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

IMPLEMENTATION OF TRANSFER LEARNING MOBILENETV2 ARCHITECTURE FOR IDENTIFICATION OF POTATO LEAF DISEASE

TIKA ADILAH M[1], DINAR AJENG KRISTIYANTI[2*]

[1]Department of Information Technology, Universitas Nusa Mandiri, Jakarta 10450 Indonesia

[2]Department of Information System, Universitas Multimedia Nusantara, Tangerang 15111, Indonesia

E-mail: [1]14002364@nusamandiri.ac.id, [2*]dinar.kristiyanti@umn.ac.id

ABSTRACT

Potatoes are one of the third most important food crops in the world. Potato farming has problems in the form of diseases that attack the leaves. These diseases can affect the quality of potato plants, resulting in crop failure. Digital image processing is a method that can be used to assist farmers in identifying potato leaf diseases. The development of digital image processing has been carried out, one of which is by using the Convolutional Neural Network (CNN) algorithm. CNN requires big data. CNN architecture will experience overfitting if it uses little data, where the classification model has high accuracy on training data but poor accuracy on test data. This research utilizes Transfer Learning and Augmentation methods to avoid overfitting on too little data. Transfer Learning method used in this research is MobileNetV2. The results of the trials in this study indicate that the MobileNetV2 Transfer Learning method has good classification performance results and produces a high accuracy value of 99.6%.

Keywords: *Convolutional Neural Network, Leaf Disease Classification, MobileNetV2, Potato Leaf Disease, Transfer Learning*

1. INTRODUCTION

Potato (Solanum tuberosum L.) is one of the third most important food crops in the world originating from the Andes highlands, South America [1]. Potatoes are of great interest in agricultural production systems because they consolidate an extremely high yield potential with high nutritional value. Potatoes are grown in more than 125 countries and consumed in almost every part of the world [2]. In Indonesia, potatoes have become a popular staple food. Potatoes grow well in highlands or mountainous areas with a slope of 800-1,500 m above sea level (asl). Potatoes can become infected with diseases that affect the stage of the crop before and after harvest [3].

Plant diseases are a common cause of loss of crop production which can have a significant economic impact resulting in reduced income for producers and distributors and increased prices for consumers. Diseases that often attack potato plants include dry blight (early blight) and late blight (late blight) [4]. Based on the experience of potato farmers, the disease appears when potato plants are 5-6 weeks after planting [5]. This disease has the potential to spread to other parts of the potato plant, such as stems, stalks and tubers of potatoes. Therefore, potato farmers must trim the leaves that have contracted the disease so that the disease does not spread so that farmers can get good harvests [6].

Early identification of potato leaf disease is very important, therefore new methods for potato disease identification are needed to prevent loss of production and improve quality. Early identification is a challenge for farmers, this is because they have to face various disease problems that cannot be overcome by manual prediction. If they can spot the disease early, it will help farmers make decisions that will increase potato production. Indexing and abstracting services depend on the accuracy of the title, extracting from it keywords useful in cross-referencing and computer searching. An improperly titled paper may never reach the audience for which it was intended, so be specific. Manual identification of potato diseases is not only difficult and error-prone but also time-consuming while technology-based systems are more efficient and cost-effective [2]. The development of the industrial revolution 4.0 is emphasized in various fields, one of which is in the agricultural sector [7]. In agriculture, digital technology can be used, one of which is for automatic detection of diseases in potato leaves.

Deep Learning systems are currently widely used to automate processes [8], [9], [10].

Deep Learning to search for disease patterns is one solution that can be provided and is easily accessible to small farmers [11]. Given enough image data of infected plants, it is possible to train a Deep Learning system that is able to identify the given disease according to the data[12].

In this study, the purpose of this study was to classify diseases on potato leaves. To ensure the right type of disease, currently the use of deep learning is the optimal way with the aim of getting an algorithm that can classify the types of leaf diseases in potato plants [13]. The method that will be used is Convolutiona Neural Network which will show the accuracy of the results of potato leaf disease classification. The CNN model is a modern and efficient recognition method[14]. In contrast to traditional image classification [15], Convolutiona Neural Network uses multilayer convolution which automatically extracts and combines features[16]. This algorithm is designed to be carried out independently and decided to complete a specific task [17]. CNN is very suitable for classification on large amounts of data[18]. However, the data obtained for use in image classification does not always have a large amount of data, the data obtained may have an unbalanced proportion for each class and this affects the level of accuracy generated in the classification process. The solution in overcoming scanty data can be used augmentation techniques [19] and in the classification process can use transfer learning models on the CNN architecture, transfer learning models or also known as pretrained models is a CNN architecture that has been trained with a previous dataset which can later be used to train another dataset [20].

This study classified the disease on potato leaves using the Convolutional Neural Network (CNN) architecture. The data used in this study has an imbalanced and small amount, the problem that can arise when using less data is the occurrence of overfitting [21], so in this study using data augmentation techniques to overcome data imbalances and using a transfer learning model of the MobileNetV2 architecture to solve the problem of scanty data. Among the many ways to implement a deep learning model, implementing it on a mobile device is one of the easiest. The ease of using popular mobile devices is an advantage. With a few touches the user can get the answer. In addition, thanks to the camera phone deep learning models can receive large amounts of data in real time. Two aspects should be considered when implementing a Deep Learning model on mobile devices namely the model file size and processing speed. The model cannot be applied to mobile devices if the size is too large, and it will cause inconvenience to the user when the process is slow. Therefore, in this study a real-time mobile application was developed to classify types of potato leaf disease using the CNN Model method based on transfer learning and augmented images. To clarify the aims of this study, we suggest the following hypotheses:

- perform augmentation to handle data imbalance
- designed a CNN-based transfer learning model that can classify pseudo-endangered parrot species with high color similarity.
- Applications developed can embed CNN-based training designs)

Novelty in this study is that the proposed technique was also compared with the state-of-the-art model and achieved significant significance in terms of accuracy and computational cost.

