Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

TRANSFER LEARNING WITH VGG16 AND INCEPTIONV3 MODEL FOR CLASSIFICATION OF POTATO LEAF DISEASE

[1]DEWI ROSMALA, [2]MOCHAMMAD REVALDI PRAKHA ANGGARA, [3]JUNIARTI P. SAHAT
[1,2]Department of Informatics, Institut Teknologi Nasional, Bandung, Indonesia
[3]Indonesian Vegetable Research Institute (IVEGRI), Bandung, Indonesia
E-mail: [1]d_rosmala@itenas.ac.id, [2]revaldiangga@gmail.com, [3]juniartisahat@gmail.com

ABSTRACT

Early diagnosis of plant diseases carried out by experts in laboratory tests is often not applicable for fast and inexpensive implementation. Using deep learning, leaf images are used as data input. Training deep learning models require large, hard-to-come datasets to perform the task to achieve optimal results. In this study, the PlantVillage dataset was used totaling 2700 training data and 300 validation data. Data were trained using 100 epoch iterations using the transfer learning method with the VGG16 and InceptionV3 models. At the top layer of both models, the same MLP is applied to several parameters, namely the size of FC and the dropout rate to compare the model's performance. Based on testing using 150 IVEGRI data, the VGG16 model can generalize data better than InceptionV3. VGG16 by tuning block-3 using parameters 4096x2 and Dropout 0.4 shows the best performance with an average score of 1 precision, an average recall of 1, an average f1-score of 1, and 100% accuracy. Then, with the same parameters, the Inception-v3 model with tuning in the mixed6 inception module shows the best performance with an average score of 0.93 precision, an average recall of 0.92, an average f1-score of 0.92, and an average accuracy of 92%.

Keywords: *Deep Learning, Transfer Learning, VGG16, InceptionV3, Potato Leaf Diseases Classification*

1. INTRODUCTION

Potato plants are the most common plants in the highlands of Indonesia with a slope of 800 - 1500 meters above sea level (masl) [1]. Potato (Solanum tuberosum L) is a vegetable commodity that has a high economic value [2]. Dimyati, et.al [3] stated that in Southeast Asia, Indonesia occupies the first position as the largest potato producer and is in the second position after China among priority countries in the International Potato Center CIP in the East Asia region, Asia. Southeast, and the Pacific. One of the potato developments in Indonesia is located at the Balai Penelitian Tanaman Sayuran, Jl Tangkuban Perahu, Lembang, Kabupaten Bandung. Plants are generally attacked by leaf disease. Leaf disease is one of the problems faced by farmers which can reduce production yields. One of the main diseases affecting potato plants is late blight and early blight. Diseased leaves should be handled as early as possible so as not to spread to other plants. If the classification process is still done manually, the classification stage may be very expensive and it also depends on the farmer's subjectivity which sometimes occurs human errors.

The classification of plant diseases is a very important job for producers or farmers. By using deep learning for image-based leaf disease classification, plant diseases can be detected accurately and at an affordable cost, because to classify using experts may be very expensive. Classification using vision technology has been widely used to increase accuracy and reduce work costs. The Convolutional Neural Network method is proposed by many researchers to classify data in the form of images with large data input. Examples of CNN models that are widely used by researchers today are AlexNet [4], VGGNet [5], GoogleNet [6], Inceptionv3 [7], ResNet [8], and DenseNet [9] followed. All of these CNN algorithms are the result of the ImageNet Large Scale Visual Recognition Challange (ILSVRC) competition which annually creates and searches for new algorithms for object detection and classification in recognizing ImageNet objects containing 1000 classes [10].

Training a deep learning model requires large amounts of data, large computational resources, which takes hours, in some cases days [11]. Then in practice, collecting data in the form of imagery in the agricultural world in large numbers is very difficult.

To overcome this problem, the transfer learning method is used [12].

In this study, implemented the transfer learning method using the CNN VGG16 [5] and InceptionV3 [7] models for the classification of potato leaf disease. The dataset in this study is a picture of Potato Leaf plants published by Geetharamani [13]. In this study, the upper layers of the two models were equated using 3 FC layers and 2 dropouts, where the last FC layer used 3 neurons with a softmax classifier, each unit representing the potato leaf data class category (early blight, late blight, and non-disease or healthy).

This study aims to implement the transfer learning method that can produce the best performance for classifying 3 classes of potato leaf disease with a fine-tuning approach using the parameters of the number of neurons (Size of FC) and the dropout rate. And the implementation is expected to be applied to applications that can cut algorithms from producers to experts to classify types of plant diseases, especially potatoes, because providing experts may be very expensive.

2. RELATED WORK

Research related to the classification of plant diseases has been carried out by several previous studies. Rakhmawati et al [1] used 300 training images and 90 test images consisting of 3 classes of potato leaves using GLCM and Color Moment for feature and color extraction and then SVM classifier. The results of this study get an accuracy of up to 80%. Also, in the study, it was concluded that the pattern of diseased leaves greatly influenced identification. Therefore, the selection of non-disease or healthy leaves should be more selective with no spots at all. Then other research related to plant diseases was discussed by Arya [14]. This study uses 2 architectures, namely CNN with 3 layers of feature learning and 2 layer classification and the AlexNet model with transfer learning. The dataset used in this study is from the PlantVillage website and is taken in real-time from the GBPUAT plantation. The total data amounted to 4004 consisting of 4 classes, 2 classes of mango plants, and 2 classes of potato plants. 80:20, 3523 random splits were performed for training data and validation, while 481 was for testing. The results of the study obtained an accuracy of 90.85% for CNN and 98.33% for AlexNet. Research on grape leaves using transfer learning was conducted by Gangwar et al [15] using the InceptionV3 model with SVM classifier, logistic regression, and neural network. This study uses a PlantVillage dataset totaling 4062 consisting of 4 classes where the training data is 3209 and the test data is 853. The best classification results for InceptionV3 with logistic regression classifier achieve an accuracy of 99.4%. Agarawal, M. et al [16] conducted a study on potato leaves with a PlantVillage dataset of 3000 datasets using the caviar approach [17] to train the model. that. The training accuracy was 99.47% and testing accuracy was 98% using the CNN architecture using four layers with 32, 16, and 8 filters in each layer against 150 test data.

3. THE PROPOSED WORK

In this study, there are several stages to identify potato leaf disease. Figure 1 shows the general stages in this study to classify potato leaf disease. At the training stage, the transfer learning method uses the pre-trained VGG16 and Inceptionv3 models found in the hard repository. Both models have previously been trained using the ImageNet large dataset, where previously trained models on the ImageNet large dataset can help solve computational problems and relatively few datasets at the training stage [18].

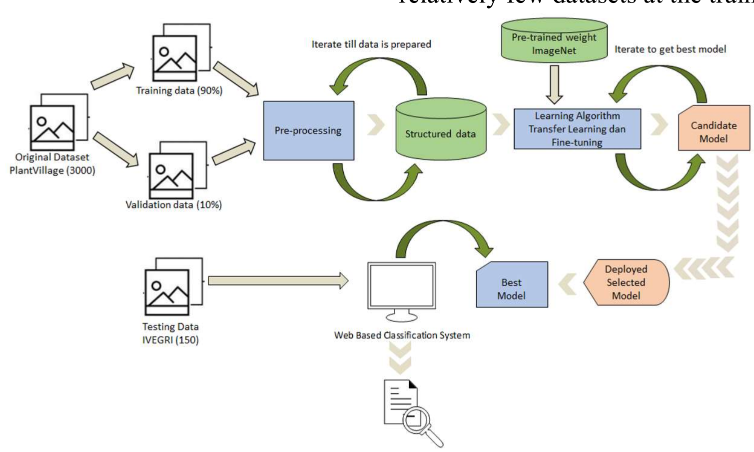

Figure 1. Block diagram of the proposed potato leaf disease classification system research

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.1 Data Pre-processing

In the training and testing stages, the image input size must be the same as the input size applied to the model. In this study, the image is resized to 224x224. Then scale the pixel value (between 0 and 255) to the interval [0, 1], this operation is because neural networks prefer to handle small input values [18].

