Waste Image Segmentation Using Convolutional Neural Network Encoder-Decoder with SegNet Architecture

Urratul Aqyuni
Dept. of Computer Science
Universitas Diponegoro
Semarang, Indonesia
urratulaqyuni@students.undip.ac.id

Sukmawati Nur Endah
Dept. of Computer Science
Universitas Diponegoro
Semarang, Indonesia
sukmane@lecturer.undip.ac.id

Priyo Sidik Sasongko
Dept. of Computer Science
Universitas Diponegoro
Semarang, Indonesia
priyoss_undip@yahoo.co.id

Retno Kusumaningrum
Dept. of Computer Science
Universitas Diponegoro
Semarang, Indonesia
retno@live.undip.ac.id

Khadijah
Dept. of Computer Science
Universitas Diponegoro
Semarang, Indonesia
khadijah@live.undip.ac.id

Rismiyati *
Dept. of Computer Science
Universitas Diponegoro
Semarang, Indonesia
*Corresponding author:
rismiyati@live.undip.ac.id

Hanif Rasyidi
College of Engineering & Computer Science
The Australian National University
Canberra, Australia
hanif.rasyidi@anu.edu.au

***Abstract*—Sustainable waste management has become one of the great concerns in many countries to reduce the negative environmental and health impact caused by the increase of unmanaged household garbage. One of the key elements is waste recycling, that relies on the process of garbage sorting to separate the recycled garbage into different categories for further processing. The sorting process in waste recycling is usually done manually by hand-picking. Therefore, a system that can recognize waste automatically is needed so that the waste sorting process can be done more quickly and accurately. In this paper, we propose a waste segmentation method using Convolutional Neural Network based on the Encoder-Decoder approach of SegNet architecture [5]. We compare two different setups of the architecture based on the number of filters in each convolutional layer, then evaluate our model using TrashNet benchmark dataset. Our experiment shows that one of our proposed architecture managed to achieve 82.95% intersection over union (IoU) value, which is higher than the previous work by the TrashNet developer.**

Keywords—Convolutional Neural Network, Encoder-Decoder, SegNet, waste, image segmentation

I. INTRODUCTION

The increase of population and new residential areas raise various problems in developing country due to the high volume of household garbage produced each year. In 2016, Indonesia Central Bureau of Statistics recorded that there were 65,200,000 tons of waste per year with a total population of 261,115,546 people in Indonesia [1]. Various strategies have been done by the government as an effort to solve the waste problem. One of them is waste recycling. The process of waste recycling starts with garbage sorting, a process that is still done manually in Indonesia. This manual sorting makes the waste sorting process ineffective and inefficient, so a faster and more accurate vision-based waste classification or recognition process is needed to support sustainable management.

The automatic garbage classification process starts by taking an image of a certain batch of mixed household waste. The system then extracts the image features to use them for object segmentation and classification. Segmentation is a process for dividing an image based on the similarity of a pixel to its neighbor pixels to separate an object with other objects and the image background. The segmented image then can be used to further classify the object in more detail classification process, where good segmentation result will provide a positive impact on more optimal classification results [2].

Recently, deep learning networks have resulted in strong performance improvements in image segmentation. Several deep learning methods that have been proposed by researchers for image segmentation such as convolutional neural networks (CNN), recurrent neural networks (RNNs), long short term memory (LSTM), and generative adversarial networks (GANs). CNN is the most successful method and is widely used in deep learning, especially in solving problems that are related to computer vision [3]. In performing image segmentation, CNN is modified to feature map upsampling and it is called CNN encoder-decoder [8]. There are several types of CNN encoder-decoder architectures that are often used in image segmentation, such as SegNet [4] and U-Net [5]. SegNet uses a CNN encoder-decoder which consists of 13 convolution layers and uses pooling layer to transfer the feature map from the encoder network to the decoder network that can increase the resolution of the feature map. This paper proposes the use of convolutional neural network encoder-decoder SegNet architecture for segmentation of the waste image.

