COMPARISON OF THRESHOLDING OTSU AND MORPHOLOGY OPENING PREPROCESSOR FOR IMAGE TEXT DETECTION USING TESSERACT

[1]Oei Hizkia Renat Guntur Sugiarto,[2]Yonathan Purbo Santosa
[1,2]Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Katolik Soegijapranata
[2]yonathansantosa@unika.ac.id

ABSTRACT

Now everything is completely digital, so image processing is very necessary for this era, one of which is image text processing. processing image text and converting to text using OCR and Tesseract help. There are many helpful applications like this on the internet, but if we want to use them on personal documents it will be very dangerous to use applications from the internet. so it is safer if you use your offline application. This program will not only convert the text converter on the image into text format but will also recognize the text later. So data processing will be easier when the final result is in txt format then for the detection of text in the image will use the help of an open source library from python, namely tesseract. but before using this tesseract library, we have to do preprocessing on our image, because tesseract is very sensitive if there is noise in the image, so here I will use preprocessing thresholding binary otsu method and use dilation. This method will later be compared the final result will be more accurate if detected using tesseract. The final result of this project will detect the text in the image, from here it will detect every word in the image, not every sentence, and will also save the text in the image that has been detected using tesseract

Keywords: Opening, Threshold Otsu, Preprocessing, Image Processing, Text Detection

BACKGROUND

Now everything is completely digital, so image processing is very necessary for this era, one of which is image text processing. processing image text and converting to text using OCR and Tesseract help. There are many helpful applications like this on the internet, but if we want to use them on personal documents it will be very dangerous to use applications from the internet. so it is safer if you use your offline application.

This program will not only convert the text converter on the image into text format but will also recognize the text later. So data processing will be easier when the final result is in txt format then for the detection of text in the image will use the help of an open source library from python, namely tesseract. but before using this tesseract library, we have to do preprocessing on our image, because tesseract is very sensitive if there is noise in the image, so here I will use preprocessing thresholding binary otsu method and use dilation. This method will later be compared the final result will be more accurate if detected using tesseract

The final result of this project will detect the text in the image, from here it will detect every word in the image, not every sentence, and will also save the text in the image that has been detected using tesseract

LITERATURE REVIEW

the text on the image is important information, but often the image is made with a foreground and background, so the text on the image is often difficult to recognize, Gilvatta, et al. [1] apply appropriate local boundary rules for line histogram difference sequences to improve text detection security. then Gunawan, et al. [2] shows a recognition system to detect text in images by changing the appearance of images to grayscale to determine the threshold limit, then to detect the display area of a kWh meter, applying a smearing approach that relates labeling components to character segmentation. and to recognize each number using template matching, for the final results of this study the results are not optimal because there are several factors such as there is still noise in the image which interferes with character detection, light effects, camera angles, and smudge boundary effects then Sarief, et al. [3] also detects characters on license plates using template matching, but in this study the image data is processed first using grayscale, smoothing, dilatation, erosion and after that, the results will be compared, and the results of this study succeeded in detecting vehicle license plates. Siregar, [4] conducted research on the OCR method for detecting characters in digital images, in this research he normalized it first before going to the next stage, namely segmentation and assistance, Siregar used Thresholding Otsu before performing text detection using tesseract, then the final results compared the detection results obtained the Otsu thresholding process has been carried out and those that have not, the final results show that the dataset that has been normalized before using the Otsu threshold is better than those that have not been. Ibnutama and Suryanata [5] detect vehicle license plates using LPR (License Plate Recognition). LPR contains 3 main steps, namely the area of plate detection, segmentation, and classification of characters and digits. in this tesseract, research is used for text detection, Binarization Threshold for the segmentation process in digital images. The final results show that Tesseract can detect text in digital images properly. Hoang [6] researched to detect edges or cracks in an image of a building using the Thresholding Otsu preprocessing method. The purpose of this gray intensity adjustment method is to improve the accuracy of the cracks. disappointing results. and the final results of this study indicate that the Otsu thresholding is good for image processing for crack recognition in building images. Nina, et al. [7] stated that the new approach to this problem includes two main variations. One combines Otsu's threshold recursive extension and selective bilateral filtering to allow automatic binarization and segmentation of handwritten text images. Everything else is also built on top of Otsu's method is recursive and adds better background normalization and post-processing steps to the algorithm to make it more robust with very good binary results compared to existing garden boundary algorithms.

Then Said, et al.[8] started a project on image processing using opening and closing morphological processes, Opening and closing processes are those that manipulate erosion and dilation the opening process removes the smaller foreground structure from the styling element, while the