This paper is structured as follows. The proposed method is described in the ‘Method’ section in more detail. Furthermore, the experimental results are discussed and analyzed in the ‘Results and Discussion’ section. Finally, the conclusions of this paper are presented in the ‘Conclusion’ section.

2. LITERATURE REVIEW

Much research has been done on potato leaf disease. Research [22] proposed a data augmentation method based on an image-to-image translation model to overcome the problem of bias by complementing these less diseased leaf images. The proposed augmentation method translates images of healthy and diseased leaves and utilizes the attention mechanism to create images that reflect a more realistic texture of the disease. Augmentation is important, such as Utpal's research [23] on potato leaf images aimed at overcoming data imbalances, Utpal research obtained the highest accuracy reaching 97%. The small number of potato leaf image datasets made Faranbee et al [15] conduct research to identify potato leaf diseases using the CNN method with transfer learning and obtain an accuracy of 99.43%, the transfer learning method is a promising method and can be applied in mobile applications.

In the study of [24] Proposing an Efficient DenseNet Model pre-training has been used by utilizing an additional transition layer in DenseNet-201 to classify potato leaf diseases efficiently. In addition, the use of the reweighted cross-entropy loss function makes our proposed algorithm more robust because the training data is very unbalanced. Dense joints with regularizing strength help minimize overfitting during small training sets of potato leaf samples. Algorithm performance was evaluated on the test set and provided 97.2% accuracy. In the study of Tiwari et al. (Tiwari et al., 2020), 2152 pictures of potato leaves were taken from the plant village. Various pre-trained models such as inceptionV3, VGG16, and VGG19 are used for feature extraction, of which VGG19 gives optimal results. Multiple classifiers namely KNN, SVM, Neural Network, and logistic regression are used for classification. Among them, logistic regression provides an up-to-date solution with a classification accuracy of 97.8%.

Research by Rozaki et al., [27] uses the Transfer Learning method to avoid model overfit when the data used is not ideal or too little. Transfer Learning is a method that uses a CNN architecture that has been turned off by other data before which is then used for image classification on new data. The purpose of this study is to use the Transfer Learning method on the CNN architecture to classify potato leaf images in identifying potato leaf disease, this method produces the highest accuracy value of 95%. Thus, most of the applicable techniques rely solely on the existing “The Plant Village” dataset. In addition, the data set has a large understanding which significantly affects the performance of the model. Therefore, to address the problem to the class psychosis algorithm we propose using a better deep learning pattern. we also propose a system for classifying image data obtained in real time using a mobile device camera.

3. METHOD

This section describes the research method applied in this study. The diagram in Figure 1 shows the general process of classifying potato dun disease. In this study consisted of six important processes. The first is dataset collection, next is image augmentation, preprocessing, data split process to determine training data, validation and testing, then continued with the image classifier process, after that converting the model into a FlatBuffer File (tflite) and the final results are displayed on mobile devices.

3.1 Data Collection

In this study, images from public data were used. Potato leaf images are obtained from ‘kaggle’ which is a publicly available database. This dataset contains 2152 color image potato leaf images. Image resolution is 256×256 pixels. Potato leaves which consist of 3 disease classes can be seen in Figure 2.

3.2 Augmentation

The need for large data sets is one of the biggest limitations in developing deep learning models, as building a reliable Deep Learning model requires Millions or even more data samples. by manipulating and changing small amounts of data the limitations can be overcome. The technique is data augmentation [28]. Data augmentation techniques have been used in many studies [19], [20]. These includes random cropping, horizontal flipping, brightness modification and contrast modification. In addition, augmentation is also commonly used in dealing with unbalanced data. In this study, there were more images in the early and late flash classes than healthy ones. This is a type of unbalanced data set that can cause the applied model to be overfitting or underfitting. To overcome the problem in this study, augmentation was only carried out on the healthy leaf class.

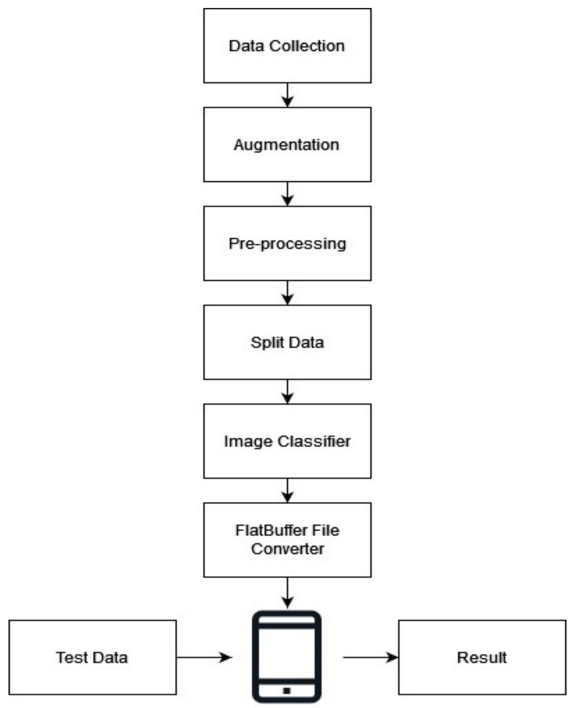


Figure 1: Research Design Model

(a) (b) (c)

Figure 2: Potato Leaf (a) Early Blight (b) Healthy, and (c) Late Blight

3.3 Data Pre-Processing

After the augmentation process, pre-processing is carried out. All potato leaf images in the dataset are color images with dimensions of 256x256. In this study, the image size is changed to a new dimension of 224x224. The purpose of resizing the image to a smaller dimension is to help the performance of the model to do the job correctly with less time [29]. After resizing, then

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

the conversion of BGR to RBG is carried out. This conversion is done because OpenCV uses the BGR image format. so that when we read an image using cv2.imread() it will be interpreted in BGR format by default. We can use cvtColor (im, cv2.COLOR_BGR2RGB) method to convert BGR image to RGB and vice versa. After that, rescale = 1/255. This is done to change the size of the image RGB pixel data (0-255) into a range of numbers (0-1) to facilitate the data training process.