In this paper, we compare two experimental trash segmentation scenarios using the same basic architecture. The first scenario uses a different number of filters in each

978-1-7281-9526-1/20/$31.00 ©2020 IEEE

convolution layer and the second scenario uses the same number of filters in each convolution layer. The reason for comparing those scenarios is based on the research of image segmentation which states that the changes in the number of filters will affect the architectural performance [12]. In the first scenario, we use a different number of filters based on the original SegNet architecture [4] with 64, 128, 256, and 512 at the end of each feature extraction block. The second scenario which uses the same number of filters in each convolution layer refers to the work on [3] on biomedical image segmentation with 128 filters in each feature extraction block. In each scenario, there are several parameters tested to determine the effect of these parameters on the performance of the SegNet architecture. These parameters are learning rate, number of epochs, usage of dropout layers, and kernel size.

In the future, this method is expected to be useful in solving waste classification. for example, this method can be applied to a smart trash can that able to classify waste automatically.

II. METHODOLOGY

This research is divided into six main stages, data collection, preprocessing, division of training and testing data, data augmentation, segmentation using CNN encoder-decoder SegNet architecture, and architecture evaluation. The research stages can be seen in Fig. 1.

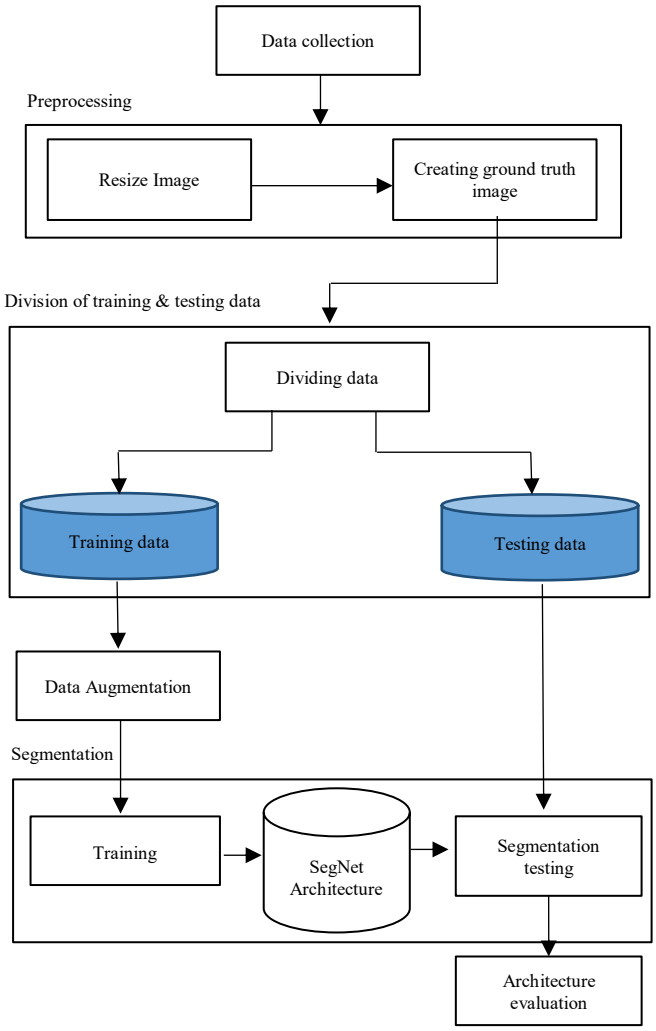


Fig. 1. Research stages

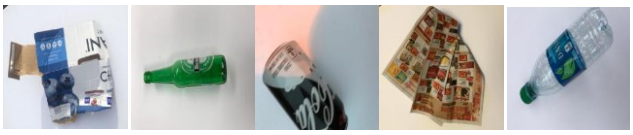


Fig. 2. Example of Trashnet dataset

A. *Data collection*

The dataset used in this research is waste image dataset called Trashnet. The Trashnet dataset is published by Mind Yang and Gary Thung through the Github website. The Trashnet dataset contains images of waste objects with six classes totalling 2,527 images. The six classes include paper, glass, metal, cardboard, plastic, and small trash. The trash object in the image is taken against a white cardboard background using room lighting. The original size of the image in the downloaded dataset is 512x384 [6]. The example of data can be seen in Fig. 2.

B. *Preprocessing*

In the preprocessing stage, there are two processes, image resizing and creating ground truth image. In the image resizing process, all images that were originally 512 x 324 pixels are converted into 96 x 96 pixels. The process of creating ground truth images is done manually using GIMP (GNU Image Manipulation Program).

C. *Division Training and Testing Data*

The division of training and testing data is used to separate data for training and testing. The 2,527 images of dataset Trashnet are divided into 2,021 as training data and 506 as testing data. The division of training and testing data is done by selecting data randomly with the number adjusted to the ratio of the number of images in each class in the dataset.

