Computer Engineering and Applications Vol. 9, No. 3, October 2020

Deep Convolutional Neural Networks-Based Plants Diseases Detection Using Hybrid Features

Budiarianto Suryo Kusumo*, Ana Heryana, Dikdik Krisnandi, Sandra Yuwana, Vicky Zilvan, and Hilman F Pardede

Research Center for Informatics, Indonesian Institute of Sciences
**rbud001@lipi.go.id*

ABSTRACT

With advances in information technology, various ways have been developed to detect diseases in plants, one of which is by using Machine Learning. In machine learning, the choice of features affect the performance significantly. However, most features have limitations for plant diseases detection. For that reason, we propose the use of hybrid features for plant diseases detection in this paper. We append local descriptor and texture features, i.e. linear binary pattern (LBP) to color features. The hybrid features are then used as inputs for deep convolutional neural networks (DCNN) Support and VGG16 classifiers. Our evaluation on Based on our experiments, our proposed features achieved better performances than those of using color features only. Our results also suggest fast convergence of the proposed features as the good performance is achieved at low number of epoch.

Keywords: Deep convolutional neural networks, Hybrid features, Linear binary pattern, Deep learning.

1. INTRODUCTION

Tea (Camellia Sinensis) is one of the main agricultural industry commodities in Indonesia, this industrial plant needs special attention in managing it, including detecting damage as early as possible. The problem that often attacks this plant is Exobasidium vexans massee causes a disease called blister blight, leaf hoppers (Empoasca sp.), and the looper caterpillar (Hyposidra talaca). The pests and diseases have different characteristics that can be distinguished from the leaves. To detect them experts still rely on traditional separation, by looking directly at and still involving the knowledge of a tea expert. Diagnosing the disease manually requires large costs and a long time. Therefore, a way to detect diseases in plants more efficiently and accurately is needed. This can be done with current technological advances. One way is to apply machine learning. There have been a lot of studies that have been proposed in the literature to detect diseases in plants. In the process of image classification used to detect tea disease there are several algorithms in Machine learning that have been implemented, namely SVM (Support Vector Machine), K-Nearest Neighbors, Case-based reasoning, Naıve Bayes, and Decision Tree.

In study [15] classical machine learning algorithms rely on image pre-processing and the extraction of features which are then fed into one of the ML algorithms. Multi-objective Bayesian optimization [16] applied to deep learning model and the average number of evaluations of each objective including time and error are investigated. For training and validating the deep network, a number of images present various diseases in leaves are provided from Plant Village data set. While they have been implemented with some success, they are mostly unsatisfactory for real implementation in the field. Recently, deep learning technology is predominant method for plant diseases detection, especially deep convolutional neural networks (DCNN) [1, 2, 3]. In [4], DCNN architectures like VGG16, VGG19, Inception, and ResNet50, is used for classifying types of flowers. In [1], concatenated DCNN is used for tea diseases detection. The role of features is important for machine learning. Features that are extracted from data, usually pass through a removal operations to reduce redundancy and reduce unwanted informations [1]. Various study employs many features from image processing and computer vision for object classification tasks. Interestingly, most deep learning systems employs only color features such as red-green-blue (RGB) for object classification. Color descriptors is used for face recognition [2]. In study of image segmentation [3], RGB features are also employed. In study [14], Comprised of 35,000 images of healthy plant leaves and infected with the diseases by using plant village dataset, they use data augmentation techniques in CNN, also in [13] they use AlexNet to proof better than SqueezeNet in accuracy result. Texture features such as LBP has also been implemented. LBP is a texture descriptor that is powerful and simple yet [11]. In [5], LBP is used to detect defect surfaces. They adopted LBP to detect the gray difference value between single points as color features may not be robust to noise and illumination. Therefore, this paper improves the traditional LBP method and proposes a surface defect detection method based on gradient local binary patterns (GLBP), which uses image sub-blocks to reduce the dimensionality of the LBP data matrix. However, oftentimes feature extraction process reduce some of important informations for classification. Furthermore, diseases on plants may not only be identified by the difference in colors but also changes in shape and textures. For that reason, we propose to use hybrid features in this paper.

