Comparison of CNN Transfer Learning in Detecting Superior Local Fruit Types in Bali

By Nyoman Purnama

WORD COUNT	4362	TIME SUBMITTED	22-JUL-2024 09:00AM
		PAPER ID	110587856

Comparison of CNN Transfer Learning in Detecting Superior Local Fruit Types in Bali

Nyoman Purnama[1]
[1]Information System Department, Faculty of Information Technology and Design, Primakara University, Bali, Indonesia

Corresponding author: Nyoman Purnama (e-mail: purnama@primakara.ac.id).

ABSTRACT Bali Province is an island that has unique geographical conditions, as well as the diversity of fruit it has. The specialty of local fruit is not only of economic value for food needs but also for religious ceremonial needs. Bali provincial government is currently actively promoting local fruit so that it can be used as consumption for Bali's increasingly rapid tourism. Several superior fruits were developed as an effort to raise the potential of local fruit in the tourism sector. Some of the superior fruits are Balinese snake fruit and sapodilla. However, snake fruit is one of the superior local fruits in Bali which has not experienced degradation over time. This research aims to detect the types of snake fruit in Indonesia. This fruit is not popular compared to imported fruit. Therefore, an application is needed to recognize this type of snake fruit automatically. This research uses a deep learning method with the CNN (Convolutional Neural Network) algorithm. This algorithm is able to recognize and classify an image well. The fruit images used were 400 fruits for 4 types of snake fruit. Where the training data for snake fruit is special because it has different skin and fruit contents. In this research, 2 transfer learning models from the CNN algorithm were also compared, namely mobilenetv2 and ResNet152. Based on the test results, it was found that the best level of accuracy was obtained using the ResNet152 model with an accuracy value of 92% in identifying images of Balinese snake fruit.

KEYWORDS CNN, Local Bali Fruits, ResNet152, VGG16

I. INTRODUCTION

Indonesia is known for its diverse range of fruit plants, facilitated by its archipelagic geography where each region boasts unique fruit varieties. The diversity of native Indonesian fruits plays a crucial role not only in meeting nutritional needs due to their high vitamin content, beneficial for health [1], but also holds religious significance. In religious ceremonies, fruits are often used as offerings alongside leaves and flowers. This practice is prominently observed in Hindu religious rituals, where various fruits are utilized, each carrying profound philosophical meanings [2].

Bali, as one of Indonesia's provinces with distinctive geographical features, indirectly fosters unique diversity in its fruit varieties. The uniqueness of fruits in Bali extends beyond their nutritional benefits to their role as ceremonial complements in religious events. However, local Balinese fruit faces marketing challenges due to the presence of fruits from outside the region and imported fruits. To address this, the Bali provincial government promptly issued Regional Regulation No. 99 of 2018, mandating all tourism components to use local Balinese fruits [3]. Some local fruits from Bali have even been designated as superior varieties that can compete with fruits from other regions. These superior local fruits include Siam Orange, mangosteen, bananas, and Balinese Snake Fruit (Salak Bali). Based on research by Made Tamba, snake fruit and mangosteen will remain superior fruits and will not degrade into non-superior fruits in Bali. Snake fruit reflects the diversity of flavors and textures found across various regions in Indonesia, making it a popular and highly regarded fruit nationwide. Specific areas are known for distinctive varieties of snake fruit, such as Pondoh Snake Fruit (Salak Pondoh) from Yogyakarta, Balinese Snake Fruit from Bali, Condet Snake Fruit (Salak Condet) from Jakarta, and Sidempuan Snake Fruit (Salak Sidempuan) from Magelang.

The similarity in texture and shape among different varieties of snake fruit poses a significant challenge in identifying the specific type originating from Bali compared to other regions. In this study, researchers employed a classification method. Classifying fruit types is a task that requires time and expertise [4]. The advancement of computer vision allows for efficient and accurate automation of fruit-type classification. One effective method for automated classification is through deep learning, a rapidly evolving field within machine learning [5]. A prominent deep learning method capable of processing image information is the Convolutional Neural Network.

Convolutional Neural Network (CNN) is an extension of the Multilayer Perceptron (MLP) designed specifically for processing two-dimensional data. Unlike MLP, where each neuron is one-dimensional, CNN represents neurons in a two-dimensional form [6]. CNN is a deep learning architecture tailored for structured data arrays. Widely employed in computer vision, CNNs have become pivotal in various visual applications such as image classification and have demonstrated effectiveness in natural language processing for text classification as well. CNN is renowned for its automatic feature extraction capabilities. In contrast to traditional machine learning methods that rely on manual feature extraction, CNNs automatically extract features in layers like convolutional, pooling, and Rectified Linear Unit (ReLU) activation. Following feature extraction, classification tasks are performed in the Fully Connected Layer (FCL) with softmax activation [6].

