JOIV : Int. J. Inform. Visualization, 7(2) - June 2023 535-541

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

# A Combination of Transfer Learning and Support Vector Machine for Robust Classification on Small Weed and Potato Datasets

Faisal Dharma Adhinata [a,*], Nur Ghaniaviyanto Ramadhan [a], Muhammad Dzulfikar Fauzi [b], Nia Annisa Ferani Tanjung [a]

[a] *Department of Software Engineering, Faculty of Informatics, Institut Teknologi Telkom Purwokerto, Indonesia*
[b] *Department of Informatics, Faculty of Information Technology and Business, Institut Teknologi Telkom Surabaya, Indonesia*
*Corresponding author: [*]faisal@ittelkom-pwt.ac.id*

***Abstract*— Agriculture is the primary sector in Indonesia for meeting people's daily food demands. One of the agricultural commodities that replace rice is potatoes. Potato growth needs to be protected from weeds that compete for nutrients. Spraying using pesticides can cause environmental pollution, affecting cultivated plants. Currently, smart agriculture is being developed using an Artificial Intelligence (AI) approach to classifying crops. The classification process using AI depends on the number of datasets obtained. The number of datasets obtained in this research is not too large, so it requires a particular approach regarding the AI method used. This research aims to use a combination of feature extraction methods with local and deep feature approaches with supervised machine learning to classify small datasets. The local feature method used in this research is Local Binary Pattern (LBP) and Histogram of Oriented Gradients (HOG), while the deep feature method used is MobileNet and MobileNetV2. The famous Support Vector Machine (SVM) uses the classification method to separate two data classes. The experimental results showed that the local feature HOG method was the fastest in the training process. However, the most accurate result was using the MobileNetV2 deep feature method with an accuracy of 98%. Deep features produced the best accuracy because the feature extraction process went through many neural network layers. This research can provide insight into how to analyze a small number of datasets by combining several strategies.**

***Keywords*— Smart agriculture; weed; potatoes; small dataset; machine learning; local and deep feature.**

*Manuscript received 23 Jun. 2022; revised 28 Aug. 2022; accepted 8 Nov. 2022. Date of publication 30 Jun. 2023.*


## I. INTRODUCTION

Indonesia is an agrarian country that is rich in natural wealth because of vast agricultural land or plantations. One of the agricultural products is potato, and the demand for its crops continues to rise in tandem with Indonesia's growing population [1]. In the land around potato plants, weeds often grow, which farmers must eradicate. Weeds that are not eradicated can reduce agricultural yields. Weed plants compete with potato plants to get the nutrients contained in the soil [2]. Farmers usually spray pesticides randomly on land with weeds. This random spraying can also affect potato plants as cultivated plants. The use of pesticides can also pollute the environment by leaving chemical residues [3], and weeds can be resistant to pesticides [4]. Therefore, it is essential to keep cultivated plants growing well by applying modern agricultural technology, especially in identifying weeds and cultivated plants.

Accurate methods in the classification process usually use deep learning. However, deep learning usually requires large amounts of data [5][6]. In the proposed research, the data obtained is not too much. This not too much data can result in inaccurate classification accuracy. In several previous studies, this not too much data was carried out by a data augmentation process to improve accuracy results [7]. The classification process used uses the deep learning method. Before the data augmentation process, the resulting accuracy was around 82%-86%. After augmenting the data, the accuracy results are 89%-91%. In this research, the amount of data was less than 1000. In the following study, using two classes of datasets carried out by the data augmentation process resulted in an accuracy of 74%-90% [8]. The classification process also uses deep learning. The results of this classification are not optimal. Therefore, we use an approach to the feature extraction method used. This research aims to use an appropriate feature extraction method to classify small datasets.

Recently, machine learning has been popularly used in various research topics. Machine learning is part of Artificial Intelligence that learns from datasets that are matched with new data [9]. The data in this research has been labelled, so the machine learning used is supervised learning. The supervised learning classification method needs good features for good accuracy. Feature extraction is critical in getting an object's feature interest [10]. In addition, the small amount of data requires a good combination of feature extraction and supervised learning. Some feature extraction methods are Local Binary Pattern (LBP), Histogram of Oriented Gradients (HOG), and deep features. A researcher by Hussein [11] used the LBP method for human ear recognition as human identification. The results showed an accuracy of 99%. Then researchers Mu'jizah and Novitasari [12] used the HOG method for breast cancer classification. The classification results show accuracy above 98%. Recently, deep features have become popular in combination with machine learning. Researchers Michele et al. [13] used MobileNetV2 for palmprint recognition. The results of research using this method produce 100% accuracy. The proposed research will test the three methods, namely LBP, HOG, and deep feature MobileNet, for the feature extraction process of weeds and potatoes.