The remainder of this paper is organized as follows. Section 2 describes the experimental framework used to evaluate the performance of the proposed method of hybrid features. Section 3 introduces terminology feature descriptor that we append LBP with 4 channel RGB as feature Descriptors. Sections 4 introduces 4 channel RGB. Section 5 then testing data and classification model. Section 6 experimental setups and parameters adjustment and data-set as an input. Section 7 result and discussions then combines the results from VGG 16 in combination with LBP 4 channel RGB. Finally, Section 8 presents our conclusions.

2. PROPOSED METHOD

2.1 HYBRID FEATURES HYBRID FEATURES

Local Binary Pattern (LBP) was proposed in 1996 [2]. The detail steps of this algorithm are as follows. First, the image is converted into gray scale. For each point p, select r points that surround the point p. Every point that surrounds the point p is compared to the point p. If the intensity of the p point is greater, then set the point to 1. If otherwise, set it to 0. Calculate the value of P from point p. The way to calculate it is to combine the binary values obtained and convert them to decimal values. Replace point p with the decimal value. Make a histogram of the overall

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

results that have been obtained. While studies have shown that LBP is an efficient texture descriptor [6, 7, 9], it lacks robustness to image rotation and noise. This is because there are as many as 2P patterns. In this paper, we append LBP as the fourth channel to RGB features. As diseases on plants may not only be identified by the difference in colors but also changes in shape and textures. Furthermore, by hybriding both RGB and LBP, more information is fed into the classifier, making it more discriminative to differentiate between class labels.

2.2 DEEP CONVOLUTIONAL NEURAL NETWORKS

In this paper, we employ VGG16 as our classifiers [12]. Due to our previous findings, VGGNet is quite competitive to other CNN architectures such ResNet and DenseNet. Unfortunately, we have limitations on their implementation since DenseNet is memory exhaustive. The VGG16 architectures is shown in Figures 1 and 2. The flow of data in VGG16 can be explained as follows. First, the input, which resolutions are set at 224 x 224 x 4, pass the several convolutional layers. The input features comprise of 3 RGB channels and 1 LBP channel. In VGG16, all filters has kernel size of 3 3. In one of the configurations, it also utilizes 1 1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. the input layer is such that the spatial resolution is preserved after convolution, i.e. the padding is 1-pixel for 3 3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2 2 pixel window, with stride 2. Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has a different depth in different architectures). The first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all network the convolution stride is fixed to 1 pixel; the spatial padding of conv. the input layer is such that the spatial resolution is preserved after convolution, i.e. the padding is 1-pixel for 3 3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2 2 pixel window, with stride 2. Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has a different depth in different architectures). The first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks.

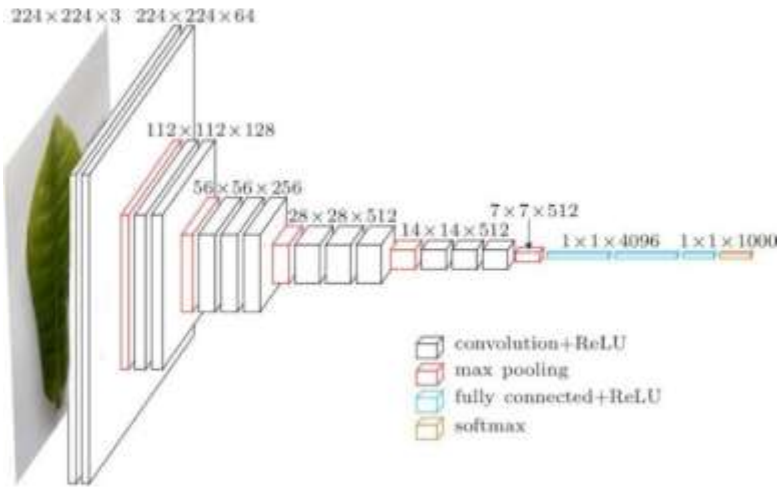

FIGURE 1. VGG16 Architecture

All hidden layers are equipped with the rectification (ReLU) non-linearity. It is also noted that none of the networks (except for one) contain Local Response Normalizations (LRN). Such normalization does not improve the performance on the ILSVRC dataset, but leads to increased memory consumption and computation time.