D. *Data Augmentation*

In this stage, the training data which has been divided based on the predetermined ratio will go through the augmentation process to accumulate image data. In this paper, the augmentation strategies were rotation and flipping. Rotation is done by rotating the image according to the specified degrees. In this research, rotations degree applied is 90°, 180°, and 270°. Flipping is done by mirroring the image horizontally and vertically. The number of training data which is originally 2,021 images becomes 12,126 images after performing data augmentation.

E. *Segmentation*

Segmentation stage consists of training and testing process. The training process is used to build a model by using training data. The testing process is used to evaluate the performance of the segmentation model by using the architecture that has been built and testing data. The segmentation is done by using the CNN encoder-decoder SegNet architecture.

CNN encoder-decoder consists of the encoder part that consists of convolutional layers and the decoder part that takes the feature vector generated by encoder part as input and produces a pixel map of the segmentation results.

Research by Noh et al. about semantic segmentation with deconvolution networks [8] and Badrinayan et al. about semantic segmentation with SegNet [4] are done by using CNN VGG16 type architecture. The CNN VGG16 architecture consists of 13 layers of convolution. At the end of the architecture, there is the Softmax classifier.

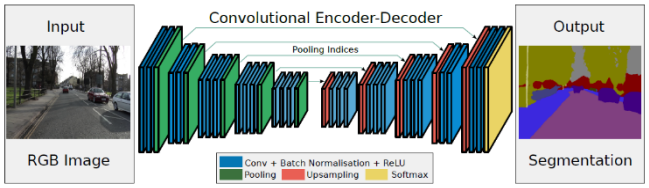


Fig. 3. CNN encoder-decoder SegNet architecture[4]

The form of the CNN encoder-encoder SegNet architecture can be seen in Fig. 3. Each encoder layer in the encoder part performs a convolution process with a certain number of filters to produce a feature map. The convolution results are normalized using the batch normalization layer and followed by the rectified linear unit (ReLU) activation function. Then, the results are sub-sampled in the pooling layer.

The decoder layer in the decoder section up-samples the feature map that has been generated by the encoder part. The feature map is convoluted using many filters on the decoder to get a feature map with a higher dimension. Then the convolution result is normalized using batch normalization followed by the ReLU activation function. Furthermore, the feature map with a higher dimension is classified for each pixel using the softmax classifier layer. The output from the softmax classifier layer is the classification probability of each pixel as N channels, where N is the number of classes. The segmentation prediction results are adjusted to the class that has the highest probability for each pixel.

The convolution layer consists of neurons which are arranged to form a filter with length and height, namely pixels [9]. The convolutional layer works by performing convolutional operations on the output of the previous layer. The convolution process can be seen in Fig. 4. In Fig. 4, the kernel (yellow box) will be executed by multiplying the filter on the image (green box) from the left corner then moving it according to the stride value. If the value of stride is 1 then the kernel will be shifted by 1 pixel continuously.

The normalization process is carried out using batch normalization. It is called batch normalization because during the training process each input layer is normalized using the mean and standard deviation or variance values in the batch. The batch normalization formula can be seen in (1) where $x_i$represents a pixel value, then μ_B represents the mean and σ_B^2represents the standard deviation with the epsilon $\in$.

$$\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \quad (1)$$

The pooling layer uses a feature map as input and processes it with various statistical operations based on the nearest pixel value. In the encoder part, the statistical process is used to take the maximum pixel value.

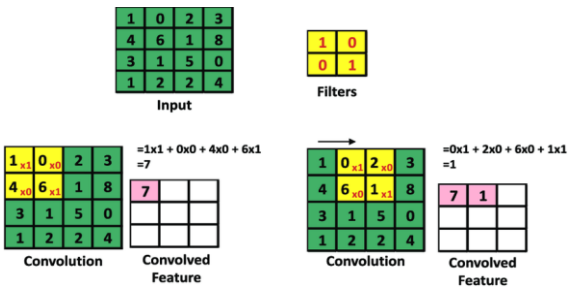


Fig. 4. Convolution process[8]

