Detection of Palm Oil Leaf Disease with Image Processing and Neural Network Classification on Mobile Device

Alham F. Aji, Qorib Munajat, Ardhi P. Pratama, Hafizh Kalamullah, Aprinaldi, Jodi Setiyawan, and Aniati M. Arymurthy

***Abstract*—Palm oil is an important agricultural commodity as it has great contribution in producing vegetable oil. Being important commodity, palm oil is threatened by diseases that can attack since early stage. The diseases certainly disrupt the palm tree growth and may decrease the palm oil production. Technology can help in identifying disease in early stage so that effective treatment can be given immediately. This paper propose application of image processing and machine learning in identifying three palm oil diseases based on visual appearances. We design a method with linear complexity process in order to minimize processing time so that it can be implemented in mobile device. Using image processing, 6 types of features are extracted from palm leaf image then the pattern was learned using Neural Network method in machine learning process. The learning process yielded a classification model with 87.75% average accuracy and then a mathematical equation was formulated based on classification model in order to be implemented in mobile device.**

***Index Terms*—Image processing, machine learning, mobile device, neural network, palm oil disease.**

I. Introduction

Palm oil is an important agricultural commodity that has high production level. Palm oil plantation could reach more than 30 million metric ton per year in 2006 [1]. Moreover this plantation contributed more than 70% on oil food productions [1].

In the development stage, palm oil plantations are threatened by pests, diseases and weeds which could decrease plantation's production [2]. Early detection of palm disease is essential to save the palm trees from excessive damage. It is also necessary to reduce human error in the disease identification process. Technology implementation offers a great benefit in disease identification process. Technology can provide precise analysis so that can reduce the risk of human error.

Some diseases of palm tree show visual symptoms, for example Anthracnose diseases, Hawar leaf diseases, and purple spot caused by *Culvularia*. Those diseases show typical visual appearance on the leaves [3]. To learn the visual pattern in the disease identification process, image processing and machine learning were used. Considering the benefit of mobility of mobile application this paper also designed an algorithm and pseudo code of disease identification process in a way that can be implemented in mobile device. The purpose of implementation in mobile device is to help farmers identify and detect disease conveniently.

This research was focused on detection of three diseases that attack palm oil trees in the early cultivation stage. All of three diseases can be detected based on visual appearance of palm oil leaves. The diseases are *Anthracnose*, *Hawar Leaf*, and *Leaf Spot* because of *Pestalotiopsis Palmarum*.

A. Anthracnose

Fig. 1. Palm oil leaf with anthracnose disease

Anthracnose disease is infection caused by three pathogenic fungi: *Botryodiplodia* spp., *Melanconium elaeidis* and *Glomerella cingulate* [4]. This disease was reported in various oil palm plantations in Indonesia. Anthracnose mainly attacks seeds in age of 2 month. The symptom usually found at center or side part of leaves, as a bright spot, and then grows wider and became yellow and dark brown [4].

B. Hawar Leaf

Fig. 2. Palm oil leaf with hawar leaf disease

Hawar leaf disease caused by several species of fungi, including *Curvularia eragrostidis*, *Curvularia spp*., *Drechslera halodes*, *Cochliobolus carbonus*, *Cochliobolus sp*, and *Pestalotiopsis sp* [5]. Fungus is spread within spores via wind or splashing water.

Initial symptoms appear in the form of yellow spots on the leaves or the spear that had opened, spots enlarge and become somewhat elliptical with a length of 7-8 mm, light brown color with yellow edges or not, the center sometimes appear oily spotting. The further symptom looks become necrotic spots, some spots fused to form large irregular area. In some cases, the center of the spots becomes dry, brittle, gray or brown [5].

Manuscript received October 24, 2012; revised December 17, 2012.

The authors are with the aculty of Computer Science University of Indonesia (e-mail: alham.fikri@ui.ac.id, qorib.munajat@ui.ac.id, hafizh.kalamullah@ui.ac.id, ardhi.putra@ui.ac.id, aprinaldi@ui.ac.id, jodi.setiyawan@ui.ac.id, aniati@cs.ui.ac.id).