FIGURE 2. Description of layers

The block diagram of our Disease identification system is shown in Figure 3. We developed the method by adding hybrid features to the classifier process and SVM is used as the classifier.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

Computer Engineering and Applications Vol. 9, No. 3, October 2020

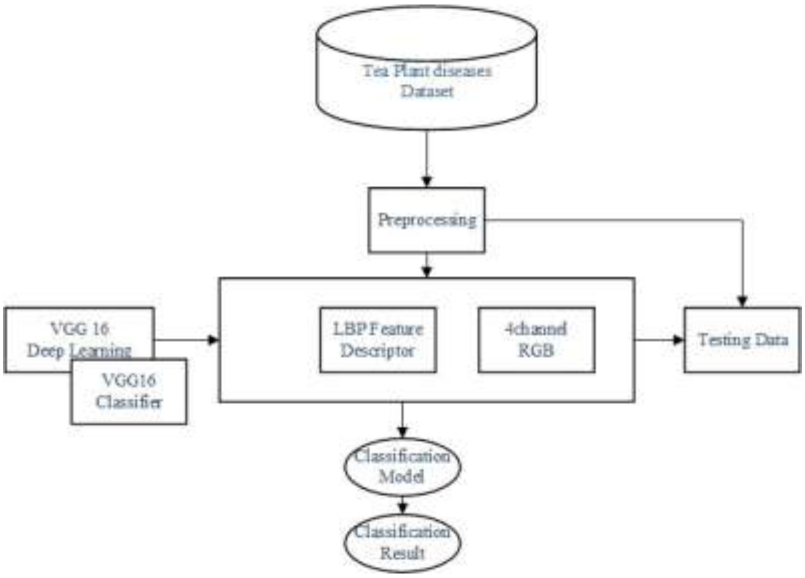

FIGURE 3. The Tea leaf disease identification system flow diagram

3. FEATURE DESCRIPTORS

The feature descriptor is an algorithm that takes an image and generates an image feature vector. The feature descriptor displays the required string of information in vector form and acts as a kind of numerical ”fingerprint” that can be used to distinguish one feature from another. In this study, we modified the LBP by adding 4 RGB channels as feature descriptors. The hybrid process that will be highlighted in this study is a combination of the classification features, descriptors and 4 RGB channels into the main algorithm.

4. FOUR CHANNEL RGB

Local binary pattern with 4-channel RGB implementation is a grayscale RGB binary pattern which is then defined into 4 image transformations, namely Gaussian blur, median blur, rotate blur, and transpose blur. The entire image transformation is then compared with the original image to get the damaged area from more detail. The image resolution processed is 64x64 with 4 kernel features.

5. TESTING DATA AND CLASSIFICATION MODEL

We separate the testing data to test the 4 classifications (Healthy, Blister blight, Empoasca, and Looper caterpillars) that have been trained and recognized by the system. We load the data from the test data tuples sequentially with a size of 64 x 128 for RGB np zeros, and we load test data originally with size 64 x 64 and the same size for test data in Gaussian blur, median blur rotate blur and transpose blur.

6. EXPERIMENTAL SETUPS

To carry out this experiment, we collected tea leaf image as dataset, parameter settings and image size adjustments, cross validation values, training epoch, image resolution from input to output, optimizer value and cross entropy.

6.1. DATASET

In this study, we collected 5569 data on tea leaf images. The dataset was collected using two digital cameras and five smartphone cameras. All images are taken indoor and under an uncontrolled environment and use the autofocus feature. The distance of shooting from the camera to the leaf is not determined or randomly. Samples of the data is shown in Figure 4.