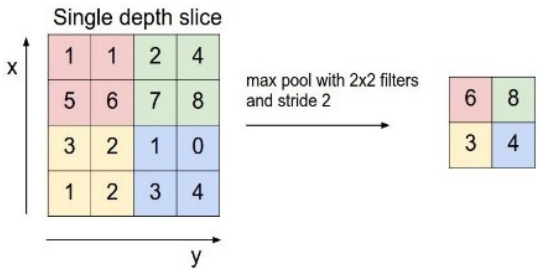


Fig. 5. Max-pooling process [9]

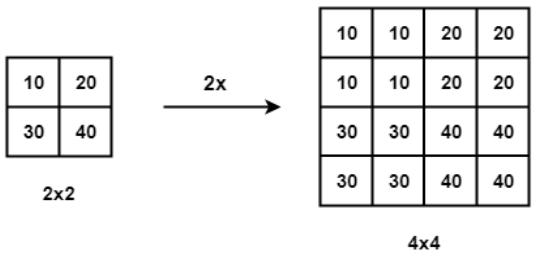


Fig. 6. Nearest neighbour interpolation [11]

The pooling layer works by reducing the size of the feature map so the pooling layer can reduce the risk of overfitting and speed up computation. The max-pooling process can be seen in Fig. 5.

The upsampling layer uses mathematical operations such as interpolation to increase the size of the image. Interpolation is the process of resampling (re-creation) of the initial image by determining the values between the specified pixels. This research uses nearest-neighbour interpolation to achieve its purpose. An example of the nearest neighbour interpolation results can be seen in Fig. 6.

In this research, we also perform the experiments using a dropout layer. The dropout layer is used to reduce the risk of overfitting in deep learning models. The dropout layer is a layer that regularizes the neural network where the neurons used during training will be randomly selected and not all neurons are used during training. The dropout layer on the CNN encoder-decoder SegNet architecture in this paper is used after the ReLU activation function.

The Softmax classifier can be used to classify two or more classes. Softmax classifier is another form of logistic regression algorithm. Logistic regression algorithms are usually used to solve problems in binary class classification. The formula used in the softmax classifier can be seen in (2) where f_j is the result of the function of the j element, e is the exponential function, z represents the model hypothesis, and k is the number of input dimensions

$$f_j(z) = \frac{e^{z_j}}{\sum_{k=1}^{k} e^{z_k}} \quad (2)$$

The detail of SegNet Architecture that is used in the experimental scenario can be seen in Table 1 where E represents the encoder, D represents the decoder, and BN represents batch normalization.

TABLE I. ARCHITECTURE DETAIL FOR EACH SCENARIO

Set	Scenario 1		Scenario 2	
	Type of layer, repetition	*Num filter*	*Type of layer, repetition*	*Num filter*
E 1	Conv+BN+ReLU,2	64	Conv+BN+ReLU,2	128
E 2	Conv+BN+ReLU,2	128	Conv+BN+ReLU,2	128
E 3	Conv+BN+ReLU,3	256	Conv+BN+ReLU,3	128
E 4	Conv+BN+ReLU,3	512	Conv+BN+ReLU,3	128
E 5	Conv+BN+ReLU,3	512	Conv+BN+ReLU,3	128
D 1	Deconv+BN+ReLU,3	512	Deconv+BN+ReLU,3	128
D 2	Deconv+BN+ReLU,3	512	Deconv+BN+ReLU,3	128
D 3	Deconv+BN+ReLU,3	256	Deconv+BN+ReLU,3	128
D 4	Deconv+BN+ReLU,2	128	Deconv+BN+ReLU,2	128
D 5	Deconv+BN+ReLU,2	64	Deconv+BN+ReLU,2	128

F. *Architecture Evaluation*

In this study, the quality of the architecture is evaluated using an intersection over union (IoU). Intersection over Union (IoU) is a type of evaluation that calculates the overlap area between the segmented object's prediction and the object's area on ground truth. The intersection over union formula can be seen in (3).

$$IoU = \frac{TP}{TP+FP+FN} \quad (3)$$

TP is the number of correct pixels in the area of the segmented object, while TN is the number of correct pixels in the background area. FP represents the number of pixels that are classified incorrectly in the segmented object area, while FN represents the number of pixels that are classified incorrectly in the background area.

