J-COSINE (Journal of Computer Science and Informatics Engineering)
Vol. 8, No. 2, December 2024 E-ISSN:2541-0806
Accredited Sinta-4 by RISTEKDIKTI Decree No. 79/E/KPT/2023 P-ISSN:2540-8895

Classification of Local Fruit Types using Convolutional Neural Network Method

(Study Case: Lombok Island)

Moh. Azzam Al Husaini*, Ario Yudo Husodo, Fitri Bimantoro
Dept Informatics Engineering, Mataram University
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA
Email: azzamkasavani4@gmail.com, *:* [ario, bimo]@unram.ac.id

Corresponding Author

Abstract **Indonesia, with its natural beauty and abundant resources, has significant potential for producing food and horticultural crops, particularly on Lombok Island, West Nusa Tenggara. This region is crucial in supplying tropical fruits such as Mangosteen, Pisang Kepok, and Rambutan Lebak Bulus. However, the agricultural sector in NTB faces challenges in post-harvest handling, especially in classifying fruit ripeness, impacting distribution and supply sustainability. To address this, researchers developed a fruit classification model using digital image processing with the Convolutional Neural Network (CNN) method. This model serves as a preliminary step before creating a fruit maturity classification model. Evaluation results showed that the RGB format model achieved 95% accuracy, while the HSV format reached 97%. Comparing three models in HSV format revealed: the proposed model (0.97), MobileNetV2 (0.96), and ResNet50 (0.97). These results indicate that implementing this model could enhance post-harvest efficiency in NTB, ensuring better fruit supply management.**

***Key words*: CNN, HSV, MobileNetV2, RGB, ResNet50**

I. Introduction

The large population, known for its natural beauty and wealth, is one of the great potentials of Indonesia. Each population requires the number of nutritional needs that come from food as the main requirement. In addition, there is also a need for horticultural crops that are consumed as an additional companion to staple foods to meet the nutritional needs of the entire population of Indonesia. In general, meeting the needs of existing land is utilized by the Indonesian people to meet nutritional needs whose sources are plants with a variety of content and benefits for Indonesian horticulture [1].

In the context of meeting the nutritional needs of the population, West Nusa Tenggara has an important role in providing various types of food and horticulture that contribute to meeting the nutritional needs of the Indonesian population. The geographical location of West Nusa Tenggara in the Indonesian archipelago provides its own advantages in the development of the agricultural sector, especially in fruit production [2].

Fruit plantations in NTB generally include various types of tropical fruits such as Mangosteen Varietas, Banana Kepok and Rambutan Lebak Bulus. In addition, diverse geographical conditions provide opportunities for fruit farming with seasonal variations that allow harvesting throughout the year. Despite its great potential, the agricultural sector in the West Nusa Tenggara region, especially on Lombok Island, still faces a number of challenges such as limited water for irrigation, suboptimal land management, and market access and distribution issues. Efforts to develop and improve the quality of agriculture including fruit plantations need to continue to ensure that the nutritional needs of the population can be met in a sustainable manner [2].

Of the various types of fruit produced on Lombok Island, there are still challenges in terms of post-harvest handling and management. One of the main obstacles faced is the difficulty of classifying the ripeness level of the fruit, which in turn can affect the distribution, storage and sustainability of the fruit supply. In response to this challenge, the researchers designed a fruit type classification model which is the initial stage to help design a fruit ripeness classification system based on mobile applications, websites and even for IoT devices through digital image processing. This research builds a model to perform fruit type classification and does not directly address fruit ripeness classification because accurately identifying the type of fruit is a very important first step to assess fruit ripeness. Without knowing the specific fruit type, the fruit ripeness classification model may be inaccurate because ripeness indicators vary between fruit types. Therefore, it is essential to perform a preliminary classification of fruit types so that a robust and reliable fruit ripeness classification system can be built. This fruit type classification also serves as the basis for fruit ripeness classification by providing precise fruit identification, which is necessary for applying specific ripeness criteria. Once the fruit types are accurately classified, the system can then apply a model that performs customized fruit ripeness classification, which considers unique ripeness indicators such as color, texture, and other features specific to each fruit type.

Digital Image Processing is the study of techniques for transforming images, whether in the form of photographs or videos. The term "digital" refers to the use of computers to process images. RGB stands for Red-Green-Blue, the three basic colors that form the basis for all other colors. Using the RGB color model, colors can be converted into numeric codes, making them universally displayable. Computers have organized color information into a consistent model, making RGB color processing easy [3]. One of the various methods that can manage images is the *Convolutional Neural Network* (CNN) method.

from previous research that compares CNN method with HSV using the dataset available at Kaggle a total of 11,219 data divided into 17 classes and CNN method gets higher accuracy results of 96.87% compared to HSV which gets results of 93.09% However, this research has shortcomings, namely details on the varieties of fruit types used and limited color formats at the time of testing [4].

The use of CNN method in the classification research of three types of fruit with varying degrees of ripeness offers many advantages, including automatic feature extraction capabilities, handling of color and texture variations, and robustness to data variations. These advantages make CNN a highly effective choice for complex classification tasks that require high accuracy, such as fruit recognition from unripe to rotten [5].

CNN (Convolutional Neural Network) plays an important role in fruit type classification by processing images as input and using a series of layers such as SoftMax to produce an output capable of recognizing objects in the image. CNN architecture consists of neurons that have weights, biases, and activation functions. What distinguishes CNNs from ordinary neural networks is their ability to load information from multiple image scales, not just length and height. By utilizing information from the entire image scale, CNN can classify fruit types more accurately. Therefore, in this study researchers used the *Convolutional Neural Network* (CNN) method as one of the image processing methods that can classify fruit types to help improve accuracy in recognizing various types of fruit in digital images [6].