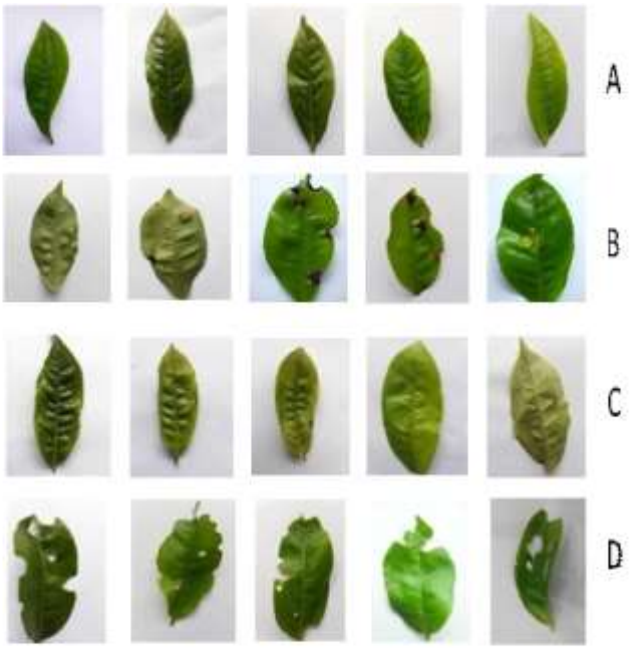

FIGURE 4. The dataset: (a) Healthy, (b) Blister blight, (c) Empoasca sp. (leafhoppers), and (d) Looper

Distributions of each class are as follows. 1104 images of Blister, 2055 Emphoasca, 1448 healthy tea leaves, and 1025 looper caterpillars. As shown in Figure 3. We obtained data from the Tea and Cinchona research center of The Research Institute for Tea and Cinchona, Gambung, West Java, Indonesia. The distribution of the data for each.

Class is summarized in Table 1. We involved researchers from tea and Cinchona research center The Research Institute for Tea and Cinchona, Gambung, West Java, Indonesia. The collected tea leaves and to determine the classification of disease, because only researchers as an Expert from the research center of the Tea and Cinchona Research Institute can only determine that classification of the diseases.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

TABLE 1
The Dataset Used For The Classification

No	Type of Tea Plant Diseases	Number
1	Healthy	1448
2	Blister blight	1104
3	Empoasca sp. (leafhoppers)	2055
4	Looper caterpillars	1025
	Total	5632

6.2. PARAMETERS SETUP

In this study, we use the LBP setting with the ’uniform’ method P variable value = 16 and R = 2. Library runs in Python programming. For VGG16, we use batch size 10 and Adam optimizer. Due to memory limitations, we limit the batch to small sizes While the use of large mini-batches increases the available computational in parallelism. We divided the data into three parts: there are 3.933 training data, 589 validations data and 1.110 testing data with each distribution of 70%, 10% and 20% respectively.

7. RESULTS AND DISCUSSIONS

In our experiments seen in Table 2, to calculate accuracy, we use the following formula [15]:

$$\frac{(TP+TN)}{TP+TN+FP+FN} \tag{1}$$

True Positive (TP): It refers to the number of predictions where the classifier correctly predicts the positive class as positive. True Negative (TN): It refers to the number of predictions where the classifier correctly predicts the negative class as negative.

False Positive (FP): It refers to the number of predictions where the classifier incorrectly predicts the negative class as positive. False Negative (FN): It refers to the number of predictions where the classifier incorrectly predicts the positive class as negative. We revealed very significant results, VGG 16 in combination with LBP 4 channel RGB produces very high accuracy to classify four data classes with the highest value of 96.20 on epoch 200. It can even produce an increase in accuracy of 51.90 on epoch 40 this value is higher than experiment.

TABLE 2.
The comparison of accuracy proposed features on VGG16 and RGB only

Data Classes	Architecture	Epoch			
		40	50	100	200
4	RGB	36.22	84.88	91.26	93.50
4	Hybrid Features	88.12	91.43	94.16	96.20

In Figure 5 we can see the results of the training with 40 epochs with very low accuracy, not even being able to determine the classification of 4 classes with an average value of accuracy of 36.22, but in epochs of more than 50 have shown good results with an accuracy value of 84.88, the more epochs run the better results of the training accuracy obtained.