Transfer learning is a technique in neural networks where a model trained on one task is utilized to solve a different task. It serves as a foundational concept behind many popular machine learning applications such as speech recognition, object detection, and natural language processing [7]. Several examples of transfer learning in CNN algorithms include VGG16, AlexNet, MobileNet, and ResNet. MobileNet is a Convolutional Neural Network architecture specifically designed to address excessive computing resource requirements. As implied by its name, Mobile, researchers at Google developed this CNN architecture for mobile devices [8]. MobileNet released its second version in April 2017. Similar to MobileNetV1, MobileNetV2 still utilizes depthwise and pointwise convolutions but introduces two new features: 1) linear bottleneck, and 2) shortcut connections between bottlenecks. On the other hand, ResNet, or Residual Network, is another artificial neural network architecture that introduces shortcut connections across layers and applies activation functions to the preceding layers [9]. There are several variants of ResNet, with ResNet152 being one of them. ResNet152 comprises 152 layers in its network architecture. Due to its complexity, this model achieved success in the ILSVRC competition in 2015 for its minimal error rate [10].

Research on fruit classification using the YOLOv3-based CNN method has been conducted by Mr. Wibi Bagas et al. [11]. In their study, they classified 10 types of fruits using 2333 images. The training process involved 5000 iterations and achieved an accuracy of 90% in the first test of each fruit and 70% for the second test on out-of-test data images. Fruit classification using the fruit-360 dataset has been conducted by Febian Fitra Maulana. They utilized 15 out of 111 classes available in the dataset, achieving an accuracy of 91.42%. From these studies, it is evident that CNN is an effective method for image classification. Fruit classification research has also been undertaken by Myongkyoon Yang [12], titled "Fruit Classification using Convolutional Neural Network." They classified 7 categories of fruits with a dataset of 1000 images. Based on their findings, CNN demonstrates strong classification capability with an error rate of 10%. Research on the accuracy of CNN algorithms using various architectures has been conducted by Wahyudi Setiawan in the paper titled “Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus” [13].

Research on the performance of ResNet152 and AlexNet in classifying types of skin cancer has been conducted by Tommy Saputra [10]. In their study, an accuracy of 87.85% in skin cancer classification was achieved using ResNet152. Another study on classification using the RESNET model was conducted by Vijay Gadre, titled "Waste Classification using ResNet152" [9]. In their research, waste was classified based on various characteristics using ResNet152. The results of the study indicate that the success of waste classification using ResNet152 depends on the quality and diversity of the training data. Research on classification using MobileNetV2 for butterfly image classification has been conducted by Desi Ramayanti. The research dataset consisted of 4955 images labeled with 50 butterfly species, each sized 224 x 224 x 3. The best accuracy achieved by MobileNetV2 without fine-tuning reached 96% [14].

Based on the background and previous research, this study develops a CNN architecture using fruit images as test data, focusing on a prominent local fruit from Bali province, Balinese Snake Fruit. Four types of snake fruit will be classified based on their origin: Balinese Snake Fruit, Pondoh Snake Fruit, Condet Snake Fruit, and Sidempuan Snake Fruit. Transfer learning CNN models ResNet152 and MobileNetV2 will be compared. The diverse textures and shapes of various snake fruit in Indonesia pose a challenge in distinguishing their origins. Detection of the prominent local fruit, Balinese Snake Fruit, will be conducted using CNN classification methods. By comparing transfer learning CNN models ResNet152 and MobileNetV2, the study aims to effectively classify these local Bali fruits and achieve the highest accuracy possible.

II. RESEARCH METHODS

In this study, an experimental research method was employed with the following stages:

A. DATA COLLECTION

The data utilized in this research consists of digital images. These images were collected from various sources and collectively referred to as the Dataset. The Dataset was obtained from search engine datasets such as Google and Bing. Selected images were chosen based on adequate lighting conditions, backgrounds with minimal noise, and intact or peeled snake fruit images. The collected Dataset includes images of four types of snake fruit: Balinese Snake Fruit, Pondoh Snake Fruit, Condet Snake Fruit, and Sidempuan Snake Fruit. Before using this data, preprocessing was conducted by categorizing each image into respective folders according to its category. The collected image Dataset remains in RGB color format with JPG extension.

Before using this image Dataset in the classification process using transfer learning CNN models, the data underwent preprocessing. Pixel size standardization was applied to all images, followed by data normalization. To increase the Dataset size, augmentation of these snake fruit images was performed initially. The subsequent process involved converting pixel values of the images into array form, aiming to standardize input sizes for the CNN network.

B. METHOD USED

The method employed in this study for fruit classification is one of the Deep Learning methods. Deep Learning can address problems with large amounts of data [15]. By utilizing Deep Learning, it allows us to create systems capable of learning at desired speeds and accuracies. One example of a Deep Learning method used in this research is Convolutional Neural Network.