With this classification system, it is expected that producers, distributors, and consumers of fruit on Lombok Island can improve efficiency in post-harvest management, including in the process of storage, packaging, and distribution of fruits.

II. Literature Review and Basic Theory

A. Related Research

This research is designed based on several studies that have previously existed and used as references in conducting this research. The following is research that becomes a reference according to the object in the form of fruits and CNN methods.

The first research is research that analyzes two classification methods between SVM and CNN where the SVM method gets an accuracy of 93.09% and CNN method gets a greater accuracy of 96.87% [4].

The second research related to research conducted to find out how Deep Learning works by using CNN method in classifying the types of fruit. by using Deep Learning and by using CNN method get a fairly high average result of 90% [7].

The third study of this research aims to improve the performance of CNN architecture and uses fruit images obtained from Kaggle, namely the Fruit-360 dataset as a testing dataset which gets a result of 97.97% [8].

The fourth study which aims to make it easier for people to distinguish between dry and fleshy fruit during the Covid-19 pandemic using datasets from crawling on google search and CNN method as the method used to get an accuracy of 90% [9].

This fifth study aims to distinguish the type of fruit from the skin and size of the fruit using CNN architecture as the method used and get results using the SVM method has a sensitivity of 0.990 after fully connected modification in CNN architecture, followed by KNN with 0.981 and LDA with 0.976. For accuracy, SVM also excels with 96.3%, followed by CNN without fully connected modification with 94%. With these results, the SVM method was chosen because of better sensitivity, specificity, and accuracy. as for the hope of the study to add class types to the existing fruit dataset as well as perform various architectures on CNN method [5].

The sixth, research that compares 2 methods and 2 color formats, namely the CNN and KNN methods using the GMI and SVM color formats in classifying types of herbal leaves which are divided into 2 classes, namely Tap leaves and Moringa leaves totaling 480 images and getting the results that the HSV color format is superior to GMI in both methods tested by HSV-CNN = 0.98%, GMI-CNN = 0.96%, HSV-KNN = 0.94, and GMI-KNN = 0.56% [10].

From previous research, it can be concluded that deep learning using CNN method has a relatively high accuracy compared to several other methods. Many previous studies have used datasets from Kaggle, specifically the Fruit-360 dataset. However, no studies have focused on classifying fruit types found on Lombok Island, such as Mangosteen Variety, Banana Kepok, and Rambutan Lebak Bulus.

B. Supporting Theory

The following are general theories that are used as support in this research:

B.1. Classification

Classification is the process of evaluating data objects to place them in one of several available classes. It involves building a model based on existing training data and then using that model to classify new data. More specifically, classification can be described as training or learning to understand a target function that will map each set of attributes to one of several available class labels [11].

Classification involves two main stages, namely the learning stage and the classification stage. In the learning stage, the classification algorithm builds a classification model by analyzing training data. This stage can be considered as a function formation or mapping stage Y = F(X), where Y is the predicted class and X is the tuple, whose class is to be predicted. Then, in the classification stage, the model that has been created is used to perform classification. Classification is a process in which a series of models are found to describe and distinguish data classes, so that the model can be used to predict the class of an object whose class is unknown [12].

B.2. Deep Learning

Deep Learning is an approach in machine learning that uses artificial neural networks with multiple layers, also known as multi-layer neural networks. This model is inspired by the complex structure of the human brain, where neurons are interconnected to form a network capable of performing very powerful computations. The uniqueness of Deep Learning lies in its ability to perform complex non-linear transformations on data, allowing the model to understand complex and abstract patterns. This makes Deep Learning extremely useful in a wide range of tasks, from image and text recognition to natural language processing and speech recognition. By combining machine learning techniques with artificial intelligence, particularly through artificial neural networks, Deep Learning has become a very exciting field in the world of artificial intelligence. Various algorithms such as Convolutional Neural Networks (CNN), Restricted Boltzmann Machines (RBM), Deep Belief Networks (DBN), and Stacked Autoencoders are used in Deep Learning to overcome various challenges in data analysis and machine learning. This makes Deep Learning a very promising option for researchers and practitioners in various industries around the world [13].

B.3. Convolutional Neural Network

Convolutional Neural Network (CNN) is a type of neural network that is often used in image data analysis with the aim of detecting and recognizing objects in images. CNN is specifically designed to process data in the form of arrays, especially in color images consisting of three 2D arrays that represent pixel intensities in three color channels. CNNs use the natural properties of signals, such as local connections, weight sharing, pooling, and the use of multiple layers to perform analysis.

