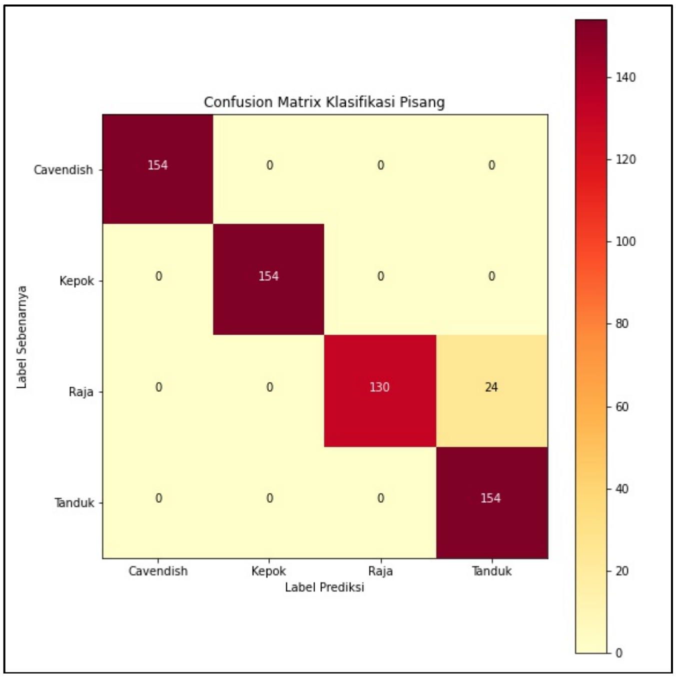


Figure 12. Confusion Matrix Prediction Results

https://doi.org/10.31849/digitalzone.v16i1. 19680

	precision	recall	f1-score	support
Cavendish	1.00	1.00	1.00	154
Kepok	1.00	1.00	1.00	154
Raja	1.00	0.84	0.92	154
Tanduk	0.87	1.00	0.93	154
accuracy			0.96	616
macro avg	0.97	0.96	0.96	616
weighted avg	0.97	0.96	0.96	616

Figure 13. Accuracy, Precision, Recall, and F1 Score

Figure 13 presents the model's average accuracy of 96%, indicating that it can accurately determine the four different banana varieties: Cavendish, Kepok, Raja, and Tanduk. This classification model shows a better level of accuracy than earlier researches conducted by F. Huda and M. P. K. Putra [15] as well as by J. Halim and A. N. Fajar [16], according to these findings. It can be stated that the model can performs better as a result of the higher number of datasets and also various testing parameters used.

In this study, several findings were obtained, namely: first, high model accuracy. The developed CNN model achieved an average accuracy of 96% in classifying four types of bananas (Cavendish, Kepok, Raja, and Tanduk). This accuracy is higher than previous studies, such as that conducted by F. Huda [15], which only reached 78%. Second, the influence of the dataset splitting scheme. The 80:10:10 dataset splitting scheme gave the best classification results compared to other schemes (70:15:15 and 60:20:20). This scheme allows the model to obtain sufficient training data, without reducing the effectiveness of validation and testing. Third, the effectiveness of the adam optimizer. The adam optimizer gave better results compared to the SGD optimizer, producing higher accuracy and validation accuracy. Fourth, optimal epoch. 15 epochs produced the best performance, where the greatest increase in accuracy occurred in the early epochs (1-2), while after that it tended to be stable. loss and validation loss continued to decline, with the fastest decline in the first three epochs.

The following is a discussion of the interpretation of the results obtained in this study. First, CNN model is effective for banana classification. The results show that CNN can automatically extract features from banana images without the need for manual feature extraction. The model can distinguish four types of bananas with high accuracy, especially in the Cavendish, Kepok, and Tanduk classes. Second, prediction error in the Raja Class, 24 out of 154 Raja images were predicted incorrectly, indicating that the model had little difficulty in classifying this type of banana. This error could come from the similarity of visual features between Raja bananas and other types, or the lack of samples for this class. Third, selection of the right parameters affects model performance. 80:10:10 as the best scheme shows that proportional splitting of the dataset is very important in minimizing overfitting and increasing model generalization. Adam optimizer and learning rate 0.001 are proven to provide optimal performance in this model.