This algorithm is efficient in image processing and widely used in image recognition [16]. CNN is not fundamentally different from other neural networks like artificial neural networks, as they all have biases, weights, and activation functions. However, what sets CNN apart from other neural networks is its specialized layer called the Convolutional Layer [17]. Image processing for leaf classification is performed using kernel filters. These filters are used to obtain fragments (strides) from an image. The process of obtaining these fragments/strides is called convolution. The process of this convolution is depicted in Figure 1 below.

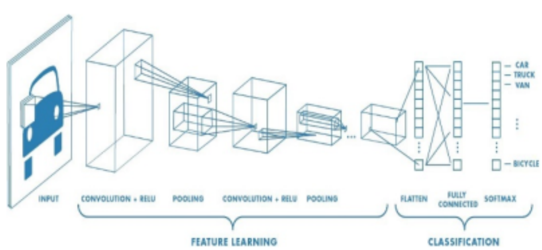


FIGURE 1. Process in Convolutional Neural Network.

MobilenetV2 and ResNet152 are transfer learning CNN architectures used in image classification applications [18]. Each architecture has its own strengths and weaknesses. Based on previous research, MobilenetV2 is a transfer learning model suitable for devices with limited resources. On the other hand, ResNet152 achieves high accuracy due to its deep convolutional layers [10]. In this study, each architecture will be used to train the same dataset. There are 500 fruit image data points, which will be split into training, validation, and test sets with a ratio of 80:10:10.

C. EXPERIMENT, EVALUATION, AND VALIDATION OF RESULTS

In this study, the Tensorflow framework developed by Google is utilized for developing fruit image classification using CNN with transfer learning models ResNet152 and MobilenetV2. Tensorflow offers numerous features related to image classification, utilizing Keras as a high-level interface for machine learning development. The classification results will be evaluated using precision, recall, and F1 Score metrics.

Comparison is applied to accuracy, precision, recall, and F1-score values in prediction results during testing, calculated using a confusion matrix. True Positive (TP) indicates instances where actual and predicted values are both positive, while True Negative (TN) indicates instances where both are negative. False Positive (FP) shows instances where actual values are negative but predicted as positive, and False Negative (FN) indicates instances where actual values are positive but predicted as negative [19].

Precision measures the accuracy of the system in providing requested information compared to the system's responses. Recall, on the other hand, measures the system's success in retrieving information [20]. Broadly, the process of detecting prominent local Bali fruit types is depicted in Figure 2.

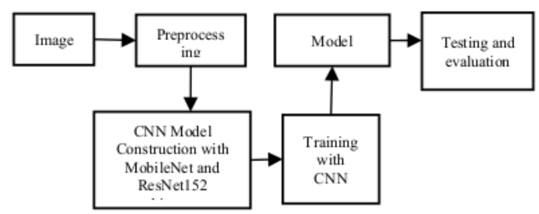


FIGURE 2. Research diagram

III. RESULT AND DISCUSSION

A. DATA PREPARATION

This study utilizes 4 images of local snake fruit: Balinese Snake Fruit, Pondoh Snake Fruit, Condet Snake Fruit, and Sidempuan Snake Fruit. These image data are collectively referred to as the dataset. Classification processes are conducted using transfer learning convolutional neural networks, namely ResNet152 and MobileNetV2. Evaluation of test results employs precision, F2-score, and recall metrics. In the initial stage, data collection involved gathering 250 images from search engines for these 4 types of snake fruit. These image data underwent augmentation to increase the training set to 400 images. Augmentation methods included adjustments for brightness, rotation, and vertical flips.

The composition of training, validation, and test data is set at 80:10:10 ratio. Thus, each fruit category comprises 80 training, 10 validation, and 10 test data points. Each type of snake fruit has an equal number of 100 training images. After collection, the data was categorized accordingly, with both training and test data placed into folders named after each subset. Examples of these images for each type of snake fruit used in this study are shown in Figure 3.

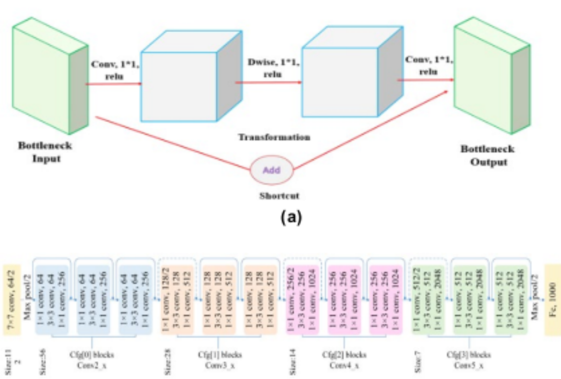
FIGURE 3. Initial images of Balinese Snake Fruit, Pondoh Snake Fruit, Condet Snake Fruit, and Sidempuan Snake Fruit

The next step involves image processing. In this stage, Python programming language is utilized within the Google Colab IDE environment to aid in processing. The image processing library used is OpenCV, a free library designed for image processing in Python. Since OpenCV uses the BGR (Blue, Green, Red) mode, the images collected in each folder are initially converted to the RGB (Red, Green, Blue) mode using the `cv2.cvtColor` function. The resulting images are then resized to 224x224 pixels. This step is necessary due to the varying sizes of the collected images, and resizing to 224x224 pixels accelerates the training process. Similarly, images for testing undergo the same image processing as those for training. Each test set consists of 20 images per fruit category. Before input into the CNN network, pixel values of both training and test images are normalized.