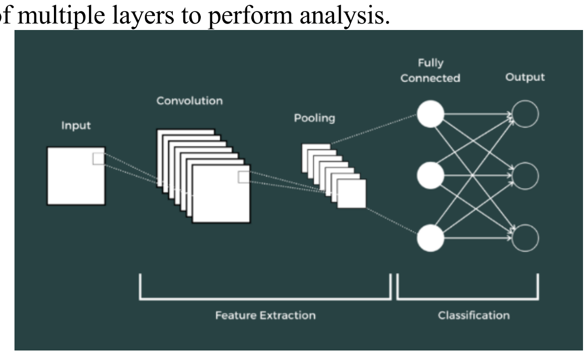


Fig. 1. CNN Architecture [14]

Based on Fig.1, the image processing process using CNN consists of four main models, namely Convolution Layer, Pooling Layer, Dropout Layer, and Fully Connected Layer. Each of these models is tasked with processing the image with various methods, such as filtering and data dimension reduction through taking the largest value from each grid. In addition, overfitting is prevented before transforming the dimensionality of the data for better classification. It is important to note that each of these models requires quality training data to obtain accurate classification results. Thus, CNN is one of the popular choices in analyzing image data for the purpose of object recognition [15]. Here are some layers in the convolutional neural network method:

B.3.1. Convolution Layer

The Convolutional Layer is the key element in a CNN that is directly connected to the input. This layer consists of a series of filters that are responsible for extracting features from the input. These filters are defined by their width, height, and number of channels. For example, the first layer is usually referred to as a 5x5x3 convolutional layer, which means it has a width of 5 pixels, a height of 5 pixels, and 3 channels corresponding to the color channels of the image. Each of these filters or channels is shifted across the image area, and at each shift, a dot operation is performed between the input value and the filter value to produce the output, which is often referred to as the activation map or feature map [16].

B.3.2. Pooling Layer

Pooling layer or sub-sampling is part of CNN architecture that is responsible for reducing the dimensionality of the feature map generated by the convolution layer. The function of this layer is to take a portion of the feature map and produce a single output depending on the type of pooling used. The main purpose of the pooling layer is to reduce the number of parameters calculated in the computational process [17].

The pooling layer in CNN uses the feature map or activation map as input to be processed with various types of statistical operations based on the nearest pixel value. The advantage of layer pooling is its ability to gradually reduce the dimension of the output volume on the feature map, which is beneficial for controlling overfitting. In general, there are two types of pooling used, namely average pooling and max pooling [18].

B.3.3. Dropout Layer

Dropout is a regulation technique in neural networks that prescribes that some neurons will be randomly selected and will not be active during the network model training process. The neurons selected for dropout will be randomly deactivated, so they will temporarily not contribute to the learning process of the model. This will result in the ignored neurons not receiving or transmitting signals, and new weights will not be applied to them. The use of dropouts aims to prevent overfitting of the neural network model. In addition, dropouts can also reduce the complexity or number of parameters used in the model, thus speeding up the training process. When enforcing dropout on the hidden layer, each neuron will be assigned a probability to be ignored, whose value can be predetermined and is in the range between 0 and 1 [19].

B.3.4. Fully Connected Layer

The fully connected layer is the component in a neural network that is responsible for resizing the data so that it can be classified linearly. To produce the output of this layer, no convolution operation is used, but rather a matrix multiplication process followed by the addition of a bias. With this approach, each neuron has a full connection to all activations from the previous layer, so this layer is often referred to as a fully connected layer [20].

The layer is commonly used in Multilayer Perceptron (MLP) and aims to resize the data so that it can be classified linearly. Each neuron in the convolutional layer needs to be converted into one-dimensional data before being fed into the fully connected layer. This causes a loss of spatial information and makes it irreversible. Therefore, the fully connected layer can only be placed at the end of the network. The convolution layer with a 1x1 kernel size has a similar function to the fully connected layer, but retains the spatial character of the data [14].

B.4. TensorFlow

TensorFlow, developed by the Google Brain team in 2015, is a framework for numerical computation. Today, it has become very popular and is widely used by major companies to develop artificial intelligence applications, including image classification, word embedding, and chatbot development. TensorFlow provides an interface that makes it possible to express machine learning algorithms and applications to execute those algorithms. The framework supports modeling various types of neural networks such as Recurrent Neural Network (RNN), Restricted Boltzmann Machine (RBM), Convolutional Neural Network (CNN), and Dynamic Bayesian Network (DBN), and supports parallel execution. The name "TensorFlow" comes from the concept of tensors used throughout the network, where a tensor is a type of multidimensional array [21]. TensorFlow has the following main features:

- Use of mathematical expressions for definition, optimization, and calculations involving multidimensional arrays (tensors).
- Programming support for deep neural networks and machine learning techniques.
- Efficient use of the GPU, as well as automation of management and optimization of the same memory against the data used. TensorFlow can write and run the same code on CPU or GPU, and is able to identify the parts that should be moved to GPU.
- Scalable computing capabilities across machines for large data sets.

B.5. Confusion Matrix

Confusion Matrix is an important evaluation tool in classification problems, which is very useful for measuring the values of Recall, Precision, Accuracy, F-Measure, and most importantly, provides information about the number of class predictions given, regardless of the correctness or error of the prediction [22]. There are four terms used to describe the results of the classification process, namely: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). The four terms can be represented in the following Fig.2.

Predicted Values

Actual Values	Predicted A	Predicted B	Predicted C
Actual A	Mean (TP_A1, TP_A2, TP_A3)	Mean (FP_BA1, FP_BA2, FP_BA3)	Mean (FP_CA1, FP_CA2, FP_CA3)
Actual B	Mean (FN_AB1, FN_AB2, FN_AB3)	Mean (TP_B1, TP_B2, TP_B3)	Mean (FP_CB1, FP_CB2, FP_CB3)
Actual C	Mean (FN_AC1, FN_AC2, FN_AC3)	Mean (FN_BC1, FN_BC2, FN_BC3)	Mean (TP_C1, TP_C2, TP_C3)

Fig. 2. Terms on the confusion matrix

In Fig. 2, above is an image of the Confusion Matrix which contains the predicted value and the actual value in tabular form. The following is a description of the figure above [23].