2.2 Data Augmentation

To generalize the data, data augmentation is applied during the training process. Data augmentation is an operation to prevent overfitting by generating more training data. The goal is that the model will not see the same image twice and be able to adapt to real-world problems. This operation is performed using Keras with the ImageDataGenerator function by applying a transformation operation consisting of rotation, width and height shifting, zoom, and horizontal flip. Augmentation was performed on training data and not performed on data validation [18].

4.1 Transfer Learning

Transfer learning is a learning technique method using pre-trained neural networks by taking part in a model that has been trained to be reused in recognizing new models [12]. As shown in Figure 2, there is a source task and a target task, the knowledge gained from the source task model training process is transferred to handle the target task, in this case, the trained weight model from a large dataset of ImageNet images is transferred to recognize potato leaf disease.

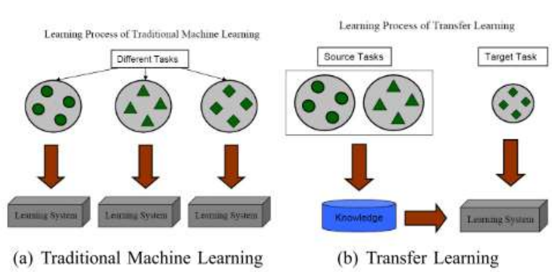

Figure 1. Differences in traditional learning and transfer learning

By using transfer learning, there is no need to train a model with a large amount of data because the parameters transferred from the source model (ImageNet) allow the model to predict accurately even when using a small training data set. Also, transfer learning can reduce training complexity and shorten neural network training time. Figure 3 illustrates the scenario of transfer learning in tuning which is described in the literature [19].

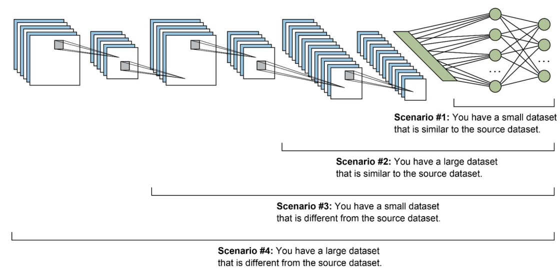

Figure 2. guidelines for the appropriate fine-tuning level to use in each of the four scenarios

4.1.1 Transfer Learning VGG16

VGG16 was developed by the Visual Geometry Group at the University of Oxford [5]. This model won the ILSVRC contest as the second winner in image classification and the winner of image localization in 2014 in recognizing 1000 ImageNet object classes. VGG16 is designed to reduce the large kernel size on AlexNet 1x1 and 5x5 which is replaced by several 3x3 kernels with 1 stride which is useful for extracting complex features with low computation. VGG16 has 5 convolutional blocks consisting of 2 to 3 convolutional layers with ReLu activation. At the end of each block, MaxPooling 2x2 with 2 strides is used. Further descriptions of the VGG16 architecture have been described in the literature [5].

In this study, the pre-trained VGG16 convolutional base was maintained, then after the 5th block convolution layer, Global Average Pooling was used. The top layer in this study was modified using three fully connected layers with dropouts. where the last fully connected layer uses the softmax classifier. Figure 4 shows the VGG16 architectural model proposed in this study.

4.1.2 Transfer Learning InceptionV3

The Inception-v3 model is the development of the GoogleNet or Inception-v1 model developed in research [6] which won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) contest in the image classification category in 2014. Then it was refined by adding batch normalization (BN) called Inception-v2 model. With the development of additional factorization ideas for convolutional operations, this architecture is named after the Inception-v3 model which won the first runner-up in the ISLVRC contest in 2015 in recognizing 1000 ImageNet objects classes. Inception-v3 consists of 5 basic convolutional layers (stem) consisting of conv2d_0 to conv2d_4 where each convolutional

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

operation is followed by ReLu activation and BatchNormalization. Then followed by 11 inception modules consisting of mixed0 modules to mixed10 modules. The 11 Inceptionv3 module blocks are designed with 1x1, 3x3, 1x3, 3x1, 5x5, 1x7, and 7x7 convolutional kernels.

In this study, the Inceptionv3 convolutional base was maintained, while the upper layer in this study was modified consisting of GAP and three dense layers where the last dense layer used the softmax classifier. A complete description of Inceptionv3 is described in the literature [7]. In Figure 5 is the InceptionV3 architectural model proposed in this study.