B. CONSTRUCTION OF RESNET152 AND MOBILENETV2 TRANSFER LEARNING MODELS

The next process involves constructing transfer learning models using CNN architectures, specifically MobileNetV2 and ResNet152. MobileNetV2 offers the advantage of being deployable on devices with low resources, especially mobile devices. It utilizes convolutional layers with filter thickness tailored to the input image thickness. On the other hand, the ResNet152 architecture is based on the concept of residual learning, comprising convolutional, normalization, and ReLU activation layers followed by residual blocks. In addition to transfer learning, this study employs the Sequential model in the model creation process. The Sequential model is a type of deep learning architecture that enables sequential layer-by-layer model building. This approach is commonly used for constructing deep learning models with Keras, particularly in TensorFlow.

Figure 4 illustrates the comparison of the MobileNetV2 and ResNet152 models utilized in this research. The models were constructed using the TensorFlow and Keras libraries in Python, with programming conducted in the Google Colab IDE environment.



(a)

(b)

FIGURE 4. Comparison of MobileNetV2 (a) and ResNet152 (b) architecture models.

Regarding the parameters used in this model creation process, the learning rate was set to 0.001, with 100 epochs and a batch size of 32. The activation function employed was ReLU, optimized using Adam, and the loss function used was categorical_crossentropy, suitable for datasets with more than one label. The training was conducted in two phases: initially, by freezing or maintaining the pre-trained layers that had previously learned general features from classification tasks. Only the last few layers designated for fruit class classification were trained. This approach leverages the knowledge already captured in the lower layers of the transfer learning pre-trained model.

Model: "mobilenetv2_1.00_224"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	[]
Conv1 (Conv2D)	(None, 112, 112, 32)	864	['input_1[0][0]']
bn_Conv1 (BatchNormalization)	(None, 112, 112, 32)	128	['Conv1[0][0]']
Conv1_relu (ReLU)	(None, 112, 112, 32)	0	['bn_Conv1[0][0]']
expanded_conv_depthwise (DepthwiseConv2D)	(None, 112, 112, 32)	288	['Conv1_relu[0][0]']
expanded_conv_depthwise_BN (BatchNormalization)	(None, 112, 112, 32)	128	['expanded_conv_depthwise[0][0]']
expanded_conv_depthwise_relu (ReLU)	(None, 112, 112, 32)	0	['expanded_conv_depthwise_BN[0][0]']
expanded_conv_project (Conv2D)	(None, 112, 112, 16)	512	['expanded_conv_depthwise_relu[0][0]']
expanded_conv_project_BN (BatchNormalization)	(None, 112, 112, 16)	64	['expanded_conv_project[0][0]']
block_1_expand (Conv2D)	(None, 112, 112, 96)	1536	['expanded_conv_project_BN[0][0]']
block_1_expand_BN (BatchNormalization)	(None, 112, 112, 96)	384	['block_1_expand[0][0]']
block_1_expand_relu (ReLU)	(None, 112, 112, 96)	0	['block_1_expand_BN[0][0]']
block_1_pad (ZeroPadding2D)	(None, 113, 113, 96)	0	['block_1_expand_relu[0][0]']
block_1_depthwise (DepthwiseConv2D)	(None, 56, 56, 96)	864	['block_1_pad[0][0]']
block_1_depthwise_BN (BatchNormalization)	(None, 56, 56, 96)	[illegible]	['block_1_depthwise[0][0]']

(a)