- True Positive (TP) is the correct predicted value and the actual value is correct.
- False Positive (FP) is the correct predicted value while the actual value is wrong. [Error Type 1]
- False Negative (FN) is an incorrect predicted value while the actual value is correct. [Error Type 2]
- True Negative (TN) is a false predicted value and a false actual value.

III. Research Methodology

A. Research Flow

The following is a flowchart that explains the flow of this research.

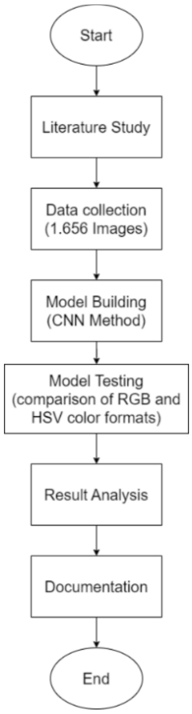


Fig. 3. System Flow

J-COSINE (Journal of Computer Science and Informatics Engineering)
Vol. 8, No. 2, December 2024 E-ISSN:2541-0806
Accredited Sinta-4 by RISTEKDIKTI Decree No. 79/E/KPT/2023 P-ISSN:2540-8895

In Fig. 3, the first stage in conducting this research is looking for a Literary Study which aims to study and recognize various methods and can compare the advantages and disadvantages of each method that has been done before. Then for the second stage, namely data collection where the data that will be used during the research, for the data needed is image data from 3 types of fruit that have been determined, namely Mangosteen Varietas, Banana Kepok and Rambutan Lebak Bulus, each of which amounts to 552 data in JPG format. For the third stage, namely modeling which at this stage researchers design and build models using CNN method in accordance with the literature that has been studied. The fourth stage of testing is carried out to see the performance of the model that has been built whether it is as expected or not. The fifth stage, namely analyzing the results, is a stage where after the testing stage is carried out and the results are analyzed for the performance of the model that has been made and the last is the documentation stage, namely taking documentation of the entire research from the start of the research to completion and collected in report format.

B. Model Building

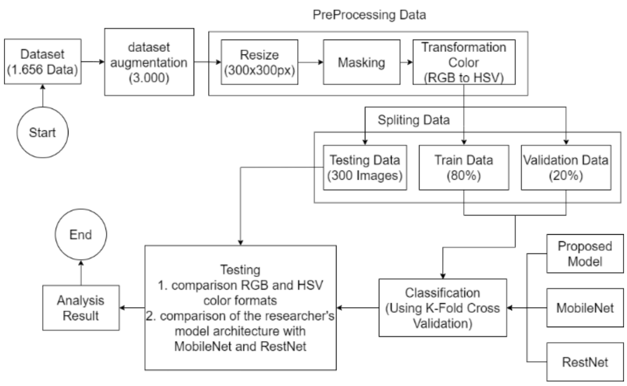


Fig. 4. CNN Modeling Flow

Fig. 4 is the flow of the model that will be made in this study which will be explained at the next points:

B.1. Dataset Collection

Capturing image datasets in research is important to provide the data needed to train, test, and evaluate the model. The model learns from the given image data in order to classify the images. Image datasets help researchers develop accurate and reliable models, and allow them to experiment and compare the results of different methods or models.

This research will use 1,656 datasets which are divided into 3 classes namely Mangosteen Variety, Banana Kepok and Rambutan Lebak Bulus and taken using a smartphone at a mini product studio with a white background from a distance of 30 cm with a zoom size of 3x. Then the camera settings used are with Aperture f/1.7 and ISO speed 100 as well as light brightness of 1667 lux in JPG format

B.2. Dataset Augmentation

At this stage the researcher performs dataset augmentation which is carried out to increase the number of datasets and also aims to help the model recognize each class with more varied data while the augmentation performed is "rescale", "horizontal flip", "rotation range" and "zoom range" with the initial data of 1,656 images to 3000 images.

B.3. Dataset Preprocessing

At this stage, the dataset will go through several processing processes that make the data easier to process and recognize by the model. The following are the preprocessing stages that will be carried out in this study:

B.3.1. Resize

Resizing the dataset is done for various reasons such as first, to align the image size to fit the input layer. Secondly, to reduce the computational load as reducing the image resolution can speed up processing and save memory and lastly to ensure uniform data as data that has different dimensions can cause errors during training.

Therefore, this research will change the image size to 300x300 pixels because it is considered to be the most ideal size to be the optimal resolution[9].

B.3.2. Masking

Image masking is used to highlight or hide certain parts of an image, allowing focus on relevant areas for further analysis. This is useful in various applications such as object segmentation, where the main goal is to separate objects from the background such as in datasets that use a white background.

B.3.3. Color Transformation RGB to HSV

The color space transformation from RGB to HSV is done to simplify color processing by separating color information (hue) from intensity and saturation, making color detection easier. In addition, HSV is more stable to lighting changes than RGB, making it better for image analysis. Hue in HSV provides a more intuitive and relevant color representation for many computer vision applications.

B.4. Classification

At this stage the researcher conducts training on the model using 3 types of models, namely those designed by the researcher, MobileNetV2 and RestNet50. at this training stage the researcher uses the Cross Validation Techniques, namely K-Fold Cross Validation which is used to reduce bias in the model that has been trained. for the K value used in this method is 3 because in research that has used K-Fold Cross Validation concluded that this value can already be a validation that the validation results are unbiased [25].

