LONTAR KOMPUTER VOL. 14, NO. 1 APRIL 2023
DOI : 10.24843/LKJITI.2023.v14.i01.p01
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021
p-ISSN 2088-1541
e-ISSN 2541-5832

# Optimization Strategy on Deep Learning Model to Improve Fruit Freshness Recognition

I Gusti Agung Indrawan[a1], Putu Andy Novit Pranartha[a2], I Wayan Agus Surya Darma[a3], I Putu Eka Giri Gunawan[a4]

[a]Department of Informatics, Faculty of Technology and Informatics, Institut Bisnis dan Teknologi Indonesia
Denpasar, Indonesia
[1]agung.indrawan@instiki.ac.id
[2]tuandy27@gmail.com
[3]surya@instiki.ac.id (Corresponding author)
[4]eka.giri@instiki.ac.id

***Abstract***

*The high fruit production during the harvest season is a challenge in the process of sorting fresh fruit and rotten fruit in plantations. Automatic fruit freshness classification based on deep learning can speed up the sorting process. However, building a model with high accuracy requires the right strategy based on the dataset's characteristics. This research aims to apply optimization strategies to deep learning models to improve performance. The optimization strategy is implemented by optimizing the model using the fine-tuning approach by selecting the best parameters based on learning rate, optimizers, transfer learning, and data augmentation. The transfer learning process is applied based on the dataset's characteristics by training some parameters with a size of 30% and 60%, which were tested in four scenarios. The fine-tuning strategy is applied to three Deep Learning models, i.e., MobileNetv2, ResNet50, and InceptionResNetV2, which have various parameter sizes. Based on test results, fine-tuning strategy produces the best performance up to 100% with a learning rate of 0.01. The SGD optimizers on the InceptionResNetV2 model are trained on 60% of the parameters.*

***Keywords:*** *Fruit Freshness, Deep Learning, Fine Tuning, Data Augmentation, Transfer Learning*

## 1. Introduction

Indonesia is a tropical country that produces a lot of apples, oranges, and bananas. During the harvest season, fruit production on plantations increases, so a fresh fruit sorting process is needed to maintain the quality of the fruit supply. The fruit classification process is required to distinguish between rotten fruit and fresh fruit parts to minimize errors in categorizing fruit, which still uses human labor and takes longer. Computer-based automatic fruit freshness classification can speed up sorting by utilizing fruit objects' color and texture characteristics. Machine learning and deep learning methods are currently widely applied for this task. However, building a high-performance model requires the right strategy based on the dataset's characteristics.

The classification process can be done automatically by utilizing deep learning models. Convolutional Neural Network (CNN) is a type of deep learning focused on being used in image processing. Convolutional Neural Network (CNN) is an artificial neural network used in image recognition and processing [1]. CNN works hierarchically so that the output in the first convolution layer is used as input in the next convolution layer [2]. Related studies have proposed deep learning methods in similar domains. In [3], proposed apple freshness detection using ResNet50. This research can identify fresh and rotten apples. The transfer learning strategy using the ResNet50 model can produce a classification performance of up to 97%. MobileNetV2 also shows high performance in identifying similar cases [4]. This study tested the model's performance on the Kaggle dataset and achieved the highest accuracy in the training data at 99.46% and in the validation set at 99.61% by implementing MobileNetV2. In [5], the Single Shot Detector (SSD) Model had an accuracy of 49.2%, significantly behind the 80.4% accuracy of the Faster R-CNN.

LONTAR KOMPUTER VOL. 14, NO. 1 APRIL 2023 p-ISSN 2088-1541
DOI : 10.24843/LKJITI.2023.v14.i01.p01 e-ISSN 2541-5832
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

In [6], the Gabor CNN approach can generate an image classification accuracy of 89%, which is the highest. In [7], to achieve a test accuracy result of 95.24%, the VGG19 architecture is paired with epoch variations and data augmentation in the testing settings.

Transfer learning is a deep learning model training strategy to improve model performance. Several researchers have proposed transfer learning strategies for various research objects. On Balinese carving objects, a transfer learning strategy has been proposed in four pre-trained models, MobileNet, Inception-v3, VGG16, and VGG19. This research improved the model's performance to achieve the highest accuracy of 87.5% by retraining 90% of the parameters [8]. The research improved the model's performance to achieve the highest accuracy of 87.5% by retraining 90% of the parameters. Fine-tuning is a strategy to improve the performance of deep learning models by modifying the model's hyperparameters. The hyperparameter model is tuned by testing a variety of values to produce the best performance, in [9], the proposed Balinese characters recognition method using transfer learning and fine-tuning on the MobileNet model. This research increased the model's performance to 86.23% using stochastic gradient descent (SGD) as the optimizer. Data augmentation is a technique used to overcome the problem of overfitting the model due to the limited dataset in deep learning model training. Several studies have proposed the application of data augmentation in recognizing various objects, in [10], applied a geometric transformation to improve the performance of the MobileNet by up to 16.2%.