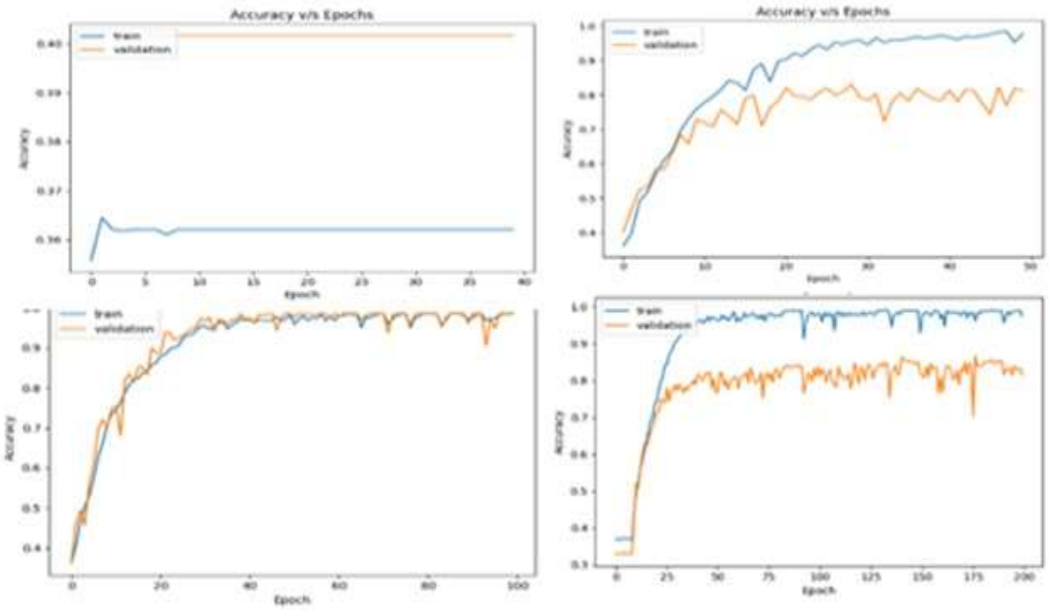

FIGURE 5. Training result VGG16 without LBP 4 channel
(RGB only)

In Figure 6, the results of validation accuracy are very good even in epoch 40, the average accuracy is 88.12, and the more epochs used the higher the accuracy obtained. This method minimizes the use of epoch in deep learning, thus shortening the computation in deep learning.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