B.5. Testing Model

At this stage, testing will be carried out on the model that has been designed with the MobileNetV2 and ResNet50 architectures using the testing data that has been provided at the time of splitting the data. at this stage, checking the model that has been made whether it can recognize the appropriate fruit type class through the results in the form of accuracy and comparing the performance of the 2 color formats, namely RGB and HSV which will be analyzed at a later stage.

B.6. Analysis Result

The last stage is evaluation, at this stage the researcher evaluates the model that has been created and has been tested by taking the accuracy values of each color format which can later be processed using a confusion matrix because the confusion matrix can accumulate the results of the processed data to be more easily understood and with the results of the calculation of the confusion matrix the researcher can better recognize the shortcomings and advantages of the model that has been built and trained.

IV. Results and Discussion

A. Dataset Collection

This research uses a dataset taken by the researchers themselves using a cellphone camera in a mini product photo studio with a white background from a distance of 30 cm with a zoom size of 3x. Then the camera settings used are with Aperture f/1.7 and ISO speed 100 as well as light brightness of 1667 lux in JPG format and has various dimensions with a total of 1,656 images which are divided into 3 classes, namely Mangosteen Varieties, Banana Kepok and Rambutan Lebak Bulus with each class totaling 552 photos. The following is a visualization of the dataset that has been taken.

TABLE I. Visualization of Datasets from Each Class

Mangosteen Varietas	**Rambutan Lebak Bulus**	**Banana Kepok**
		

The dataset can be accessed via the following google drive link: (Dataset Link).

B. Dataset Augmentation

TABLE II. Dataset Augmentation

Parameter	Value
Rescale	1.0/255
rotation range	40
zoom range	0.2
horizontal flip	True

Based on Table II, researchers performed augmentation Rescale = 1.0/255, rotation range = 40, zoom range = 0.2 and horizontal flip = True, in order to increase the number of datasets from 1,656 images to 3,000 images and also help the model to train with varied datasets.

C. Dataset Preprocessing

The dataset that has been collected will be processed to make it easier to manage and recognize by the model.

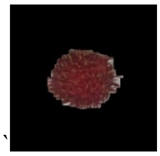
B.1. Resize

The first data processing is resized, which is to equalize the size or dimension to 300x300 pixels to ensure input consistency, reduce computational and memory load, speed up the training process, and ensure the model's capacity to handle images effectively. The smaller image size helps to reduce noise and unimportant details, so that the model focuses more on the main relevant features.

B.2. Masking

Then for the second data processing is masking which is done to remove the white background of the mini product studio on the dataset taken by converting the color to HSV so that it can easily detect white and gray colors to be removed and converted back to RGB format so that the dataset color returns to its original color but with a white and gray background that has been removed, which can be seen in Table III.

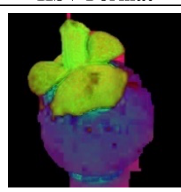
TABLE III. Dataset After and Before Masking

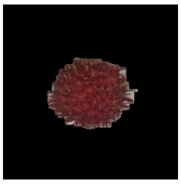
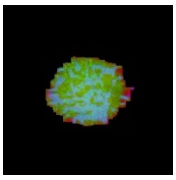
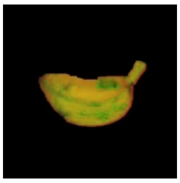
Class	Before Masking	After Masking
Mangosteen Varietas		
Rambutan Lebak Bulus		
Banana Kepok		

B.3. Color Transformation RGB to HSV

And the last data processing is Color Transformation RGB to HSV is done to change the color features in the dataset which initially used RGB format into HSV which aims to make it easier for the model to recognize the dataset because the HSV color format is sensitive to light and represents colors that are intuitive and relevant to many known computer vision applications.

TABLE IV. Dataset from RGB Format to HSV Format

Class	RGB Format	HSV Format
Mangosteen Varietas		

Rambutan Lebak Bulus		
Banana Kepok		

D. Model Building

Model building in this study uses the Convolutional Neural Network (CNN) method to perform image classification and will use the TensorFlow library for model initialization and configuration. The following is the configuration of the hyperparameters used in model building.

TABLE V. HYPERPARAMETER CONFIGURATION

Hyperparameter	**Value**
Optimizer	Adam (default, 0.001)
Batch Size	64
Epochs	50
Activation Function	SoftMax

Table V, this hyperparameter configuration was chosen to balance the efficiency, stability, and generalization ability of the model. The Adam optimizer with default learning rate (0.001) provides fast and stable convergence, batch size 64 maximizes computational efficiency and generalization ability, 50 epochs provides sufficient training time without overdoing it, and the SoftMax activation function in the output layer ensures interpretable probabilities for multiclass classification. This combination is the most ideal combination that researchers have achieved from various experiments [26].