This research aims to apply optimization strategies to deep learning models to improve performance. The optimization strategy is implemented by optimizing the model by fine-tuning model by selecting the best parameters based on learning rate, optimizers, transfer learning, and data augmentation. The fine-tuning strategy is applied to three Deep Learning models, i.e., MobileNetv2, ResNet50, and InceptionResNetV2, which have various parameter sizes. The transfer learning process is applied based on the dataset's characteristics by training some parameters with a size of 30% and 60%, which were tested in four scenarios.

## 2. Research Methods

In this research, we proposed a fine-tuning strategy to improve the deep learning model for fruit freshness recognition. The strategy consists of fine-tuning learning rates and optimizers and transferring learning based on dataset characteristics and size-similarity matrix. We applied data augmentation to enrich the data variation. The recognition model is built based on three pre-trained models. These three pre-trained models were selected based on various parameter sizes. The MobileNetV2 model is the smallest model with 3.4 million parameters. The ResNet50 model has 23 million parameters. The InceptionResNetV2 model has 54 million parameters [5]. The model's selection with variations in the size of these parameters aims to determine each model's performance on various parameters. At this stage, model training is carried out by applying transfer learning and fine-tuning strategies to improve the performance of the classification model. Figure 1 shows the research methods.

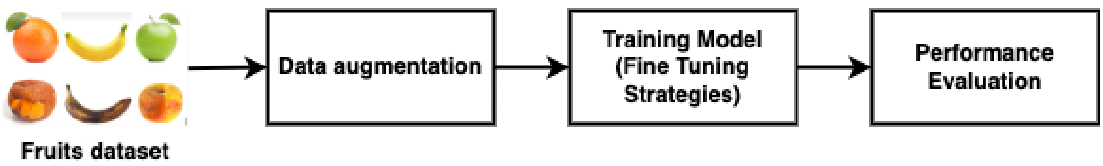



**Figure 1.** Research Methods

### 2.1. Dataset

This study uses the fruit freshness dataset obtained from the Kaggle repository (https://www.kaggle.com/competitions/dlai5-bad-apples/data) based on the DA5 Hackathon competition collected by each competition participant. This dataset comprises 13,737 images divided into 10,901 training images, 138 validation images, and 2,698 testing images with various resolutions in *.png and *.jpg format. Validation data is an image that experts in the DLAI5 Hackathon competition have validated. In the testing phase, the training vs. testing ratio is 75:25 based on the DLAI5 Hackathon competition. This dataset consists of 13,737 images. Figure 2 shows a sample image of the fruit dataset.

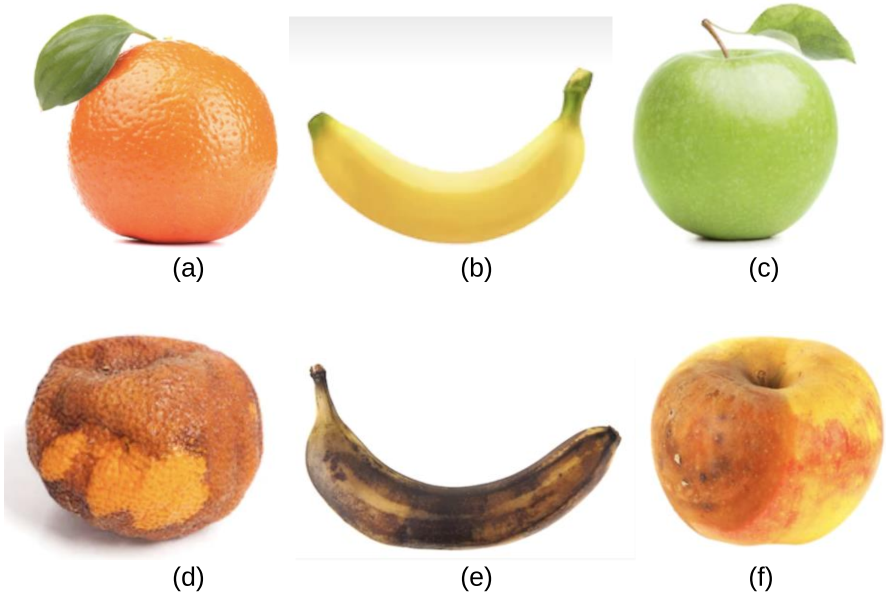



**Figure 2**. Sample images from the dataset. (a) fresh orange, (b) fresh banana, (c) fresh apple, (d) rotten orange, (e) rotten banana, (f) rotten apple

The dataset consists of 6 classes, i.e., fresh apples, fresh oranges, fresh bananas, rotten apples, rotten oranges, and rotten bananas. Table 1 shows the number of images in each class.