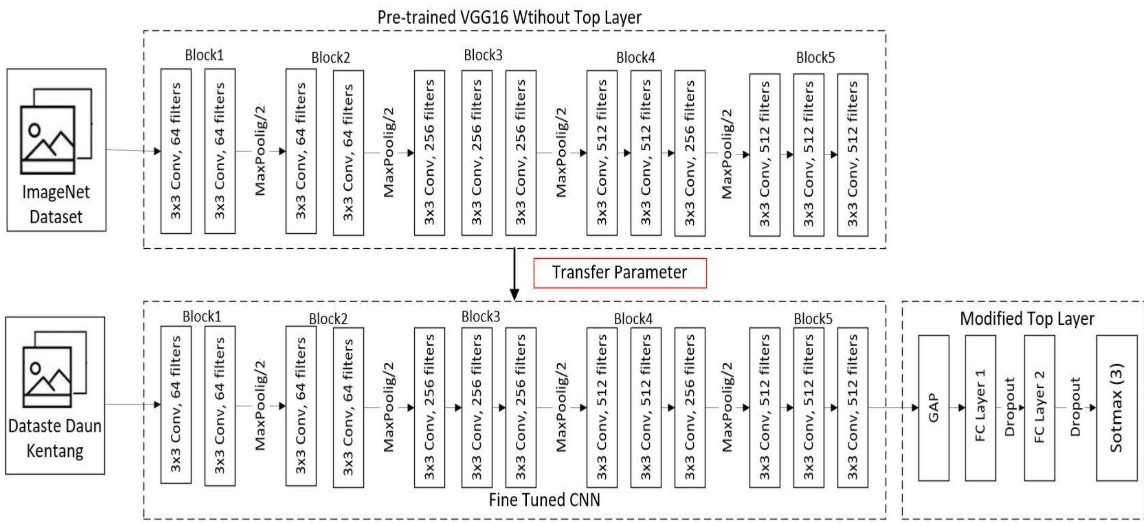

Figure 4. Transfer learning architecture of VGG16 with the modified top layer

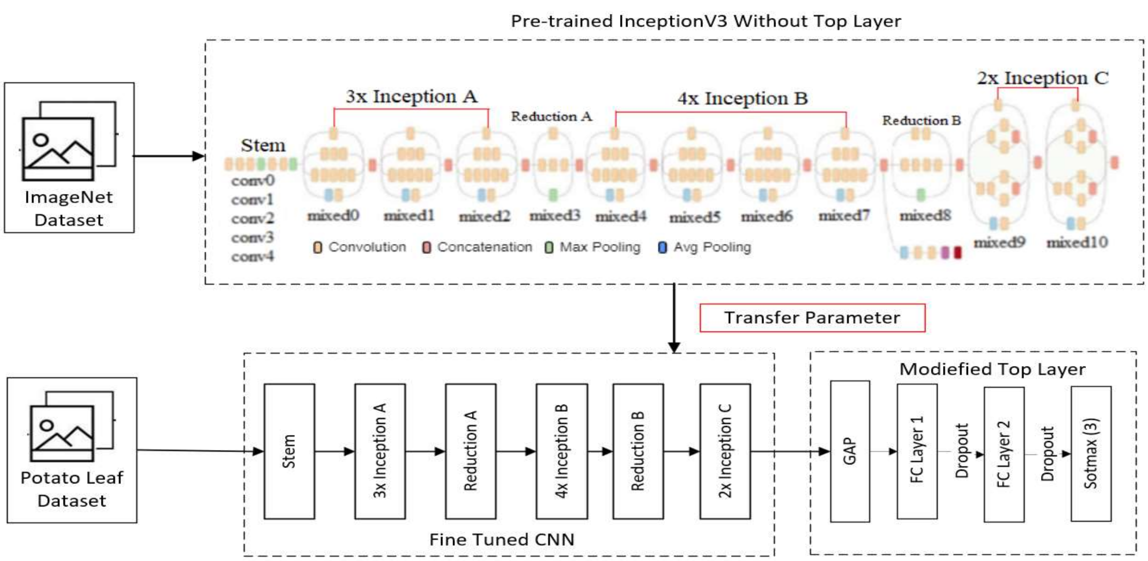

Figure 5. Transfer Learning architecture InceptionV3 with the modified top layer

4.2 Dropout

Dropout is a regularization technique that can help CNN withstand overfitting and also speed up the training process [20]. The way it works is to temporarily remove hidden layers as well as visible layers that are randomly located in the network and redirect to more trained neurons to reduce interdependence learning in each neuron [19]. Figure 6 below is an example of an Artificial Neural Network before and after the dropout process.

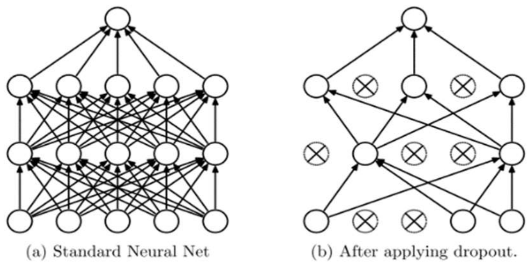

Figure 6. Dropout

4.3 Performance Measure

To evaluate the performance of the classification model and object recognition in machine learning, deep learning, and information retrieval, evaluation based on precision, recall, f1-score, and accuracy is used to test the model performance of the test dataset test process [21].

Accuracy is the ratio of correct predictions to the overall data. To calculate accuracy, we use Equation 1. precision is the part of the object that is predicted to be correct, calculated using Equation 2. Recall / Sensitivity is used to find out how accurate is the model's performance to classify correctly, or in other words, how many times the model misclassifies false negative using Equation 3. The F1-score is the comparison of the mean precision and recall, in other words summarizing the classification performance with a single metric representing precision and recall. The F1-score is calculated using Equation 4.

$$\text{Acc} = \frac{\textit{correctly predicted samples}}{\textit{total samples in the set}} \text{x}100 \tag{1}$$

$$\text{Precision} = \frac{\sum_{i=1}^{l} TP_i}{\sum_{i=1}^{l}(TP_i + FP_i)} \tag{2}$$

$$\text{Recall} = \frac{\sum_{i=1}^{l} TP_i}{\sum_{i=1}^{l}(TP_i + FN_i)} \tag{3}$$

$$F_1 score = 2 \text{ x} \frac{(\text{presisi x recall})}{(\text{presisi+recall})} \tag{4}$$

Where TP (True Positive) is positive data that is proven true, FP (False Positive) is negative data that is proven true, TN (True Negative) is positive data that is not proven true, and FN (False Negative) is negative data that is not proven right.

5 EXPERIMENTS AND RESULT

In this section, the results obtained are displayed based on the method used. The first machine used for training has specifications of a Xeon processor, 25 GB RAM, Nvidia Tesla P400, and storage of 147 GB from the colab.research.google.com server [22]. The second machine used to test the model has the 7th generation Intel i5 specification, 12GB RAM, Nvidia GeForce 940MX, and 1 TB HDD memory specification. Experiment using Tensorflow and Keras to implement the VGG16 and Inceptionv3 pre-trained models. Stochastic Gradient Descent Optimizer with a learning rate of 0.0001, batch size 32, and 100 epoch were used to train both models.

5.1 Dataset

This study uses a PlantVillage training dataset obtained from research [10]. The image used in this research is a potato leaf that has an image resolution of 256x256. The training data amounted to 3000 with 3 classes, each class consisting of 1000 pictures. Training data is divided into 90:10 for training data and validation data. While the test data is data obtained from the Research Group on Germplasm Breeding and Germination, the Indonesian Vegetable Research Institute, totaling 150 data taken using a Nikon D3200 DSLR camera and smartphone from various angles and using white cardboard as a background. The amount of data in this study is shown in Table 1 and Table 2.

Table 1. Training and validation data

Nama Class	Sample of image	Num of training	Num. of validation
Early Blight		900	100
Late Blight		900	100
Healthy		900	100
Total		2700	300

Table 2. Testing Data

Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Class	Sample	Num. of image
Early Blight		50
Late Blight		50
Healthy		50
Total		150

5.2 Experiment Parameters

The top layer of the VGG16 and InceptionV3 models in this study was carried out by tuning trial and error on the parameters to obtain the best performance for training the data. The tuning performed on the VGG16 model is replacing Flatten with Global Average Pooling to produce one feature map for each output. Whereas in the InceptionV3 model, which originally had one FC layer, in this study tuning was carried out by adding one FC layer to the top layer so that the performance of the two models based on MLP on the same top layer could be compared.

After using GAP, two FC layers followed with dropouts. The use of GAP aims to reduce prone overfitting when using a fully connected layer and summarize spatial information to speed up the training process [23]. The fully connected layer is useful for getting a feature map then using it to classify images into the appropriate labels. The FC layer has a function to represent image features into appropriate labels, therefore this study also looks for the performance of the number of nodes/neurons in the upper layer. On the other hand, the FC layer tends to experience overfitting especially when large neural networks are trained on relatively few datasets [24]. Because of that, after the dense layer, a dropout parameter is used. This research applies dropout parameter tuning in both models starting from not applying it at all then applying dropout 0.2, 0.3, 0.4, and 0.5.

The two architectures are divided into several blocks based on Figure 4 and Figure 5. Fine-tuning of the model is done by experimenting with different numbers of neurons and dropouts. Each network is tuned backward. In VGG16, the first tuned block is the 5th block, then the backward tuning until the entire network is the 1st block. Whereas the InceptionV3 model has 11 inception model blocks, backward tuning is performed starting from the mixed10 inception module then backward tuning to the entire basic convolutional network. This adjustment is done to find at what layer the model gets the best performance or converges faster (training accuracy and validation are not corrugated).