D.1. Proposed Model Architecture

TABLE VI. PROPOSED MODEL ARCHITECTURE

Layer Type	**Description**	**Output Shape**
Input Layer	Input shape: (300, 300, 3)	(300, 300, 3)
Convolutional Layer 1	32 filters, kernel size: 3x3, activation: ReLU	(298, 298, 32)
Max Pooling Layer 1	Pool size: 2x2	(149, 149, 32)
Convolutional Layer 2	64 filters, kernel size: 3x3, activation: ReLU	(147, 147, 64)
Max Pooling Layer 2	Pool size: 2x2	(73, 73, 64)
Convolutional Layer 3	64 filters, kernel size: 3x3, activation: ReLU	(71, 71, 64)
Max Pooling Layer 3	Pool size: 2x2	(35, 35, 64)
Convolutional Layer 4	128 filters, kernel size: 3x3, activation: ReLU	(33, 33, 128)
Max Pooling Layer 4	Pool size: 2x2	(16, 16, 128)
Dropout Layer 1	Dropout rate: 0.5	(16, 16, 128)
Flatten Layer	Flatten input	(32768)
Dropout Layer 2	Dropout rate: 0.5	(32768)
Dense Layer 1	16 units, activation: ReLU	(16)
Dropout Layer 3	Dropout rate: 0.5	(16)
Dense Output Layer	3 units, activation: SoftMax	(3)

Table VI, is the model that has been designed for this research. The model starts with an input layer that receives a 300x300 pixel image with three RGB color channels. This is followed by four consecutive convolution layers, each followed by a max pooling layer that reduces the spatial dimension. After each convolution and pooling layer, there is a dropout layer to prevent overfitting. After that, a flatten layer is used to convert the data into a one-dimensional vector form, which is followed by an additional dropout layer and two dense (fully connected) layers, where the output layer uses a SoftMax activation function to classify the image into three classes. Each layer in the model is designed to progressively extract features from the input image and combine them to produce accurate class predictions.

D.2. MobileNetV2 Architecture

TABLE VII. MOBILENETV2 ARCHITECTURE

Layer Type	**Description**	**Output Shape**
Input	Input shape	(300, 300, 3)
Base Model	MobileNetV2 (pretrained on ImageNet, no top)	(10, 10, 1280)
Flatten	Flatten input	(12800)
Dropout	Dropout rate: 0.5	(12800)
Dense	16 units, activation: ReLU	(16)
Dropout	Dropout rate: 0.5	(16)
Dense	3 units, activation: SoftMax	(3)

Table VII, displays the architecture of the MobileNetV2 model used in this research and has been adapted to the designed model.

D.3. ResNet50 Architecture

TABLE VIII. RESNET50 ARCHITECTURE

Layer Type	**Description**	**Output Shape**
Input	Input shape	(300, 300, 3)
Base Model	ResNet50 (pretrained on ImageNet, no top)	(10, 10, 2048)
Flatten	Flatten input	(204800)
Dropout	Dropout rate: 0.5	(204800)
Dense	16 units, activation: ReLU	(16)
Dropout	Dropout rate: 0.5	(16)
Dense	3 units, activation: SoftMax	(3)

Table VIII, displays the architecture of the ResNet50 model used in this research and has been adapted to the designed model

E. Model Testing

At this stage, researchers use two test scenarios as a comparison, namely using datasets that are still in RGB color format and HSV format as test data to find out which color format is more suitable for the model that has been designed by researchers. then the second is to test the model that has been made with the MobileNetV2 and ResNet50 architecture models This time the researchers used 300 test data with 100 test data for each class and 3 folds for cross testing.

E.1. Testing by Color Format

for the first test, namely the comparison between the two RGB and HSV color formats from the model that has been designed in this research using 3 folds and using a total of 300 datasets.

E.1.1. RGB

The following are the results of testing using the RGB color format that has been carried out according to the predetermined scenario.

TABLE IX. MODEL TESTING USING RGB FORMAT

	Predicted A	Predicted B	Predicted C	Acc
Actual A	100	0	0	95.22%
Actual B	7.33	92.67	7	
Actual C	7	7.33	93	

Table IX, showing the average of the three folds shows the model's excellent performance in classifying the three types of fruit: Mangosteen, Banana, and Rambutan. Mangosteen has 100 True Positives (TP) with no False Positives (FP) and False Negatives (FN), meaning the model is always correct in identifying Mangosteen. For Banana, the average TP was 92.67, with 7.33 FN and 7 FPs, showing some errors but still with a high level of accuracy. Rambutan had an average of 93 TP, with 7 FN and 7.33 FP. for an average accuracy of 95.22%.

E.1.2. HSV

The following are the results of testing using the HSV color format that has been carried out according to the predetermined scenario.

TABLE X. MODEL TESTING USING HSV FORMAT

	Predicted A	Predicted B	Predicted C	Acc
Actual A	100	0	0	97.55%
Actual B	7	93	0.33	
Actual C	0.33	7	99.67	

Table X, which shows the average of the three folds, shows the model's excellent performance in classifying three types of fruit: Mangosteen, Banana, and Rambutan. Mangosteen has 100 True Positives (TP) with no False Positive (FP) and False Negative (FN), which means the model is always correct in identifying Mangosteen. For Banana, the average TP was 93, with 7 FN and 0.33 FP, indicating some errors but still with a high level of accuracy. Rambutan has an average TP of 99.67, with 0.33 FN and 7 FP. with an average accuracy of 97.55%.

E.2. Testing by Comparing Models

in this second test, we will test the models that have been created and the models used in the CNN method, namely MobileNetV2 and ResNet50 using a predetermined scenario for each model.

E.2.1. Proposed Model

For testing using the model under study, the results can be seen in the previous table, Table X, which gets an accuracy result of 97.55%.

E.2.2. MobileNetV2

TABLE XI. MODEL TESTING USING MOBILENETV2

	Predicted A	Predicted B	Predicted C	Acc
Actual A	99.33	1.33	1.67	96.66%
Actual B	1.67	97	4	
Actual C	6.33	3.67	93.67	

Table X, which shows the average of the three folds, shows the model's excellent performance in classifying three types of fruit: Mangosteen, Banana, and Rambutan. Mangosteen has an average of 99.33 True Positives (TP) with 1.33 False Positives (FP) and 1.67 False Negatives (FN). For Banana, the average TP was 97, with 1.67 FN and 4 FP, showing a few errors but still maintaining a high level of accuracy. Rambutan has an average of 93.67 TP, with 6.33 FN and 3.67 FP. with an average accuracy of 96.66%.