**Table 1.** Number of Images in each Class

| Classes | #images |
|---|---|
| Fresh Orange | 1,860 |
| Fresh Banana | 1,986 |
| Fresh Apple | 2,122 |
| Rotten Orange | 2,002 |
| Rotten Banana | 2,790 |
| Rotten Apple | 2,967 |

## 2.2. Data Augmentation

Data augmentation is a technique of artificially extending a labeled training set by changing data points in a way that preserves class labels [11]. In this study, the data augmentation technique used is Geometric Transformation. Geometric Transformations (GT) is a technique for multiplying images originating from one main image and can become many new images utilizing new adjustments such as rotation, translation, flip, and zoom. This augmentation method can provide a more diverse variety of fruit object positions to represent fruit positions during the automatic fruit classification process. This method implements augmentation based on the Keras library by performing four operations, which are (i) rotation, (ii) flip, (iii) translation, and (iv) zoom. Figure 3 shows the types of operations performed on geometric transformation-based data augmentation.

LONTAR KOMPUTER VOL. 14, NO. 1 APRIL 2023 p-ISSN 2088-1541
DOI : 10.24843/LKJITI.2023.v14.i01.p01 e-ISSN 2541-5832
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

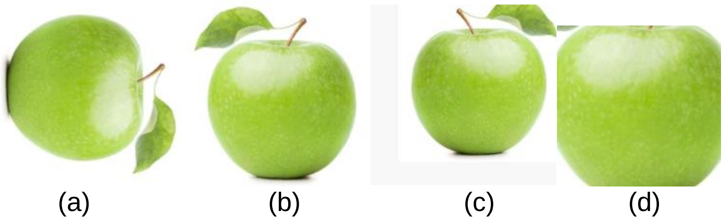


**Figure 3.** Geometric Transformation on fruits dataset: (a) Rotation, (b) Flip, (c) Translation, (d) Zoom

## 2.3. Optimization Strategies

### 2.3.1. Fine Tuning

Fine-tuning means training a model with a specific task using a model trained with a large dataset [12]. Fine-tuning strategies are good at helping to improve performance and speed up network training [13]. The fine-tuning strategy is carried out by using a variety of learning rate values and the types of optimizers used in the model training process [14]. This strategy is implemented by conducting experiments on the learning rate value of each pre-trained model to produce the model with the best performance. Table 1 shows three deep learning models' learning rate and optimizer variation.

**Table 1.** Learning rate and Optimizers Variation on Deep Learning Models

| **Model** | **Strategi *Fine-tuning*** | |
|---|---|---|
| | ***Learning rate*** | ***Optimizers*** |
| *MobileNetV2* | *Learning rate 1* | *Optimizers 1* |
| | *Learning rate 2* | *Optimizers 2* |
| *ResNet50* | *Learning rate 1* | *Optimizers 1* |
| | *Learning rate 2* | *Optimizers 2* |
| *InceptionResNetV2* | *Learning rate 1* | *Optimizers 1* |
| | *Learning rate 2* | *Optimizers 2* |

### 2.3.2. Transfer Learning

Transfer learning is a method that utilizes a model that has been trained on a dataset to solve other similar problems by using it as a starting point, modifying and updating its parameters so that they match the new dataset [15]. The meaning of transfer learning is transferring existing knowledge to be adapted to a new dataset so that it can add the latest features. The main advantages of transfer learning are increased classification accuracy and acceleration of the training process [16]. Transfer learning by training a classification model based on a dataset size-similarity matrix. The dataset size-similarity matrix describes how the transfer learning strategy is carried out based on the characteristics of the data. This characteristic is based on the size and type of data trained on the pre-trained model. In this study, the dataset has different characteristics from the ImageNet dataset used in the pre-trained model. In addition, the data size is only 13,737 images, far less than the ImageNet dataset, which consists of 1.2 million images. This is why we implement, based on the Quadrant 3 strategy, by training the model with 30% and 60% parameters by modifying the number of layers trained in each model. Figure 4 shows the dataset size-similarity matrix.