5.3 Experimental results using IVEGRI testing data

The test in this study used the IVEGRI testing dataset with the same 3 leaf classes, namely dry spot, phytophthora blight, and non-disease or healthy leaves. The best accuracy performance of both models is obtained when using the size of the FC 4096x2 parameter with a dropout rate of 0.4. The accuracy of the VGG16 model reaches 100% when tuning the 3rd block and InceptionV3 by 92% when tuning the Inception mixed6 module. The accuracy performance of the model using the tuning layer experiment, the number of neurons, and the different dropout rates in the VGG16 and InceptionV3 models is shown in Table 3, Figure 7, and Figure 8.

The precision score performance is shown in Table 4 and Figure 9, and Figure 10. The highest precision score was obtained when tuning the VGG16 model in the 3rd block using the size of FC 4096x2 parameter and a dropout rate of 0.4, with an average precision score of 1.While the model Inception V3 got the greatest precision value when tuning the mixed6 inception module using the size of FC 4096x2 parameter and a dropout rate of 0.4, with an average precision score of 0.93.

The recall score performance is shown in Table 5, Figure 11, and Figure 12. The highest recall value obtained is the same as the parameter setting on the precision side, namely the VGG16 model when tuning in the 3rd block while Inceptionv3 in the mixed6 inception module using the size parameter of FC 4096x2 and dropout 0.4. The average recall score for VGG16 is 1 and InceptionV3 is 0.92.

From the precision and recall results, the f1-score was obtained using equation 4 and the results are shown in Table 6, Figure 13, and Figure 14. The highest average f1-score obtained in the VGG16 model is 1 and InceptionV3 is 0.92.

Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 3. Accuracy (%)

Model	Fine-tune	Parameter									
		Size of FC 2048x2					Size of FC 4096x2				
		Dropout									
		None	0.2	0.3	0.4	0.5	None	0.2	0.3	0.4	0.5
VGG16	Block5	60.67	60.56	61.33	54.67	63.00	64.33	65.33	60.67	64.00	57.00
	Block4	94.00	96.00	80.67	84.00	71.33	97.33	94.00	94.00	84.00	56.00
	Block3	96.00	88.67	93.33	94.00	97.33	94.67	99.33	97.33	**<u>100.00</u>**	96.67
	Block2	98.00	98.00	98.67	95.33	88.00	93.33	94.67	96.67	97.33	98.67
	All Layers	96.67	97.73	93.33	92.67	88.00	98.67	96.00	97.33	96.67	98.00
Inception V3	Mixed10	78.67	82.67	84.67	84.67	86.00	86.67	82.00	84.67	78.67	80.67
	Mixed9	84.67	86.67	87.33	83.33	86.00	89.33	84.67	80.67	84.67	89.33
	Mixed8	82.00	86.00	88.67	82.00	82.67	88.00	82.00	83.33	82.67	81.33
	Mixed7	86.00	84.00	84.00	81.00	84.00	85.33	88.00	84.67	87.33	82.00
	Mixed6	82.00	86.67	90.67	86.67	80.67	89.33	90.00	88.00	**<u>92.00</u>**	88.00
	Mixed5	90.67	86.67	89.33	80.67	88.00	90.67	90.67	80.67	73.33	86.67
	Mixed4	86.00	87.33	91.33	80.67	88.67	82.67	84.00	80.00	79.33	73.33
	Mixed3	82.67	88.00	85.33	88.67	72.00	85.33	84.67	88.67	80.00	83.33
	Mixed2	91.33	88.00	86.67	88.00	73.33	88.00	83.33	80.00	75.33	78.00
	Mixed1	82.67	84.67	84.67	76.67	76.00	84.00	78.67	75.33	80.00	82.00
	All Layers	74.67	80.00	82.00	77.33	66.67	82.00	72.67	77.33	74.67	74.67

The best accuracy score is marked in bold and underlined

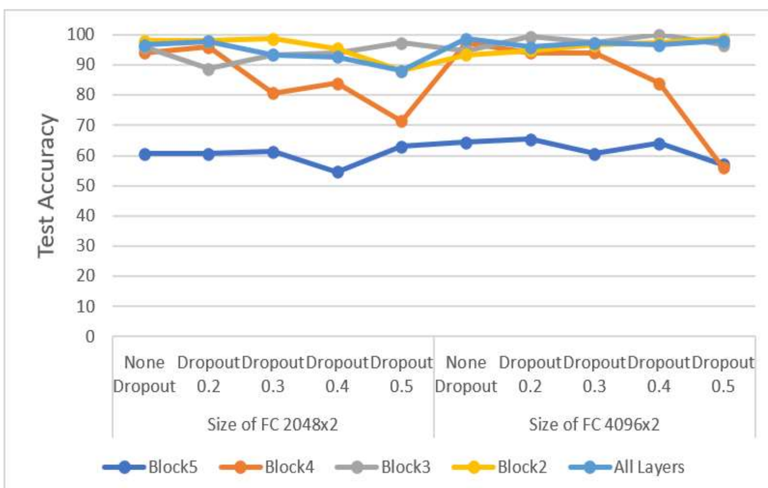

Figure 7. Graph of testing accuracy of fine-tuning VGG16

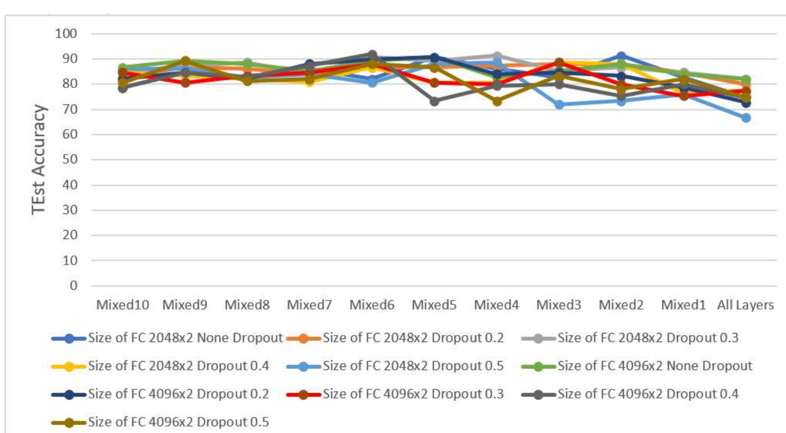

Figure 7. Graph of testing accuracy of fine-tuning InceptionV3

Tabel 4. Precision

Model	Fine-tune	Class Type	Parameter									
			Size of FC 2048x2					Size of FC 4096x2				
			Dropout									
			None	0.2	0.3	0.4	0.5	None	0.2	0.3	0.4	0.5
VGG16	Block5	EB	0.94	0.90	0.94	0.90	0.90	0.92	0.93	0.92	0.93	0.64
		LB	0.45	0.46	0.46	0.42	0.47	0.49	0.49	0.46	0.48	0.43
		H	1	1	1	1	1	1	1	1	1	1
	Block4	EB	0.98	0.98	1	1	1	0.94	1	1	1	1
		LB	0.86	0.91	0.63	0.85	0.54	0.98	0.85	0.85	0.68	0.43
		H	1	1	1	1	1	1	1	1	1	1
	Block3	EB	0.98	1	0.98	0.98	0.94	0.88	0.98	0.94	**<u>1</u>**	0.96
		LB	0.96	0.75	0.89	0.86	0.98	0.98	1	0.98	**<u>1</u>**	0.94
		H	0.94	1	0.94	1	1	1	1	1	**<u>1</u>**	1
	Block2	EB	0.98	0.94	1	0.96	0.95	0.88	0.90	0.98	0.93	1
		LB	0.96	1	0.96	0.91	0.75	0.92	0.94	0.92	1	0.96
		H	1	1	1	1	1	1	1	1	1	1
	All Layers	EB	0.92	0.98	0.98	0.98	0.77	0.98	0.94	1	1	0.98
		LB	0.98	0.89	0.94	0.83	0.90	0.94	0.98	0.92	0.92	0.96
		H	1	0.94	1	1	1	1	1	0.96	0.98	1
Inception V3	Mixed10	EB	0.97	0.96	0.91	0.96	0.93	0.95	0.98	1	0.9	0.93
		LB	0.62	0.69	0.72	0.72	0.77	0.74	0.67	0.7	0.65	0.66
		H	1	0.94	1	0.95	1	0.98	0.94	0.95	0.97	0.97
	Mixed9	EB	0.87	0.95	0.83	0.86	0.91	0.90	0.95	0.91	0.92	0.95
		LB	0.76	0.74	0.84	0.74	0.75	0.82	0.70	0.67	0.73	0.78
		H	0.97	1	1	1	1	0.87	1	1	0.97	1
	Mixed8	EB	0.89	0.87	0.85	0.87	0.88	0.87	0.93	0.91	0.85	0.80
		LB	0.69	0.77	0.84	0.70	0.70	0.81	0.68	0.70	0.72	0.72
		H	1	1	1	1	1	1	1	1	1	1
	Mixed7	EB	0.90	0.93	0.87	0.92	0.93	0.92	0.90	0.88	0.94	0.95
		LB	0.75	0.77	0.74	0.98	0.70	0.74	0.79	0.74	0.75	0.66
		H	1	1	1	1	1	1	1	1	1	1
	Mixed6	EB	1	0.94	0.96	0.87	0.94	0.96	1	0.98	**<u>0.98</u>**	0.94
		LB	0.65	0.75	0.80	0.78	0.83	0.79	0.77	0.75	**<u>0.83</u>**	0.77
		H	1	1	1	1	0.98	0.98	1	0.97	**<u>0.98</u>**	1
	Mixed5	EB	0.93	0.97	0.80	0.91	0.92	0.93	0.98	0.91	0.93	0.95
		LB	0.83	0.75	0.98	0.66	0.78	0.87	0.80	0.66	0.57	0.74
		H	0.98	0.95	0.97	0.95	0.96	0.98	0.98	0.95	1	0.98
	Mixed4	EB	0.97	0.93	0.97	1	0.95	0.97	0.97	1	0.94	0.96
		LB	0.73	0.78	0.83	0.78	0.8	0.68	0.70	0.64	0.64	0.57
		H	0.96	0.96	0.96	0.94	0.94	0.96	0.96	0.78	0.95	0.93
	Mixed3	EB	0.97	0.97	0.95	0.97	0.56	0.94	0.94	0.95	1	0.82
		LB	0.68	0.77	0.73	0.75	0.95	0.73	0.71	0.80	0.64	0.69
		H	0.95	0.96	0.96	0.98	1	0.96	0.98	0.94	0.93	0.96
	Mixed2	EB	0.98	0.97	0.94	0.96	1	0.97	0.96	0.96	0.94	0.96
		LB	0.84	0.77	0.74	0.60	0.57	0.75	0.70	0.64	0.59	0.63
		H	0.94	0.96	0.98	0.98	0.96	0.98	0.94	0.96	0.94	0.94
	Mixed1	EB	0.97	0.94	0.94	1	0.95	0.96	1	1	0.97	0.93
		LB	0.66	0.71	0.72	0.60	0.60	0.69	0.63	0.60	0.64	0.68
		H	1	0.98	0.96	1	0.96	0.98	0.94	0.93	0.96	0.96
	All Layers	EB	0.93	0.89	0.96	0.89	0.88	0.93	0.85	0.95	0.89	0.82
		LB	0.59	0.67	0.67	0.47	0.51	0.67	0.57	0.62	0.59	0.61
		H	0.93	0.94	0.96	0.94	0.97	0.98	0.96	0.93	0.94	0.94

EB: Early Blight; LB: Late Blight; H: Healthy
the best precision scores are marked in bold and underlined

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Tabel 5. Recall

Model	Fine-tune	Class Type	Parameter									
			Size of FC 2048x2					Size of FC 4096x2				
			Dropout									
			None	0.2	0.3	0.4	0.5	None	0.2	0.3	0.4	0.5
VGG16	Block5	BK	0.64	0.56	0.68	0.54	0.86	0.68	0.82	0.7	0.82	0.64
		BD	0.92	0.94	0.96	0.94	0.9	0.94	0.94	0.94	0.94	0.94
		S	0.22	0.32	0.2	0.16	0.14	0.34	0.2	0.18	0.16	0.12
	Block4	BK	0.84	0.9	0.42	0.56	0.14	0.98	0.82	0.82	0.52	0.2
		BD	0.98	0.98	1	1	1	0.94	1	1	1	1
		S	1	1	1	0.96	1	1	1	1	1	0.48
	Block3	BK	0.92	0.66	0.84	0.84	0.98	0.98	1	0.98	**1**	0.94
		BD	0.96	1	0.96	0.98	0.94	0.86	0.98	0.94	**1**	0.96
		S	1	1	1	1	1	1	1	1	**1**	1
	Block2	BK	0.96	1	0.96	0.9	0.7	0.92	0.94	0.92	1	0.9
		BD	0.98	0.94	1	0.96	0.96	0.88	0.9	0.98	0.92	1
		S	1	1	1	1	0.98	1	1	1	1	0.98
	All Layers	BK	0.98	0.94	0.84	0.8	0.92	0.98	0.9	0.98	0.92	0.96
		BD	0.92	0.98	0.94	0.98	0.72	0.98	0.98	0.94	0.98	0.98
		S	1	1	1	1	1	1	1	1	1	1
Inception V3	Mixed10	BK	0.74	0.88	0.84	0.89	0.82	0.84	0.8	0.72	0.74	0.82
		BD	1	0.96	1	0.98	0.98	0.96	0.98	0.98	1	0.98
		S	0.62	0.64	0.7	0.7	0.74	0.8	0.68	0..84	0.62	0.62
	Mixed9	BK	0.9	0.82	0.96	0.84	0.86	0.9	0.82	0.8	0.92	0.82
		BD	0.94	1	0.96	1	1	1	1	1	0.98	1
		S	0.7	0.78	0.7	0.66	0.72	0.92	0.72	0.62	0.64	0.86
	Mixed8	BK	0.8	0.82	0.94	0.82	0.74	0.92	0.74	0.82	0.82	0.72
		BD	0.98	1	0.92	0.98	1	0.96	1	1	0.98	0.98
		S	0.68	0.76	0.8	0.66	0.74	0.76	0.72	0.68	0.68	0.74
	Mixed7	BK	0.86	0.84	0.92	0.9	0.82	0.88	0.9	0.86	0.9	0.82
		BD	0.98	0.98	0.98	1	0.98	1	0.96	0.98	0.96	0.98
		S	0.74	0.7	0.62	0.54	0.72	0.68	0.78	0.7	0.74	0.66
	Mixed6	BK	0.78	0.88	0.86	0.96	0.94	0.88	0.88	0.88	**0.9**	0.88
		BD	1	1	0.98	0.92	0.96	0.96	1	0.98	**0.98**	0.98
		S	0.68	0.72	0.88	0.72	0.82	0.84	0.82	0.78	**0.88**	0.78
	Mixed5	BK	0.82	0.76	0.82	0.64	0.7	0.84	0.8	0.64	0.78	0.7
		BD	0.96	1	0.98	0.94	0.94	0.96	0.98	0.94	0.98	0.96
		S	0.94	0.84	0.88	0.84	1	0.98	0.94	0.84	0.44	0.94
	Mixed4	BK	0.64	0.74	0.78	0.74	0.72	0.58	0.58	0.48	0.64	0.48
		BD	0.98	0.98	0.98	0.94	0.96	0.98	0.98	0.98	0.98	1
		S	0.96	0.9	0.98	0.94	0.98	0.92	0.96	0.94	0.76	0.72
	Mixed3	BK	0.66	0.7	0.7	0.78	0.38	0.66	0.64	0.7	0.58	0.56
		BD	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
		S	0.84	0.96	0.88	0.88	0.8	0.92	0.92	0.98	0.84	0.96
	Mixed2	BK	0.8	0.72	0.68	0.48	0.32	0.66	0.52	0.46	0.34	0.48
		BD	0.98	0.98	0.98	0.98	1	0.98	0.98	0.98	0.98	0.98
		S	0.96	0.94	0.94	0.84	0.88	1	1	0.96	0.94	0.88
	Mixed1	BK	0.56	0.62	0.58	0.6	0.42	0.54	0.38	0.28	0.56	0.5
		BD	0.98	0.98	0.96	1	0.98	0.98	0.98	0.98	0.98	0.98
		S	0.94	0.94	1	0.72	0.88	1	1	1	0.86	0.98
	All Layers	BK	0.28	0.5	0.5	0.16	0.3	0.5	0.34	0.93	0.32	0.36
		BD	0.96	0.98	0.98	0.98	0.98	0.98	0.94	0.96	0.94	0.98
		S	1	0.92	0.98	0.68	0.72	0.98	0.9	1	0.98	0.9

EB: Early Blight; LB: Late Blight; H: Healthy
the best recall scores are marked in bold and underlined

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Tabel 6. F1-score

Model	Fine-tune	Class Type	Parameter									
			Size of FC 2048x2					Size of FC 4096x2				
			Dropout									
			None	0.2	0.3	0.4	0.5	None	0.2	0.3	0.4	0.5
VGG16	Block5	BK	0.76	0.69	0.79	0.68	0.88	0.78	0.87	0.8	0.87	0.75
		BD	0.62	0.61	0.62	0.58	0.62	0.64	0.64	0.61	0.64	0.59
		S	0.36	0.48	0.33	0.28	0.25	0.51	0.33	0.31	0.28	0.21
	Block4	BK	0.9	0.94	0.59	0.72	0.25	0.96	0.9	0.9	0.68	0.33
		BD	0.92	0.94	0.78	0.81	0.7	0.96	0.92	0.92	0.81	0.6
		S	1	1	1	0.98	1	1	1	1	1	0.65
	Block3	BK	0.95	0.8	0.9	0.9	0.96	0.92	0.99	0.96	**1**	0.95
		BD	0.96	0.85	0.92	0.92	0.96	0.91	0.99	0.96	**1**	0.95
		S	0.97	1	0.97	1	1	1	1	1	**1**	1
	Block2	BK	0.97	0.97	0.98	0.93	0.8	0.9	0.92	0.95	0.96	0.98
		BD	0.97	0.97	0.98	0.93	0.84	0.9	0.92	0.95	0.96	0.98
		S	1	1	1	1	0.99	1	1	1	1	0.99
	All Layers	BK	0.95	0.96	0.9	0.88	0.84	0.98	0.95	0.96	0.96	0.97
		BD	0.95	0.96	0.96	0.9	0.8	0.98	0.95	0.96	0.95	0.97
		S	1	1	0.97	1	1	1	0.98	1	0.99	1
Inception V3	Mixed10	BK	0.84	0.92	0.87	0.91	0.87	0.89	0.88	0.84	0.81	0.87
		BD	0.76	0.8	0.84	0.83	0.86	0.83	0.8	0.82	0.79	0.79
		S	0.77	0.76	0.82	0.8	0.85	0.88	0.79	0.89	0.76	0.76
	Mixed9	BK	0.88	0.88	0.89	0.85	0.89	0.9	0.88	0.85	0.92	0.88
		BD	0.84	0.85	0.9	0.85	0.85	0.88	0.83	0.8	0.84	0.88
		S	0.81	0.88	0.82	0.8	0.84	0.89	0.84	0.77	0.77	0.92
	Mixed8	BK	0.84	0.85	0.9	0.85	0.8	0.89	0.82	0.86	0.84	0.76
		BD	0.81	0.87	0.88	0.82	0.83	0.88	0.81	0.83	0.83	0.83
		S	0.81	0.86	0.89	0.8	0.85	0.86	0.84	0.81	0.81	0.85
	Mixed7	BK	0.88	0.88	0.89	0.71	0.87	0.9	0.9	0.87	0.92	0.88
		BD	0.85	0.88	0.84	0.81	0.82	0.85	0.86	0.84	0.85	0.79
		S	0.85	0.82	0.77	0.7	0.84	0.81	0.88	0.82	0.85	0.82
	Mixed6	BK	0.88	0.91	0.91	0.91	0.94	0.92	0.94	0.93	**0.94**	0.91
		BD	0.79	0.85	0.88	0.84	0.89	0.86	0.87	0.85	**0.9**	0.86
		S	0.81	0.84	0.94	0.84	0.89	0.9	0.9	0.87	**0.93**	0.88
	Mixed5	BK	0.87	0.85	0.87	0.75	0.8	0.88	0.88	0.75	0.85	0.8
		BD	0.89	0.85	0.88	0.78	0.85	0.91	0.88	0.78	0.72	0.83
		S	0.96	0.89	0.93	0.89	0.98	0.98	0.96	0.89	0.61	0.96
	Mixed4	BK	0.77	0.82	0.87	0.85	0.82	0.72	0.72	0.65	0.76	0.64
		BD	0.84	0.87	0.9	0.87	0.87	0.8	0.82	0.78	0.78	0.73
		S	0.96	0.93	0.97	0.94	0.96	0.94	0.96	0.94	0.84	0.82
	Mixed3	BK	0.79	0.81	0.8	0.87	0.54	0.78	0.76	0.8	0.73	0.71
		BD	0.8	0.86	0.84	0.85	0.71	0.84	0.82	0.88	0.78	0.81
		S	0.89	0.96	0.92	0.93	0.87	0.94	0.95	0.96	0.88	0.96
	Mixed2	BK	0.88	0.83	0.79	0.64	0.48	0.79	0.68	0.62	0.5	0.64
		BD	0.91	0.86	0.84	0.74	0.72	0.85	0.82	0.62	0.74	0.77
		S	0.95	0.95	0.96	0.9	0.92	0.99	0.97	0.96	0.95	0.91
	Mixed1	BK	0.71	0.75	0.72	0.75	0.58	0.69	0.55	0.44	0.71	0.65
		BD	0.79	0.82	0.82	0.75	0.74	0.81	0.77	0.74	0.78	0.8
		S	0.97	0.96	0.98	0.84	0.92	0.99	0.97	0.96	0.91	0.97
	All Layers	BK	0.43	0.64	0.66	0.27	0.45	0.65	0.49	0.52	0.47	0.5
		BD	0.73	0.8	0.8	0.63	0.67	0.8	0.71	0.76	0.72	0.75
		S	0.96	0.93	0.97	0.79	0.83	0.98	0.93	0.96	0.96	0.9