III. EXPERIMENT RESULT

In this research, we perform two main scenarios for waste image segmentation. Scenario 1 uses the same number of filters as proposed by Badrinayan et al., while scenario 2 uses 128 filters in each convolution layer as proposed by Mittal et al. Parameters used in each scenario are kernel size, the number of epochs, learning rate, and dropout. Kernel size parameters used are 3x3 and 5x5. The number of epochs used is 15, 30, and 45 where each number of epochs are tested using a learning rate of 0.002, 0.005, and 0.008. The experiments are also conducted using 0.1 dropout value and without dropout. The combinations of parameters are chosen based on the previous study that uses CNN. The reason to use dropout layer is based on a previous study by Mittal et al. about lung segmentation.

By using different combinations of parameters for each experiment, there are 36 types of experiments for scenario 1 and scenario 2, so that the total number of experiments is 72 experiments. The 36 types of experiments can be seen in Table 2.

TABLE II. TYPES OF EXPERIMENTS IN EACH SCENARIO

Num	Dropout (0.1)	Kernel		Epoch (15,30,45) & learning rate(2=0.002, 5=0.005, 8=0.008)								
		3	*5*	*15*			*30*			*45*		
				2	*5*	*8*	*2*	*5*	*8*	*2*	*5*	*8*
1			■			■						
2			■						■			
3			■									■
4			■		■							
5			■					■				
6			■								■	
7			■	■								
8			■				■					
9			■							■		
10	■		■			■						
11	■		■						■			
12	■		■									■
13	■		■		■							
14	■		■					■				
15	■		■								■	
16	■		■	■								
17	■		■				■					
18	■		■							■		
19		■				■						
20		■							■			
21		■										■
22		■			■							
23		■						■				
24		■									■	
25		■		■								
26		■					■					
27		■								■		
28	■	■				■						
29	■	■							■			
30	■	■										■
31	■	■			■							
32	■	■						■				
33	■	■									■	
34	■	■		■								
35	■	■					■					
36	■	■								■		

Each experiment is evaluated by calculating the intersection over union (IoU) value. The IoU value is then compared to determine the effect of each parameter on architectural performance. Best IoU parameter combination can be seen in table 3 and table 4.

In scenario 1, the kernel size parameter does not affect IoU value as shown in table 3 while the use of dropouts greatly affects the IoU value. This condition may happen because the use of dropouts in each encoder and decoder layer causes many convolutional neurons to be inactive, hence reducing the architectural performance. The highest IoU value generated in scenario 1 is using the experiment number 3 that is without dropout, kernel size 5x5, epoch 45, and learning rate 0.008. Those combinations generated 82.95% IoU value.

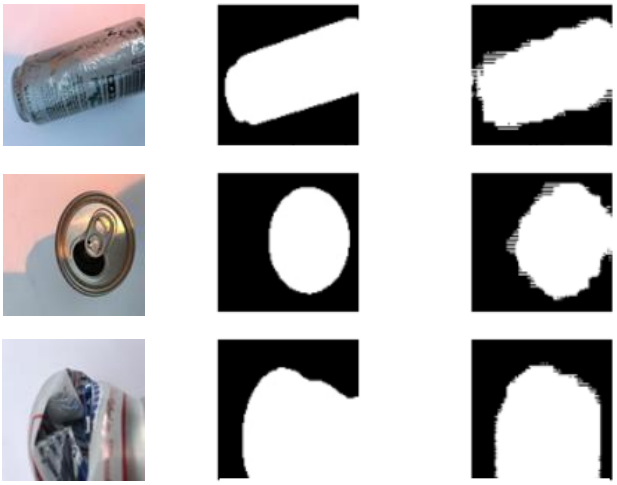
In the results of scenario 2, the kernel size parameter does not have an effect on IoU value as shown in table 4. But IoU value that generated using kernel size 3x3 has higher than IoU value that generated using kernel size 5x5. This condition may happen because scenario 1 uses a greater number of filters, whereas scenario 2 uses a smaller number of filters so that the smaller kernel size will increase the IoU value. The highest IoU value generated in scenario 2 is using the experiment number 21 that is without dropout, kernel size 5x5, epoch 45, and learning rate 0.008. That combination generated 82.78% IoU value.

TABLE III. COMPARISON OF BEST IOU VALUES IN SCENARIO 1

Parameter Combination	Experiment	IoU Value
Without Dropout-kernel 5x5	3	**82.95%**
With Dropout-kernel 5x5	12	73.82%
Without Dropout-kernel 3x3	21	82.94%
With Dropout-kernel 3x3	33	70.12%

TABLE IV. COMPARISON OF BEST IOU VALUES IN SCENARIO 2

Parameter Combination	Experiment	IoU Value
Without Dropout-kernel 5x5	2	82.02%
With Dropout-kernel 5x5	12	72.01%
Without Dropout-kernel 3x3	21	**82.78%**
With Dropout-kernel 3x3	30	67.78%



(a) (b) (c)

Fig. 7. (a)Input image (b)ground truth image (c)predicted image segmentation

Overall, the average of IoU value obtained in scenario 1 is higher than the average of IoU value obtained using scenario 2. The difference between the average value obtained is up to 3%. This result is because scenario 1 uses a greater number of filters compared to scenario 2 so that the convolution process carried out in scenario 1 is deeper and more complex than in scenario 2.

From the whole experiment, the highest IoU value is achieved by scenario 1, experiment number 3. The IoU value obtained is 82.95%. The results of segmentation can be seen in Fig. 7. In Fig. 7 it can be seen that the prediction of segmentation results is not very good. However, according to the IoU value obtained, around 70-80% of the pixels are close to the ground truth pixel.

This proposed method can only handle properly images where the color of the trash object contrasts with the background. Meanwhile, for garbage objects whose color is similar to the background (white), the segmentation results are not very good. This method has only been tried on the Trashnet dataset where the waste category is paper, glass, metal, cardboard, plastic, and small trash, for other waste categories this method has not been tested.

IV. Conclusion

Based on the experimental results in this paper, it can be concluded that the parameters tested have different effects on the IoU value obtained for image segmentation of waste objects. The use of a 5x5 kernel size provides a better IoU value than the 3x3 kernel size in scenario 2, while in scenario 1 the difference in results obtained is not much different. The use of the dropout layer causes a significant decrease in the IoU value in the architecture. For the whole experiment, scenario 1 which uses the SegNet architecture with a different number of filters obtains higher average IoU value than scenario 2 which uses the same number of filters in each convolution layer. The best combination that generated the highest IoU value is using kernel size 5x5, without dropout, epoch 45, and learning rate 0.008 that got 82.95% of IoU value.

Acknowledgement

The authors would like to acknowledge the research funding supported by the Faculty of Science and Mathematics, Diponegoro University under the Grant of Primary Research– Contract Number 2007/UN7.5.8/PP/2020.

References

[1] Badan Pusat Statistik. (2018). Statistik Lingkungan Hidup Indonesia (SLHI) 2018. *Badan Pusat Statistik/BPS–Statistics Indonesia*, 1–43.

[2] Azhar, R., Arifin, A. Z., & Khotimah, W. N. (2016). Integrasi Density-Based Clustering dan HMRF-EM Pada Ruang Warna HSI untuk Segmentasi Citra IkanTuna. *Inspiration: Jurnal Teknologi Informasi dan Komunikasi*, *6*(1), 28–37.

[3] Mittal, A., Hooda, R., & Sofat, S. (2018). LF-SegNet: A Fully Convolutional Encoder-Decoder Network for Segmenting Lung Fields from Chest Radiographs. *Wireless Personal Communications*, *101*(1), 511–529.

[4] Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(12), 2481–2495.

[5] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9351, 234–241.

[6] Yang, M., & Thung, G. (2016). *Classification of Trash for Recyclability Status*. 1–6.

[7] Karimpouli, S., & Tahmasebi, P. (2019). Segmentation of digital rock images using deep convolutional autoencoder networks. *Computers and Geosciences*, *126*(October 2018), 142–150.

[8] Noh, H., Hong, S., & Han, B. (2015). Learning deconvolution network for semantic segmentation. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 1520–1528.

[9] Sena, S. (2017). *Pengenalan Deep Learning Part 7 : Convolutional Neural Network (CNN)*.

[10] Ng, W., Minasny, B., Montazerolghaem, M., Padarian, J., Ferguson, R., Bailey, S., & McBratney, A. B. (2019). Convolutional neural network for simultaneous prediction of several soil properties using visible/near-infrared, mid-infrared, and their combined spectra. *Geoderma*, *352*(July), 251–267.

[11] Achmad, B., & Firdausy, K. (2005). Teknik Pengolahan Citra Digital Menggunakan Delphi. Ardi.

[12] Chen, L.-C., Zhu, Y., Papandreou, G., Schrof, F., & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *European Conference on Computer Vision*.