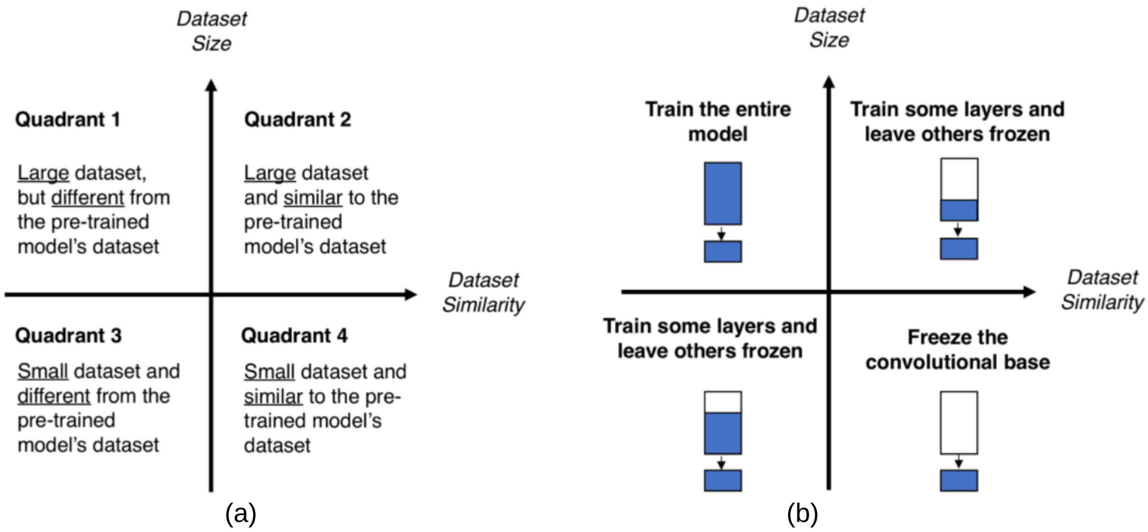



**Figure 4.** Dataset Size-Similarity Matrix. (a) quadrant characteristics of the dataset, (b) strategy applied to model training [10]

Table 2 shows the transfer learning strategy in the fresh and rotten fruit classification model training. Based on the dataset size-similarity matrix, the process trains each model with two parameter variations, i.e., 30% and 60%. These two transfer learning strategies were selected based on the dataset size-similarity matrix on the transfer learning strategy. Based on the four quadrants in this matrix (Figure 3), quadrant 3 is a strategy suitable for training models by training some of the model's parameters with a size of 30% and 60% of the total parameters in each model.

**Table 2.** Strategi Transfer Learning

| Model | Trainable Params. |
|---|---|
| *MobileNetV2* | 30% |
| | 60% |
| *ResNet50* | 30% |
| | 60% |
| *InceptionResNetV2* | 30% |
| | 60% |

### 2.4. Performance Evaluation

Model performance evaluation is carried out by measuring the classification results. The performance evaluation is based on accuracy, precision, recall, and f1-score [17]. Evaluation is done by measuring the performance of the classification model based on recall, precision, f1-score, and accuracy. The performance of each model is evaluated for each fine-tuning, transfer learning, and data augmentation strategy applied to the model training process. Three models were used to run these scenarios, i.e., MobileNetV2, ResNet50, and InceptionResNetV2, trained in each scenario for ten epochs. Equations calculate the accuracy, precision, recall, and f1-score:

$$Accuracy = \frac{TP + FN}{P + N} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

## 3. Result and Discussion

In this Section, the results of each stage are discussed to compare the model's performance in each scenario. Model training and testing using the NVIDIA K80 GPU with 12GB memory.

### 3.1. Models Training

This model training consists of three strategies, i.e., transfer learning (TL), fine-tuning (FT), and data augmentation (AD). The three strategies form four model training scenarios:

i. The first scenario only uses the fine-tuning strategy (FT).
ii. The second scenario uses fine-tuning and data augmentation strategies (FT+AD).
iii. The third scenario uses fine-tuning and transfer learning strategies (FT+TL).
iv. The fourth scenario uses fine-tuning, data augmentation, and transfer learning strategies (FT+AD+TL).

The optimization strategy was carried out on three models, i.e., MobileNetV2, ResNet50, and InceptionResNetV2, in the model training scenario using epoch (E) 10 combined with a learning rate (LR) of 0.01 SGD optimizers and 0.001 optimizers Adam in each model training scenario with the distribution of 10,901 training images and 138 validation images. Table 3 shows the results of the first scenario.

**Table 3.** The Results of First Scenario (i)

| ***LR*** | | ***Model*** | | |
|---|---|---|---|---|
| | | ***MobileNetV2*** | ***ResNet50*** | ***InceptionResNet V2*** |
| 0,01 | *val acc.* | 97.22% | 99.22% | **100%** |
| | *training acc.* | 99.84% | 99.83% | **99.92%** |
| 0,001 | *val acc.* | 86.84% | 96.07% | 97.74% |
| | *training acc.* | 98.44% | 98.93% | 99.67% |

The optimization strategy is applied to the MobileNetV2, ResNet50, and InceptionResNetV2 models to get the best results from the three models in scenario (i). Based on the experimental results in scenario (i), the InceptionResNetV2 model (A) produced the highest training accuracy of 99.92% and a validation accuracy of 100%.

**Table 4.** The Results of Second Scenario (ii)

| ***AD*** | ***LR*** | | ***Model*** | | |
|---|---|---|---|---|---|
| | | | ***MobileNetV2*** | ***ResNet50*** | ***InceptionResNet V2*** |
| *GT* | 0,01 | *val acc.* | 81.47% | 85.84% | **99.89%** |
| | | *training acc.* | 99.54% | 84.87% | **99.94%** |
| | 0,001 | *val acc.* | 87.25% | 97.03% | 81.95% |
| | | *training acc.* | 99.41% | 98.73% | 97.01% |

Table 4 shows the results of the second scenario (ii) in the model training. The second scenario applied geometric Transformation as data augmentation. We perform four types of geometric Transformation that are shown in Figure 3. Based on the data augmentation process, the dataset number increases four times in the training process, based on four types of data augmentation techniques. MobileNetV2, ResNet50, and InceptionResNetV2 were trained to produce the best results out of the three models. Based on the experimental results in scenario (ii), the InceptionResNetV2 model (B) produced the highest training accuracy of 99.94% and a validation accuracy of 99.89%.