EB: Early Blight; LB: Late Blight; H: Healthy
the best f1-score are marked in bold and underlined

Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

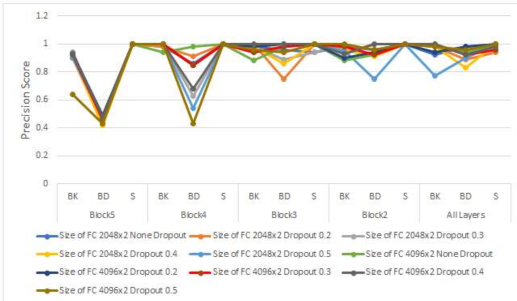

Figure 9. Graph of testing precision fine-tuning model VGG16

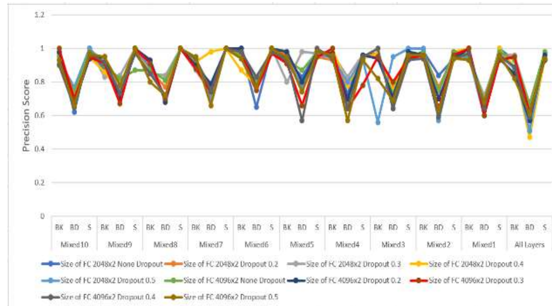

Figure 10. Graph of testing precision fine-tuning model InceptionV3

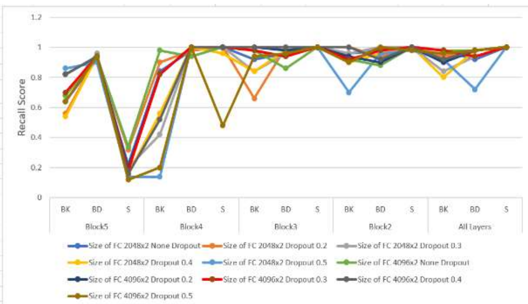

Figure 11. Graph of testing recall fine-tuning model VGG16

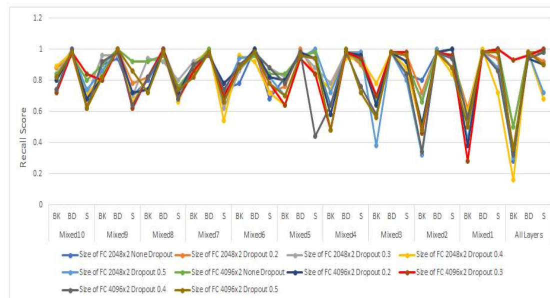

Figure 12. Graph of the testing recall fine-tuning of the InceptionV3 model

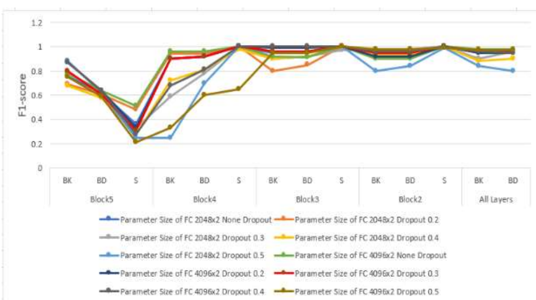

Figure 13. Graph of the testing f1-score fine-tuning of the VGG16 model

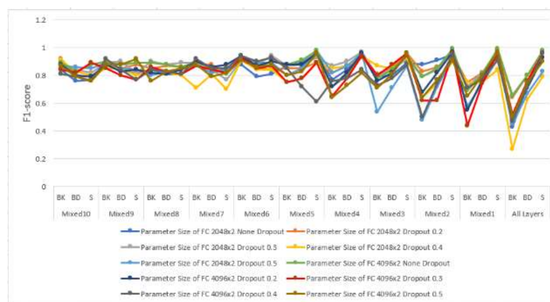

Figure 14. Graph of testing f1-score fine-tuning of InceptionV3 model

The process of retraining CNN with the trained weights for targeted data is called fine-tuning. It takes a lot of time to fine-tune the entire CNN network and does not guarantee good performance. As explained by [17], [23], the lower layer learns general features, such as edges that are common in all image data sets, while the upper layer is a very specific feature. To train a model with a relatively small number of sets, it's a good idea to freeze a third or half of the pre-trained layer. Because the initial layer contains a very general feature map that is useful for the target data set even though the domains are very different.

From the experiment doing fine-tuning using the dropout rate and the number of neurons on the Fully Connected Layer, the best performance was obtained by the VGG16 model with a dropout of 0.4 and the number of neurons 4096x2 by tuning in the 3rd block and the test results on the test dataset, namely the average precision score. 0.97, average recall 0.97, mean F1-score 0.97, and 100% accuracy. While the experiment using the dropout rate and the number of neurons in the FC layer obtained the best performance. InceptionV3 model using dropout 0.4 and the number of neurons 4096x2 with tuning in the mixed6 inception module which got a test result score on the test dataset, namely an average score of 0.93 precision, an average recall of 0.92, the average f1-score is 0.92 and the accuracy is 92%. The graph of the training results of the VGG16 model on the 3rd block with the number of neurons

Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4096x2 and a dropout rate of 0.4 is shown in Figure 15 and Figure 16. Then the graph of the results of training for InceptionV3 on the mixed6 inception module with a dropout rate of 0.4 and using the Size of FC 4096x2 is shown in Figure 17 and Figure 18.

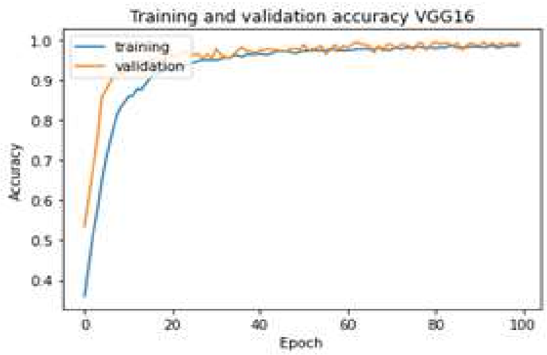

Figure 16. Results of training and validation accuracy for VGG16 with tuning block 3, Size of FC 4096x2, and dropout rate 0.4

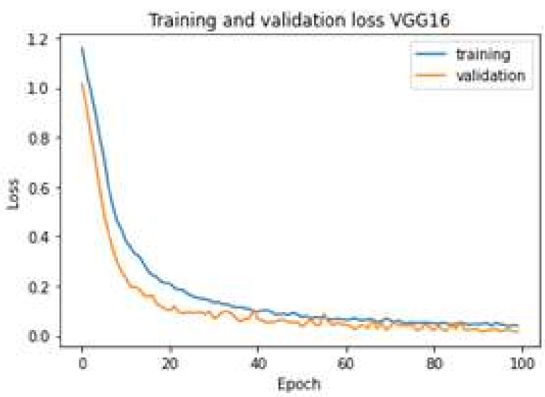

Figure 17. Results of training and validation loss VGG16 with tuning block 3, Size of FC 4096x2, and dropout rate 0.4