DOI: 10.7763/IJCTE.2013.V5.743

C. *Leaf Spot* by *Pestalotiopsis Palmarum*

Pestalotiopsis Palmarum is fungus that attack faded leaves that haven't opened yet or young leaves that already opened. First indicator is the appearance of small round translucent spot that can be viewed from both sides of leaves. Spot will grow and spreads. Furthermore the color will change to mud brown and usually surrounded by orange or yellowish halo.

In heavy infection, oldest leaf will be dry, wavy, and become brittle. But, in this brittle leaf, the spots keep seen clear as old brown spot on pale brown net [7].

Fig. 3. Leaf spot by *pestalotiopsis palmarum*

II. Palm Leaf Image Processing

Disease detection from oil palm leaf image can be done by observing leaf spot color and shape. Every spot will be classified into certain disease. In this research we focused on detection using RGB color intensity. RGB is additive color system based on tri-chromatic theory. Often found in systems that use a CRT to display images [6].

Leaf image will be captured using digital camera. In this research, the leaf image must be captured in white background. The captured image will be converted to matrix that contains RGB value of each pixel.

A. *Pixel Color Identification*

First step in classifying palm leaves disease is classifying all pixels in the image. Every pixel will be classified as one of the three categories, which is *background pixel, spot pixel,* and *leaf pixel.* As mentioned before that the palm image must be taken with white and clear background, so a pixel will be classified as a background pixel if intensity level of red, green and blue color is above 120, as defined in (1):

$$is\_bg(RGB) = \begin{cases} 1, \min(R, G, B) \geq 120 \\ 0 \quad ,otherwise \end{cases} \quad (1)$$

Equation (1) will return 1 if the pixel is classified as background and will return 0 if the pixel is not classified as background. If the pixel is not classified as background, it will be checked again to be classified as spot pixel or as leaf pixel. Equation 2, 3, and 4 is the equations needed to classify the spot pixel.

$$is\_yellow(RGB) = \begin{cases} 1, R > G; R > B \\ 0 \ ,otherwise \end{cases} \quad (2)$$

$$D = (R + G)/2 - B \quad (3)$$

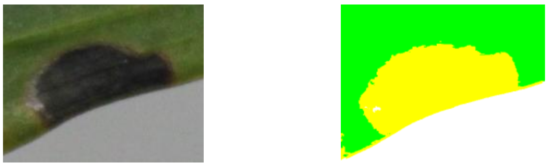
$$is\_purple(RGB) = \begin{cases} 1, \max(R, G, B) < 100; D < 20 \\ 0 \quad ,otherwise \end{cases} \quad (4)$$

A pixel will be classified as spot disease if the pixel color is yellow or purple. To test whether the pixel is yellow, red color intensity must be higher than green color and yellow color as stated in (2). To test if the pixel is purple, we will define *D* to represent distance of the blue color to red and green as defined in (3). Then (4) is applied to test whether the pixel is purple or not. The pixel is classified as spot pixel if (2) or (4) returns 1. These equations are based on our observation on RGB color value which is represented in Table I.

TABLE I: RGB Value of Some Yellow Color [8]

Color	Red	Green	Blue
	189	183	107
	240	230	140
	238	232	170
	255	236	139
	238	220	130
	205	190	112
	238	221	130
	218	165	032
	184	134	011

Classified pixel will be modeled for visual testing. The result can be viewed at Fig. 4.



(a) (b)

Fig. 4. Result from identifying pixel color in palm leaf image. (a) Original image. (b) Processed image, yellow color represent spot pixel, yellow color represent leaf pixel and white color represent background pixel.

B. *Image Filtering*

The identification result that represent background pixel or leaf pixel will be removed from image. Then all remaining adjacent spot pixel will be joined to form a spot image. *Breath First Search* or *BFS* algorithm is used to form all spot pixels.