**Table 5.** The Results of Third Scenario (iii)

| *TL (%)* | *LR* | | *Model* | | |
|---|---|---|---|---|---|
| | | | ***MobileNet V2*** | ***ResNet50*** | ***InceptionResNet V2*** |
| 30 | 0,01 | *val acc.* | 9203% | 99.33% | 98.89% |
| | | *training acc.* | 99.86% | 99.71% | 99.63% |
| | 0,001 | *val acc.* | 81.95% | 98.15% | 98.85% |
| | | *training acc.* | 97.01% | 99.41% | 99.38% |
| 60 | 0,01 | *val acc.* | **99.37%** | 80.73% | 85.25% |
| | | *training acc.* | **99.95%** | 81.56% | 84.19% |
| | 0,001 | *val acc.* | 95.63% | 96.11% | 95.85% |
| | | *training acc.* | 99.54% | 97.79% | 99.32% |

Table 5 shows the third scenario (iii) results in the model training. MobileNetV2, ResNet50, and InceptionResNetV2 were trained to obtain the best results from the three models. Based on the experimental results in scenario (iii), the MobileNetV2 model (C) produced the highest training accuracy of 99.37% and a validation accuracy of 99.95%.

**Table 6.** The Results of Fourth Scenario (iv)

| *AD* | *TL (%)* | *LR* | | *Model* | | |
|---|---|---|---|---|---|---|
| | | | | ***Mobile NetV2*** | ***ResNet50*** | ***InceptionResNet V2*** |
| ***GT*** | 30 | 0,01 | *val acc.* | 99.18% | 98.93% | 85.73% |
| | | | *training acc.* | 99.90% | 99.86% | 85.12% |
| | | 0,001 | *val acc.* | 97.14% | 89.77% | 99.48% |
| | | | *training acc.* | 99.79% | 99.10% | 99.57% |
| | 60 | 0,01 | *val acc.* | 79.80% | 83.69% | **99.89%** |
| | | | *training acc.* | 98.55% | 82.99% | **99.56%** |
| | | 0,001 | *val acc.* | 87.99% | 71.76% | 99.11% |
| | | | *training acc.* | 99.18% | 87.27% | 99.48% |

Table 6 shows the results of the fourth scenario (iv) in the model training. The models trained were MobileNetV2, ResNet50, and InceptionResNetV2 to get the best results out of the three models. Based on the experimental results in scenario (iv), the highest accuracy of the InceptionResNetV2 (D) model resulted in a training accuracy of 99.89% and a validation accuracy of 99.56%.

Based on the experimental results of the four model training scenarios, the model with the highest training accuracy and validation accuracy was selected from each scenario. Table 7 shows the model with the highest performance in each scenario.

**Table 7.** The Best Model of Each Scenario

| **Scenario** | ***Model*** | **AD** | ***TL (%)*** | ***LR*** | ***Validation Acc.*** | ***Training Acc.*** |
|---|---|---|---|---|---|---|
| ***FT*** | *InceptionResNetV2 (A)* | - | - | 0.01 | 100% | 99.92% |
| ***FT*+AD** | *InceptionResNetV2 (B)* | *GT* | - | 0.01 | 99.89% | 99.94% |
| ***FT+TL*** | *MobileNetV2 (C)* | - | 60 | 0.01 | 99.37% | 99.95% |
| ***FT*+AD+*TL*** | *InceptionResNetV2 (D)* | *GT* | 60 | 0.01 | 99.89% | 99.56% |

### 3.2. Performance Evaluation

The performance of the four models was evaluated using testing data from 2,698 testing images divided into six classes. This testing phase was carried out based on the model with the best performance in each scenario in Table 7. Figure 5 shows the confusion matrix based on the test results for each model.