E.2.3. ResNet50

TABLE XII. MODEL TESTING USING RESNET50

	Predicted A	Predicted B	Predicted C	Acc
Actual A	97.33	2.67	0	97.55%
Actual B	2.67	97.33	2	
Actual C	5.67	1.67	94.67	

Table XII, which shows the average of the three folds, shows the model's strong performance in classifying three types of fruit: Mangosteen, Banana, and Rambutan. Mangosteen has an average of 97.33 True Positives (TP), 0 False Positives (FP), and 2.67 False Negatives (FN). For Banana, the average TP was 97.33, with 2.67 FNs and 2 FPs, which indicates a slight error but still maintains a high level of accuracy. Rambutan has an average TP of 98, with 2 FN and 5.67 FP. with an average accuracy of 97.55%.

F. Result Analysis

The first test, namely the comparison of the performance of the proposed model using 3 folds in RGB and HSV color formats shows a clear difference in performance where the model using the RGB color format gets an average accuracy of 95.22% and the HSV color format gets an average accuracy of 97.55% where the model with the RGB color format is more difficult to identify banana and rambutan compared to the HSV color format which can be seen in Table XI and Table X and it can be concluded in this study that the HSV color format is more suitable on the dataset used in this study.

For the second test, the comparison between the proposed model, MobileNetV2, and ResNet50 shows the performance variation in classifying Mangosteen, Banana, and Rambutan. The proposed model showed excellent accuracy, especially for Mangosteen with 100 perfect TP, and an average overall accuracy of 97.55%. MobileNetV2 showed strong performance, but with slightly more errors, achieving 99.33 TP for Mangosteen and an overall accuracy of 96.66%. ResNet50 also performed well, with a TP of 97.33 for Mangosteen and a comparable average accuracy of 97.55%. Although the proposed model and ResNet50 show similar accuracy levels, MobileNetV2 slightly lags behind, making the proposed model the most reliable for this task.

V. Conclusion and Suggestion

In the research that has been done, it can be concluded that the classification of local fruit types using the CNN method with a dataset of 3000 images which are divided into 3 classes namely Mangosteen, Pisang Kepok and Rambutan Lebak Bulus varieties get higher accuracy in HSV color format with 97.55% accuracy compared to RGB color format with 97.22% accuracy and for the designed model get high accuracy of 97.55%. 55% compared to the RGB color format with an accuracy of 95.22% and for the designed model to get a high accuracy of 97.55% which is comparable to the accuracy of the ResNet50 model which also gets 97.55% compared to the MobileNetV2 model which gets 96.66% accuracy. With this it can be concluded that the initial model of fruit type classification for fruit maturity classification has been designed by researchers can be used in mobile-based applications, websites and even for IoT devices that will be built to sort fruits.

It is recommended for further research to increase the number of classes for various types of fruit on the island of Lombok so that they can be classified, try several other color formats as a comparison in order to get the most accurate color format for classification on the type of fruit used as a dataset and develop the architecture that has been designed to be more accurate in testing.

Acknowledgment

We as the authors would like to thank the faculty of food engineering for providing facilities for this research activity and the fruit farmers who have provided materials for research activities.

References

[1] R. M. Hasibuan and S. I. Sinambela, "Kerjasama Pertanian Indonesia Dengan China Dalam Kerangka ACFTA," vol. 2, no. 1, pp. 1–16, 2024.

[2] J. Purwohandoyo, B. T. Lubis, Y. J. Pambudi, and M. S. Wicaksono, "Analisis Sumberdaya Wilayah Untuk Pengembangan Industri Di Provinsi Nusa Tenggara Barat," *J. Pembang. Wil. Kota*, vol. 15, no. 4, p. 275, 2019, doi: 10.14710/pwk.v15i4.21732.

[3] S. Ratna, "Pengolahan Citra Digital Dan Histogram Dengan Phyton Dan Text Editor Phycharm," *Technol. J. Ilm.*, vol. 11, no. 3, p. 181, 2020, doi: 10.31602/tji.v11i3.3294.

[4] B. W. Kurniadi, H. Prasetyo, G. L. Ahmad, B. Aditya Wibisono, and D. Sandya Prasvita, "Analisis Perbandingan Algoritma SVM dan CNN untuk Klasifikasi Buah," *Semin. Nas. Mhs. Ilmu Komput. dan Apl. Jakarta-Indonesia*, no. September, pp. 1–11, 2021.

[5] I. A. Sabilla, "Arsitektur Convolutional Neural Network (Cnn) Untuk Klasifikasi Jenis Dan Kesegaran Buah Pada Neraca Buah," *Tesis*, no. 201510370311144, pp. 1–119, 2020.

[6] Rima Dias Ramadhani, A. Nur Aziz Thohari, C. Kartiko, A. Junaidi, T. Ginanjar Laksana, and N. Alim Setya Nugraha, "Optimasi Akurasi Metode Convolutional Neural Network untuk Identifikasi Jenis Sampah," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 2, pp. 312–318, 2021, doi: 10.29207/resti.v5i2.2754.

[7] I. R. W. Putra, "Deteksi Jenis Buah-Buahan Menggunakan Deep," *Fak. Teknol. dan Inform. Univ. Din.*, p. 60, 2020.