(a) (b)

Fig. 5. Removing background and leaf in palm leaf image, (a) Original image (b) Image after background and leaf removal.

With that method to classify each pixels, the algorithm need only to check every pixel once, and do flood-filling based on *BFS* algorithm with $\Theta(|V|+|E|)$ where $|V|$ is number of vertex and $|E|$ is the number of edges [9]. On pixel matrix with adjacent pixels as the edges, every pixels except pixels at the side of matrix will have 4 edges, means total edges in the matrix is equal to four times total vertex which is all the pixels in the matrix, thus this algorithm runs in linear complexity and possible to be implemented in mobile devices.

C. Feature Extraction

Features were extracted from each spot that had been successfully detected. The features value will be used in machine learning process to learn the visual pattern. All extracted features are represented in Table III.

TABLE II: FEATURE DATA FOR MACHINE LEARNING

Features	Explanation
Median of RGB	Median of red, green, and blue value in all contained pixel on selected spot.
Quartile 1 of RGB	First quartile of red, green, and blue in all contained pixel, which is the value in 25% position of sorted value.
Quartile 3 of RGB	Third quartile of red, green, and blue in all contained pixel, which is the value in 75% position of sorted value.
Average brightness	$B = 0.299r + 0.587g + 0.114b,$ Where: *rgb* = mean of RGB value in all contained pixel each [10].
Standard Deviation of RGB	$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(\mu - x_i)}{n}}$ Where : n = the total contained pixel; μ = mean red, green or blue of all contained pixel, depends on what color is being calculated. x = red, green or blue value of pixel, depends on what color is being calculated.
Image Shape	$S = \frac{a}{L}$ Where: a = spot area *size*; L =spot perimeter length

III. NEURAL NETWORK

Neural network is used to learn the pattern and generate a classification model. Neural Network is popular method in pattern recognition case and its performance in pattern recognition has been proven. Some research about pattern recognition used Neural Network [11-18]. The machine learning process with neural network is depicted in Fig. 6.

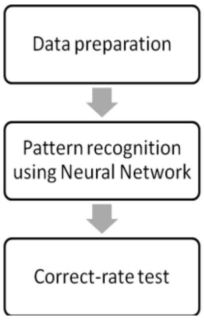


Fig. 6. flow of machine learning process.

A. Training Process

Data was divided into three groups, Training Data, Validation Data, and Testing Data. Validation data is repeatedly used to estimate the non-training performance error of candidate designs, also used to stop training once the non-training validation error estimate stops decreasing Test data is used once and only once on the best design to obtain an unbiased estimate for the predicted error of unseen non-training.

It is necessary to specify the number of hidden neurons in Neural Network. Equation (5) is used to determine number of neurons in the hidden layer [18].

$$n = \sqrt{n_1 + n_2} + a \tag{5}$$

where:

n_1: number of neurons in the input layer

n_2: number of neurons in the output layer

a: constant number between 1 and 10.

This process was done using Matlab application. The setting applied in the training process is described below:

Data Allocation

- Training: 70%
- Validation: 15%
- Testing: 15%

Neural Network Setting

- Number of Hidden Neurons: 3,6, and 12
- Transfer function: tansig (tan sigmoid)

The performance of the model generated from training process can be measured from its Mean Squared Error (MSE) value and from Percentage Error value. MSE is the average squared difference between outputs and targets. Lower values are better, it means the error difference between target and output is close so it can be concluded that the accuracy is better.

Percent Error indicates the fraction of samples which are misclassified. A value of 0 means no misclassifications, 100 indicates maximum misclassifications. Higher values show that the model has low accuracy because there are high number misclassification incident.

However performance of the model cannot be concluded only from single experiment because there might a coincidental event in the testing or validation process. To assess how effective some features in generating model a number of experiments were done to compute the average of model's correct-rate. The process to assess features effectiveness in generating model is depicted in Fig. 7.

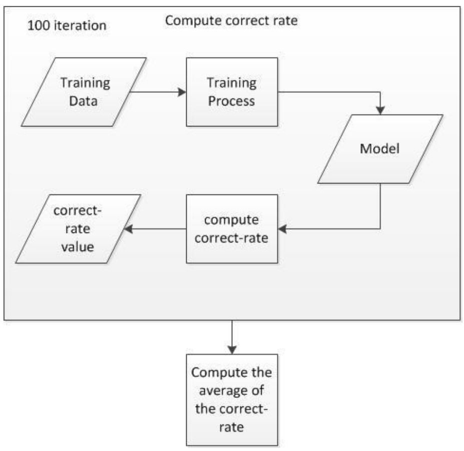


Fig. 7. Process flow to calculate correctness rate.

The average of correct rate values derived from models generated using certain features show the significance influence of the features in determining disease. Higher values is better, it means that features have high relevance in disease identification.

B. Mathematical Function Formulation

In order to make the detection process is performable in mobile device; it is necessary to formulate the classifier into mathematical formula.

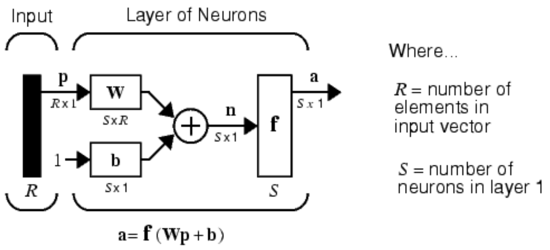


Fig. 8. The concept model of neural network.

Neural network is represented as a net in layers as in Fig. 8. The input of neural network is a value or a set of values that will be processed in each layer of neurons. Every input will be multiplied with its weight and then added with the bias. The result of the computation will be processed further using certain transfer function. The weight and the bias are constant value obtained from trained model.

There are several transfer functions. Each function has its own characteristic so that the implementation of certain function will depend on the case of the neural network classification problem. In case of pattern recognition using neural network, the transfer function is tan sigmoid function. Furthermore a mathematical equation is formulated based on Fig. 8.

$$out = tansig(W2 \times (tansig(W1 \times input + b1) + b2) \tag{6}$$

where:

$W2$: Weight of the output layer (Layer Weight)
$W1$: Weight of the hidden layer (Input Weight)
input: input (features value) which need to be computed
$b1$: bias of the hidden layer
$b2$: bias of the output layer

(The value of $W1$, $W2$, $b1$, and $b2$ are obtained from the actual model) *tansig(n)* in (6) is a tan sigmoid function which has already been provided in Matlab. Therefore, in order to implement the function outside Matlab environment, the tansig function needs to be converted into general form as in (7).

$$a = tansig(n) = \frac{2}{(1+\exp(-2 \times n)} - 1 \tag{7}$$

Data preprocessing is an important process in neural network classification process. Data preprocessing help neural network in generating better model. Values input mapping is one of data preprocessing phase. In this process, the input values will be mapped into the range of (-1) and 1. In Matlab application, to map certain values into certain range *mapminmax* function is provided. The function and algorithm of *mapminmax* function is described in (8) and (9).

$$[y1, PS] = mapminmax(x1) \tag{8}$$

where:

$x1$: value to be mapped (can be an array variable)
$y1$: mapped value
PS: mapping setting
(The default range is -1,1 if it is not specified)

$$y = (ymax - ymin) \times \frac{(x - xmin)}{(xmax - xmin)} + ymin \tag{9}$$

where:

x: the value to be mapped
xmax: the maximum value of x in the population
xmin: the minimum value of x in the population
ymax: the maximum value of the mapped value
ymin: the minimum value of the mapped value

IV. RESULT

The result of the experiment is shown in Table III, Table IV, and Table V.

TABLE III: CORRECT-RATE VALUE OF MODEL GENERATED WITH NUMBER OF HIDDEN NEURONS = 3

Features	Mean CR	Max CR	Min CR	Stdev CR
median RGB, shape feature	72.08%	86.99%	15.07%	0.15
median RGB, quartil 1, quartil 3, shape feature	80.92%	93.15%	31.51%	0.14
median RGB, quartil 1, quartil 3, brightness, standard deviation, shape feature	84.92%	96.58%	31.51%	0.14

TABLE IV: CORRECT-RATE VALUE OF MODEL GENERATED WITH NUMBER OF HIDDEN NEURONS = 6

Features	Mean CR	Max CR	Min CR	Stdev CR
median RGB, shape feature	74.99%	86.30%	26.71%	0.13
median RGB, quartil 1, quartil 3, shape feature	82.38%	93.84%	17.81%	0.13
median RGB, quartil 1, quartil 3, brightness, standard deviation, shape feature	87.75%	96.58%	34.25%	0.10

TABLE V: CORRECT-RATE VALUE OF MODEL GENERATED WITH NUMBER OF HIDDEN NEURONS = 12

Features	Mean CR	Max CR	Min CR	Stdev CR
median RGB, shape feature	74.89%	85.62%	34.93%	0.12
median RGB, quartil 1, quartil 3, shape feature	80.53%	95.89%	22.60%	0.16
median RGB, quartil 1, quartil 3, brightness, standard deviation, shape feature	83.51%	96.58%	15.75%	0.17

It can be concluded from the Table III, IV, V that median RGB, quartil 1, quartil 3, brightness, standard deviation, shape feature generated better model than the others. Further analysis show that 6 is an optimal number of hidden neurons to generate model for this case. The best model's MSE and % Error values is depicted in the Fig. 9.

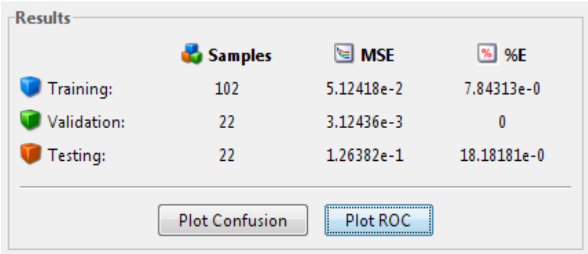


Fig. 9. MSE and %Error values of model performance.

That set of features was able to generate model with MSE values 0.005 in the training process, 0.0003 in the validation process, and 0.1 in the testing process. As for the % error, it is shown that there was 7.8% misclassified incident in training process, 0% misclassified incident in validation process and 18% misclassified incident in testing process. These values indicate that this model have best performance compared to the others.

V. Conclusion

Early detection of palm oil disease can be carried out by implementing image processing and machine learning. The average accuracy is 87.75% and the features are median of RGB value, quartile 1 of RGB value, quartile 3 of RGB value, average brightness, standard deviation, and shape. The number of hidden neurons generating best model is 6. There are still many possibilities that can be elaborated more in this research. Higher accuracy are better, the aim is near 100% accuracy therefore further research is needed to increase accuracy.

This research designed an image processing algorithm that runs in linear time complexity. The classifier generated from machine learning with neural network method was also formulated in general mathematical function. It means that palm oil disease early detection can be implemented at various application including mobile devices.

References

[1] P. Thoenes (2012). Biofuels and Commodity Markets – Palm Oil Focus. [Online]. Available: http://www.fao.org/es/ESC/common/ecg/122/en/full_paper_English.pdf.

[2] S. Haryono, "Penyakit-penyakit kelapa sawit," in *Penyakit-Penyakit Tanaman Perkebunan di Indonesia*, Yogyakarta, Indonesia: Gajah Mada University Press, pp. 150, 1987.

[3] S. Agus (February 2011). Hama Kelapa Sawit. [Online]. Available: http://kliniksawit.com/hama-sawit/46-hama-kelapa-sawit.¬html

[4] S. Agus (February 2011). Busuk Daun. [Online]. Available: http://kliniksawit.com/penyakit-sawit/busuk-daun.html

[5] A. Ford and A. Roberts (2010). Colour Space Conversions. [Online]. Available: http://www.poynton.com/PDFs/coloureq.pdf

[6] S. Haryono, "Penyakit-penyakit kelapa sawit," in *Penyakit-Penyakit Tanaman Perkebunan di Indonesia*, Yogyakarta, Indonesia: Gajah Mada University Press, pp. 172-176, 1987.

[7] Wessendorf, Guido RGB. (December 2008). Color Table. [Online]. Available: http://gucky.uni-muenster.de/cgi-bin/rgbtab-en

[8] E. K. Donald, *The Art of Computer Programming*, 3rd ed., Boston: Addison-Wesley, vol. 1, 1997.

[9] P. Bourov and D. Ilinih, "Brightness calculation in digital image processing Sergey Bezryadin," KWE Int. Inc., San Francisco, CA, USA; UniqueIC's, Saratov, Russia.

[10] X. D. Ma and G. Y. Qi, "Investigation and recognition on diseased spots of soybean laminae based on neural network," J. of Heilongjiang August First Land Reclamation University, vol. 18, pp. 84-87, April 2006.

[11] H. A. Hiary, S. B. Ahmad, M. Reyalat, M. Braik, and Z. A. Rahamneh, "Fast and accurate detection and classification of plant diseases," *International Journal of Computer Applications*, vol. 17, pp. 31-38, March 2011.

[12] D. W. Zhang and J. Wang, "Design on image features recognition system of cucumber downy mildew based on BP algorithm," *Journal of Shenyang Jianzhu University (Natural Science)*, vol. 25, pp. 574-578,

[13] L. B. Liu and G. M. Zhou, "Identification method of rice leaf blast using multilayer perception neural network," *Transactions of the CSAE*, vol. 25, pp. 213-217, October 2009.

[14] H. Ding and J. Y. Ai, "The analysis of black rot spot based on computer vision," *Computing Technology and Automation*, vol. 26, pp. 115-118, March 2007.

[15] Z. R. Li and D. J. He, "Research on identify technologies of apple's disease based on mobile photograph image analysis," *Computer Engineering and Design*, vol. 31, pp. 3051-3053, July 2010.

[16] Y. P. Wang, X. P. Dai, H. Huang, and R. Zhang, "Identification of tomatoes with diseases based on mathematical morphology and neural network," *Journal of Hunan Agricultural University (Natural Sciences)*, vol. 32, pp. 344-346, June 2006.

[17] H. G. Wang, G. L. Li, Z. H. Ma, and X. L. Li. "Application of neural networks to image recognition of plant diseases," *International Conference on Systems and Informatics.* 2012.

Qorib Munajat was born in Kebumen, Central of Java, Indonesia in 24th February 1989. Qorib Munajat took information system for his undergraduate degree in Faculty of Computer Science, University of Indonesia. Qorib Munajat finished his undergraduate study in 2011. Currently Qorib Munajat is taking Master Program in Faculty of Computer Science University of Indonesia. During his study, he also works as teaching assistant in Faculty of Computer Science University of Indonesia. Several courses he was involved as teaching assistant were Software Engineering, Web Programming, and Enterprise Resource Planning. He also had working experience in Telecommunication company in 2010. Regarding his work in his studies, his undergraduate thesis was about Enterprise Application Integration. Currently his research interests are about enterprise system, software engineering, and machine learning.

Alham Fikri Aji was born in Jambi, Indonesia in 1992. He currently studied in computer science major at Faculty of Computer Science, University of Indonesia. He became lecturer assistant in Faculty of Computer Science University of Indonesia at 2011. He also had working experience as mobile application developer since 2011, and became mentor in mobile application student interest group at his faculty. He also a joined ACM Student membership in 2012.

Jodi Setiyawan was born in Kebumen, Indonesia in August 27th 1992. He is undergraduate student in Faculty of Computer Science University of Indonesia. He also works as teaching assistance in discrete mathematic course. Currently he is active in research about image processing and machine learning.