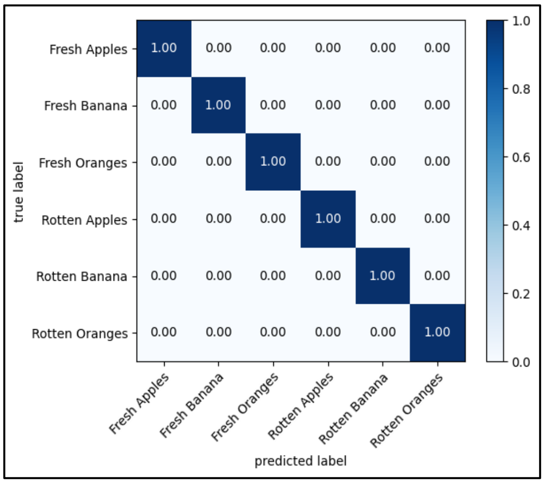
Figure 5 (a) is the confusion matrix of the InceptionResNetV2 model (Model A), which has been trained and tested using the first scenario, namely the fine-tuning strategy. The confusion matrix shows that the Fresh apple class (1) produces 100% accuracy, the Fresh Banana class (2) produces 100% accuracy, the Fresh Oranges class (3) produces 100% accuracy, the Rotten apple class (4) produces accuracy 100%, Rotten Banana class (5) produces 100% accuracy, Rotten Oranges class (6) produces 100% accuracy.

Figure 5(b) is the confusion matrix of the InceptionResNetV2 model (Model B), which has been trained and tested using the second scenario, namely the fine-tuning strategy and data augmentation. The confusion matrix shows that the Fresh apple class (1) produces 100% accuracy, the Fresh Banana class (2) produces 100% accuracy, the Fresh Oranges class (3) produces 100% accuracy, the Rotten apple class (4) produces 100% accuracy, in the Rotten Banana class (5) it produces 100% accuracy, in the Rotten Oranges class (6) it produces 100% accuracy.

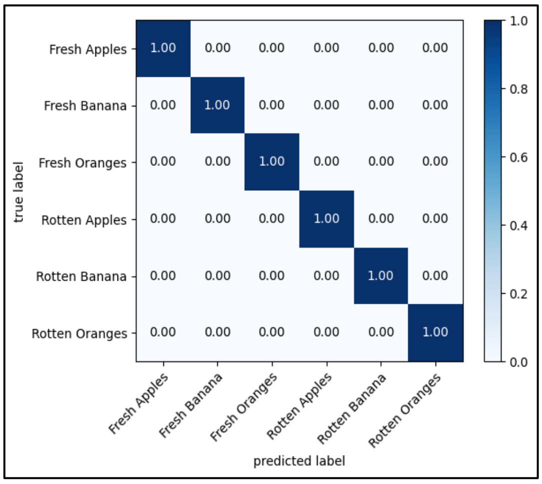
Figure 5(c) is an image of the confusion matrix of the MobileNetV2 model (Model C), which has been trained and tested using the third scenario, namely the fine-tuning strategy and transfer learning. The confusion matrix shows that the Fresh apple class (1) produces 100% accuracy, the Fresh Banana class (2) produces 97% accuracy, the Fresh Oranges class (3) produces 99% accuracy, the Rotten apple class (4) produces an accuracy of 100 %, Rotten Banana class (5) produces 99% accuracy, Rotten Oranges class (6) produces 100% accuracy.

Figure 5(d) is the confusion matrix of the InceptionResNetV2 model (Model D), which has been trained and tested using the third scenario, namely fine-tuning strategy, data augmentation, and transfer learning. Figure 15 The confusion matrix shows that the Fresh apple class (1) produces 100% accuracy, the Fresh Banana class (2) produces 100% accuracy, the Fresh Oranges class (3) produces 100% accuracy, the Rotten Apples class (4) produces accuracy 100%, Rotten Banana class (5) produces 100% accuracy, Rotten Oranges class (6) produces 100% accuracy. Based on the confusion matrix results, it is concluded in the model testing table below. Table 8 shows a model testing table based on the four best models.

Based on the model performance evaluation, the model optimization strategy shows that it can improve model performance to recognize all classes correctly. Table 8 shows the results of the model performance evaluation and details of the test results for each class. Based on the tests carried out according to each scenario, the best testing accuracy results are obtained and evaluated based on testing accuracy, precision, recall, and f1-score. The following are three models that managed to achieve 100% performance.

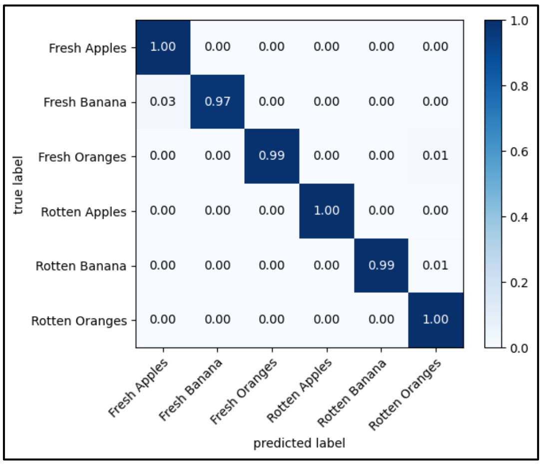



(a)