3.4 Split Data

After the rescale, then split the data. In this study, the dataset is divided into three parts, namely training, validation and testing data. Training data is the data used to train the classification algorithm, validation data is used for the model validation process and testing data is used to determine the performance of the previously trained classification algorithm. Initially the data was divided into 2 parts, namely train data and test data, with a comparison of 90% for train data and 10% for test data. The train data is then divided into 2 parts, namely train data and validation data. The validation data is takes as many as 300 images from the train data or as much as 10% of the total data. The distribution of images is shown in Table 1.

Table 1: Dataset Splitting

Dataset	Total
Train	2400
Test	300
Valid	300

3.5 Image Classifier

After the pre-processing stage, next is the image classification process using the MobileNetV2 Transfer Learning model. At this stage there are two steps, the first step is feature-extraction and the second step are the classification process. For feature extraction, we import the MobilenetV2 model from the hardware library. The imported model uses a convolutional layer that is initialized with predefined weights and pre-trained using ImageNet[30]. Furthermore, for classification used Global Average Pooling layer and dense layer with SoftMax added after convolutional layer, not fully connected layer [31]. According to [32] fully connected layer is usually used in the traditional CNN model and tends to be overfitting even though it uses dropout. Therefore, in this study, Global Average Pooling (GAP) [33] is used which inputs the average value of each feature and maps it into a vector and links it directly to the SoftMax layer input[34]. Figure 3 is a diagram of the image classifier process.

The main foundation of the MobileNetV2 architecture [35] is the bottleneck residual block which is shown in Table 2 where h, w, d, t and s are length, width, depth, expansion factor and stride respectively. When compared to the residual block belonging to the ResNet model [36], the bottleneck residual block is more memory efficient and resistant to loss of important features during the model training process (vanishing gradient) [37]. The type of convolution layer used in this block is the depthwise (dwise) separable convolution layer which uses 8 to 9 times less computation than the standard convolution layer [35].

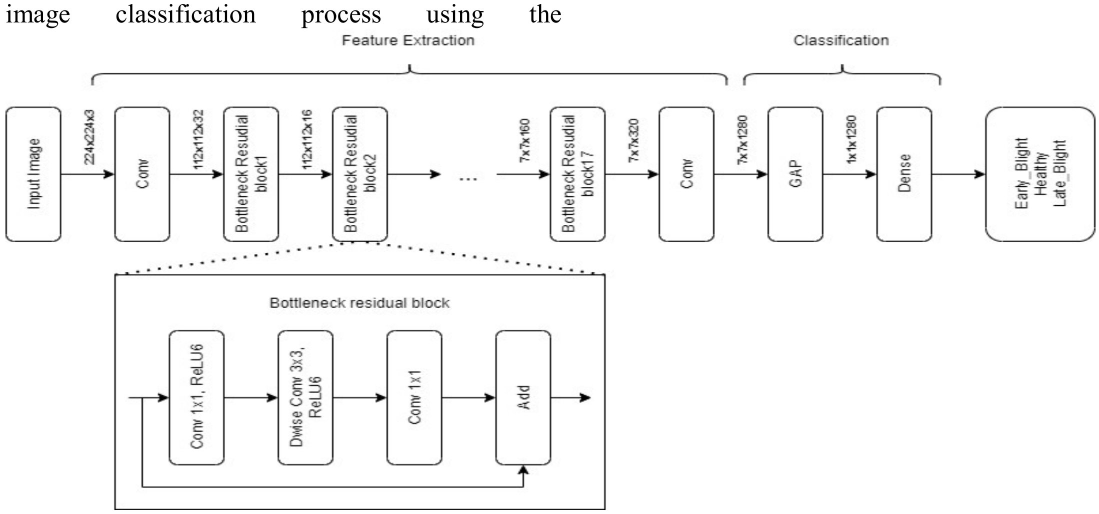


Figure 3: Image Classifier Process

The main foundation of the MobileNetV2 architecture [35] is the bottleneck residual block which is shown in Table 2 where h, w, d, t and s are length, width, depth, expansion factor and stride respectively. When compared to the residual block belonging to the ResNet model [36], the bottleneck

Journal of Theoretical and Applied Information Technology
31st August 2023. Vol.101. No 16

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

residual block is more memory efficient and resistant to loss of important features during the model training process (vanishing gradient) [37]. The type of convolution layer used in this block is the depthwise (dwise) separable convolution layer which uses 8 to 9 times less computation than the standard convolution layer [35].

Table 2: Bottleneck Resudial Block

Input	Operator	Output
h x w x d	1 x 1 conv2d, ReLU6	h x w x t*d
h x w x t*d	3 x 3 dwise stride = s, ReLU6	h/s x w/s x t*d
h/s x w/s x t*d	Linear 1 x 1 conv2d	h/s x w/s x d'

The MobileNetV2 architecture uses an input resolution of 224 x 224 pixels, as shown in Table 3. This resolution was chosen because 224 x 224 pixels is the standard resolution of the MobileNetV2 model that has been trained on the ImageNet dataset. Each bottleneck layer in the architecture is built by n bottleneck residual blocks. The first bottleneck residual block for each bottleneck layer has a stride with a value of s, and then uses a stride with a value of 1. The channel/depth of the output in each layer is denoted by c. With the exception of the first layer, an expansion rate (t) of 6 is used for all layers. With this model, each potato leaf image will be represented as 1280 features in global average pooling. 1280 this feature will be input to the output/classifier layer. In the classifier layer, the softmax activation function is used which produces the object class to be classified.

Table 3: MobileNetV2 Architecture

Input	Layer	*t*	*c*	*N*	*s*
224 x 224 x 3	Conv2d	-	32	1	2
112 x 112 x 32	Bottleneck	1	16	1	1
112 x 112 x 16	Bottleneck	6	24	2	2
56 x 56 x 24	Bottleneck	6	32	3	2
28 x 28 x 32	Bottleneck	6	64	4	2
14 x 14 x 64	Bottleneck	6	96	3	1
14 x 14 x 96	Bottleneck	6	160	3	2
7 x 7 x 160	Bottleneck	6	320	1	1
7 x 7 x 320	Conv2d 1x1	-	1280	1	1
7 x 7 x 1280	GlobalAvgPool	-	1280	1	-

The MobileNetV2 model was trained with 30 epochs and 32 batch sizes. The chosen optimizer was Adam. 'categorical_crossentropy' is used as a loss function. At the end of each epoch, the accuracy value of the classification results will be calculated and the weight from MobileNetV2 will be stored as a model checkpoint if the accuracy value obtained is higher than the previous checkpoint. With a training scheme like this, a model will be obtained that can provide the highest accuracy value in the training process.