[8] F. F. Maulana and N. Rochmawati, "Klasifikasi Citra Buah Menggunakan Convolutional Neural Network," *J. Informatics Comput. Sci.*, vol. 1, no. 02, pp. 104–108, 2020, doi: 10.26740/jinacs.v1n02.p104-108.

[9] Y. Jumaryadi, A. Muhammad Ihsan, and B. Priambodo, "KLIK: Kajian Ilmiah Informatika dan Komputer Klasifikasi Jenis Buah-Buahan Menggunakan Citra Digital Dengan Metode Convolutional Neural Networks," *Media Online*, vol. 4, no. 3, pp. 1737–1746, 2023, doi: 10.30865/klik.v4i3.1421.

[10] Alfitriana Riska, Purnawansyah, H. Darwis, and W. Astuti, "Studi Perbandingan Kombinasi GMI, HSV, KNN, dan CNN pada Klasifikasi Daun Herbal," *Indones. J. Comput. Sci.*, vol. 12, no. 3, pp. 1201–1215, 2023, doi: 10.33022/ijcs.v12i3.3210.

[11] D. P. Utomo and M. Mesran, "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung," *J. Media Inform. Budidarma*, vol. 4, no. 2, p. 437, 2020, doi: 10.30865/mib.v4i2.2080.

[12] Heliyanti Susana, "Penerapan Model Klasifikasi Metode Naive Bayes Terhadap Penggunaan Akses Internet," *J. Ris. Sist. Inf. dan Teknol. Inf.*, vol. 4, no. 1, pp. 1–8, 2022, doi: 10.52005/jursistekni.v4i1.96.

[13] P. A. Nugroho, I. Fenriana, and R. Arijanto, "Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) Pada Ekspresi Manusia," *Algor*, vol. 2, no. 1, pp. 12–21, 2020.

[14] F. N. Cahya, N. Hardi, D. Riana, and S. Hadiyanti, "Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network (CNN)," *Sistemasi*, vol. 10, no. 3, p. 618, 2021, doi: 10.32520/stmsi.v10i3.1248.

[15] F. Ramadhani, A. Satria, and S. Salamah, "Implementasi Algoritma Convolutional Neural Network dalam Mengidentifikasi Dini Penyakit pada Mata Katarak," *sudo J. Tek. Inform.*, vol. 2, no. 4, pp. 167–175, 2023, doi: 10.56211/sudo.v2i4.408.

[16] E. Y. Prastowo, "Pengenalan Jenis Kayu Berdasarkan Citra Makroskopik Menggunakan Metode Convolutional Neural Network," *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 2, pp. 489–497, 2021, doi: 10.28932/jutisi.v7i2.3706.

[17] A. B. Handoko, I. K. Timotius, and D. Utomo, "Klasifikasi Citra X-Ray Covid-19 Menggunakan Three-layered CNN Model," *Techné J. Ilm. Elektrotek.*, vol. 21, no. 2, pp. 155–168, 2022, doi: 10.31358/techne.v21i2.316.

[18] Radikto, D. I. Mulyana, M. A. Rofik, and Mo. Z. Z. Zakaria, "Klasifikasi Kendaraan pada Jalan Raya menggunakan Algoritma Convolutional Neural Network (CNN)," *J. Pendidik. Tambusai*, vol. 6, no. 1, pp. 1668–1679, 2022.

[19] T. Dwi Antoko, M. Azhar Ridani, and A. Eko Minarno, "Klasifikasi Buah Zaitun Menggunakan Convolution Neural Network," *Komputika J. Sist. Komput.*, vol. 10, no. 2, pp. 119–126, 2021, doi: 10.34010/komputika.v10i2.4475.

[20] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," *J. Algoritm.*, vol. 1, no. 1, pp. 45–56, 2020, doi: 10.35957/algoritme.v1i1.434.

[21] N. Wiranda, H. S. Purba, and R. A. Sukmawati, "Survei Penggunaan Tensorflow pada Machine Learning untuk Identifikasi Ikan Kawasan Lahan Basah," *IJEIS (Indonesian J. Electron. Instrum. Syst.*, vol. 10, no. 2, p. 179, 2020, doi: 10.22146/ijeis.58315.

[22] M. Z. Ersyad, K. N. Ramadhani, and A. Arifianto, "Pengenalan bentuk tangan dengan convolutional neural network (CNN)," *eProceedings Eng.*, vol. 7, no. 2, pp. 8212–8222, 2020.

[23] S. Frangky Handono, F. Tri Anggraeny, and B. Rahmat, "Implementasi Convolutional Neural Network (CNN) Untuk Deteksi Retinopati Diabetik," *J. Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 669–678, 2020.

[24] B. P. Pratiwi, "Pengukuran Kinerja Sistem Kualitas Udara," *J. Inform. UPGRIS*, vol. 6, no. 2, pp. 66–75, 2020.

[25] C. Y. Sari, "Implementasi Metode K-Nearest Neighbor Dalam Memprediksi Bau, Rasa Dan Warna Air Sumur Di Kelurahan Selat Baru Bengkalis," 2019, [Online]. Available: https://repository.uir.ac.id/8862/%0Ahttps://repository.uir.ac.id/8862/1/153510266.pdf

[26] Sari, Citra Yunela. Implementasi Metode K-Nearest Neighbor Dalam Memprediksi Bau, Rasa Dan Warna Air Sumur Di Kelurahan Selat Baru Bengkalis. Diss. Universitas Islam Riau, 2019.