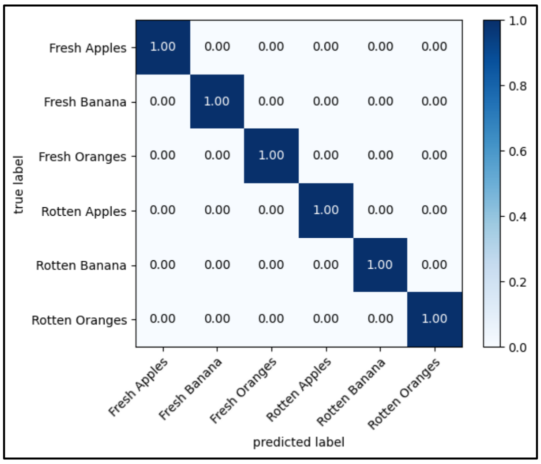



(b)

LONTAR KOMPUTER VOL. 14, NO. 1 APRIL 2023
DOI : 10.24843/LKJITI.2023.v14.i01.p01
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021
p-ISSN 2088-1541
e-ISSN 2541-5832




(c)




(d)

**Figure 5.** Confusion Matrix of Each Model Testing: (a) InceptionResNetV2 (A), (b) InceptionResNetV2 (B), (c) MobileNetV2 (C), (d) *InceptionResNetV2 (D)*

Based on the experimental results, testing the four models at the evaluation stage shows that the proposed optimization strategy can recognize 2,698 testing images correctly. The proposed optimization strategy can produce fruit freshness classification performance with 100% accuracy. These results show that the optimization strategy can improve performance not only on models with a small number of parameters, e.g., MobileNetV2 with 3.4M but also on models with large parameter sizes, e.g., InceptionResNetV2 with 54M parameters.

**Table 8.** Detailed Results of Model Evaluation

| ***Model*** | **AD** | ***TL (%)*** | ***LR*** | **Class** | **#Data** | **#True** | **#False** | ***Acc.*** | ***P*** | ***R*** | ***F1*** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *A* | - | - | 0,01 | 1 | 395 | 395 | 0 | **100%** | 100% | 100% | 100% |
| | | | | 2 | 381 | 381 | 0 | | | | |
| | | | | 3 | 388 | 388 | 0 | | | | |
| | | | | 4 | 601 | 601 | 0 | | | | |
| | | | | 5 | 530 | 530 | 0 | | | | |
| | | | | 6 | 403 | 403 | 0 | | | | |
| *B* | *GT* | - | 0,01 | 1 | 395 | 395 | 0 | **100%** | 100% | 100% | 100% |
| | | | | 2 | 381 | 381 | 0 | | | | |
| | | | | 3 | 388 | 388 | 0 | | | | |
| | | | | 4 | 601 | 601 | 0 | | | | |
| | | | | 5 | 530 | 530 | 0 | | | | |
| | | | | 6 | 403 | 403 | 0 | | | | |
| *C* | - | 60 | 0,01 | 1 | 395 | 395 | 0 | **99,37 %** | 99% | 99% | 99% |
| | | | | 2 | 381 | 371 | 10 | | | | |
| | | | | 3 | 388 | 384 | 4 | | | | |
| | | | | 4 | 601 | 601 | 0 | | | | |
| | | | | 5 | 530 | 527 | 3 | | | | |
| | | | | 6 | 403 | 403 | 0 | | | | |
| *D* | *GT* | 60 | 0,01 | 1 | 395 | 395 | 0 | **100%** | 100% | 100% | 100% |
| | | | | 2 | 381 | 381 | 0 | | | | |
| | | | | 3 | 388 | 388 | 0 | | | | |

| *Model* | AD | *TL (%)* | *LR* | Class | #Data | #True | #False | *Acc.* | *P* | *R* | *F1* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 4 | 601 | 601 | 0 | | | | |
| | | | | 5 | 530 | 530 | 0 | | | | |
| | | | | 6 | 403 | 403 | 0 | | | | |

## 4. Conclusion

Deep learning model optimization strategies based on fine-tuning, transfer of learning, and data augmentation can produce high-performance models for fruit freshness recognition tasks. The fine-tuning strategy using a combination of learning rate = 0.01, SGD as optimizers, transfer learning with 60% parameters, and geometric Transformation as data augmentation yields up to 100% performance on InceptionResNetV2. The test results show that the resulting model can recognize all test images consisting of 2,698 images. In future work, further research can be carried out by applying the model to mobile devices for real-world applications.

## References

[1] A. Kholik, "Klasifikasi Menggunakan Convolutional Neural Network (Cnn) Pada Tangkapan Layar Halaman Instagram," *Jurnal Data Mining dan Sistem Informasi*, vol. 2, no. 2, pp. 10–20, 2021, [Online]. Available: https://ejurnal.teknokrat.ac.id/index.php/JDMSI/article/view/1345/673%0Ahttps://ejurnal.teknokrat.ac.id/index.php/JDMSI/article/view/1345

[2] E. N. Arrofiqoh and H. Harintaka, "Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi," *Geomatika*, vol. 24, no. 2, p. 61, 2018, doi: 10.24895/jig.2018.24-2.810.