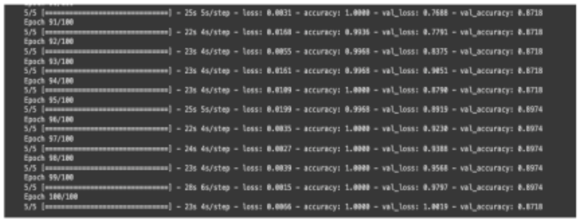
Model: "resnet152v2"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	[]
conv1_pad (ZeroPadding2D)	(None, 230, 230, 3)	0	['input_1[0][0]']
conv1_conv (Conv2D)	(None, 112, 112, 64)	9472	['conv1_pad[0][0]']
pool1_pad (ZeroPadding2D)	(None, 114, 114, 64)	0	['conv1_conv[0][0]']
pool1_pool (MaxPooling2D)	(None, 56, 56, 64)	0	['pool1_pad[0][0]']
conv2_block1_preact_bn (BatchNormalization)	(None, 56, 56, 64)	256	['pool1_pool[0][0]']
conv2_block1_preact_relu (Activation)	(None, 56, 56, 64)	0	['conv2_block1_preact_bn[0][0]']
conv2_block1_1_conv (Conv2D)	(None, 56, 56, 64)	4096	['conv2_block1_preact_relu[0][0]']
conv2_block1_1_bn (BatchNormalization)	(None, 56, 56, 64)	256	['conv2_block1_1_conv[0][0]']
conv2_block1_1_relu (Activation)	(None, 56, 56, 64)	0	['conv2_block1_1_bn[0][0]']
conv2_block1_2_pad (ZeroPadding2D)	(None, 58, 58, 64)	0	['conv2_block1_1_relu[0][0]']
conv2_block1_2_conv (Conv2D)	(None, 56, 56, 64)	36864	['conv2_block1_2_pad[0][0]']
conv2_block1_2_bn (BatchNormalization)	(None, 56, 56, 64)	256	['conv2_block1_2_conv[0][0]']
conv2_block1_2_relu (Activation)	(None, 56, 56, 64)	0	['conv2_block1_2_bn[0][0]']
conv2_block1_0_conv (Conv2D)	(None, 56, 56, 256)	16640	['conv2_block1_preact_relu[0][0]']

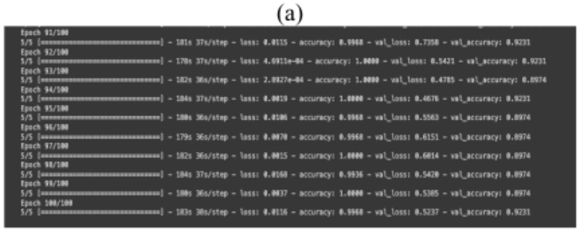
(b)

FIGURE 5. Summary of MobileNetV2 (a) and ResNet152 (b) models.

To mitigate overfitting, the training process implemented early stopping, a technique used to halt training early if signs of overfitting or a lack of performance improvement on validation data occured. The objective of early stopping is to prevent the model from excessively memorizing the training data. Figure 5 presents a summary of the model results generated by the Keras library in Python, used in this fruit classification process.



(a)

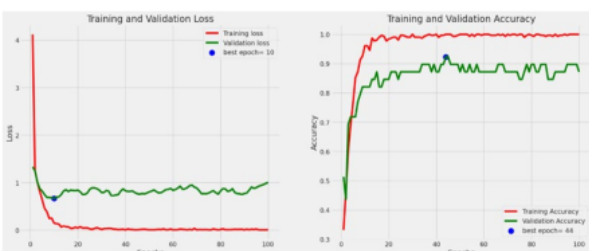


(b)

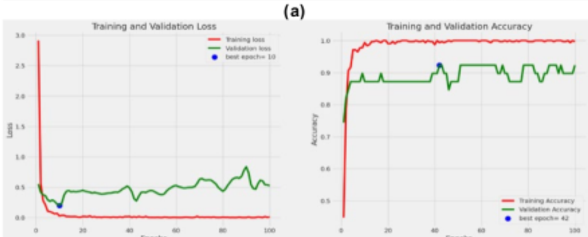
FIGURE 6. Comparison of training results for MobileNetV2 (a) and ResNet152 (b) models.

C. TESTING RESULTS AND EVALUATION

Following the creation of two models using the same parameters, the networks were trained for 100 epochs. The initial training process utilized the MobileNetV2 architecture. As shown in Figure 6 Part B, the training results using the MobileNetV2 model yielded commendable accuracy, aligning with its intended use on resource-constrained devices. In part c of Figure 6, the training process in the final epoch with the ResNet152 model is depicted. The performance results of ResNet152 demonstrated superior accuracy compared to MobileNetV2.



(a)



(b)

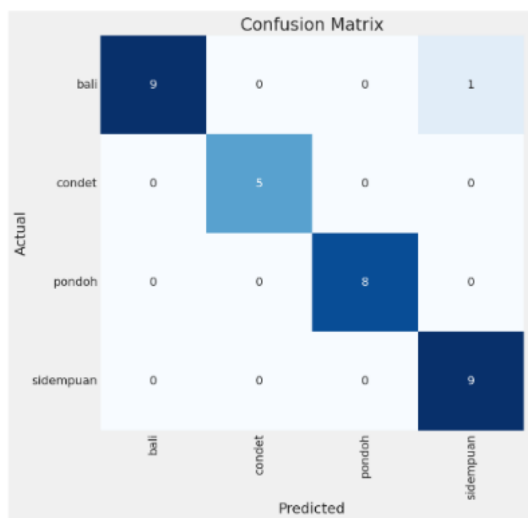
FIGURE 7. Comparison of training and validation loss data for MobileNetV2 (b) and ResNet152 (c) models.

