2021 International Seminar on Intelligent Technology and Its Applications (ISITIA) | 978-1-6654-2847-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISITIA52817.2021.9502258

# Robustness of Convolutional Neural Network in Classifying Apple Images

Dzalfa Tsalsabila Rhamadiyanti
*Department Informatics*
*Telkom University*
Bandung, Indonesia
dzalfatr@student.telkomuniversity.ac.id

Suyanto Suyanto
*Department Informatics*
*Telkom University*
Bandung, Indonesia
suyanto@telkomuniversity.ac.id

*Abstract*—Apple is one of the popular fruits for public consumption. People can distinguish many apples based on their colors and shapes, such as the Braeburn Apple with skin color varies from orange to red, the Pink Lady Apple that is red with pseudo pink, the Crismon Snow Apple that has dark red skin. Recently, computers can automatically recognize them using digital image processing techniques such as Convolutional Neural Networks (CNN). In this paper, a CNN-based classification model of apple types is developed using 1856 apple images from three classes derived from the fruit-360 dataset on the Kaggle website, and its robustness is then examined. Two types of testing have been carried out in this study: testing five scenarios for sharing training data and testing five scenarios for robustness to noise. An examination based on 5-fold cross-validation shows that CNN is robust to decreasing the portion of training set size up to 50% to get high accuracy of 99.97% in classifying 50% testing set, which is better than previous models that use VGG16, faster R-CNN, and Tanh. Decreasing the portion training set to 40% and 30% reduces the accuracy to 95.97% and 95.29%, respectively. Adding low-level noises of 10% into the testing images decreases the accuracy slightly to 99.17%. However, high-level noises of 50% drastically make the accuracy drastically drops to 63.93%.

*Keywords—apple classification, convolutional neural network, image processing, robustness, skin color*

## I. INTRODUCTION

Apple (Malus Domestica) is a pomaceous fruit from the Rosacease family originating from Central Asia. Apple is one of the most abundantly grown horticultural crops in Central Asia because of the favorable climate [1]. Apples thrive in subtropical regions such as America, Russia, the Netherlands, and Italy. Apples are a rich source of phytochemicals. According to epidemiological studies, the consumption of apples can provide various benefits, such as a reduced risk of several types of cancer, cardiovascular disease, and diabetes. Different types of apples have different phytochemical compositions [2].

Apples are one type of fruit that is superior and very popular for public consumption. There are many varieties of apples in the world, namely Pink Lady Apples which are red with pseudo pink, Granny Smith Apples which are light green, Crismon Snow apples with dark red skin, Braeburn apples whose skin colors vary from orange to red and above yellow, Rome Beauty Apples, which has shiny red color, Idared Apples with a layer bright red over a green-red skin, Golden Delicious apples with light green to golden yellow with small lenticels (spots), Cortland apples which are bright red, deep red, sometimes green, blushing, and many other types. Several types of apples, such as Roman Beauty, Idared, Cortland, and Golden Delicious apples, are usually provided in the form of applesauce [3]. Because several types of apples have almost the same skin color, apple distributors and the general public may find it difficult to classify them according to their needs. One of the things that can help people classify apples is digital image processing. With this research, it is hoped that apple distributors and the wider community will find it easier to find types of apples according to their needs.

Image classification is a reasonably easy task for humans to do. The development of science in Computer Vision allows computers to have intellectual abilities that can work like humans in general. Fruit image recognition has been carried out using a variety of techniques over the past few years [4].

In the field of digital image processing, several methods are often used, such as K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and CNN. Classification of apple types has been done using an image-based approach with a small dataset, and the results are still not optimal. Accordingly, this study will build a CNN model to classify apple images into three classes: Braeburn, Crimson Snow, and Pink Lady. A 100x100 image will be trained with three convolution layers of 8, 16, and 32. It is then evaluated using *k*-fold cross-validation (*k*FCV). Testing will be carried out in 100 epochs with a batch size of 32 in every fold. This research model was designed using a different dataset from previous studies that use CNN with an activation function of Tanh, VGG16, and comparing the faster R-CNN with the SSD500 and SSD300. A comprehensive evaluation of the CNN robustness to reducing the training set portion and noise levels is also discussed.

In this paper, section 1 discusses the introduction of apples, a brief explanation regarding digital image classification, and a little explanation regarding previous research. Section 2 will discuss research related to image classification, apple varieties, and the method used in this study, namely CNN, which will be discussed further in section 3 and explain the dataset used. Section 4 defines the results obtained in this study, and section 5 has the conclusions from this study.

## II. RELATED STUDIES

There have been several previous studies conducted using the CNN method to classify images. In this section, several studies related to image classification using an image-based approach is described. In [5], Chandrashekhar S. Janadri et al. show that a multi-class SVM (MSVM) produces a higher result of 80.0655% than an artificial neural network or ANN (13.688%) in the classification of Kirlian images. In [6], informs that deep learning (DL) gives much higher accuracy for recognizing fruit through images with a dataset of Fruit360. It reaches accuracies of 100% for the validation set and 96.3% for the testing set.

Yingqiong Peng et al. in [7], show that CNN obtains a testing accuracy of 97.19% using a ratio of 80% training data and 20% testing data from a total dataset of 1167 fruit images. The dataset is augmented by rotated, turned images by left, right, up, and down, also adjusted the brightness and darkness. Research by Kavish Sanghvi et al. informs that a classification of fauna images using CNN with 15962 datasets achieves an accuracy of 91.84% and succeeded in predicting testing data accrual value of 99.77%. This study uses VGG16, Tensorflow, and Leaky ReLU. to train a model [8]. Ashik Kumar et al. in [9] inform that VGG16 yields an accuracy of 99.77% up to epoch 40 in the classification of succulent plants. This research identified succulent plant species with 3632 succulent plant image datasets and 200 non-succulent plant images using VGG19, 3 CNN layers, and five layers of CNN.

The research by Marcus Guozong LIM et al. with the topic of durian classification using existing techniques in deep learning, namely convolutional neural network using a total of 800 pictures from 3 classes mala in a predictive value of 82.50% for Durio Zibethinus images and when added with an image next to the durian, the accuracy drops to 81.25% [10]. Shadman Sakib's research, which raised the topic of fruit recognition using a CNN with various hidden layers using a total dataset 17823, resulted in a training result of 99.79% and a testing result reaching 100% [11]. In [12], discusses Pure CNN (PCNN) with minimum parameters as a framework for fruit image classification. This study uses a fruit-360 dataset and produces an accuracy value of 98.88%.

Research on apple classification using CNN with a dataset of 100 images is conducted by Pichate Kunarkornvong et al. [13]. They compare two activation functions, namely ReLU and Tanh. Using ReLU, an accuracy of 95% is obtained, while using Tanh is 90%. Research by QiaokangLiang et al. in [14] discusses the introduction of apples using VGG16 and compares Faster R-CNN with SSD300 and SSD500. It can be concluded that the SSD algorithm is superior to the Faster R-CNN and the SSD500 is better than the SSD300 for getting the detection accuracy value.

In [15], describes the classification of the appearance of apples with low sample quality. This study uses the Imp-ResNet50 model for the classification process and produces an accuracy value of 96.5%. In [16], which contains real-time detection of disease in apple leaves using the INAR-SSD model, succeeded in obtaining a performance value of 78.80% mAP and a detection speed of 23.13 FPS [16]. All the previous studies show that CNN is the most excellent method for image processing. However, no comprehensive evaluation of the CNN robustness to reducing the training set portion and noise levels. Therefore, this paper discusses a more detailed evaluation of the CNN robustness.

## III. Literature Studies

### A. Apple Varieties

Apple (Malus Domestica) is a pomaceous fruit from the Rosacease family originating from Central Asia. Apples thrive in subtropical regions such as America, Russia, the Netherlands, and Italy.

There are many varieties of apples in the world, namely Pink Lady Apples which are red with pseudo pink, Granny Smith Apples which are light green, Crismon Snow apples with dark red skin, Braeburn apples whose skin colors vary from orange to red and above yellow, Rome Beauty Apples, which has shiny red color, Idared Apples with a layer bright red over a green-red skin, Golden Delicious apples with light green to golden yellow with small lenticels (spots), Cortland apples which are bright red, deep red, sometimes green, blushing, and many other types. Several types of apples, such as Roman Beauty, Idared, Cortland, and Golden Delicious apples, are usually provided in the form of applesauce [3].

### B. CNN

CNN is known as a deep learning method because of its depth. Deep learning can be called a branch of machine learning intended for computers to do human work, for example computers can learn from the training process. CNN can be used to view image features. The deeper the image layer, CNN will work more complex in studying these features, then CNN can classify images according to the actual class [17]. CNN is often used to solve several problems related to computer vision, such as detecting objects, classifying images, and annotating images [18]. The network consists of three layers: convolution layer, subsampling layer, and the output layer [19].

CNN has two steps, namely the feature extraction layer and classification. The feature extraction stage consists of two parts, namely, the convolutional layer and the pooling layer. Meanwhile, the classification stage consists of flattening and fully-connected. CNN works are sequential, which means that the first convolutional layer will be used as input for the next layer. Meanwhile, the classification process consists of a fully connected and activation process (softmax), producing classification results [20].

*1) Convolution Layer:* The convolution layer is the first layer of the CNN network [19]. The convolution layer uses filter weights to separate items from the input. This Weight filter contains the weight used to detect the character of the item. Layer convolution will result in a linear transformation of the input image. This layer requires stride to perform a convolutional operation. It helps control the flow of useful data across the network with information repetition and computing power [21].

*2) Pooling Layer:* There are two sorts of Pooling, namely Max Pooling and Average Pooling. Max-pooling works to apply a pooling cover over the n x n (2 x 2) region of the input matrix and then select the highest value from that area to be output [22]. Whereas Average Pooling is the movement of n x n (2 x 2) size kernels across the matrix, each average or average position is taken from all values and entered into the output matrix.

*3) Flatten:* Flatten is used to convert the feature map into a multidimensional array and then a vector to be used as input for the fully connected layer.

*4) Fully Connected layer:* A Fully connected layer is connected with the output of the previous layer. This layer is ultimately connected to the previous layer. This layer mathematically adds the previous layers' weights to predict the class label. This layer is the last layer of CNN [21].

*5) Softmax Activation Function:* The Softmax function is the last layer used to get the classification result [23]. Softmax activation is usually done at the last layer in the CNN. The activation function generates a value interpreted as the probability that has not been normalized for each class.

### C. k-fold cross-validation

A *k*FCV is the process of dividing data into *k* folds of the same sample size. The data distribution in the *k*FCV process based on k-1 from the dataset section is called train data or training data, and the rest as test data or test data. The process is carried out as much as *k* [24]. The advantage of using this method is that all the data in the dataset has been used as training data and test data at least once [25].

### D. Confusion Matrix

The evaluation will be carried out using the Confusion matrix method. With the confusion matrix, the values for accuracy, precision, recall, and f1-score will be obtained. Below is a table of possible predictions [26].

- True Positive (TP) is if the data is classified correctly, it also comes out as valid values.
- False Positive (FP) is if the data is classified incorrectly, but the output is the correct value.
- False Negative (FN) is if the data is classified incorrectly and the output is also wrong.
- True Negative (TP) is if the data is classified appropriately, but the output is wrong.

### E. Data Augmentation

Data augmentation is one way that can be done to reduce overfitting in the model [27]. Data augmentation is the process of modifying an image so that the computer will detect that the changed image is a different image. Data augmentation was done to increase the size and quality of the limited training dataset. Data augmentation including rotation, flip, scaling, and brightness adjustment [28]. The use of augmentation can be seen in Fig. 1.

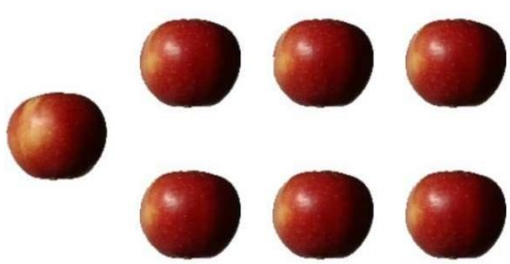

Fig. 1 Augmentation by rotating the image horizontally

## IV. Developed System

In the chapter above, related studies have been described regarding the theories applied in this chapter. This section will explain how the system is built and a more detailed explanation regarding the stages to be used. The stages that will be applied to the system are One, the preprocessing stage of the dataset. The preprocessing stage that will be carried out resizes the dataset size—two, dividing the dataset into two parts, namely train data and test data. The augmentation process will be carried out on the data train, and then the train data will enter the CNN Model training. The training results will be used to test the data testing until the accuracy is finally obtained. This study uses the softmax activation function to classify the test data from Fruit-360. Several previous studies implemented VGG16, Tanh, and compared the faster R-CNN with SSD300 and SSD500.

### A. Pre-processing

In this process, the image input will be processed in two stages: resizing and cropping processes. In the cropping process, the image pixel size will be cut. This cropping process is done so that all images are the same size. After resizing, the image dataset size will be 100x100 pixels so that the training process does not take too long.

### B. Data Split

This research performs five scenarios with different data split. This study applies a *k*FCV with k = 5. From a total dataset of 1856, five dataset divisions will be carried out as follows:

- Scenario 1: 70% Training (1299) - 30% Testing (557)
- Scenario 2: 60% Training (1113) - 40% Testing (743)
- Scenario 3: 50% Training (928) - 50% Testing (928)
- Scenario 4: 40% Training (743) - 60% Testing (1113)
- Scenario 5: 30% Training (557) - 70% Testing (1299)

### C. Dataset

The dataset from Kaggle is provided in .jpg or .jpeg format, grouped according to each class. Three classes used in this research are Braeburn Apples, Crimson Snow Apples, and Pink Lady Apples, as illustrated in TABLE. I.

TABLE. I LIST OF THREE CLASSES IN THE DATASET

| No. | Class | Total | Image |
|---|---|---|---|
| 1. | Braeburn Apples | 656 |  |
| 2. | Crimson Snow Apples | 592 |  |
| 3. | Pink Lady Apples | 608 |  |

### D. CNN classification

In the image input section, the sized image will be inserted 100x100x3. When the input image has a different size, it must first be resized at the preprocessing stage. The layer convolution used is to increase the total filter in the next layer.

In several experiments that have been carried out, the first layer used a filter value of 8, then increased in the next layer to 16 filters. However, the accuracy results obtained are still not good at detecting test data. In the next experiment, the last layer was added with a filter value of 32. In this experiment the results obtained were quite good compared to the previous experiment. Softmax is applied in the classification process.

Based on these experiments, it is concluded that the model in the apple image classification using the dataset from Kaggle uses three convolution layers with filter values in the first layer, namely 8, and then increased in the next two layers to 16 and 32. The summary of the model can be seen in Fig. 2.

## V. RESULT AND DISCUSSION

This study's performance test was used using *k*FCVwith a *k* = 5. The following are the test scenarios that will be carried out in this study: testing five scenarios for sharing training data and testing for model robustness to noise

### A. Impact of the training set sizes

Training is carried out using image data of 100x100 size with an epoch value of 100 and a total depth of 3 layers with a value of 8, 16, 32. The first training with a data comparison of 70% train and 30% test yields an accuracy of 97.84%, and the loss value is 0.08. This research conducted the second training with a data comparison of 60% train and 40% test, resulting in an accuracy of 96.74% and a loss of 0.09. This research conducted the third training with a data comparison of 50% train and 50% test resulting in an accrual value of 95.88% and a loss value of 0.13. This research conducted the fourth training with a data comparison of 40% train and 60% test resulting in an accrual value of 96.69% and a loss value of 0.16. Finally the fifth training with a data comparison of 30% train and 70% test resulting in an accrual value of 94.44% and a loss value of 0.22. This research will predict all

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_11 (Conv2D)           (None, 98, 98, 8)         224
_________________________________________________________________
max_pooling2d_11 (MaxPooling (None, 49, 49, 8)         0
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 47, 47, 16)        1168
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 23, 23, 16)        0
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 21, 21, 32)        4640
_________________________________________________________________
max_pooling2d_13 (MaxPooling (None, 10, 10, 32)        0
_________________________________________________________________
flatten_5 (Flatten)          (None, 3200)              0
_________________________________________________________________
dense_10 (Dense)             (None, 64)                204864
_________________________________________________________________
dense_11 (Dense)             (None, 3)                 195
=================================================================
Total params: 211,091
Trainable params: 211,091
Non-trainable params: 0
_________________________________________________________________
```

Fig. 2 Summary model CNN


Fig. 3 Accuracy of the training set

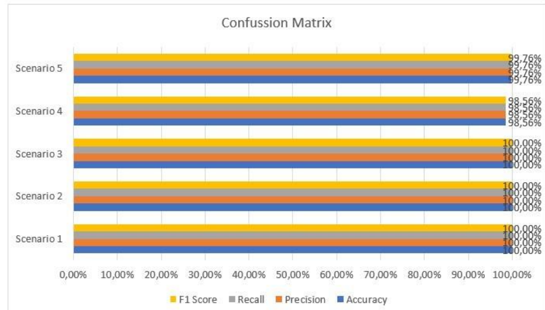


Fig. 4 Confusion matrix

scenarios with two testing data types: noise testing data and noise-free testing data.

In this research, the training is carried out using five scenarios with different training set sizes. The first scenario uses a portion ratio of 70% training and 30% testing sets. The model reaches the highest accuracy and loss of 99.82% and 0.02, respectively, on Fold 2; and gets the lowest accuracy and loss of 93.35% and 0.24, respectively, on Fold 4. With a ratio of training data and testing data of 60% training and 40% testing sets, the model reaches the highest accuracy and loss of 98.78% and 0.06, respectively, on Fold 3; and the lowest accuracy and loss of 95.28% and 0.13, respectively, on Fold 5. The third scenario uses the same portion of the training and the testing sets of 50% and 50%.

TABLE. II ACCURACY OF THE PROPOSED MODEL FOR FIVE SCENARIOS WITH DIFFERENT TRAINING SET SIZES

| | Scenario 1 | Scenario 2 | Scenario 3 | Scenario 4 | Scenario 5 |
|---|---|---|---|---|---|
| Fold 1 | 100% | 100% | 100% | 98.56% | 93.22% |
| Fold 2 | 100% | 100% | 100% | 95.86% | 92.60% |
| Fold 3 | 100% | 100% | 99.89% | 95.05% | 94.45% |
| Fold 4 | 100% | 100% | 100% | 97.66% | 96.45% |
| Fold 5 | 100% | 100% | 100% | 92.72% | 99.76% |
| Average | 100% | 100% | 99.97% | 95.97% | 95.29% |

TABLE. III ACCURACY OF THE MODEL FOR THE TESTING SETS ADDED FIVE DIFFERENT NOISE LEVELS

| | Noise 10% | Noise 20% | Noise 30% | Noise 40% | Noise 50% |
|---|---|---|---|---|---|
| Fold 1 | 100% | 95.04% | 87.71% | 81.35% | 78.01% |
| Fold 2 | 100% | 100% | 95.04% | 92.24% | 85.99% |
| Fold 3 | 100% | 79.31% | 45.47% | 38.36% | 33.18% |
| Fold 4 | 96.22% | 91.48% | 85.23% | 80.49% | 76.83% |
| Fold 5 | 99.67% | 86.42% | 77.15% | 61.42% | 45.68% |
| Average | 99.17% | 90.45% | 78.12% | 70.77% | 63.93% |

The model reaches the highest accuracy and loss of 97.30% and 0.08, respectively, on Fold 2; and gets the lowest accuracy and loss of 93.64% and 0.20, respectively, on Fold 3. With a ratio of training and testing sets of 40:60, the fourth scenario reaches the highest accuracy and loss of 98.20% and 0.08, respectively, on Fold 1; and achieves the lowest accuracy and loss of 94.79% and 0.22 for Fold 5. Finally, with a ratio of 30:70, the fifth scenario gets the highest accuracy and loss of 98.38% and 0.13, respectively, on Fold 5; and yields the lowest accuracy and loss of 91.38% and 0.24, respectively, on Fold 1.

TABLE. II shows that Scenario 1 and 2 has the highest accuracy compared to scenarios 3, 4, and 5. In Scenario 1, Fold 2 gets the highest accuracy from the 557 test data and successfully predicts 557 images. Even in the Fold with the lowest accuracy, it still manages to predict 557 images correctly. Likewise, with scenario 2, Fold 3 achieves the highest accuracy and Fold 5 the lowest accuracy. These two Folds successfully predicted the 743 test data from the total 743 data correctly.

In scenario 3, the system starts predicting a bit of the test data incorrectly. Fold 2 gets the highest accuracy from the 928 test data and successfully predicts 928 images. Meanwhile, the lowest accuracy is obtained in Fold 3 that successfully predicts 927 images correctly. The lowest class prediction is in the pink lady apple class, with 310 correctly predicted from the total image (311). In Scenario 4, using 1113 test data, Fold 1 gets the highest accuracy and successfully predicts 1097 images. Meanwhile, the lowest accuracy is obtained at Fold 5 that successfully predicts 1032 images. The lowest prediction comes from the Pink Lady Apple class, where 328 of 371 images are correctly predicted. Using 1299 test data in scenario 5, Fold 5 gets 1296 predicted images correctly. Meanwhile, the lowest accuracy is obtained in Fold 2 that successfully predicts 1203 images. The lowest prediction is in the Crimson Snow Apple class, where 368 of 433 images are correctly predicted. It can be seen from the explanation of the test results above, for testing the sharing of training data, scenarios 1 and 2 are the best scenarios because they successfully predict all apple data according to their type and get 100% accuracy values in all folds.

Using Scenario 3, the model successfully predicts almost all the test data since the dataset is clear to recognize. Therefore, in the next subsection V-B, the model is then tested using noisy images with different noise levels to examine the model robustness.

### B. *Model robustness to noisy images*

The model produced by Scenario 3 described in subsection V-A is then examined to predict two apple datasets: the original dataset (without noise) and the dataset added with Gaussian noises using five different levels of 10%, 20%, 30%, 40%, and 50%. The result in TABLE. III shows that the model gives a high result of 99.17% for the low-level noise of 10%. Its performance drastically decreases to 90.45% for the noise level of 20% and quickly declines for noise level 30%, 40%, and 50%. It can be seen in the explanation of the results of testing the model's resistance to noise above, by adding noise by 10%, the model still gets an accuracy value above 99%, which means the model still works very well at this figure, but when noise is added by 50%, the accuracy value fell drastically to 63.93% and the model started to be wrong when predicting many apple data.

These performance decrements are caused by the noise spots that cover the texture of the test data image. Due to the high-level noise, the model starts to match the noise covered texture with the other classes testing images. The noisy testing images significantly affect the model since it does not use a regularization technique commonly applied to avoid overfitting.

### C. *Comparison to other models*

The developed model was finally compared with three previous research. All researches used the fruit-360 dataset.

Research by Horea Muresan et al. (2018) used 38409 images. From the total dataset, 75% was used as train data and 25% as test data and produced a training result of 100% and testing result of 96.3%. Research by Shadman Sakib et al. (2019) used 17823 images. This study used a ratio of 80% for training data and 20% for testing data and produced a testing accuracy of 100% and a training accuracy of 99.79%. Research by Dang Thi Phuong Chung et al. (2019) used 17624 images. From the total dataset, 75% was used for train data and 25% for test data. This study produced a testing accuracy of 98% and a training accuracy of 96.79%.

The three studies above discuss the classification of fruit images using one comparison scenario, unlike this study which uses five dataset comparison scenarios and checks the robustness of the model against the noise image. The three studies above produce good training and testing accuracy values for only one dataset comparison, unlike this study where all comparison scenarios produce good training and testing accuracy, which can be seen in Fig. 3 and TABLE. II.

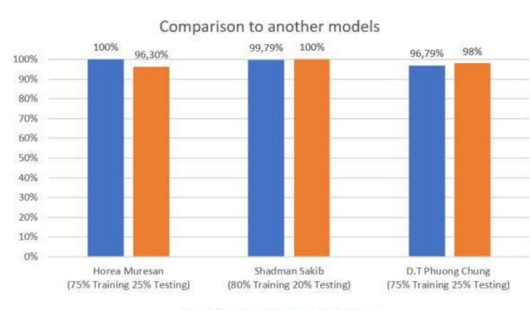


Fig. 5 Comparison to another models

## VI. Conclusion

A CNN-based classification model of apple images into five classes has been developed. A 5-FCV analysis informs that CNN is robust to decreasing the portion of training set size up to 50% to get high accuracy of 99.97% in classifying 50% testing set. This result is higher than the previous models of VGG16, faster R-CNN, Tanh and better than the existing model in comparison research. The model in the comparison research produced good training accuracy values for only one dataset comparison scenario, unlike the model in this study which produced good training accuracy values for the five dataset comparison scenarios. The three dataset comparison scenarios in this study also managed to get a testing accuracy value of 100%. Decreasing the portion training set to 40% and 30% reduces the accuracy to 95.97% and 95.29%. Correctly selecting the dataset is crucial in the process of classifying apple images. If the dataset is clean, then the model can predict all images correctly. Hence, adding a high-level noise of 50% to the testing set significantly reduces the CNN accuracy to 63.93%. In the future, advanced augmentation and regularization methods can be incorporated to tackle this problem.

## References


[1] M. Akbari, M. Yamaguchi, T. Maejima, S. Otagaki, K. Shiratake, and S. Matsumoto, "Apple cultivation and breeding in Afghanistan: S - RNase genotypes and search system for suitable cultivar combination," *Int. J. Agron.*, vol. 2016, 2016, doi: 10.1155/2016/3101864.

[2] J. Boyer and R. H. Liu, "Apple phytochemicals and their health benefits," *Nutr. J.*, vol. 3, no. 5, 2014, doi: 10.1186/1475-2891-3-5.

[3] K. Wolfe, X. Wu, and R. H. Liu, "Antioxidant activity of apple peels," *J. Agric. Food Chem.*, vol. 51, no. 3, pp. 609–614, 2003, doi: 10.1021/jf020782a.

[4] D. T. P. Chung and D. Van Tai, "A fruits recognition system based on a modern deep learning technique," *J. Phys. Conf. Ser.*, vol. 1327, no. 1, 2019, doi: 10.1088/1742-6596/1327/1/012050.

[5] C. S. Janadri, B. G. Sheeparamatti, and V. Kagawade, "Multiclass classification of kirlian images using SVM technique," *2017 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2017*, vol. 2017-Janua, pp. 2246–2250, 2017, doi: 10.1109/ICACCI.2017.8126180.

[6] H. Mureşan and M. Oltean, "Fruit recognition from images using deep learning," *arXiv*, vol. 1, pp. 26–42, 2017, doi: 10.2478/ausi-2018-0002.

[7] Y. Peng, M. Liao, W. Huang, H. Deng, L. Ao, and J. Hua, "Fruit Fly Classification via Convolutional Neural Network," *Proc. 2018 Chinese Autom. Congr. CAC 2018*, pp. 3395–3399, 2019, doi: 10.1109/CAC.2018.8623178.

[8] P. Kunakornvong and D. M. Asriny, "Fauna Image Classification using Convolutional Neural Network," *Int. J. Futur. Gener. Commun. Netw.*, vol. 13, no. June 2019, pp. 8–16, 2020.

[9] A. K. Das, M. A. Iqbal, B. Paul, A. Rakshit, and M. Z. Hasan, "Classification of succulent plant using convolutional neural network," *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng. LNICST*, vol. 325 LNICST, no. January, pp. 695–704, 2020, doi: 10.1007/978-3-030-52856-0_55.

[10] M. G. Lim and J. H. Chuah, "Durian types recognition using deep learning techniques," *2018 9th IEEE Control Syst. Grad. Res. Colloquium, ICSGRC 2018 - Proceeding*, no. August, pp. 183–187, 2019, doi: 10.1109/ICSGRC.2018.8657535.

[11] S. Sakib, Z. Ashrafi, and M. A. B. Sidique, "Implementation of fruits recognition classifier using convolutional neural network algorithm for observation of accuracies for various hidden layers," *arXiv*, pp. 8–11, 2019, doi: 10.13140/RG.2.2.31636.14723.

[12] A. Kausar, M. Sharif, J. Park, and D. R. Shin, "Pure-CNN: A framework for fruit images classification," *Proc. - 2018 Int. Conf. Comput. Sci. Comput. Intell. CSCI 2018*, pp. 404–408, 2018, doi: 10.1109/CSCI46756.2018.00082.

[13] P. Kunakornvong and D. M. Asriny, "Apple image classification using Convolutional Neural Network," *34th Int. Tech. Conf. Circuits /Systems, Comput. Commun.*, no. June 2019, 2019.

[14] Q. Liang, J. Long, W. Zhu, Y. Wang, and W. Sun, "Apple recognition based on Convolutional Neural Network Framework," *2018 13th World Congr. Intell. Control Autom.*, vol. July, no. 4–8, pp. 1751–1756, 2018, doi: 10.1109/WCICA.2018.8630705.

[15] L. I. Sun, K. Liang, Y. Song, and Y. Wang, "An Improved CNN-Based Apple Appearance Quality Classification Method With Small Samples," vol. 9, 2021, doi: 10.1109/ACCESS.2021.3077567.

[16] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019, doi: 10.1109/ACCESS.2019.2914929.

[17] S. Sugianto and S. Suyanto, "Voting-Based Music Genre Classification Using Melspectogram and Convolutional Neural Network," in *2019 2nd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2019*, Dec. 2019, pp. 330–333, doi: https://doi.org/10.1109/ISRITI48646.2019.9034644.

[18] A. A. Nugraha, A. Arifianto, and Suyanto, "Generating Image Description on Indonesian Language using Convolutional Neural Network and Gated Recurrent Unit," in *2019 7th International Conference on Information and Communication Technology (ICoICT)*, Jul. 2019, pp. 1–6, doi: https://doi.org/10.1109/ICoICT.2019.8835370.

[19] K. . S. Deepika Jaswal, Sowmya.V, "Image classification using convolutional neural networks with multi-stage feature," *Adv. Intell. Syst. Comput.*, vol. 345, no. 6, pp. 587–594, 2015, doi: 10.1007/978-3-319-16841-8_52.

[20] A. L. Katole, K. P. Yellapragada, A. K. Bedi, S. S. Kalra, and M. S. Chaitanya, "HIERARCHICAL DEEP LEARNING ARCHITECTURE FOR 10K OBJECTS CLASSIFICATION," *Comput. Sci. Inf. Technol. (CS IT)*, vol. September, pp. 77–93, 2015, doi: 10.5121/csit.2015.51408.

[21] A. G. Alharbi and M. Arif, "Detection and classification of apple diseases using convolutional neural networks," *2020 2nd Int. Conf. Comput. Inf. Sci. ICCIS 2020*, pp. 1–5, 2020, doi: 10.1109/ICCIS49240.2020.9257640.

[22] Rismiyati and S. N. Azhari, "Convolutional Neural Network implementation for image-based Salak sortation," *Proc. - 2016 2nd Int. Conf. Sci. Technol. ICST 2016*, pp. 77–82, 2017, doi: 10.1109/ICSTC.2016.7877351.

[23] K. Z. Thet, K. K. Htwe, and M. M. Thein, "Grape Leaf Diseases Classification using Convolutional Neural Network," *Proc. 4th Int. Conf. Adv. Inf. Technol. ICAIT 2020*, pp. 147–152, 2020, doi: 10.1109/ICAIT51105.2020.9261801.

[24] Y. Jung and J. Hu, "A K -fold Averaging Cross-validation Procedure," *J. Nonparametr. Stat.*, vol. 27, no. 2, pp. 1–13, 2015, doi: 10.1080/10485252.2015.1010532.

[25] K. S. Raju, M. R. Murty, M. V. Rao, and S. C. Satapathy, "Support Vector Machine with K-fold Cross Validation Model for Software Fault Prediction," *Int. J. Pure Appl. Math.*, vol. 118, no. 20, pp. 321–334, 2018, [Online]. Available: https://acadpubl.eu/hub/2018-118-21/articles/21b/36.pdf.

[26] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[27] J. Yang, Y. Zhao, J. C. W. Chan, and C. Yi, "Hyperspectral image classification using two-channel deep convolutional neural network," *Int. Geosci. Remote Sens. Symp.*, vol. 2016-Novem, pp. 5079–5082, 2016, doi: 10.1109/IGARSS.2016.7730324.

[28] J. Li *et al.*, "A Shallow Convolutional Neural Network for Apple Classification," *IEEE Access*, vol. 8, pp. 111683–111692, 2020, doi: 10.1109/ACCESS.2020.3002882.