[3] D. Srinivasan and Dr. M. Yousef, "Apple Fruit Detection and Maturity Status Classification," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 9, no. 2, pp. 1055–1059, 2020, doi: 10.35940/ijrte.b4063.079220.

[4] S. Chakraborty, F. M. J. M. Shamrat, M. M. Billah, M. al Jubair, M. Alauddin, and R. Ranjan, "Implementation of Deep Learning Methods to Identify Rotten Fruits," *Proceedings of the 5th International Conference on Trends in Electronics and Informatics, ICOEI 2021*, pp. 1207–1212, 2021, doi: 10.1109/ICOEI51242.2021.9453004.

[5] Y. Adiwinata, A. Sasaoka, I. P. Agung Bayupati, and O. Sudana, "Fish Species Recognition with Faster R-CNN Inception-v2 using QUT FISH Dataset," *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, vol. 11, no. 3, p. 144, Dec. 2020, doi: 10.24843/lkjiti.2020.v11.i03.p03.

[6] I. P. B. G. Prasetyo Raharja, I. M. Suwija Putra, and T. Le, "Kekarangan Balinese Carving Classification Using Gabor Convolutional Neural Network," *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, vol. 13, no. 1, p. 1, Apr. 2022, doi: 10.24843/lkjiti.2022.v13.i01.p01.

[7] A. A. J. V. Priyangka and I. M. S. Kumara, "Classification Of Rice Plant Diseases Using the Convolutional Neural Network Method," *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, vol. 12, no. 2, p. 123, Aug. 2021, doi: 10.24843/lkjiti.2021.v12.i02.p06.

[8] I. W. A. S. Darma, N. Suciati, and D. Siahaan, "Balinese Carving Recognition using Pre-Trained Convolutional Neural Network," *ICICoS 2020 - Proceeding: 4th International Conference on Informatics and Computational Sciences*, 2020, doi: 10.1109/ICICoS51170.2020.9299021.

[9] N. P. Sutramiani, N. Suciati, and D. Siahaan, "Transfer Learning on Balinese Character Recognition of Lontar Manuscript Using MobileNet," *ICICoS 2020 - Proceeding: 4th International Conference on Informatics and Computational Sciences*, pp. 0–4, 2020, doi: 10.1109/ICICoS51170.2020.9299030.

[10] I. W. A. S. Darma, N. Suciati, and D. Siahaan, "Neural Style Transfer and Geometric Transformations for Data Augmentation on Balinese Carving Recognition using MobileNet," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 6, pp. 349–363, 2020, doi: 10.22266/ijies2020.1231.31.

[11] A. J. Ratnerÿ, H. R. Ehrenbergÿ, and J. Dunnmon, "Belajar Menulis Domain-Spesifik Transformasi untuk Augmentasi Data," no. Nips, 2017.

[12] T. Fatyanosa, "Fine-Tuning Pre-Trained Transformer-based Language Model | by Tirana Fatyanosa | Medium," *www.medium.com*, 2020. https://fatyanosa.medium.com/fine-tuning-pre-trained-transformer-based-language-model-c542af0e7fc1 (accessed Mar. 31, 2022).
[13] J. Margeta, A. Criminisi, R. Cabrera Lozoya, D. C. Lee, and N. Ayache, "Fine-tuned convolutional neural nets for cardiac MRI acquisition plane recognition," *Comput Methods Biomech Biomed Eng Imaging Vis*, vol. 5, no. 5, pp. 339–349, 2017, doi: 10.1080/21681163.2015.1061448.
[14] X. Xu, B. Shi, Z. Gu, R. Deng, X. Chen, A.S. Krylov, Y. Ding, "3D No-Reference Image Quality Assessment via Transfer Learning and Saliency-Guided Feature Consolidation," *IEEE Access*, vol. 7, pp. 85286–85297, 2019, doi: 10.1109/ACCESS.2019.2925084.
[15] S. Sena, "Pengenalan Deep Learning Part 8 : Gender Classification using Pre-Trained Network (Transfer Learning) | by Samuel Sena | Medium," *www.medium.com*, 2018. https://medium.com/@samuelsena/pengenalan-deep-learning-part-8-gender-classification-using-pre-trained-network-transfer-37ac910500d1 (accessed Apr. 06, 2022).
[16] A. Saber, M. Sakr, O. M. Abo-Seida, A. Keshk, and H. Chen, "A Novel Deep-Learning Model for Automatic Detection and Classification of Breast Cancer Using the Transfer-Learning Technique," *IEEE Access*, vol. 9, pp. 71194–71209, 2021, doi: 10.1109/ACCESS.2021.3079204.
[17] I. W. A. S. Darma, N. Suciati, and D. Siahaan, "GFF-CARVING: Graph Feature Fusion for the Recognition of Highly Varying and Complex Balinese Carving Motifs," *IEEE Access*, 2022, doi: 10.1109/ACCESS.2022.3228382.