In Figure 6, the performance results of each model after the training process are shown. In the figure 6 a is the result of training for mobilenetv2 and figure 6 b is for resnet152. To evaluate the models on the training dataset, the evaluate() function available in the Keras library is used. This function takes the same input and output as used to train the model. It generates predictions for each input-output pair and collects scores, including the average loss and any configured metrics such as accuracy. The evaluate() function returns a list with two values: the first value is the model's loss on the dataset, and the second value is the model's accuracy on the dataset. From the figure, it is evident that the best epoch for achieving the highest accuracy was obtained in the ResNet152 transfer learning model.

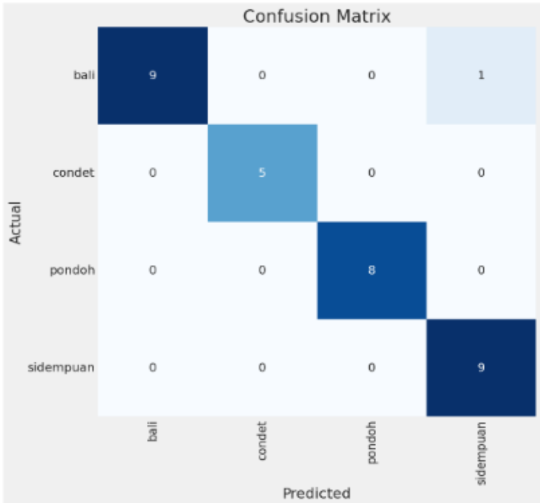
TABLE I
COMPARISON OF MODEL PERFORMANCE BETWEEN RESNET152 AND MOBILENETV2.

	Resnet152	MobilenetV2
Accuraccy	94.1%	84.72%
F1-Score	90%	87%
Recall	91%	89%
Precission	92%	92%

In Figure 7, the graph illustrates the accuracy and loss results for both training and validation data during the model creation process. Accuracy represents the ratio of correct predictions (both positive and negative) to the total number of instances for each class. Meanwhile, the loss function indicates how well the model's predictions match the actual results. In model creation, the goal is to minimize the loss value. The lowest accuracy value was observed in the MobileNetV2 model. The predictions are then evaluated to determine accuracy, recall, precision, and F1-score. Below, Table I compares the training accuracy performance results of the three models tested in this study.



(a)



(b)

FIGURE 8. Comparison of confusion matrices for MobileNetV2 (a) and ResNet152 (b) models.

In Figure 8, the confusion matrix for the MobileNet and ResNet152 models is depicted. Based on the confusion matrix, MobileNetV2 correctly classified 33 images, while ResNet152 correctly classified 46 images. The confusion matrix indicates that ResNet152 outperformed the other model in image classification. This outcome also reflects the longer training time required for ResNet152 compared to other models. MobileNetV2 had an average training time of 0.04 ms, whereas ResNet152 required 1.4 ms. This difference is due to ResNet152's more numerous and complex layers compared to MobileNetV2. However, MobileNetV2 strikes a balance between accuracy and the time required for training and testing.

Based on the research, evaluation scores for each fruit image were obtained using ResNet152, as shown in Table II below:

TABLE II
EVALUATION RESULTS OF THE TRAINING DATA FOR THE RESNET152 MODEL.

Snake Fruit Type	Precission	Recall	F1-score
Pondoh Snake Fruit	1.00	0.86	0.87
Balinese Snake Fruit	0.7	1.00	0.85
Condet Snake Fruit	1.00	0.87	0.93
Sidempuan Snake Fruit	0.55	0.84	0.68

Meanwhile, the evaluation results using the MobileNetV2 model yielded the following outcomes, as shown in Table III below:

TABLE III
EVALUATION RESULTS OF THE TRAINING DATA FOR THE MOBILENETV2 MODEL.

Class	Precission	Recall	F1-score
Pondoh Snake Fruit	1.00	0.75	0.86
Balinese Snake Fruit	0.75	1.00	0.86
Condet Snake Fruit	0.70	1.00	0.82
Sidempuan Snake Fruit	0.83	0.91	0.87

Based on the above evaluation results, it is evident that Balinese Snake Fruit exhibits higher precision values in both the ResNet152 and MobileNetV2 models. On the other hand, Pondoh Snake Fruit demonstrates higher recall values in both models. Both types of snake fruit are characterized by distinct shapes and skin colors compared to other fruit varieties.