3.6 Split Data FlatBuffer File Converter

To develop applications, TensorFlow Lite provides a method that converts the resulting model into a Tensor- FlatBuffer Flow Lite (.tflite) file format, which can be used on mobile devices. FlatBuffer is an open-source cross-flatform serialization library that serializes data efficiently. TensorFlow Lite supports file conversions created by TensorFlow, concrete functions, and Keras Keras [38]. In this study, the modified model file was included in the demo project provided by TensorFlow Lite. After this step, next create an android package file (APK) and install the app on the device always. Figure 4 shows the whole process.

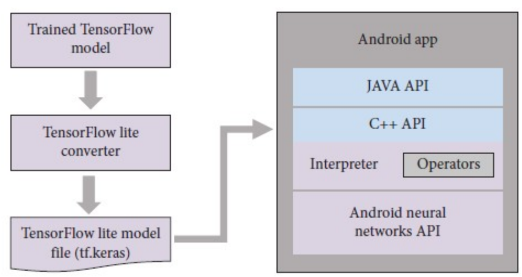


Figure 4: Tensorflow Lite Conversion Process

3.7 Evaluation and Validation

At this stage, the process of testing the proposed method is carried out by detecting potato leaf disease on a new image, so that it is known how precise the proposed method is to identify potato leaf disease.

4. RESULTS AND DISCUSSION

In this study, several types of experiments were carried out to determine the results of each method used.

4.1 Augmentation

In this study, image augmentation was carried out using six techniques, this augmentation process was not carried out in previous studies, namely research [2][25][26], the results of augmentation in this study were used to overcome data imbalances. The number of healthy leaf images which were originally only 152 was augmented so that the number became balanced with the other 2 classes, namely 1000 images. The healthy potato leaf dataset was augmented using:

1 horizontal flip = true. Function to flip the image horizontally randomly.
2 vertical flip = true. Function to flip the image vertically randomly.
3 zoom_range = 0.2. Serves to enlarge the image, 0.2 indicates the magnification intensity of the image.
4 fill_mode = 'nearest'. Serves to fill empty area where 'nearest' means replacing the empty area with the nearest pixel value.
5 brightness_range = [0.5, 1.5]. Function to change brightness randomly.

The results of the augmentation are shown in Figure 5 and Table 4 shows the number of datasets after the augmentation process.

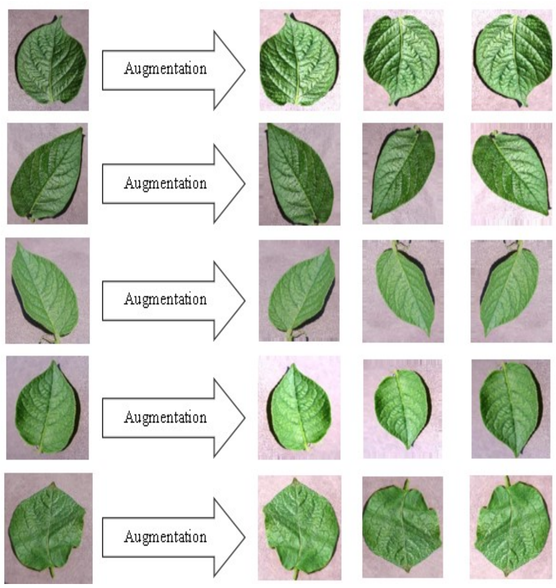


Figure 5: Augmentation Results

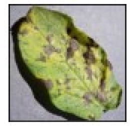
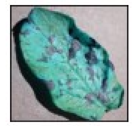
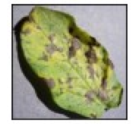
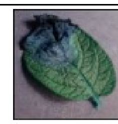
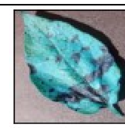
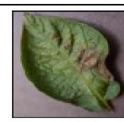
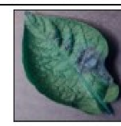
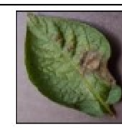
Table 4: Augmentation Result Dataset

Label	Category	Total
1	Early Blight	1000
2	Healthy	1000
3	Late Blight	1000

4.2 Data Pre-processing

The pre-processing stage is conducted by resizing, rescale and converting the BGR image to RGB. Some images from BGR to RGB conversion can be seen in Table 5.

Table 5: BGR to RGB Convesion Results

Original Image	RBG	RGB
		
		
		
		
		

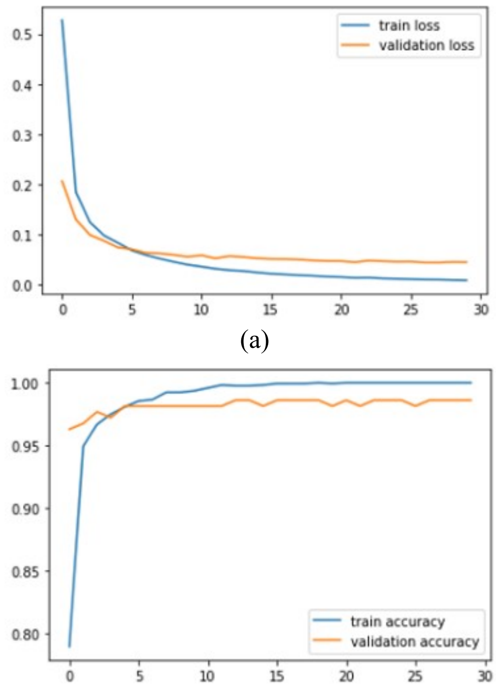
4.3 Image Classifier

This study conducted several experiments according to the research stages and experiments using several Transfer Learning methods [39] including MobileNetV2, InceptionResNetV2, VGG16 and Inception V3.

4.3.1 Model Testing with MobileNetV2 Transfer Learning

Testing of the MobileNetV2 transfer learning model was carried out with 2 experiments. The first experiment was carried out using the MobileNetV2 transfer learning method with the original dataset or without augmentation. The second experiment used the MobileNetV2 transfer learning method with the augmented dataset on the healthy leaf class. Figure 6 shows a graph of the performance of the loss and accuracy results of the MobileNetv2 architecture with the original dataset or not augmented, where the blue line shows the training data and the orange color shows the validation data. In this test, the MobileNetV2 model has been trained using a potato leaf dataset. The classification results on training data and data validation using the MobileNetV2 transfer learning model show good performance. If seen in the graph of the loss value (a), the training loss and validation loss decreased steadily. Likewise, if you look at the graph of the accuracy value (b), the accuracy value in the training data and data validation increases steadily.

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



(a)

(b)

Figure 6: The results of the MobileNetV2 classification model without Augmentation (a) Loss (b) Accuracy

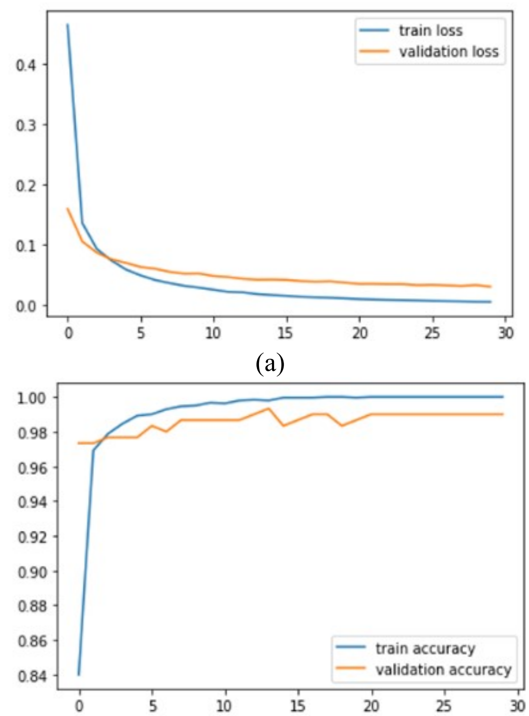
Considering that the validation loss value is stable after the 5th epoch and does not decrease until the 30th epoch and the highest accuracy value obtained is at the 14th epoch, the maximum number of epochs will be chosen of 30. The batch size used is 32, with the adam optimizer, the loss function used is categorical cross entrophy and the checkpoint model uses save best only.

The results of the evaluation and confusion matrix prediction of the MobileNetV2 model on the testing data is shown by Table 6. The prediction results of the MobileNetV2 model on the testing data get an accuracy value of 97.6%. From Table 6 it can be seen that the model gives good predictions in the Early Blight class. While in the Late Blight class there is 1 error, namely 1 image is detected as Healthy and in the Healthy class 3 images are detected Late Blight.

Table 6: Confussion Matrix MobileNetv2 without Augmentation

	Early Blight	Late Blight	Healthy
Early Blight	103	0	0
Late Blight	1	99	1
Healthy	0	3	9

From the results of the confusion matrix shown in Table 6, it can be concluded that the training dataset for the healthy class is very lacking so there are many errors in detection. Therefore, augmentation is carried out in healthy classes to handle the occurrence of data imbalances. Figure 7. shows a graph of the performance loss and accuracy results from the MobileNetv2 architecture that was trained with an authenticated dataset. The classification results on training data and data validation using the MobileNetV2 and augmentation transfer learning model showed improved performance. If seen in the graph of the loss value (a), the training loss and validation loss decreased steadily. Likewise, if you look at the graph of the accuracy value (b), the accuracy value in the training data and data validation increases steadily.



(a)

(b)

Figure 7: The results of the MobileNetV2 classification model with Augmentation (a) Loss (b) Accuracy

The results of the evaluation and confusion matrix prediction of the MobileNetV2 model on the testing data is shown by Table 7. The prediction results of the MobileNetV2 model with a dataset that has been augmented received an accuracy value of 99.6%. From Table 7, it can be seen that the model gives good predictions for the Early Blight and Healthy classes. There is only one error, namely one Late Blight image is detected as Healthy.

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

Table 7: Confussion Matrix MobileNetv2 with Augmentation

	Early Blight	Late Blight	Healthy
Early Blight	107	0	0
Late Blight	0	99	1
Healthy	0	0	93

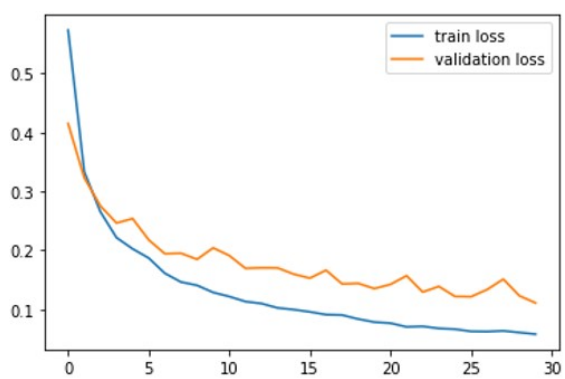
The results of the comparison of the accuracy values obtained using the Confusion Matrix is shown by in Table 8. The accuracy of the classification with the MobileNetV2 transfer learning model using the original dataset resulted in 97.6%, while the MobileNetV2 transfer learning model using the augmented dataset in the Healthy class produced an accuracy of 99.6%.

Table 8: Comparison of Mobilenetv2 Performance with Distribution of Original Data and Augmented Data

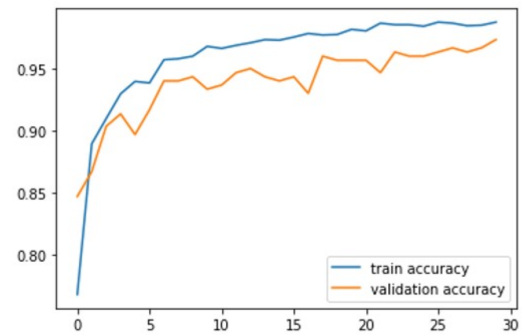
Data distribution	Correct Predic-tion	Wrong Prediction	Accu-racy
Original	211	5	97,6%
Augmentation Results	299	1	99,6%

4.3.2 Model Testing with Transfer Learning InceptionResNetV2, VGG16 and Inception V3

The next experiment was to classify using the Transfer Learning model InceptionResNetV2, VGG16 and InceptionV3 using the same hyperparameters as those applied to the experiment using transfer learning MobileNetV2.



(a)



(b)

Figure 8: The results of the InceptionResNetV2 Transfer Learning Classification Model (a) Loss and (b) Accuracy

Figure 8 shows a graph of the performance loss and accuracy of the InceptionResNetV2 architecture. If you look at the graph of the loss value (a), the training loss and validation loss are less stable. Likewise, if you look at the graph of the accuracy value (b), the accuracy value in the training data and data validation increases less stable.

The results of the evaluation and confusion matrix predictions of the InceptionResnetV2 model is shown by Table 9. The prediction results obtained an accuracy value of 96.6%. From Table 9 it can be seen that the model gives a poor prediction. In the Early Blight class, there is one image detected in the Late Blight class. Then in the Late Blight class there are six images that are not correctly detected, namely three images detected Early Blight and three images detected healthy. Then in the Healthy class there are three images that were detected incorrectly, namely one image detected Early Blight and 2 classes detected Late Blight.

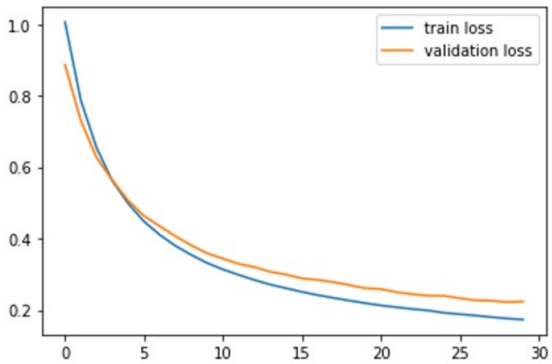
Table 9: Confusion Matrix Model Inceptionresnetv2

	Early Blight	Late Blight	Healthy
Early Blight	106	1	0
Late Blight	3	94	3
Healthy	1	2	90

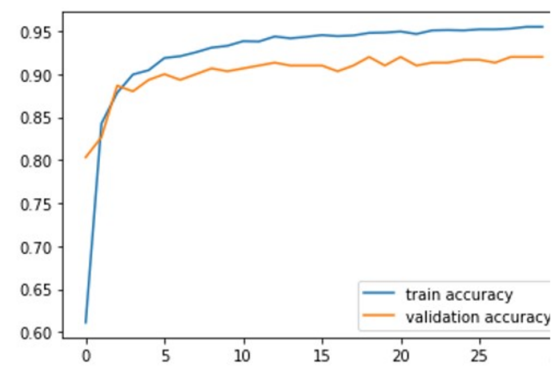
The next experiment was to classify using the Transfer Learning VGG16 model. Figure 9 shows a graph of the performance of the loss and accuracy results of the VGG16 architecture. If you look at the graph of the loss value (a), the training loss and validation loss decreased steadily. Likewise, if you look at the graph of the accuracy value (b), the accuracy value in the training data and data validation increases steadily.

Journal of Theoretical and Applied Information Technology
31st August 2023. Vol.101. No 16

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



(a)



(b)

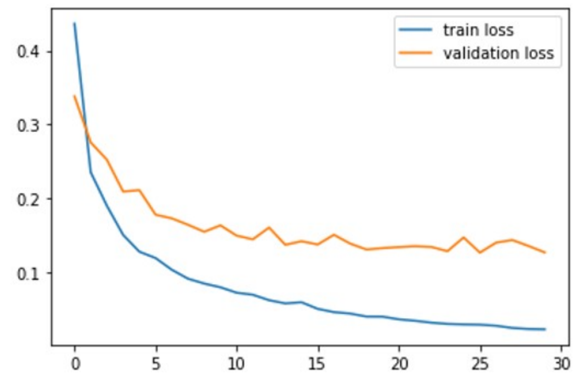
Figure 9: The Results of The VGG16 Classification Model (A) Loss and (B) Accuracy

The results of the evaluation and confusion matrix prediction of the VGG16 model is shown by Table 10. The prediction results obtained an accuracy value of 96.3%. From Table 10 it can be seen that the model gives a poor prediction.

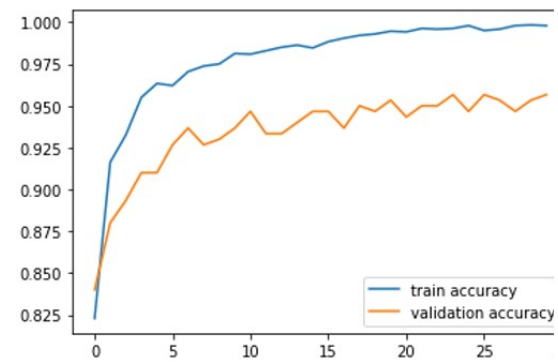
Table 10: Confusion Matrix Model VGG16

	Early Blight	Late Blight	Healthy
Early Blight	103	4	0
Late Blight	4	94	2
Healthy	0	1	92

The next experiment was to classify using the Transfer Learning InceptionV3 model. Figure 10. shows a graph of the performance loss and accuracy results from the InceptionV3 architecture. If you look at the graph of the loss value (a), the training loss and validation loss have very large differences. Likewise, if you look at the graph of the accuracy value (b). The wide value difference between the accuracy and loss values is evidence that the classification performance is overfitting and this is a poor result of the classification model.



(a)



(b)

Figure 10: The Results of the Inceptionv3 Classification Model (A) Loss and (B) Accuracy

The results of the evaluation and confusion matrix predictions of the InceptionV3 model is shown by Table 11. The prediction results obtained an accuracy value of 95.6%. From Table 11 it can be seen that the model gives a poor prediction.

Table 11: Confusion Matrix Model Inceptionv3

	Early Blight	Late Blight	Healthy
Early Blight	102	4	1
Late Blight	3	95	2
Healthy	0	3	90

The results of the comparison of the accuracy values obtained using the Confusion Matrix is described in Table 12. The accuracy of the classification using the MobileNetV2 transfer learning model resulted in 99.6% accuracy, InceptionResNetV3 got 96.6% accuracy, VGG16 produces 96.3% accuracy and InceptionV3 produces 95.6% accuracy.

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

Table 12: Comparison of Testing Results

Classification Model	Correct Prediction	Wrong Prediction	Accuracy	Time
MobileNetV2	299	1	99,6%	5m
InceptionResNetV2	292	8	96,6%	25m
VGG16	289	11	96,3%	14m

4.4 System Implementation

The implementation of the interface is an application of the model used to classify potato leaves and then applied to the android application. The android platform was chosen to make it easier to capture images, because on android you can directly use the camera to capture images of potato leaves. Some views of the application are shown in Figure 11, 12, and 13.

Figure 11: App View for Home

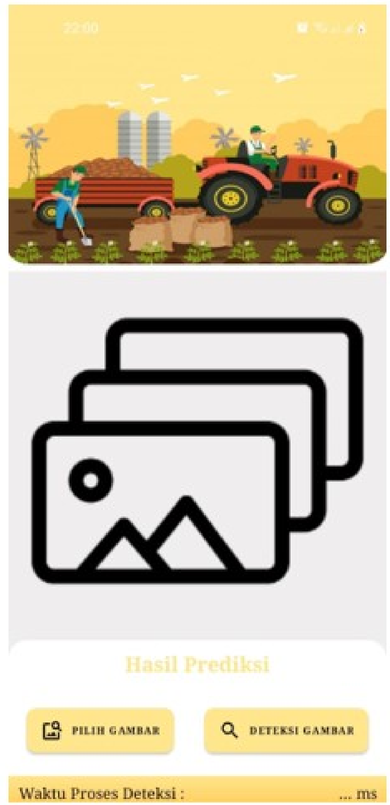


Figure 12: App View for Detect from Gallery

Figure 13: App View for Detect with Camera

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

4.5 Evaluation and Validation

At this stage, it is explained about the evaluation process from testing, how well the detection is with the android application that has been made. Table 13 is a description of the tested image.

Table 13: Description of Identified Image

Image	Class	Prediction Model	Result
	Early Blight	91,1%	Suitable
	Late Blight	97,3%	Suitable
	Healthy	99,9%	Suitable

From the discussion above, it can be seen that with augmentation, feature extraction using the MobileNetV2 transfer learning method can be carried out on potato leaf images well when compared to research [2], and an android application that can detect potato leaves can be designed automatically.

5. CONCLUSION

Based on the research that has been done, it can be concluded that the augmentation method can be used to handle data unbalances and this is proven by the increased performance of the model. The classification method used is a transfer learning approach. The model training phase is done by freezing the base model and adding several additional layers. In this study, 4 different types of transfer learning models were used to classify potato leaf diseases. The four models are MobileNetV2, InceptionResNetV2, VGG16 and Inception V3, then the evaluation is done by predicting the class of the testing data using confusion matrices to get calculations from the model assessment, namely accuracy. The model with the best performance in performing the classification is achieved by the MobileNetV2 transfer learning model using a dataset that has been augmented in the healthy class with an accuracy value of 99.6% and finally the implementation of the interface by applying the methods used for detection of potato leaf disease can be applied on android application. Detection of potato leaf disease can be achieved automatically.

In further research, the augmentation process is expected to be applied to all classes. In order for the data set to be large in order to produce better model performance, fine-tuning is carried out in conducting training to prevent overfitting and in the application, it is hoped that the segmentation process will be added to the test. This minimizes the background image being read.

ACKNOWLEDGEMENTS

Author thanks to Universitas Nusa mandiri, Universitas Multimedia Nusantara and Universitas Bina sarana Informatika for the support in this research.

REFERENCES:

[1] Kementrian Pertanian, "Kentang Sangkuriang Agrihorti Tahan Penyakit Busuk Daun," 2019. [Online]. Available: https://www.litbang.pertanian.go.id/info-teknologi/3836/.

[2] F. Islam, M. N. Hoq, and C. M. Rahman, "Application of Transfer Learning to Detect Potato Disease from Leaf Image," *2019 IEEE Int. Conf. Robot. Autom. Artif. Internet-of-Things, RAAICON 2019*, pp. 127–130, 2019, doi: 10.1109/RAAICON48939.2019.53.

[3] R. A. Sholihati, I. A. Sulistijono, A. Risnumawan, and E. Kusumawati, "Potato Leaf Disease Classification Using Deep Learning Approach," *IES 2020 - Int. Electron. Symp. Role Auton. Intell. Syst. Hum. Life Comf.*, pp. 392–397, 2020, doi: 10.1109/IES50839.2020.9231784.

[4] Pusat Penelitian dan Pengembangan Hortikultura, "Pengenalan Penyakit yang Menyerang Pada Tanaman Kentang," pp. 1–8.

[5] H. P. Susetyo, "Penyakit Busuk Daun Kentang," 2017. [Online]. Available: http://hortikultura.pertanian.go.id/?p=2025.

[6] 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.

[7] R. D. Puspitasari, "Pertanian Berkelanjutan Berbasis Revolusi Industri 4.0," *J. Layanan Masy. (Journal Public Serv.*, vol. 3, no. 1, p. 26, 2020, doi: 10.20473/jlm.v3i1.2019.26-28.

[8] M. Hassaballah and A. I. Awad, *Deep Learning in Computer Vision: Principles and Applications*. CRC Press, 2020.

[9] Z. Song, "English speech recognition based on deep learning with multiple features," *Computing*, vol. 102, no. 3, pp. 663–682, 2020, doi: 10.1007/s00607-019-00753-0.
[10] L. Lausen, "GluonCV and GluonNLP : Deep Learning in," vol. 21, pp. 1–7, 2020.
[11] H. F. Pardede *et al.*, "Plant diseases detection with low resolution data using nested skip connections," *J. Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00332-7.
[12] H. Wibowo, I. S. Sitanggang, M. Mushthofa, and H. A. Adrianto, "Large-Scale Oil Palm Trees Detection from High-Resolution Remote Sensing Images Using Deep Learning," *Big Data Cogn. Comput.*, vol. 6, no. 3, p. 89, 2022, doi: 10.3390/bdcc6030089.
[13] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agric.*, vol. 145, no. September 2017, pp. 311–318, 2018, doi: 10.1016/j.compag.2018.01.009.
[14] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *J. Physiol.*, vol. 195, no. 1, pp. 215–243, 1968.
[15] D. Choe, E. Choi, and D. K. Kim, "The Real-Time Mobile Application for Classifying of Endangered Parrot Species Using the CNN Models Based on Transfer Learning," *Mob. Inf. Syst.*, vol. 2020, 2020, doi: 10.1155/2020/1475164.
[16] Y. Altuntaş, Z. Cömert, and A. F. Kocamaz, "Identification of haploid and diploid maize seeds using convolutional neural networks and a transfer learning approach," *Comput. Electron. Agric.*, vol. 163, no. 40, pp. 1–11, 2019, doi: 10.1016/j.compag.2019.104874.
[17] Z. Huang, Z. Pan, and B. Lei, "Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data," *Remote Sens.*, vol. 9, no. 9, pp. 1–21, 2017, doi: 10.3390/rs9090907.
[18] F. Natalia *et al.*, "Automated measurement of anteroposterior diameter and foraminal widths in MRI images for lumbar spinal stenosis diagnosis," *PLoS One*, vol. 15, no. 11 November, 2020, doi: 10.1371/journal.pone.0241309.
[19] B. Liu, C. Tan, S. Li, J. He, and H. Wang, "A Data Augmentation Method Based on Generative Adversarial Networks for Grape Leaf Disease Identification," *IEEE Access*, vol. 8, pp. 102188–102198, 2020, doi: 10.1109/ACCESS.2020.2998839.
[20] R. Hu, S. Zhang, P. Wang, G. Xu, D. Wang, and Y. Qian, "The identification of corn leaf diseases based on transfer learning and data augmentation," *ACM Int. Conf. Proceeding Ser.*, pp. 58–65, 2020, doi: 10.1145/3403746.3403905.
[21] 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, doi: 10.1016/j.compeleceng.2019.04.011.
[22] B. Min, T. Kim, D. Shin, and D. Shin, "Data Augmentation Method for Plant Leaf Disease Recognition," *Appl. Sci.*, vol. 13, no. 3, Feb. 2023, doi: 10.3390/app13031465.
[23] U. Barman, D. Sahu, G. G. Barman, and J. Das, "Comparative Assessment of Deep Learning to Detect the Leaf Diseases of Potato based on Data Augmentation," *2020 Int. Conf. Comput. Perform. Eval. ComPE 2020*, pp. 682–687, 2020, doi: 10.1109/ComPE49325.2020.9200015.
[24] R. Mahum *et al.*, "A novel framework for potato leaf disease detection using an efficient deep learning model," *Hum. Ecol. Risk Assess.*, vol. 29, no. 2, pp. 303–326, 2023, doi: 10.1080/10807039.2022.2064814.
[25] D. Tiwari, M. Ashish, N. Gangwar, A. Sharma, S. Patel, and S. Bhardwaj, "Potato Leaf Diseases Detection Using Deep Learning," *Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, no. January, pp. 461–466, 2020, doi: 10.1109/ICICCS48265.2020.9121067.
[26] T.-Y. Lee, I.-A. Lin, J.-Y. Yu, J. Yang, and Y.-C. Chang, "High Efficiency Disease Detection for Potato Leaf with Convolutional Neural Network," *SN Comput. Sci.*, vol. 2, no. 4, 2021, doi: 10.1007/s42979-021-00691-9.
[27] J. Rozaqi, A. Sunyoto, and R. Arief, "Implementation of Transfer Learning in the Convolutional Neural Network Algorithm for Identification of Potato Leaf Disease," vol. 1, no. 1, 2021, doi: 10.21070/pels.v1i1.820.
[28] F. Chollet, *Introduction to Keras*. 2017.
[29] D. Ongkadinata and F. P. Putri, "Quality and size assessment of quantized images using K-means++ clustering," *Bull. Electr. Eng. Informatics*, vol. 9, no. 3, pp. 1183–1188, 2020, doi: 10.11591/eei.v9i3.1985.
[30] M. R. Karim and P. Pujari, *Practical Convolutional Neural Networks: Implement advanced deep learning models using Python*. 2018.
[31] Y. Yu, H. Lin, J. Meng, X. Wei, H. Guo, and Z. Zhao, "Deep transfer learning for modality

classification of medical images," *Inf.*, vol. 8, no. 3, 2017, doi: 10.3390/info8030091.

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

[33] Y. Chen, C. Li, L. Gong, X. Wen, Y. Zhang, and W. Shi, "A deep neural network compression algorithm based on knowledge transfer for edge devices," *Comput. Commun.*, vol. 163, no. August, pp. 186–194, 2020, doi: 10.1016/j.comcom.2020.09.016.

[34] Y. S. Tan, K. M. Lim, C. Tee, C. P. Lee, and C. Y. Low, "Convolutional neural network with spatial pyramid pooling for hand gesture recognition," *Neural Comput. Appl.*, vol. 33, no. 10, pp. 5339–5351, 2021, doi: 10.1007/s00521-020-05337-0.

[35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.

[36] 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, 2016, doi: 10.1109/CVPR.2016.90.

[37] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," 2017.

[38] Tensorflow, "TensorFlow lite converter," 2019. .

[39] 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, doi: 10.14419/ijet.v7i4.11.20781.