Some implications of this research are: first, use of deep learning in horticulture. This model shows that CNN technology can be applied to classify fruits with high precision, supporting automation in the agricultural and horticultural industries. Can be used to help farmers or distributors in sorting banana types automatically. Second, further development opportunities. This research opens up opportunities for the use of more complex models such as ResNet or EfficientNet to further improve accuracy. It can be further developed to detect banana ripeness, not just classify the type. Third, implementation in industry and market. This model has the potential to be used in the banana supply chain, where automation of classification can help packaging and distribution more efficiently.

There are still limitations that become challenges such as the model still experiences prediction errors in the Raja banana class, indicating that it may need dataset adjustments or additional augmentation techniques

4. Conclusions

The banana classification research using CNN algorithm was completed successfully. The CNN algorithm is used to classify four different types of bananas: Cavendish, Kepok, Raja, and Tanduk. The data acquisition stage of developing a banana classification model with CNN begins with capturing a dataset of 6160 banana images with a smartphone camera and splitting it into training, validation, and testing datasets. The images are then classified using 18 testing scenarios.

Based on the CNN algorithm testing findings, the research testing scenario with the 80:10:10 dataset scenario, the Adam optimization scenario, and the 15 epochs scenario is the best model, with the accuracy of 99.94% and a loss of 0.15. Using 616 test data, accuracy was 96%, with average precision and recall values of 97% and 96%, respectively. Based on these results, it is possible to conclude that the CNN algorithm accurately classifies four types of bananas. It may be possible to advance this idea by putting it into practice as an application. In this manner, consumers may realistically identify and categorize different kinds of bananas in enormous quantities. Additionally, by offering technologically based solutions, this application can help Indonesia's horticulture industry grow.

References

[1] N. Nazimah, N. Nilahayati, S. Safrizal, and S. Fachrurrazi, "Pemberdayaan Masyarakat Di Desa Baloy Kecamatan Blang Mangat Dalam Aplikasi Pupuk Hayati Untuk Budidaya Tanaman Hortikultura," *J. Vokasi*, vol. 6, no. 1, p. 40, 2022, doi: 10.30811/vokasi.v6i1.2923.

[2] N. A. Syafriani, Afiah, "PKM Petani Pisang Di Kampar," vol. 2, no. 1, pp. 113–119, 2022, doi: https://doi.org/10.31004/covit.v2i1.4525.

[3] I. F. A and P. S. Rizki, "Pengaruh Karbon Aktif Dari Kulit Pisang Tanduk Terhadap Limbah Cair Tahu Menggunakan Parameter pH, Cod (Chemical Oxygen Demand), Do (Disolved Oxygen) & Chlorida," *JIRK J. Innov. Res. Knowl. Vol.2, No.10, Maret 2023*, vol. 2, no. 10, pp. 4249–4261, 2023, doi: https://doi.org/10.53625/jirk.v2i10.5465.

[4] Badan Pusat Statistik, "Produksi Tanaman Buah-Buahan Tahun 2022," 2022. [Online]. Available: https://www.bps.go.id

[5] M. J. Akbar, M. W. Sardjono, M. Cahyanti, and E. R. Swedia, "Perancangan Aplikasi Mobile Untuk Klasifikasi Sayuran Menggunakan Deep Learning Convolutional Neural Network," *Sebatik*, vol. 24, no. 2, pp. 300–306, 2020, doi: 10.46984/sebatik.v24i2.1134.

[6] I. Arifin, R. F. Haidi, and M. Dzalhaqi, "Penerapan Computer Vision Menggunakan Metode Deep Learning pada Perspektif Generasi Ulul Albab," *J. Teknol. Terpadu*, vol. 7, no. 2, pp. 98–107, 2021, doi: 10.54914/jtt.v7i2.436.

[7] A. A. Elngar *et al.*, "Image Classification Based On CNN: A Survey," *J. Cybersecurity Inf. Manag.*, vol. 6, no. 1, p. PP. 18-50, 2021, doi: 10.5281/zenodo.4897990.

[8] S. P. Adenugraha, V. Arinal, and D. I. Mulyana, "Klasifikasi Kematangan Buah Pisang Ambon Menggunakan Metode KNN dan PCA Berdasarkan Citra RGB dan HSV," *J. Media Inform. Budidarma*, vol. 6, no. 1, pp. 9–17, 2022, doi: 10.30865/mib.v6i1.3287.

[9] Y. Amrozi, D. Yuliati, A. Susilo, N. Novianto, and R. Ramadhan, "Klasifikasi Jenis Buah Pisang Berdasarkan Citra Warna dengan Metode SVM," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 11, no. 3, pp. 394–399, 2022, doi: 10.32736/sisfokom.v11i3.1502.