IV. CONCLUSION

Based on the research conducted with a dataset of 400 images of 4 types of snake fruit, the system successfully identified fruit types based on their categories. The dataset comprised 80 images for training, 10 for validation, and 10 for testing. The training process utilized the TensorFlow and Keras libraries in the Google Colab IDE. With the same parameters, the MobileNetV2 model achieved an accuracy of 84.62%, while ResNet152 achieved 92.31%. ResNet152 demonstrated the highest accuracy in identifying local superior snake fruit. However, ResNet152 is disadvantaged by longer training times compared to MobileNetV2. MobileNetV2 exhibited a good accuracy result with faster training and testing processes. The accuracy difference between the two models was not substantial. This study has limitations, including the manual classification process. Future developments could involve mobile-based applications for real-time fruit classification. The snake fruit classification process presents its challenges, such as using training data that includes both whole fruits and peeled fruits, impacting the accuracy achieved.

AUTHORS CONTRIBUTION

Nyoman Purnama: Investigation, data collection, analysis, review writing, coding and editing.

COPYRIGHT

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

REFERENCES

[1] N. Adiputra, "Fungsi Buah Dan Daun Tanaman Dalam Budaya Bali Sebuah Kajian Terhadap Tanaman Upacara."

[2] N. Widya Utami, N. Purnama, I. Putu, And R. Prajna, "Klasifikasi Tanaman Upakara Adat Hindu Di Kebun Raya Eka Karya Bali Menggunakan Algoritma Convolutional Neural Network," 2023.

[3] I. M. Tamba, "Kajian Buah-Buahan Lokal Unggulan Provinsi Bali dan Potensi Dinamisnya," *JIA (Jurnal Ilmiah Agribisnis) : Jurnal Agribisnis dan Ilmu Sosial Ekonomi Pertanian*, vol. 9, no. 2, pp. 126–132, Apr. 2023, doi: 10.37149/jia.v9i2.1117.

[4] F. Fitra Maulana and N. Rochmawati, "Klasifikasi Citra Buah Menggunakan Convolutional Neural Network".

[5] N. E. A. Mimma, S. Ahmed, T. Rahman, and R. Khan, "Fruits Classification and Detection Application Using Deep Learning," *Sci Program*, vol. 2022, 2022, doi: 10.1155/2022/4194874.

[6] S. Juliansyah *et al.*, "Klasifikasi Citra Buah Pir Menggunakan Convolutional Neural Networks," *Jurnal Infra Petra*, vol. 7, no. 1, pp. 489–495, 2021, doi: 10.22441/incomtech.v11i1.10185.

[7] R. Pathak, "CLASSIFICATION OF FRUITS USING CONVOLUTIONAL NEURAL NETWORK AND TRANSFER

LEARNING MODELS." [Online]. Available: https://www.researchgate.net/publication/364254116

[8] Y. Miftahuddin and F. Zaelani, "Perbandingan Metode Efficientnet-B3 dan Mobilenet-V2 Untuk Identifikasi Jenis Buah-buahan Menggunakan Fitur Daun," 2022.

[9] V. Gadre, S. Sashte, and A. Sarnaik, "WASTE CLASSIFICATION USING RESNET-152," *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, vol. 07, no. 01, Jan. 2023, doi: 10.55041/ijsrem17421.

[10] T. Saputra, M. Ezar, and A. Rivan, "STRING (Satuan Tulisan Riset dan Inovasi Teknologi) Analisis Performa Resnet-152 Dan Alexnet Dalam Klasifikasi Jenis Kanker Kulit." [Online]. Available: https://challenge.isic-

[11] M. C. Wujaya and L. W. Santoso, "Klasifikasi Pakaian Berdasarkan Gambar Menggunakan Metode YOLOv3 dan CNN."

[12] "Fruit Classification using Convolutional Neural Network (CNN)," *Precision Agriculture Science and Technology*, vol. 3, no. 1, 2021, doi: 10.12972/pastj.20210001.

[13] W. Setiawan, "Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus," *Jurnal Simantec*, vol. 7, no. 2, pp. 48–53, 2020, doi: 10.21107/simantec.v7i2.6551.

[14] D. Ramayanti, D. Asri, and L. Lionie, "Implementasi Model Arsitektur VGG16 dan MobileNetV2 Untuk Klasifikasi Citra Kupu-Kupu Article Info ABSTRAK," *JSAI: Journal Scientific and Applied Informatics*, vol. 5, no. 3, 2022, doi: 10.36085.

[15] I. P. W. Prasetia and I Made Gede Sunarya, "Image Classification of Balinese Seasoning Base Genep Based on Deep Learning," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 13, no. 1, Mar. 2024, doi: 10.23887/janapati.v13i1.67967.

[16] S. A. Maulana *et al.*, "Penerapan Metode CNN (Convolutional Neural Network) Dalam Mengklasifikasi Jenis Ubur-Ubur," *Jurnal Penelitian Rumpun Ilmu Teknik (JUPRIT)*, vol. 2, no. 4, pp. 122–130, 2023, doi: 10.55606/juprit.v2i4.3084.