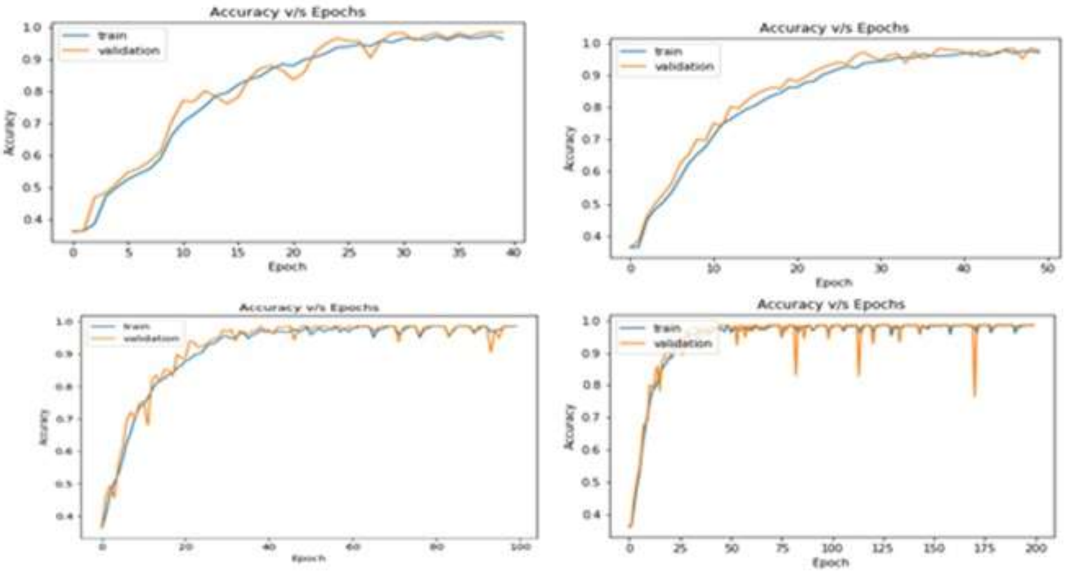

FIGURE 6. Training result VGG16 with LBP 4 Channel

In Figure 7 we see that the value of total loss is very dynamic with a trend that cannot be predicted to decrease, so it is very difficult to determine whether the system can determine the classification of images with high accuracy. With this 4-channel VGG16 LBP method in Figure 7, we succeeded in minimizing total loss from computational training, becoming very small with an average cross entropy. Value below 0.2, very different compared to vgg16 method without LBP in Figure 6 with a total value of cross entropy loss above 0.3 and loss validation value above 0.6.

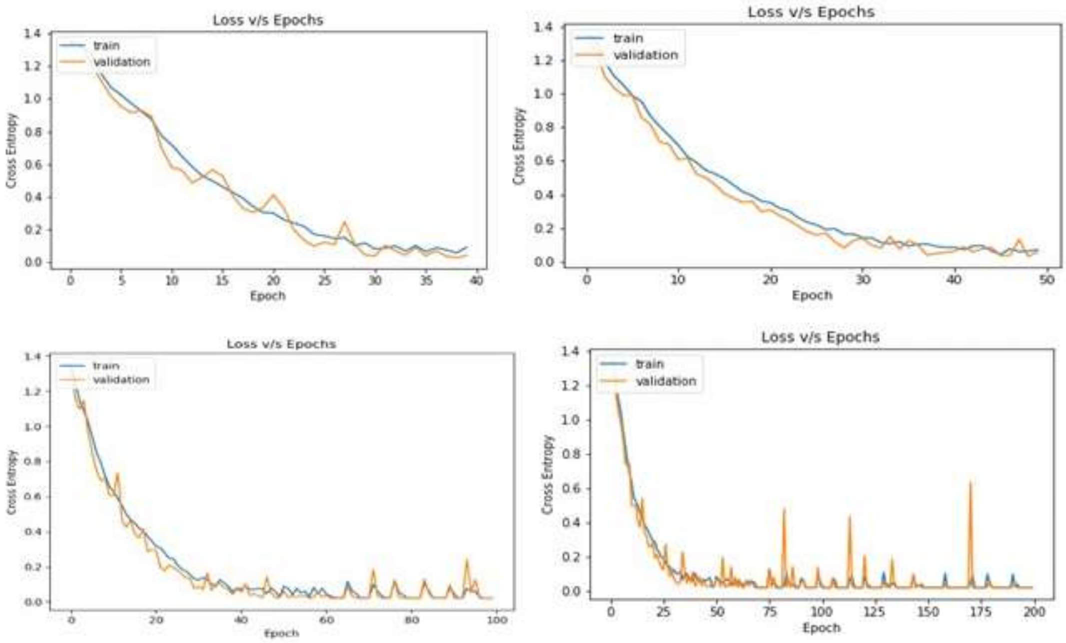

FIGURE 7. Total Loss VGG16 with LBP 4 Channel

8. CONCLUSION

In this paper we improve the method for classifying tea leaf images to identify four types of disease more precisely. LBP with 4 channels can facilitate grayscale features of binary RGB patterns. The combination of VGG16 as a learning medium, makes compatibility even more effective in each layer to look for differences in the image of damage to the area compared to the image of a healthy tea leaf. There have been many attempts to overcome limited data. In the future, we design better architectural designs for transfer of learning and use newer and more complex models. Incorporate additional data and transfer learning about our problems also in our future plans.

ACKNOWLEDGEMENTS

The experiments on this research are conducted on High Performance Computing (HPC) facility in Research Center for Informatics, Indonesian Institute of Sciences. All authors contributed equally on the paper.

REFERENCES

[1] Dikdik, Pardede, Hilman , Yuwana, R. , Zilvan, Vicky . Heryana, Ana. Fauziah, Fani . Rahadi, Vitria.*Diseases Classification for Tea Plant Using Concatenated Convolution Neural Network. CommIT (Communication and Information Technology)*,Journal. 13. 10.21512/commit.v13i2.5886, 2019.