[10] D. Widodo, A. Fauzi, and A. Sembiring, "Identification of Banana Fruit Types Using the Backpropagation Method," *J. Math. Technol.*, vol. 2, no. 2, pp. 300–307, 2023, doi:

https://doi.org/10.63893/matech.v2i2.155.
[11] Y. E. Yana and N. Nafi'iyah, "Klasifikasi Jenis Pisang Berdasarkan Fitur Warna, Tekstur, Bentuk Citra Menggunakan SVM dan KNN," *Res. J. Comput. Inf. Syst. Technol. Manag.*, vol. 4, no. 1, p. 28, 2021, doi: 10.25273/research.v4i1.6687.
[12] I. A. Sabilla, C. S. Wahyuni, C. Fatichah, and D. Herumurti, "Determining banana types and ripeness from image using machine learning methods," in *Proceeding - 2019 International Conference of Artificial Intelligence and Information Technology, ICAIIT 2019*, IEEE, 2019, pp. 407–412. doi: 10.1109/ICAIIT.2019.8834490.
[13] P. Sudharshan Duth and B. S. Shashidhara, "Banana Grading using Deep Learning Techniques," in *IEEE International Conference on Data Science and Information System (ICDSIS)*, Hassan, India: IEEE, 2022, pp. 1–5. doi: 10.1109/ICDSIS55133.2022.9915998.
[14] R. S. M. Senthilarasi, S Md.MansoorRoomi, M Maheesa, "Deep Learn Bananas: A Transfer Learning for Banana Variety Classification," in *Communications in Computer and Information Science*, Singapore: Springer, Singapore, 2020. doi: https://doi.org/10.1007/978-981-15-8697-2_53.
[15] F. Huda and M. P. K. Putra, "Klasifikasi Jenis Buah Pisang Menggunakan Algoritma Convolutional Neural Network," *J. Artif. Intell. Technol. Inf.*, vol. 1, no. 3, pp. 100–105, 2023, doi: 10.58602/jaiti.v1i3.61.
[16] J. Halim and A. N. Fajar, "Klasifikasi Pisang Berbasis Algoritma VGG16 Melalui Metode CNN Deep Learning," *Inf. (Jurnal Inform. dan Sist. Informasi)*, vol. 15, no. 1, pp. 1–17, 2023, doi: 10.37424/informasi.v15i1.190.
[17] Y. A. Suwitono and F. J. Kaunang, "Implementasi Algoritma Convolutional Neural Network (CNN) Untuk Klasifikasi Daun Dengan Metode Data Mining SEMMA Menggunakan Keras," *J. Komtika (Komputasi dan Inform.*, vol. 6, no. 2, pp. 109–121, 2022, doi: 10.31603/komtika.v6i2.8054.
[18] A. Kamilaris and F. X. Prenafeta-Boldú, "A literature survey on the use of Convolutional Neural Networks in agriculture," IRTA Torre Marimon, Inst. Food Agric. Res. Technol. Torre Marimon, Caldes Montbui, Barcelona, Cambridge Univ. Press, pp. 1–28, 2018, doi: 10.1017/S0021859618000436
[19] I. Astuti, W. W. Ariestya, and B. Solehudin, "Deteksi Objek Daun Semanggi Secara Real Time Menggunakan CNN-Single Shot Multibox Detector (SSD)," *J. Ilm. FIFO*, vol. 14, no. 1, p. 47, 2022, doi: 10.22441/fifo.2022.v14i1.005.
[20] L. Zhao and Z. Zhang, "A improved pooling method for convolutional neural networks," Nature Publishing Group UK, 2024. doi: 10.1038/s41598-024-51258-6.
[21] H. A. Akbar Fikri, "Prediksi Harga Emas Dengan Algoritma Backpropagation," *J. Sains Komput. & Inform.*, vol. 7, no. 1, pp. 182–189, 2023, doi: 10.30645/j-sakti.v7i1.582
[22] A. Julianto, A. Sunyoto, and F. W. Wibowo, "Optimasi Hyperparameter Convolutional Neural Network Untuk Klasifikasi Penyakit Tanaman Padi," Tek. Teknol. Inf. dan Multimed., vol. 3, no. 2, pp. 98–105, 2022, doi: 10.46764/teknimedia.v3i2.77.