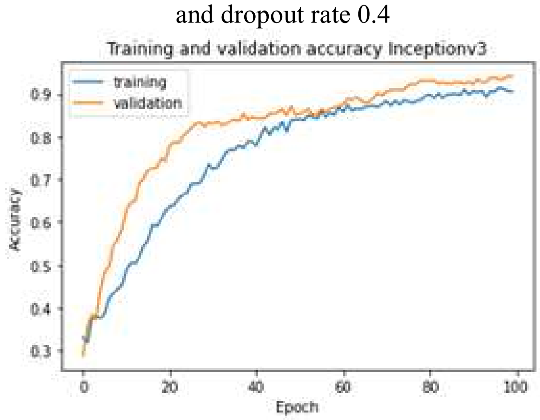

Figure 18.The results of InceptionV3 training and validation accuracy with tuning in the Inception mixed6 module, Size of FC 4096x2, and dropout rate 0.4

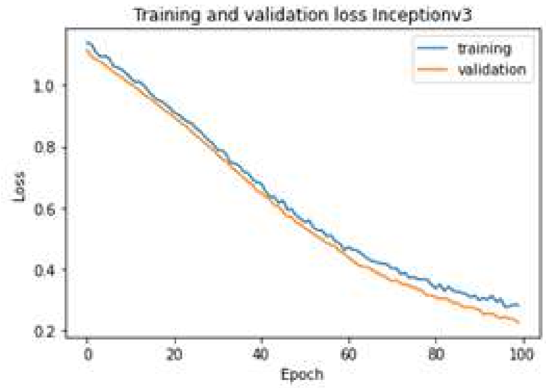

Figure 18. Results of InceptionV3 training and validation loss with tuning in the Inception mixed6 Size of FC 4096x2 module, and a dropout rate 0.4

6 CONCLUSION

In this study, it was successful in implementing Deep Learning using the Transfer Learning method with the pre-trained VGG16 and InceptionV3 models for the classification of potato leaf disease. The training process is carried out using the same top layer MLP.

In the VGG16 model, fine-tuning the last two blocks can provide good performance because of the association that sublayer features have common features such as edge features that can be used for all types of datasets. Whereas the top layer feature has more specific features, it means the bottom layer can be frozen or not retrained. In deep models such as InceptionV3, retraining the entire network results in reduced performance, because the deeper the architecture gets, the more parameters to be trained increase and the potential for overfitting increases.

It is concluded in this study that the VGG16 model can generalize data better than InceptionV3. Obtained the highest accuracy in both models using the parameters Size of FC 4096x2 and Dropout 0.4 on the top layer where VGG16 got 100% accuracy and InceptionV3 92%. The use of Size of FC 4096x2 makes the representation of image features more and helps the model to recognize more features but requires a lot of memory. As the dropout rate increases, the accuracy will increase because it reduces the learning interdependence in each neuron. The use of a dropout rate of 0.5 gives a decrease in performance because too much weight is removed, in this study a trial and error experiment was carried out.

In the future, research can be developed by adding data on types of vegetable plant diseases. And in its implementation, this research is expected

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

to help the agricultural industry in maintaining vegetable crops.

REFRENCES:

[1] P. U. Rakhmawati, Y. M. Pranoto, and E. Setyati, "Klasifikasi Penyakit Daun Kentang Berdasarkan Fitur Tekstur Dan Fitur Warna Menggunakan Support Vector Machine," *Semin. Nas. Teknol. dan Rekayasa 2018*, pp. 1–8, 2018.

[2] N. Gunadi, A. K. Karjadi, and S. Sirajuddin, "Pertumbuhan dan Hasil Beberapa Klon Kentang Unggul Asal International Potato Center di Dataran Tinggi Malino, Sulawesi Selatan," *J. Hortik.*, vol. 24, no. 2, p. 102, 2014.

[3] A. Dimyati, "Research priorities for potato in Indonesia," *Proc. CIP-Indonesia Res. Rev. Work. held Bogor, Indones.*, pp. 15–19, 2003.

[4] A. Krizhevsky and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," pp. 1–9, 2012.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[6] C. Szegedy *et al.*, "Going deeper with convolutions," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07-12-June, pp. 1–9, 2015.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2015-Decem, pp. 2818–2826, 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2015.

[9] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2261–2269, 2017.

[10] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Comput. Electron. Agric.*, vol. 161, no. October 2017, pp. 272–279, 2019.

[11] M. Shu, "Deep learning for image classification on very small datasets using transfer learning," 2019.

[12] S. J. Pan and Q. Y. Fellow, "A Survey on Transfer Learning," pp. 1–15, 2010.

[13] G. Geetharamani and A. P. J., "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Comput. Electr. Eng.*, vol. 76, pp. 323–338, 2019.

[14] S. Arya and R. Singh, "A Comparative Study of CNN and AlexNet for Detection of Disease in Potato and Mango leaf," *IEEE Int. Conf. Issues Challenges Intell. Comput. Tech. ICICT 2019*, no. Dl, 2019.

[15] N. Gangwar, D. Tiwari, A. Sharma, M. Ashish, A. Mittal, and G. K. Vishwavidyalaya, "Grape Leaf Diseases Classification using Transfer Learning," pp. 3171–3177, 2020.

[16] Agarwal M., Sinha A., Gupta S.K., Mishra D., Mishra R. (2020) Potato Crop Disease Classification Using Convolutional Neural Network. In: Somani A., Shekhawat R., Mundra A., Srivastava S., Verma V. (eds) Smart Systems and IoT: Innovations in Computing. Smart Innovation, Systems and Technologies, vol 141. Springer, Singapore. https://doi.org/10.1007/978-981-13-8406-6_37.

[17] R. Serrano-Gotarredona *et al.*, "CAVIAR: A 45k neuron, 5M synapse, 12G connects/s AER hardware sensory-processing-learning-actuating system for high-speed visual object recognition and tracking," *IEEE Trans. Neural Networks*, vol. 20, no. 9, pp. 1417–1438, 2009.

[18] F. Chollet, *Deep Learning with Python*. 2018.

[19] M. Elgendy, "Deep Learning for Vision Systems," p. 475, 2019.

[20] N. Srivastava, G. Hinton, A. Krizhevsky, & Sutskever, I., and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal Mach. Learn. Res.*, vol. 299, no. 3–4, pp. 345–350, 2014.

[21] M. A. H. Abas, N. Ismail, A. I. M. Yassin, and M. N. Taib, "VGG16 for plant image classification with transfer learning and data augmentation," *Int. J. Eng. Technol.*, vol. 7, no. 4, pp. 90–94, 2018.

[22] T. Carneiro, R. V. M. Da Nobrega, T. Nepomuceno, G. Bin Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018.

[23] M. Lin, Q. Chen, and S. Yan, "Network in network," *2nd Int. Conf. Learn. Represent. ICLR 2014 - Conf. Track Proc.*, pp. 1–10, 2014.

[24] O. Sudana, I. W. Gunaya, and I. K. G. D. Putra, "Handwriting identification using deep convolutional neural network method," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 18, no. 4, pp. 1934–1941, 2020.

[25] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," *Adv. Neural Inf. Process. Syst.*, vol. 4, no. January, pp. 3320–3328, 2014.