[2] Z. Lu, X. Jiang and A. Kot, ”A novel LBP-based Color descriptor for face recognition,” 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, 2017, pp. 1857-1861, doi: 10.1109/ICASSP.2017.7952478.. .

[3] R. Gothwal, S. Gupta, D. Gupta and A. K. Dahiya, ”Color image segmentation algorithm based on RGB channels,” Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization, Noida, 2014, pp. 1-5, doi: 10.1109/ICRITO.2014.7014669.

[4] Y. Wu, X. Qin, Y. Pan and C. Yuan, ”Convolution Neural Network based Transfer Learning for Classification of Flowers,” 2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP), Shenzhen, 2018, pp. 562-566, doi: 10.1109/SIPROCESS.2018.8600536.

[5] X. Liu, F. Xue and L. Teng, ”Surface Defect Detection Based on Gradient LBP,” 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), Chongqing, 2018, pp. 133-137, doi: 10.1109/ICIVC.2018.8492798

[6] Banerji S., Verma A., Liu C. (2012) *LBP and Color Descriptors for Image Classification. In: Cross Disciplinary Biometric Systems* Intelligent Systems Reference Library, vol 37. Springer, Berlin, Heidelberg

[7] S. Wan, X. Huang, H. Lee, J. G. Fujimoto and C. Zhou, *Spoke-LBP and ringLBP: New texture features for tissue classification,* 2015 IEEE 12th

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

International Symposium on Biomedical Imaging (ISBI), New York, NY, 2015, pp. 195-199, doi: 10.1109/ISBI.2015.7163848..
[8] Georgiou, T., Liu, Y., Chen, W. et al. *A survey of traditional and deep learningbased feature descriptors for high dimensional data in computer vision*. Int J Multimed Info Retr (2019). https://doi.org/10.1007/s13735-019-00183-w.
[9] X. Liu, F. Xue and L. Teng, ”Surface Defect Detection Based on Gradient LBP,” 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), Chongqing, 2018, pp. 133-137, doi: 10.1109/ICIVC.2018.8492798.
[10] M. H. Selamat and H. Md Rais, ”Image face recognition using Hybrid Multiclass SVM (HM-SVM),” 2015 International Conference on Computer, Control, Informatics and its Applications (IC3INA), Bandung, 2015, pp. 159-164, doi: 10.1109/IC3INA.2015.7377765.
[11] R. Hu, W. Qi and Z. Guo, ”Feature Reduction of Multi-scale LBP for Texture Classification,” 2015 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), Adelaide, SA, 2015, pp. 397-400, doi: 10.1109/IIH-MSP.2015.79.
[12] Simonyan, Karen, Zisserman, Andrew. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 1409.1556..
[13] H. Durmus¸, E. O. Gunes¸ and M. Kırcı, ”Disease detection on the leaves¨ of the tomato plants by using deep learning,” 2017 6th International Conference on Agro-Geoinformatics, Fairfax, VA, 2017, pp. 1-5, doi: 10.1109/AgroGeoinformatics.2017.8047016.
[14] S.V. Militante, B. D. Gerardo and N. V. Dionisio, ”Plant Leaf Detection and Disease Recognition using Deep Learning,” 2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), Yunlin, Taiwan, 2019, pp. 579582, doi: 10.1109/ECICE47484.2019.8942686.
[15] D. Radovanovic and S. ukanovic,´ *”Image-Based Plant Disease Detection: A Comparison of Deep Learning and Classical Machine Learning Algorithms,”* 2020 24th International Conference on Information Technology (IT), Zabljak, Montenegro, 2020, pp. 1-4, doi: 10.1109/IT48810.2020.9070664.
[16] L. Kouhalvandi, E. O. Gunes and S. Ozoguz, ”Algorithms for Speeding-Up the Deep Neural Networks For Detecting Plant Disease,” 2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics), Istanbul, Turkey, 2019, pp. 1-4, doi: 10.1109/Agro-Geoinformatics.2019.8820541.