[17] A. Prima, "Rancang Bangun Sistem Pendeteksi Aneka Ragam Buah Menggunakan MobileNetv2," *Jurnal Sistim Informasi dan Teknologi*, pp. 208–215, Jul. 2023, doi: 10.60083/jsisfotek.v5i2.217.

[18] M. Sanjaya and E. Nurraharjo, "Deteksi Jenis Rempah-Rempah Menggunakan Metode Convolutional Neural Network Secara Real Time," 2023.

[19] A. Eka *et al.*, "JEPIN (Jurnal Edukasi dan Penelitian Informatika) Klasifikasi Jenis Rempah Menggunakan Convolutional Neural Network dan Transfer Learning," 2023.

[20] C. Z. Basha, B. N. L. Pravallika, and E. B. Shankar, "An efficient face mask detector with pytorch and deep learning," *EAI Endorsed Trans Pervasive Health Technol*, vol. 7, no. 25, pp. 1–8, 2021, doi: 10.4108/eai.8-1-2021.167843.

Comparison of CNN Transfer Learning in Detecting Superior Local Fruit Types in Bali

ORIGINALITY REPORT

8%
SIMILARITY INDEX

PRIMARY SOURCES

1	ojs3.unpatti.ac.id Internet	29 words — 1%
2	Navin Kumar Manaswi. "Chapter 2 Understanding and Working with Keras", Springer Science and Business Media LLC, 2018 Crossref	27 words — 1%
3	K V Sudheesh, Kiran, Harinahalli Lokesh Gururaj, Vinayakumar Ravi, Meshari Almeshari, Yasser Alzamil. "Early Prediction of Covid-19 Samples from Chest X-ray Images using Deep Learning Approach", The Open Bioinformatics Journal, 2023 Crossref	25 words — 1%
4	Ronald Mustika Putra, Emny Harna Yossy, Suharjito, Immanuela Puspasari Saputro, Devriady Pratama, Teguh Prasandy. "Face Mask Detection using Convolutional Neural Network", 2023 8th International Conference on Business and Industrial Research (ICBIR), 2023 Crossref	24 words — 1%
5	www.frontiersin.org Internet	17 words — < 1%
6	www.atlantis-press.com Internet	16 words — < 1%

7 Luyuan Wu, Yunxiu Liu, Jianwei Zhang, Boyang Zhang, Zifa Wang, Jingbo Tong, Meng Li, Anqi Zhang. "Identification of flood depth levels in urban waterlogging disaster caused by rainstorm using a CBAM-improved ResNet50", Expert Systems with Applications, 2024
Crossref
14 words — < 1%

8 fastercapital.com
Internet
14 words — < 1%

9 e-journal.unipma.ac.id
Internet
10 words — < 1%

10 pubmed.ncbi.nlm.nih.gov
Internet
10 words — < 1%

11 tvst.arvojournals.org
Internet
10 words — < 1%

12 Rizqi Renafasih Alinra, Satryo B. Utomo, Gamma Aditya Rahardi, Khairul Anam. "Detector Face Mask using UAV-based CNN Transfer Learning of YOLOv5", 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), 2022
Crossref
9 words — < 1%

13 deepai.org
Internet
9 words — < 1%

14 jurnal.iaii.or.id
Internet
9 words — < 1%

15 Amal Abdel-raouf, Alaa Sheta, AbdelKarim Baareh, Peter Rausch. "Barcode-less Fruits Classification Using Deep Learning", IAES International Journal of Artificial Intelligence (IJ-AI), 2024
Crossref
8 words — < 1%

16	Muhlis Tahir, Abdul Azis Jakfar, Dian Budi Elnursa, Rio Meisya Resnanda. "Performance Evaluation of Pre-Trained Convolutional Neural Network and Transfer Learning for Classification of Spices and Herbal Medicines in Madura", 2023 International Conference of Computer Science and Information Technology (ICOSNIKOM), 2023 Crossref	8 words — < 1%
17	garuda.kemdikbud.go.id Internet	8 words — < 1%
18	hrcak.srce.hr Internet	8 words — < 1%
19	jurnal.polgan.ac.id Internet	8 words — < 1%
20	jurnal.umb.ac.id Internet	8 words — < 1%
21	trilogi.ac.id Internet	8 words — < 1%
22	Mohammad Farid Naufal, Selvia Ferdiana Kusuma. "Weather image classification using convolutional neural network with transfer learning", AIP Publishing, 2022 Crossref	6 words — < 1%
23	Omar E. Elejla, Mohammed Anbar, Shady Hamouda, Serri Faisal, Abdullah Ahmed Bahashwan, Iznan H. Hasbullah. "Deep-Learning-Based Approach to Detect ICMPv6 Flooding DDoS Attacks on IPv6 Networks", Applied Sciences, 2022 Crossref	6 words — < 1%

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF