Vegetation classification algorithm using convolutional neural network ResNet50 for vegetation mapping in Bandung district area

Rina Pudji Astuti[1], Ema Rachmawati[1], Edwar[1*], Simon Siregar[1], Indra Lukmana Sardi[1], Arfianto Fahmi[2], Yayan Agustian[3], Agus Cahya Ananda Yoga Putra[1], and Faishal Daffa[1]

[1] Telkom University
[2] Telkom Institute of Technology Purwokerto
[3] Dinas Pertanian Kabupaten Bandung
[1] Jl. Telekomunikasi No. 1 Bandung, West Java, Indonesia
[2] Kawasan Pendidikan Telkom, Jl. DI. Panjaitan No. 128 Purwokerto, Indonesia
[3] Jalan Raya Soreang Km 17, Kabupaten Bandung, Jawa Barat, Indonesia
*Corresponding email: edwarm@telkomuniversity.ac.id

Abstract —Bandung District is one of the crop providers for West Java Province. About 31.158,22 ha is used for the crop. However, some of them are not maintained well due to a lack of vegetation map information. The local authority has tried to map the vegetation in their area by using free license satellite images and aerial images from Unmanned Aerial Vehicles (UAV). Despite both images being able to provide large plantation area images, both cannot classify the vegetation type in those images. Telkom University with Bandung Agriculture Regional Office (Dinas Pertanian Kabupaten Bandung) has conducted joint research to develop an algorithm based on a 50-layer residual neural network (ResNet50) to classify the vegetation type. The input of this algorithm is primarily aerial images captured from different types, heights, and positions of crops. Seven different ResNet50 configurations have been set and simulated to classify the crop images. The result is the configuration with resized images, employing a triangular policy of cyclic learning rate with rate $1.10^{-7} – 1.10^{-4}$ comes out as the best setup with more than 95% accuracy and relatively low loss.

Keywords – Vegetation Mapping, ResNet50, Aerial Images

I. INTRODUCTION

Vegetation mapping is important to obtain information on the vegetation distribution in a region. That information can predict the level of food needs fulfillment in an area. Furthermore, this information can be used to manage food security management by the local authority, Bandung Agriculture Regional Office (Dinas Pertanian Kabupaten Bandung).

In Bandung District, there are 31.158,22 Ha planted by various vegetation. However, the land monitoring activity is considerably low, especially on a few commodities. Some plants are not fertilized intensively due to a lack of vegetation monitoring information. The main cause is the local authority has no proper method to monitor the vegetation area. Currently, they are using free license images from the satellite. However, the accuracy of the satellite image is low. It is hard to distinguish the crop from other non-vegetation objects such as football fields or meadows. It is even harder to recognize different vegetation types when their area is adjacent. The better monitoring result can be achieved, it is still needed to be assisted by using aerial photos from UAVs (Unmanned Aerial Vehicles). Nevertheless, their drone has no algorithm to recognize the vegetation specifically. To overcome this problem, an algorithm to detect different types of vegetation is needed to be employed in the user system so any type of input aerial image, whether it is from a satellite or UAV, the system will be able to recognize different types of vegetations.

Related research regarding vegetation mapping or land monitoring has been conducted. The vegetation classification from satellite images using multiple deep convolutional networks (DCNs) has been done [1]. It is working on classifying crop types in the Hunan Province of China. This research is working on deforestation detection from satellite images. The vegetation type is not a concern [2]. This paper has done land cover classification using grey-level co-occurrence matrix and naive Bayes. However, it only focuses on land usage only [3]. This work uses hyperspectral images and lidar data to map vegetation areas. Both data are not easy to obtain [4].

This work using ResNet is used to analyze tree mapping from UAV images. This work only works on one type of tree: date palm tree [5]. This research combines airborne lidar data and satellite images to classify tree species [6]. They use some algorithms such as ResNet18, SVN, CNN, etc., to check their performance in classifying the object. However, combining two different image sources may increase the problem of fetching the data. Another work has been done to identify trees from RGB UAV images only [7]. It uses ResNet18 and ResNet152 to predict the object type. This work is relatively similar to this paper's idea; however, the object images are captured from a homogenous altitude.

This paper explains research to classify crops by using a different approach. This research combines the various type of plants example, the target's image capturing method, and the ResNet50 algorithm to classify that image, which is less complex than [7]. In this research, rice fields and tea plantations are the targets. Both images are taken from different heights using a UAV which is a modification of this research [6-7]. Using this method, the vegetation monitoring accuracy is expected to be higher. The aerial images are classified using the 50 Layers Residual Neural Network (ResNet50) algorithm.

II. RESEARCH METHODS

A. *ResNet and ResNet50*

A convolutional neural network (CNN) is a kind of -forward neural network that can extract features from data with convolution structures [8]. Compared with the other older feature extraction methods, which have been explained by [9], [10], [11], manual feature extraction is not needed in CNN. Instead, visual perception inspires the CNN architecture [12]. A biological neuron corresponds to an artificial neuron, CNN kernels represent different receptors that can respond to various features; activation functions simulate the function that only neural electric signals exceeding a certain threshold can be transmitted to the next neuron. Fig. 1 shows the development of CNN methods.
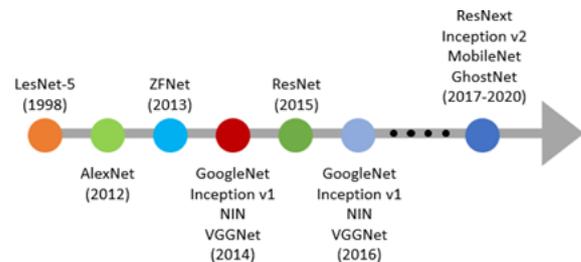

Fig.1. CNN Architecture Methods from Time to Time

CNN architecture has been developing for two decades and has resulted in many methods. One of the effective methods that are used widely is ResNet. It was proposed by [13], which is considered a continuation of deep networks. ResNet introduced a breakthrough CNN architecture comparing its predecessor using residual learning concepts in CNNs and conceived an efficient method to train a deep network. Fig. 2 shows general ResNet architecture. Like Highway Networks, it is also placed under multi-path-based CNNs. Fig. 3 shows the residual block as a basic structural unit of ResNet architecture. ResNet, which is 20 and 8 times deeper than AlexNet and VGG and can be ruwith less computational complexity than previously proposed networks [14-15]. He et al. empirically showed that ResNet with 50/101/152 layers has less error on image classification task than 34 layers plain Net. Moreover, ResNet gained a 28% improvement on the famous image recognition benchmark dataset named COCO. The good performance of ResNet on image recognition and localization tasks showed that representational depth is of central importance for many visual recognition tasks.

Residual block is a breakthrough proposed by Kaiming et al. to overcome the vanishing gradient. A vanishing gradient occurs when the model's gradient results are missing or too small due to each layer's calculation process. Residual blocks have a shortcut called a "shortcut connection." Connection shortcuts allow data to pass through one or more layers. The residual block forms the basis of the ResNet architecture. Residual blocks exist at every layer in the ResNet architecture. Residual blocks can create a variety of ResNet architectures, namely: 18, 34, 50, 101, and up to 152 layers.
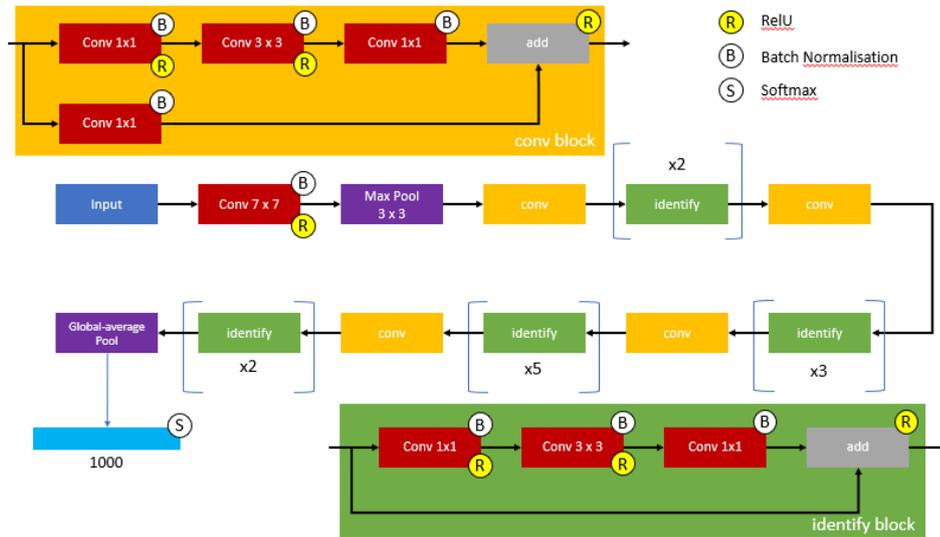
147

Fig.2. ResNet General Architecture

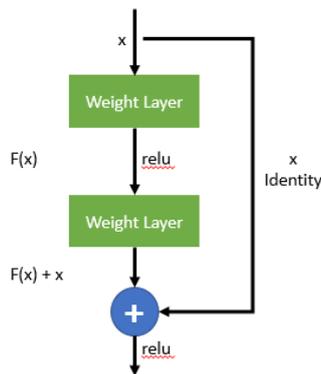Fig.3. Residual Block

ResNet50 is a type of ResNet with 50 layers deep convolutional network. It contains A convolution with a kernel size of 7 * 7 and 64 different kernels counted as one layer. The next convolution is a 1 * 1, 64 kernel, followed by a 3 * 3, 64 kernel and a 1 * 1, 256 kernel repeated three times. This block is counted to have nine layers in total. Next is kernel 1 * 1, 128 followed kernel 3 * 3, 128 and kernel 1 * 1, 512, repeated four times. This block has 12 layers. After that, there is a kernel of 1 * 1, 256 and two more kernels with 3 * 3, 256 and 1 * 1, 1024 repeated six times. This block has 18 layers. The next is a 1 * 1, 512 kernel with two more of 3 * 3, 512 and 1 * 1, 2048, repeated three times, and gives nine layers. At the end of the process, ResNet50 will do an average pool, and it will be a fully connected layer containing 1000 nodes, then produces a 1-layer output. Fig. 5 shows the ResNet50 architecture.
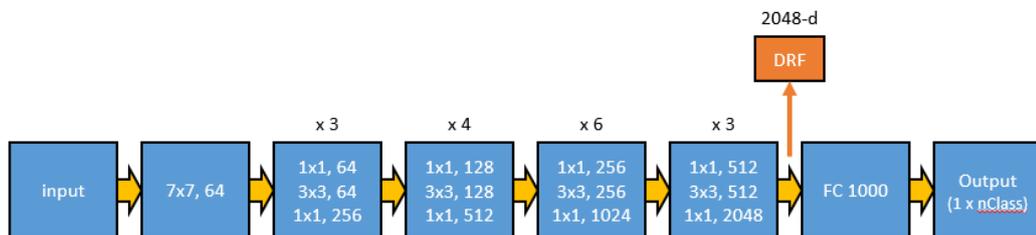
B. Research Flowchart

This research step follow the flowchart, which is shown in Fig. 4. First, it is started by collecting primary input images using drones. Then, the images are labeled based on the altitude and the vegetation type. The next step is preparing the simulation setup by configuring some of the ResNet50 parameters. The simulation is run after that step. The last activity is analyzing the simulation results.
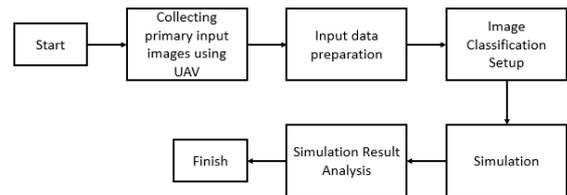

Fig.4. Research Flow Chart

C. Image Collection and Input Data Preparation

Preparing input data is an important part of running the algorithm. The type of input data may affect the image processing result. In this research, input data is divided into two, the crop type and the height of the observation point from the target. Rice field, and tea plantations have been selected for the crop type as the input. Both of them are the vegetation widely planted in the Bandung District area. Focusing the target on these can contribute to the local vegetation monitoring system.
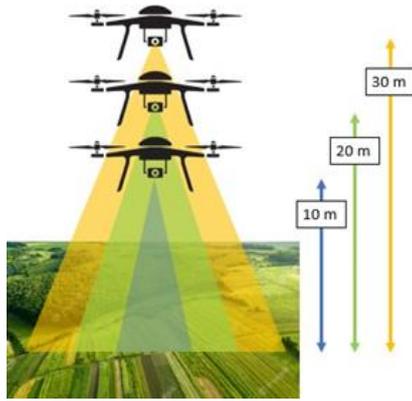

Fig.5. ResNet50 Architecture

148

Fig.6. Image Retrieval Method

them in the direction of lower loss. The current weight is determined by using Eq. 1. Then, the gradient descent *J(W)* is calculated by using Eq. 2.
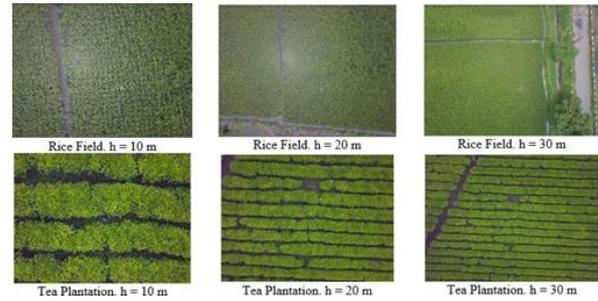

Fig.7. Example of Crop Images Captured from Different Altitudes

$$W_{i+1} = W_i - \alpha \frac{\partial}{\partial W_i} J(W) \qquad (1)$$

$$J(W) = \frac{1}{2m} \sum_{i=1}^{m} [h_W(x_i) - y_i]^2 \qquad (2)$$

Furthermore, there are three different altitude positions selected for this research regarding the altitude of observation position. Those are 10 m, 20 m, and 30 m of height from the target. This method is selected to provide diverse images that help the algorithm learn to recognize the target in different scenarios. Moreover, suppose there is a case that another unwanted object captured in an image, this method will help the algorithm to distinctis captured in an image. In that case, this method will help the algorithm distinguish it accurately. The images are taken using a drone from 10 pm – 12 pm local time. The image retrieval period is intentionally restricted to obtain the image with relatively similar illumination intensity. Fig. 6 shows how the aerial images are taken for this research need. The result of this capturing method is shown in Fig.7.

CLR is a learning rate method which is proposed by [16]. The main point of this method is various learning rates to obtain optimum results. The advantage of this method is that the time needed to achieve the optimum result is far shorter than the homogenous learning rate method. Furthermore, CLR needs no additional computation in its implementation. One of the CLR cycle formss is the triangle, which is well known as the triangular policy. Fig. 8 shows how triangular policy CLR works. In this research, the CLR method is applied in Test 1 – Test 4. Test 5 and 6 are intentionally not using the CLR to produce another result.

D.   Classification Preparation using ResNet50

After taking the crop images, the next step of this research is preparing the classification process. ResNet50 algorithm is selected as the CNN model to classify the images. The result can be optimized, and the ResNet50 configuration should be examined such as:

- The input image size
- Cyclic Learning Rate (CLR) and CLR policy
- Learning Rate
- Dropout

The original image size used in this work is $4000 \times 3000 \times 3$ pixels. Before the clasification process, this size is cropped and/or resized. . This step is named the pre-processing process. For the Test 1 – Test 4, the images are cropped into $500 \times 500 \times 3$ pixels, and then those cropped images are resized into $256 \times 256 \times 3$ pixels for the Test 1 and Test 2, and $224 \times 224 \times 3$ pixels for Test 3 and Test 4. In Test 5 and Test 6, images are only cropped into $224 \times 224 \times 3$ and $256 \times 256 \times 3$, respectively.

For training a neural network, the learning rate is a crucial parameter. It holds an important role in effective and faster network training. It determines how much of the loss gradient is needed in the current weight to move

Another configuration that is determined in this work is a dropout. Dropout is a technique to overcome the overfitting of machine learning systems. Dropout drops units from the neural network randomly during the training process. Dropout can avoid co-adapting on many units. In this work, Test 2 and Test 4 use dropout with numbers 0.5 and 0.7, respectively.

From those configuration points, 7 different test setup have been initiated. Some configurations are set to be the same for all types of tests, such as CLR policy, and epoch. The CLR policy used in this work is triangular, while the epoch is 50. The input image resolution for all setups is the same, 4000×3000 pixels of colored image. For Test 1, Test 2, Test 3, and Test 4, the input image will be cropped into $550 \times 500 \times 3$ image size. Furthermore, it will be resized into $256 \times 256 \times 3$ image sizes for Test 1 and Test 2 and $224 \times 224 \times 3$ image sizes for Test 3 and Test 4. est 5 and 6 will be cropped into $224 \times 224 \times 3$ of image size and will not be resized.

In terms of CLR usage, Test 1, Test 2, Test 3, and Test 4 use the CLR function. Meanwhile, Test 5 and Test 6 will be executed without using CLR. For the dropout, only Test 2 and Test 4 use this function with the value of 0.5 and 0.7 for Test 2 and Test 4, respectively. The other test setups are run without using

(149)

dropout. For the learning rate setup, Test 1 and Test 2 are using the same rate, which is $1\times10^{-6} - 1\times10^{-3}$. Test 3 and Test 6 are using $1\times10^{-7} - 1\times10^{-4}$ of learning rate. Test 4 and Test 5 are using $1\times10^{-7} - 5\times10^{-4}$ of learning rate. Table 1 shows the test configuration that has been done in this research.
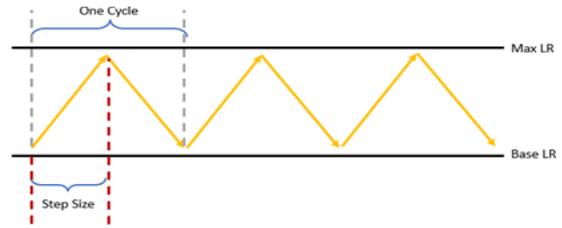

Fig.8. Triangular Policy CLR method

Table 1. ResNet50 Test Configuration

Parameter	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Input Image	$4000 \times 3000 \times 3$	$4000 \times 3000 \times 3$	$4000 \times 3000 \times 3$	$4000 \times 3000 \times 3$	$4000 \times 3000 \times 3$	$4000 \times 3000 \times 3$
Cropped Size	$550 \times 500 \times 3$	$550 \times 500 \times 3$	$550 \times 500 \times 3$	$550 \times 500 \times 3$	$224 \times 224 \times 3$	$256 \times 256 \times 3$
Resized	$256 \times 256 \times 3$	$256 \times 256 \times 3$	$224 \times 224 \times 3$	$224 \times 224 \times 3$	-	-
CLR	Yes	Yes	Yes	Yes	No	No
Dropout	No	Yes (0.5)	No	Yes (0.7)	No	No
Learning Rate	$1.10^{-6} - 1.10^{-3}$	$1.10^{-6} - 1.10^{-3}$	$1.10^{-7} - 1.10^{-4}$	$1.10^{-7} - 5.10^{-4}$	$1.10^{-7} - 5.10^{-4}$	$1.10^{-7} - 1.10^{-4}$

E. Analysing Simulation Result

In this work, the simulation is set to produce some outputs used to examine which configuration can be used to examine which configuration can classify the vegetations. Those are train accuracy, test accuracy, validation accuracy, and validation loss. The accuracy result will be calculated from the confusion matrix shown in Fig. 9. The formula used to calculate it is Eq. 3 [17]. Furthermore, the validation loss will be observed from the graph by comparing the train and validation loss curves.

$$accuracy = \frac{N1 + N5 + N9}{\sum_{i=1}^{9} N_i} \times 100\% \qquad (3)$$

Fig.9. Confusion matrix for three-class classification.

III. RESULTS

A. Accuracy

The accuracy result of the ResNet50 algorithm is presented in this section. It is exhibited in two results, train accuracy and test accuracy. Table 1 and Table 2 show the train accuracy and test accuracy results, respectively, and Fig. 10 shows the test accuracy curves. The Test 1 train accuracy is 100% which is the highest. Then, Test 2, Test 3, Test 4, Test 5, and Test 6 train accuracy result are 99.98%, 99.37%, 99.78%, 99.03%, and 96.65% respectively. Overall, the train accuracy result of every test is more than 90%. Furthermore, the test accuracy result has been obtained and reported in this paper. It shows that the test accuracy result varied from 91.01% to 93.85%. Test 4 test accuracy has come as the best with 95.42%, and Test 6 test accuracy yields the lowest result with 91.01%. Test 1, Test 2, Test 3, and Test 5 results are 93.85%, 93.14%, 94.83%, and 91.07% subsequently. According to these results, all the test result is more than 90%.

Table 1. Train Accuracy Result

	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Train Accuracy	100%	99.98%	99.37%	99.78%	99.03%	96.65%

Table 2. Test Accuracy Result

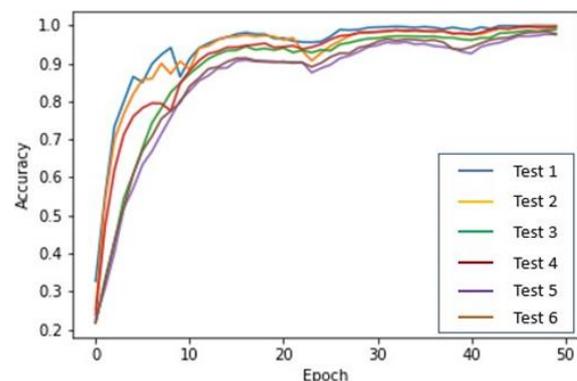|	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Test Accuracy	93.85%	93.14%	94.83%	95.42%	91.07%	91.01%

Fig.10. Comparison of Accuracy Result of Every Test Scenario

B. Loss Result

Loss Result is an observation result for the ability of the test scenario to distinguish the target's properties and recognizes them as two different objects. The loss result of every test contains two curves, the training, and the validation curve. The blue line is the loss train result, while the orange is the loss model plot result. Ideally, the loss validation plot should be as minimum as possible (less than 1). Fig. 11 shows the plot of the loss model of every test scenario. Ideally, the loss plot should be as minimum as possible (less than 1). It may cause the model to be uncertain with the classification result although the classification result is correct.
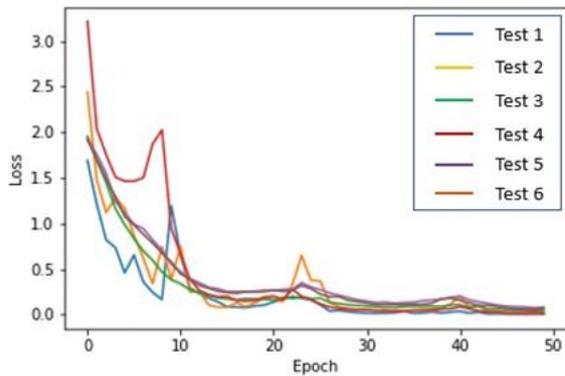

Fig.11. Loss Result Comparison of Every Test Scenario.

C. Overall Result

By considering the accuracy and loss result, another approach has been made to determine the best test. It compares every test loss result curve with their loss validation curve. Fig. 12 shows the comparison of them. The result is that Test 3 has come out as the best configuration in this research. Although Test 3 accuracy result is slightly behind Test 4, its loss performance exceeds the Test 4 loss result. It means that the Test 3 configuration can classify accurately and has a higher degree of certainty.

IV. DISCUSSION

A. Accuracy Result Discussion

Test 5 and Test 6 accuracy results, which are not using CLR and Dropout, are the lowest among the other tests. It means the CLR and dropout is important to increase the ResNet50 accuracy. The dropout value seems to have to be adjusted based on the test accuracy result. According to Test 1 – Test, only Test 2 and Test 4 use dropout with different values. However, only the Test 4 produces better accuracy than Test 1 and Test 3, that are not using dropout. Test 2, which uses dropout with a lower value than Test 4, is less accurate than Test 1 and Test 3, which do not use dropout. The learning rate plays a significant role in increasing the ResNet50 test accuracy. Among Test 1 – Test 4, Test 3 and Test 4 have a smaller learning rate, and both have a better result than Test 1 and Test 2, which have a higher learning rates.

According to previous work's results done by [6][7][18-20], this research's accuracy is notably higher. The [6] accuracy reaches 88.9%, and [7] accuracy result is about 90%. This work accuracy is more than 91%. This result also outperforms [21][22], which use different deep learning method. This result shows that the combination of multi-altitude images with the ResNet50 algorithm can increase the test accuracy.

B. Loss Result Discussion

According to result, scenarios achieve a loss value of more than one, although all accuracy scenarios' accuracy is more than 90. Test 1 and Test 4 loss results are the most unstable. The validation curve contains a lot of spikes, and the value is more than 2. Test 5 and Test 6 loss results have better results than Test 1 and Test 4. Their validation and train curves are alike compared to Test 1 and Test 4 results. However, their lost value is more than two, which means that the corresponding test setup is unsure of their result. Test 2 and Test 3 validation results are better than the rest. Both validation curves are adjacent to their train curve. However, Test 2 validation result has a significant spike. Meanwhile, the Test 3 validation curve is relatively smooth. This phenomenon can be occurred due to the outliners in the dataset. It means, among the test scenario, Test 3 setup has produced a better assurance of the object classification of the other test setups. Comparing this loss result with the previous work by [5], the loss value is higher while the last result work is within 0 and 1. It means that this loss result needs improvement.
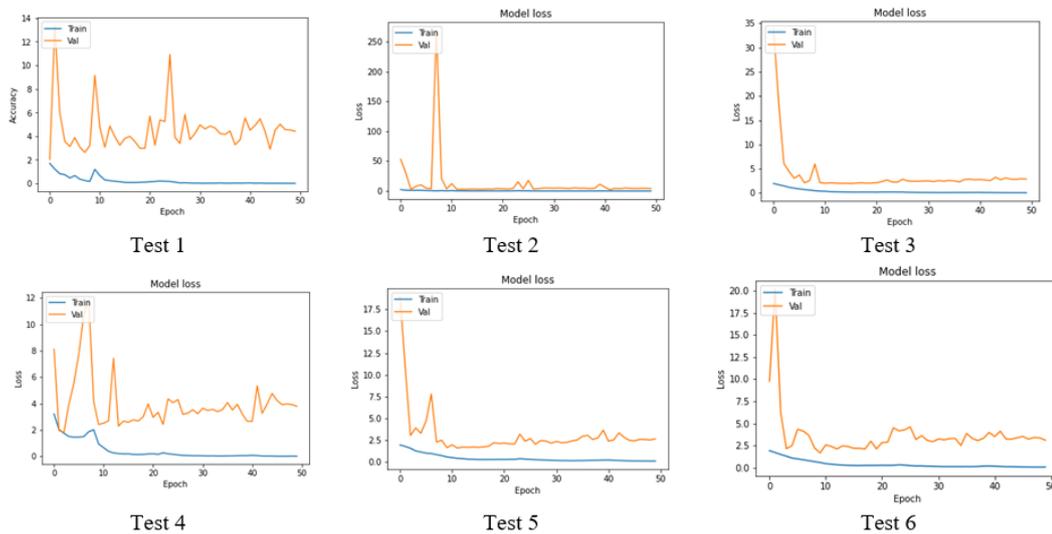
Fig.12. Loss and Validation Result Comparison of Every Test Scenario

V. CONCLUSION

The vegetation classification from different levels of aerial images using ResNet50 has been done. The aerial image has been captured using a combination of different heights and feed into the ResNet50 test setups. Some configurations have been investigated to find the best setup which can classify the object accurately. The CLR and learning rate contributed toincreasing the ResNet50 performance. The lower learning rate, the better the classification result will be. The dropout can give a good result by adjusting its value. In this research, the setup with a small CLR and without dropout has produced a good accuracy and loss result. In the future, more input images will be added to increase the accuracy of the ResNet50 algorithm. The various plants will also be used to enrich the algorithm classification capability.

REFERENCES

[1] Y. Zhou, J. Luo, L. Feng and X. Zhou, "DCN-Based Spatial Features for Improving Parcel-Based Crop Classification Using High-Resolution Optical Images and Multi-Temporal SAR Data," Remote Sensing 11, p. 13, 2019.

[2] N. Pratiwi, Y. N. Fu'adah and E. Edwar, "Early Detection of Deforestation through Satellite Land Geospatial Images based on CNN Architecture," INFOTEL, pp. 54-62, 2021.

[3] S. Sa'idah, N. K. C. Pratiwi, B. S. Aprilia, R. Magdalena and Y. N. Fu'adah, "Land cover classification using Grey Level Co-occurrence Matrix and Naive Bayes," Journal of Physics Conference Series, 2019.

[4] C. Zhang, T. A. Douglas and J. E. Anderson, "Mapping Vegetation and Seasonal Thaw Depth in Central Alaska Using Airborne Hyperspectral and LiDAR Data," in IEEE International Geoscience and Remote Sensing Symposium, 2020.

[5] M. Gibril, H. Shafri, A. Shanableh, R. Al-Ruzouq, A. Wayayok and S. Hashim, "Deep Convolutional Neural Network for Large-Scale Date Palm Tree Mapping from UAV-Based Images," Remote Sensing, p. 14, 2021.

[6] H. Li, B. Hu, Q. Li and L. Jing, "CNN-Based Tree Species Classification Using Airborne Lidar Data and High-Resolution Satellite Image," in IEEE International Geoscience and Remote Sensing Symposium, 2020.

[7] M. Onishi and T. Ise, "Explainable identification and mapping of trees using UAV RGB image and deep learning," Sci Rep 11, p. 903, 2021.

[8] Z. Li, F. Liu, W. Yang, S. Peng and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," IEEE Transactions on Neural Networks and Learning Systems, pp. 1-21, 2021.

[9] T. Lindeberg, "Scale Invariant Feature Transform," Scholarpedia, 2012.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005.

[11] T. Ahonen, A. Hadid and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006.

[12] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," The Journal of physiology, p. 106–154, 1962.

[13] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[14] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional

Neural Networks," in Advances in Neural Information Processing Systems 25, 2012.

[15] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in International Conference on Learning Representations ICLR, 2015.

[16] L. N. Smith, "Cyclical Learning Rates for Training Neural Networks," in IEEE Winter Conference on Applications of Computer Vision (WACV), 2017.

[17] R. Kusuma Putra, E. Rachmawati and F. Sthevanie, "Vision-Based Employee Activity Classification", in International Conference on Information and Communication Technology (ICoICT), Yogyakarta, Indonesia, 2021.

[18] V. Bodhwani, D.P. Acharjya, U. Bodhwani, "Deep Residual Networks for Plant Identification", Procedia Computer Science, Vol. 152, pp 186 – 194, 2019.

[19] T. Nguyen Quoc and V. Truong Hoang, "Medicinal Plant identification in the wild by using CNN," 2020 International Conference on Information and Communication Technology Convergence (ICTC), 2020, pp. 25-29.

[20] S. Chavan, J. Ford, X. Yu and J. Saniie, "Plant Species Image Recognition using Artificial Intelligence on Jetson Nano Computational Platform," 2021 IEEE International Conference on Electro Information Technology (EIT), 2021, pp. 350-354.

[21] R. Nijhawan, H. Sharma, H. Sahni and A. Batra, "A Deep Learning Hybrid CNN Framework Approach for Vegetation Cover Mapping Using Deep Features," 2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), 2017, pp. 192-196.

[22] A. Muneer and S. M. Fati, "Efficient and Automated Herbs Classification Approach Based on Shape and Texture Features using Deep Learning," in IEEE Access, vol. 8, pp. 196747-196764, 2020.