After the feature extraction stage, the classification stage also plays an essential role in producing an accurate model. In the proposed research, there are two classes: weed and potato. The supervised learning method that is powerful enough to separate the two classes is the Support Vector Machine (SVM) [14][15]. In this research, the kernel parameters of the SVM method will be evaluated. It is hoped that this research can provide insight into the appropriate feature extraction method for classifying a small number of datasets.

## II. Material and Methods

The weed and potato classification system stages begin with the input of the weed and potato dataset. Figure 1 is a step in the proposed research. The raw dataset is pre-processed by removing the non-vegetable background. Next, resize the image so that the data size is uniform. During the feature extraction stage, the pre-processing step outcomes are utilized. The Local Binary Pattern, the Histogram of Oriented Gradient, the MobileNet deep feature, and the MobileNetV2 deep feature will be employed in the feature extraction stage to extract the leaf object features. The classification process relies on the vectors produced due to feature extraction. The next step is to divide the vector data into data for training and data for testing. The Support Vector Machine approach is utilized here for the classification stage. The last stage is the evaluation of the classification system made. All these method flows are made in Python version 3 programming language.

### *A. Data Acquisition*

In this research, the data used came from Kaggle [16], which Ali Hassan collected. The data collected was small, namely data on weeds totaling 242 data, while the data on potato plants amounted to 169 data. These data were obtained from potato plantation fields captured above. All data used in this research were colored Red, Green, and Blue (RGB). The raw data was still not homogeneous in size. Figure 2 shows an example of the data used in this research.

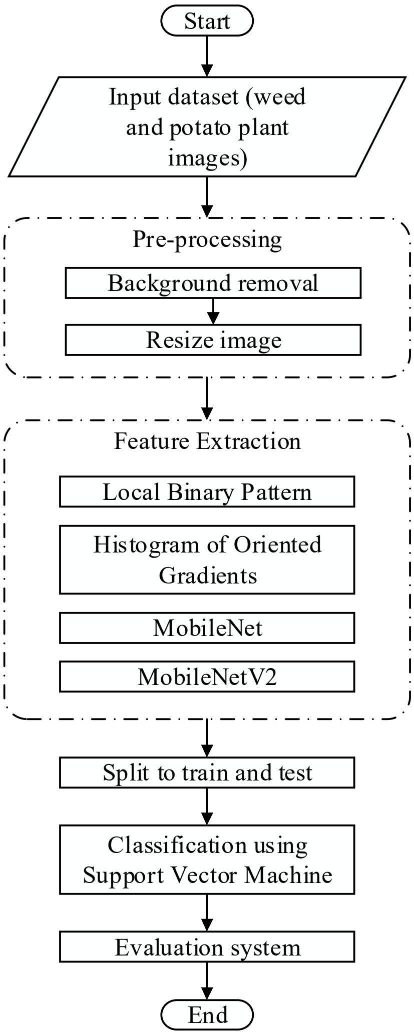
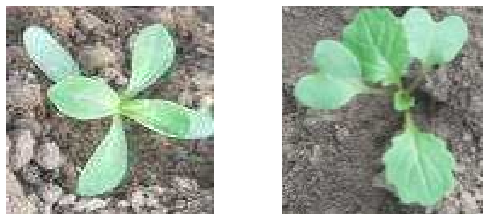



Fig. 1 Research stage diagram



Weed plant

Potato plant

Fig. 2 Example of dataset

### B. *Pre-processing Data*

Before processing the raw data into the feature extraction stage, a pre-processing process is carried out. The raw data still shows the soil background, which should be removed because the object in this research is the leaf. Figure 3 shows the pre-processing process of the raw dataset of this research.

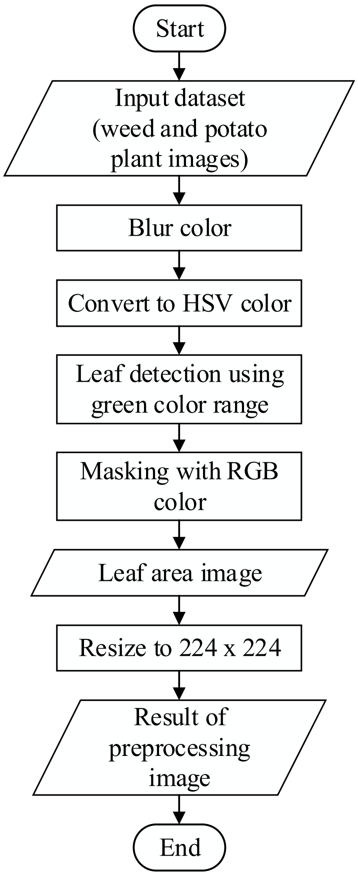



Fig. 3 Pre-processing result

Removing the background uses a blur color and converting the color to Hue Saturation Value (HSV). Detection of leaf objects uses a light green (25, 40, 50) to dark green (75, 255, 255) color range. The results of pre-processing are shown in Figure 4. The results of background removal are masked to raw data so that the color component used is RGB. The result of the pre-processing process is that there is still a small area of soil attached to the leaf. It is because the color of the soil is almost refracted by light which makes the color resemble a leaf object. After the background removal, resize the object to a size of $224\ x\ 244$ to make all image sizes uniform. This size adjustment is also a deep feature input adjustment. The results of the pre-processing data are used for input in the feature extraction stage.

Weed plant Potato plant

Fig. 4 Result of pre-processing stage

### C. *Feature Extraction*

The features of an object are its distinguishing characteristics [17]. Features are divided into two types of natural image features: brightness and object edges. Meanwhile, artificial features are those obtained from image operations such as the gray-level histogram. So, feature extraction identifies characteristics that distinguish one object from others [18]. The process of extracting a feature from a form and assessing the value acquired for use in the subsequent phase is known as feature extraction. The purpose of feature extraction is to locate relevant feature areas within an image based on the image's inherent features and the application that will be using the image. The region can be defined in either a global or a local context. It can be recognized using various features, including its shape, texture, size, intensity, statistical aspects, and many more [19]. The process of feature extraction begins with the counting of the number of points or pixels that are encountered during each check [20]. Next, the checking process is carried out in various directions by tracing the Cartesian coordinates of the digital image being analyzed. These directions include the vertical, horizontal, right, and left diagonal. The background removal results are used for the feature extraction process. This research will evaluate four feature extraction methods: Local Binary Pattern, Histogram of Oriented Gradient, deep feature MobileNet, and MobileNetV2.

*1) Local Binary Pattern:* The Local Binary Pattern is an image-clarifying descriptor based on image texture. Timo Ojala and David Harwood introduced the Local Binary Pattern in 1992 at the University of Maryland [21]. The Local Binary Pattern compares the binary value of the central pixel to the values of the eight surrounding pixels. [22]. Therefore, the central binary value of a 3x3 image is compared to its surrounding values. The value is set to 1 if the intensity of the central pixel is greater than the central binary. If the value is less than the central binary, it is set to 0 [23]. With 8 pixels surrounding it, there are $2^8 = 256$ possible Local binary pattern code combinations. Figure 5 shows the ratio of the center pixel to its neighbor.

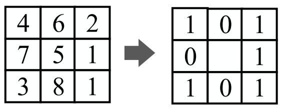



Fig. 5 Comparison of the center pixel with its neighbor

The first step in constructing a Local Binary Pattern is to compare the binary value of the pixel in the center of the image to the values of the eight surrounding pixels [24]. Next, calculate the Local Binary Pattern value for the pixel in the middle starting from the pixels around it in a clockwise or counterclockwise manner, ensuring it must be consistent. In this research, we use clockwise calculation. For example, 3x3 means there are eight binary tests. Then, the binary test results are stored in an 8-bit array converted to decimal. The binary test result is the sum of clockwise numbers, which it will place on the center pixel. The results of the LBP are shown in Figure 6.

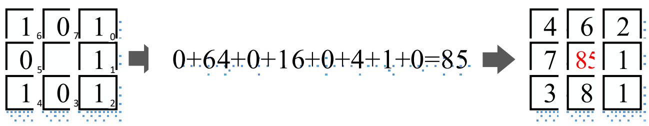



Fig. 6 Result of Local Binary Pattern

This value will be stored in the Local Binary Pattern 2D output array. Then it can be visualized, referred to as a thresholding process, namely collecting binary and storing decimal values in the output Local Binary Pattern array repeated for each pixel in the input image.

*2) Histogram of Oriented Gradients:* There are four stages in performing feature extraction with this Histogram of Oriented Gradients, first doing a gradient calculation for the input image [25]. The feature extraction process with HOG is shown in Figure 7. Gradient calculation is done by convolution with gradient operator $[-1\ 0\ 1]$. Then transform the gradient into axis coordinates with an angle between $0^0$ to $180^0$, called gradient orientation.

Second, the image is divided into cells measuring 8x8 pixels. Then the image is further divided into blocks measuring $2x2$ cells. Third, the histogram value is calculated for each cell. The block feature is obtained from a series of 4 histograms of the four cells that make up the block. Then normalization is carried out on each block feature that has been obtained. Fourth, the results of the normalization of all block features are combined into one HOG feature [26].

Then this HOG feature is normalized again with the Euclidean norm. The result of the feature extraction process with HOG is in the form of a matrix measuring $(M-b+1)x(N-b+1)$ rows and 36 columns. $M$ and $N$ are rows and columns in the matrix. Then, $b$ is the value of the feature block obtained from equation 1. Then, the variable $e$ is a small number to prevent division by 0.

$$b = \frac{b}{\sqrt{b^2+e}} \tag{1}$$

Number 36 is obtained from the number of features in each block. Because one block comprises four cells, and one comprises a histogram with nine bins [27]. The value of each bin represents one feature.

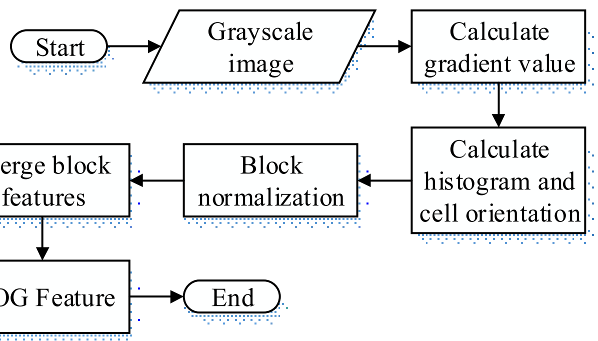



Fig. 7 HOG process

*3) Deep feature MobileNet:* MobileNet is one of the architectures of convolutional neural networks. The advantage of MobileNet is that it has the same convolution filter thickness as the image, making it more space-efficient than the model created. MobileNet proposes depth-wise and pointwise convolutions that can be separated in depth, i.e., two standard convolutions, namely depth-wise convolutions, and pointwise convolutions. The purpose of this layer is to reduce computations/to have fewer parameters, resulting in a smaller model size [28]. Figure 8 shows the difference between depth-wise convolution and standard convolution.

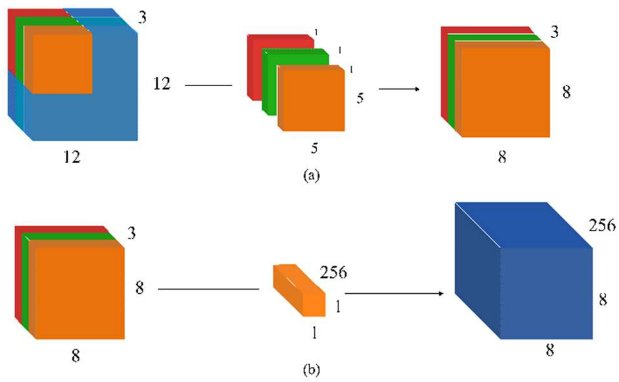



Fig. 8 Convolution process, a) Depth-wise convolution, b) standard convolution

In standard convolution, an input image is convoluted by a filter to generate a feature map. In contrast, depth-wise convolution produces a feature map with the same number of channels as the number of filters, where the number of filters equals the number of channels in the input image. In the depth-wise convolution stage, there are three filters $5x5x1$ that are shifted $8x8$ times; in other words, it means $3x5x5x1x8x8 = 4{,}800$. In the pointwise convolution stage, there are 256 filters $1x1x3$, which are shifted 8x8 times or, in another sense, means $256x1x1x3x8x8 = 49{,}152$. So the total multiplication performed during the convolution process in this depth-wise separable convolution is $4{,}800 + 49{,}152 = 52{,}952$ [29].

MobileNetV2 extends the previous MobileNetV1, which supports visual recognition, including classification, detection, and segmentation. MobileNetV2 is constructed using efficient deep separable convolutions as building blocks. MobileNetV2 introduces two new features: linear bottlenecks, which act as bottlenecks between layers, and shortcut connections, which act as links between linear bottlenecks. The bottlenecks feature compares the models' input and output in opposition. In addition, the model's capacity to transition from lower-level concepts such as pixels to image categories and higher-level descriptors is stored in the model's inner layer, the lowest layer. As a result, training more precisely and quickly using shortcuts, such as classic residual connections, is possible. In general, the MobileNetV2 model performs quicker while maintaining the same level of accuracy throughout the whole latency spectrum [30]. Thus, MobileNetV2 is very effective for object detection and segmentation. Figure 9 shows the difference between MobileNetV1 and MobileNetV2.

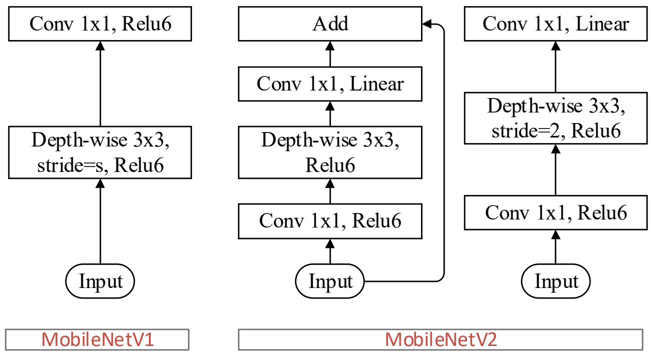



Fig. 9 Differences in MobileNetV1 and MobileNetV2

## *D. Classification*

The vector obtained from the feature extraction is used as a classification stage. Support Vector Machine (SVM) is the classification technique utilized in this research. The illustration of the SVM method in this research is shown in Figure 10.

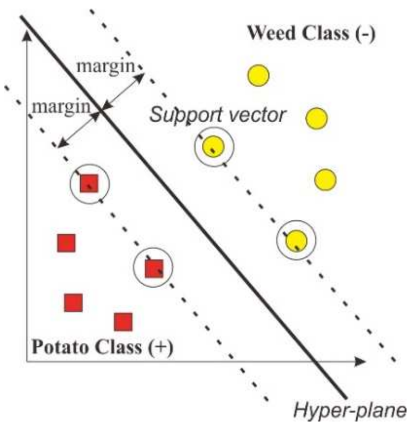



Fig. 10 The illustration of SVM method

The Support Vector Machine algorithm is one of those in the Supervised Learning category [31], which means that the data used for machine learning has already been labeled. As a result, during the decision-making process, the machine will label the testing data based on its characteristics. The Support Vector Machine method works by inserting the kernel concept into a high-dimensional space, which is especially useful in non-linear problems. The objective is to find a hyperplane or separator that minimizes the distance (margin) between data classes [32]. We can measure the margin and then find the maximum point to find the best hyperplane. The Support Vector Machine method produces the best hyperplane search process. This research will test several SVM kernels, Linear kernels, Polynomial kernels, and Radial Basis Function (RBF) or Gaussian kernels.

## *E. System Evaluation*

The system created needs to be evaluated to determine which method is the most accurate for processing small amounts of data. The LBP, HOG, MobileNetV1, and MobileNetV2 feature extraction methods will be evaluated for their accuracy and training speed. Evaluation of the feature extraction method using the default parameters of the SVM method. We will evaluate the recall, precision, and accuracy values. The best results will be used to evaluate kernel parameters on SVM. Recall, precision, and accuracy equations are shown in equations 2, 3, and 4 [33]. If the model predicts the positive class correctly, the result is a true positive (TP). In the same way, true negative (TN) is what happens when the model predicts the negative class correctly. When the model wrongly predicts the positive class, this is called a false positive (FP). When it wrongly predicts the negative class, this is called a false negative (FN).

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

# III. Results and Discussion

The pre-processing, feature extraction, classification results using potato and weed plant objects, and the discussion are all included in the results and discussion section. This study's computer configuration included an Intel Core i3 9100F processor and 8GB RAM. NVIDIA GT 1030 was used as the graphics card.

## *A. Pre-processing Result*

The pre-processing result uses segmentation that produces a leaf image with a black background, and the objects around it appear black. This segmentation is used to focus only on the leaf area, which is carried out by the feature extraction process. The segmentation results are shown in Figure 4. In this study, the calculation of the computational time required for the leaf area segmentation stage was also carried out. The computational time required for the execution of the segmentation stage is 0.02 s per image. These results indicate that the segmentation process runs quite fast.

## *B. Feature Extraction Result*

In the feature extraction stage, we conducted experiments on four feature extraction methods: LBP, HOG, MobileNet, and MobileNetV2. We evaluated the results of precision, recall, and accuracy. We used results from split data training and testing in this experiment. The best results will be used for testing the classification phase using SVM. This research also recorded processing time at the feature extraction stage.

*1. LBP Results:* The first feature extraction used the LBP method. The vector features were obtained from the LBP operations taken from the histogram results. In this research, the value of the radius parameter used was 8, while the number of neighboring points was 24. The radius utilized in the construction of the Circular Local Binary Pattern. Then the method used was uniform. The results of the experiment using LBP are shown in Table 1.

TABLE I
THE EXPERIMENT RESULT USING LBP

| **Split Data** | | **Precision** | **Recall** | **Accuracy** |
|---|---|---|---|---|
| **Train (%)** | **Test (%)** | | | |
| 50 | 50 | 0.725 | 0.64 | 0.67 |
| 60 | 40 | 0.765 | 0.665 | 0.69 |
| 70 | 30 | 0.77 | 0.64 | 0.65 |
| 80 | 20 | 0.775 | 0.655 | 0.63 |
| 90 | 10 | 0.765 | 0.63 | 0.6 |

Based on Table 1, all experiments yielded an accuracy below 0.7. Then, all precision results were higher than recall results. A system with high precision but low recall returns very few results, but most of its predicted labels match the training labels. The results obtained using the LBP method were not optimal for feature extraction with small datasets characterized by an accuracy below 0.7.

2. *HOG Results:* The second feature extraction used the HOG method. The parameter configuration used in this research used the number of bins in eight orientations. Then, the pixel size in each cell was 10x10, and the number of cells in each block was 1x1. The experimental results using the HOG method are shown in Table 2.

TABLE II
THE EXPERIMENT RESULT USING HOG

| **Split Data** | | **Precision** | **Recall** | **Accuracy** |
|---|---|---|---|---|
| **Train (%)** | **Test (%)** | | | |
| 50 | 50 | 0.805 | 0.78 | 0.79 |
| 60 | 40 | 0.855 | 0.83 | 0.84 |
| 70 | 30 | 0.88 | 0.84 | 0.85 |
| 80 | 20 | 0.865 | 0.85 | 0.84 |
| 90 | 10 | 0.83 | 0.825 | 0.81 |

The use of the HOG method produced a quite good accuracy. Even 70% of training data and 30% of testing produced 85% accuracy. The resulting precision and recall values were also quite balanced. However, precision results were still slightly higher than recall. Precision is more important than recall, which results in fewer False Positives but more False Negatives. These results indicate that the HOG method is quite suitable for small datasets.

3. *Deep Feature Results:* The last feature extraction method tested was using the deep feature. The deep feature consisted of many layers, but at this stage of feature extraction, using a layer before the classification layer. So, we took the feature extraction layer. The number of parameters of the MobileNet and MobileNetV2 models is not the same; in MobileNet, the number is 4.3M, while in MobileNetV2, the number is 3.5M [34]. Tables 3 and 4 show the experimental results using MobileNet and MobileNetV2.

TABLE III
THE EXPERIMENT RESULT USING MOBILENET

| **Split Data** | | **Precision** | **Recall** | **Accuracy** |
|---|---|---|---|---|
| **Train (%)** | **Test (%)** | | | |
| 50 | 50 | 0.925 | 0.91 | 0.92 |
| 60 | 40 | 0.92 | 0.895 | 0.91 |
| 70 | 30 | 0.915 | 0.9 | 0.91 |
| 80 | 20 | 0.94 | 0.935 | 0.94 |
| 90 | 10 | 0.875 | 0.88 | 0.88 |

TABLE IV
THE EXPERIMENT RESULT USING MOBILENETV2

| **Split Data** | | **Precision** | **Recall** | **Accuracy** |
|---|---|---|---|---|
| **Train (%)** | **Test (%)** | | | |
| 50 | 50 | 0.93 | 0.89 | 0.91 |
| 60 | 40 | 0.9 | 0.86 | 0.88 |
| 70 | 30 | 0.945 | 0.91 | 0.93 |
| 80 | 20 | 0.955 | 0.925 | 0.94 |
| 90 | 10 | 0.965 | 0.94 | 0.95 |

Based on Table 3 and Table 4, the best results were obtained using the MobileNetV2 model. The MobileNetV2 model was also faster than MobileNet, which was 12.65 s with 20.13 s. MobileNetV2 enhanced the state-of-the-art of MobileNet performance models across a spectrum of model sizes and multiple tasks and benchmarks. It was a highly efficient feature extractor for object recognition and segmentation. Results from MobileNetV2 were twice as fast as MobileNet. In addition, the resulting accuracy was also better than MobileNetV2. When compared to the LBP and HOG methods, the MobileNetV2 feature extraction yielded the best results in this experiment. Therefore, this 90% training and 10% testing data configuration will be tested with kernel parameters on SVM.

### C. Plant Classification Results

Experiments at the classification stage using multiple kernel configurations on SVM. Several SVM kernels used in this experiment were linear, polynomial, and Radial Basis Function (RBF) kernels. Table 5 shows the experimental results using the SVM kernel.

TABLE V
THE EXPERIMENT RESULT OF CLASSIFICATION USING SVM

| **SVM Kernel** | **Precision** | **Recall** | **Accuracy** |
|---|---|---|---|
| Linear | 0.97 | 0.98 | 0.98 |
| Poly | 0.97 | 0.98 | 0.98 |
| RBF | 0.965 | 0.94 | 0.95 |

The results of precision and recall in the experiment were not much different. The best results were obtained using linear and polynomial kernels. The selection of the SVM kernel was highly dependent on the dataset used. Linear kernels were usually used for data that can be separated linearly. The small amount of data made the resulting vector not too large and complex, so the results from a linear kernel can produce the best accuracy.

### D. Discussion

The best accuracy results were obtained using the MobileNetV2 deep feature model. However, using deep features also took longer than the LBP and HOG methods. The feature extraction time on the LBP was 10.3 s, while the HOG was only 3.04 s. The fastest feature extraction time was the HOG method. However, in the case research of a small number of datasets, deep features resulted in a reasonably accurate accuracy of 98%. Deep features produced the most accurate accuracy because the training process went through many neural network layers on MobileNetV2. The feature extraction time required was also not too long, which was 12.65 s. The LBP and HOG methods in previous research [11][12] can produce good accuracy because of the large data. So, assuming the amount of data was not too large, we could employ deep features in tandem with supervised learning in this study.

## IV. CONCLUSIONS

Datasets for research are not always available in large quantities. When getting a small dataset, it is necessary to carry out specific steps to keep producing good accuracy. This research conducted experiments at the feature extraction stage with local and deep feature approaches. The method used is LBP and HOG as local features, while the deep feature method was MobileNet and MobileNetV2 models. This research used the SVM classification method.

The experimental results show that HOG is the fastest in the training process. In terms of accuracy, however, using MobileNetV2's deep feature model, which achieves 98% accuracy, is the best option. The results of this research can give insight into processing small datasets. Future research can use video data to be implemented in agriculture that is integrated with Raspberry Pi devices. The use of video data can also measure processing time performance allowing real-time method performance to be determined.

## ACKNOWLEDGMENT

This article was published with funding from the PEKERTI-DN Research Grant, which LPPM Institut Teknologi Telkom Purwokerto and Institut Teknologi Telkom Surabaya provided.

## REFERENCES

[1] A. Budi Setiawan and C. Inayati, "The Analysis of Production Factors and Income of Potato Farming," *Jejak*, vol. 13, no. 1, pp. 17–29, 2020, doi: 10.15294/jejak.v13i1.21965.

[2] N. E. Korres, *Agronomic weed control: A trustworthy approach for sustainable weed management*. Elsevier Inc., 2018. doi: 10.1016/B978-0-12-809881-3.00006-1.

[3] M. Riyaz, R. A. Shah, and K. Sivasankaran, "Pesticide Residues: Impacts on Fauna and the Environment," in *Biodegradation Technology of Organic and Inorganic Pollutants*, K. F. Mendes, R. N. de Sousa, and K. C. Mielke, Eds. Rijeka: IntechOpen, 2021. doi: 10.5772/intechopen.98379.

[4] J. L. Tang, D. Wang, Z. G. Zhang, L. J. He, J. Xin, and Y. Xu, "Weed identification based on K-means feature learning combined with convolutional neural network," *Computers and Electronics in Agriculture*, vol. 135, pp. 63–70, 2017, doi: 10.1016/j.compag.2017.01.001.

[5] I. H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Computer Science*, vol. 2, no. 6, pp. 1–20, 2021, doi: 10.1007/s42979-021-00815-1.

[6] F. J. Moreno-Barea, J. M. Jerez, and L. Franco, "Improving classification accuracy using data augmentation on small data sets," *Expert Systems with Applications*, vol. 161, p. 113696, 2020, doi: 10.1016/j.eswa.2020.113696.

[7] F. J. Moreno-Barea, J. M. Jerez, and L. Franco, "Improving classification accuracy using data augmentation on small data sets," *Expert Systems with Applications*, vol. 161, p. 113696, 2020, doi: 10.1016/j.eswa.2020.113696.

[8] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.

[9] J. A. Nichols, H. W. Herbert Chan, and M. A. B. Baker, "Machine learning: applications of artificial intelligence to imaging and diagnosis," *Biophysical Reviews*, vol. 11, no. 1, pp. 111–118, 2019, doi: 10.1007/s12551-018-0449-9.

[10] K. Kanagaraj and G. G. L. Priya, "Curvelet transform based feature extraction and selection for multimedia event classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 2, pp. 375–383, 2022, doi: 10.1016/j.jksuci.2018.11.006.

[11] A. K. Hussein, "Histogram of Gradient and Local Binary Pattern with Extreme Learning Machine Based Ear Recognition," *Journal of Southwest Jiaotong University*, vol. 54, no. 6, pp. 1–6, 2019.

[12] H. Mu'jizah and D. C. R. Novitasari, "Comparison of the histogram of oriented gradient, GLCM, and shape feature extraction methods for breast cancer classification using SVM," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 3, pp. 150–156, 2021, doi: 10.14710/jtsiskom.2021.14104.

[13] A. Michele, V. Colin, and D. D. Santika, "Mobilenet convolutional neural networks and support vector machines for palmprint recognition," *Procedia Computer Science*, vol. 157, pp. 110–117, 2019, doi: 10.1016/j.procs.2019.08.147.

[14] I. Candradewi, B. N. Prastowo, and D. Lathief, "Gender Classification from Facial Images Using Support Vector Machine," *Journal of Theoretical and Applied Information Technology*, vol. 97, pp. 2684–2692, 2019.

[15] F. D. Adhinata, A. Harjoko, and Wahyono, "Object Searching on Video Using ORB Descriptor and Support Vector Machine," in *Advances in Computational Collective Intelligence*, 2020, pp. 239–251.

[16] A. Hassan, "Potato Weed Plants Classification | Kaggle," *Kaggle.com*, 2020. https://www.kaggle.com/ali7432/potato-weed-plants-classification (accessed Feb. 13, 2022).

[17] F. Y. Manik, D. S. Br Ginting, and M. I. Aldeena, "Extraction Analysis of Texture and Morphological Features as Characteristics of Cabbage Images," in *2021 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA)*, 2021, pp. 88–92. doi: 10.1109/DATABIA53375.2021.9650095.

[18] P. Zheng, D. Qin, B. Han, L. Ma, and T. M. Berhane, "Research on feature extraction method of indoor visual positioning image based on area division of foreground and background," *ISPRS International Journal of Geo-Information*, vol. 10, no. 6, 2021, doi: 10.3390/ijgi10060402.

[19] L. Liu, J. Chen, P. Fieguth, G. Zhao, R. Chellappa, and M. Pietikäinen, "From BoW to CNN: Two Decades of Texture Representation for Texture Classification," *International Journal of Computer Vision*, vol. 127, no. 1, pp. 74–109, 2019, doi: 10.1007/s11263-018-1125-z.

[20] T. Yin and Z. Lv, "Optimal Extraction Method of Feature Points in Key Frame Image of Mobile Network Animation," *Mobile Networks and Applications*, pp. 2515–2523, 2022, doi: 10.1007/s11036-022-02070-x.

[21] S. Wan *et al.*, "Integrated local binary pattern texture features for classification of breast tissue imaged by optical coherence microscopy," *Medical Image Analysis*, vol. 38, pp. 104–116, 2017, doi: 10.1016/j.media.2017.03.002.

[22] I. Muslihah and M. Muqorobin, "Texture Characteristic of Local Binary Pattern on Face Recognition with Probabilistic Linear Discriminant Analysis," *International Journal of Computer and Information System (IJCIS)*, vol. 1, no. 1, pp. 22–26, 2020, doi: 10.29040/ijcis.v1i1.10.

[23] P. A. R. Devi and R. P. N. Budiarti, "Image Classification with Shell Texture Feature Extraction Using Local Binary Pattern (LBP) Method," *Applied Technology and Computing Science Journal*, vol. 3, no. 1, pp. 48–57, 2020, doi: 10.33086/atcsj.v3i1.1745.

[24] I. Al Saidi, M. Rziza, and J. Debayle, "A New LBP Variant: Corner Rhombus Shape LBP (CRSLBP)," *Journal of Imaging*, vol. 8, no. 7, pp. 1–11, 2022, doi: 10.3390/jimaging8070200.

[25] F. D. Adhinata, M. Ikhsan, and W. Wahyono, "People counter on CCTV video using histogram of oriented gradient and Kalman filter methods," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 3, pp. 222–227, 2020, doi: 10.14710/jtsiskom.2020.13660.

[26] L. Kalake, Y. Dong, W. Wan, and L. Hou, "Enhancing Detection Quality Rate with a Combined HOG and CNN for Real-Time Multiple Object Tracking across Non-Overlapping Multiple Cameras," *Sensors*, vol. 22, no. 6, 2022, doi: 10.3390/s22062123.

[27] L. Mohammed, "Design and Implementation an Automated System for Analyzing Brushstrokes to Distinguish between Van Gogh and his Contemporaries by using Swarm Intelligent Method," *International Journal of Computer Applications*, vol. 179, no. 11, pp. 5–14, 2018, doi: 10.5120/ijca2018915896.

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

[29] M. R. R. Allaam, "Klasifikasi Genus Tanaman Anggrek Menggunakan Metode Convolutional Neural Network ( CNN ) Program Studi Sarjana Informatika Fakultas Informatika Universitas Telkom Bandung," vol. 8, no. 2, pp. 3147–3179, 2021.

[30] S. Kolonne, C. Fernando, H. Kumarasinghe, and D. Meedeniya, "MobileNetV2 Based Chest X-Rays Classification," *2021 International Conference on Decision Aid Sciences and Application, DASA 2021*, no. December, pp. 57–61, 2021, doi: 10.1109/DASA53625.2021.9682248.

[31] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, 2020, doi: 10.1016/j.neucom.2019.10.118.

[32] F. Mostafa, E. Hasan, M. Williamson, and H. Khan, "Statistical Machine Learning Approaches to Liver Disease Prediction," *Livers*, vol. 1, no. 4, pp. 294–312, 2021, doi: 10.3390/livers1040023.

[33] R. M. Alzoman and M. J. F. Alenazi, "A comparative study of traffic classification techniques for smart city networks," *Sensors*, vol. 21, no. 14, pp. 1–17, 2021, doi: 10.3390/s21144677.

[34] Keras, "Keras Applications." https://keras.io/api/applications/ (accessed May 22, 2021).