closing process removes the smaller background structure from the styling element. The results of the research show that the opening and closing processes can be used to improve the image. Tangwannawit and Saetang [9] started that presenting an algorithm to recognize lottery digits and developing an application to read and check the lottery digits. This is achieved by using the Tesseract OCR engine in a mobile application. The procedure includes the following, adaptive thresh holding, connected component analysis, location of text lines and digits, and recognition of digits pass 1 and pass 2. The operation of the application will result in 6 digits scanned from the lottery tickets. Then each generated digit will be played through a prerecorded voice. The results of Alpha and Beta testing with 100 sample tickets showed the efficacy of recognition and prize checking at 98.05% and 100% respectively. In conclusion, the results showed that the application has a good accuracy level and can be used in a real-world scenario successfully. Bai [10] has shown Infrared image enhancement is a crucial technique for improving the quality of infrared images. And, the clear image details are important information for infrared image analysis. To effectively enhance the infrared image and make image details clear, a multiscale sequential toggle operator-based algorithm is proposed in this paper. First, the sequential toggle operator, which uses opening and closing as primitives, is constructed and discussed. Secondly, the feature extraction in the infrared image through the sequential toggle operator is given, and the multi-scale extension of the feature extraction is discussed in detail. Finally, the final extracted features of the infrared image are constructed and imported into the original infrared image to produce an enhanced image. In the enhanced image, the image features are enhanced well and the image details are clear. Infrared images obtained from different environments are used in the experiment. The results show that the proposed algorithm is very effective for infrared image enhancement. using opening and closing is constructed to extract the effective image features for infrared image enhancement.

RESEARCH METHODOLOGY

Identification Problems and Literature Study

The topic of this research was obtained from journals regarding the problem of text detection algorithms using OCR, library Tesseract, and image preprocessing by different methods, sourced from various articles from Google Scholar, Research Gate, IEEE Explore, and others. These journals discuss the implementation of OCR (Optical Character Recognition) and different methods in image processing.

Collecting Dataset

The dataset of this project uses ICDAR 2003, ICDAR 2003 is a dataset created for the ICDAR 2003 Robust Reading competition organized by Prof Simon Lucas and his team. ICDAR 2003 has 3 parts of the dataset, the first is the Robust Reading and Text Locating dataset and all images are of JPEG type with various image sizes, the second Robust Word Recognition dataset in this dataset contains various images of single words and the last is the third Robust Character dataset. Recognition in this dataset contains various images of single characters, so in my project,

it is about detection text in images, the first dataset that is suitable for use for my research. There are a total of 113 image datasets

Preprocessing Images

Because the datasets in ICDAR 2003 vary in size, so I made one size first so that the images to be processed and detected have the same image size, I made the image size 400x400 pixels. so that when processed with several preprocessing methods the image sizes are all the same so that the detection results can be optimal, and there are no images that differ in size In this project, later the image will be processed with otsu thresholding and morphology opening, then the results of the two images will be used for text recognition using tesseract so that it will give better results using otsu thresholding image preprocessing or morphology opening. in the preprocessing process using morphology opening using 2 types of kernels, namely kernel cross, and square all one and with the same size, namely 3x3.

Threshold Otsu

My first step will be using Otsu thresholding for image preprocessing which in previous research stated that the threshold technique is one of the segmentation methods in the image that functions to separate the foreground and background by selecting an adequate threshold value to then be converted into a binary image. The binary image must contain all important information about the position and shape of the object contained in the image The image binarization process is carried out using the Otsu method by dividing the histogram of the gray-level image into two different areas. The threshold value to be searched is expressed by k which ranges from 1 to 255.

Opening Operator

For the second step I will use morphology opening, for image preprocessing, operator opening is a combination of erosion and dilation processes, where the image will go through the erosion process first and then go to the dilation stage, The erosion “shrinks” an image according to the shape of the structuring element, removing objects that are smaller than the shape. Then the dilation step “re-grows” the remaining objects by the same shape. which is composed of set operations (union, intersection, complement, etc.) and combinations of operations of erosion and dilation. The two most important transformation operations are opening and closing. The opening of X by structuring element S

Implementation

Using the image dataset from ICDAR 2003, because the size of the ICDAR 2003 dataset varies so I have to change the size to all the same, which is 400 x 400pixel, after changing the image size, then preprocessing the image, entering the morphology opening and thresholding otsu operators. after the image is preprocessed then it enters the image recognition process with tesseract, after the image text is successfully detected by tesseract then compares the final result of text detection it is better to use tesseract only or use images that have been preprocessed using the Morphology Opening & Thresholding Otsu.

Evaluation

The evaluation for this project is to compare the results of text detection on images that have been processed by preprocessing opening with kernel cross, opening with kernel all one, and thresholding Otsu. After that, the three model images will each detect the text image dataset from ICDAR 2003 to measure the character accuracy of the test image, accuracy formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Function 1, TP is True Positive (TP) = detect text and correct the result, TN is True Negative (TN) = partially detect text and wrong False Positive (FP) = was successfully detected and false, False Negative (FN)= not detected even though there is text. but before that, I evaluate the results of each word using Levenshtein distance using the formula:

$$Lev(a,b) = \begin{cases} |a| \\ |b| \\ lev\big(tail(a), tail(b)\big) \\ 1 + min\begin{cases} lev(tail(a), b) \\ lev\big(a, tail(b)\big) \\ lev\big(tail(a), tail(b)\big) \end{cases} \end{cases} \quad (2)$$

Function 2, where a and b correspond to the two input strings and |a| and |b| are the lengths of each respective string. The tail of a string a or b corresponds to all characters in the string except for the first. Where it indicates a[0] and b[0], that is the character in a and b at the 0th element.

ANALYSIS AND DESIGN

Analysis

The process of converting a text image into text using the tesseract OCR library. But before image processing, pre-processing is done to improve image quality and increase the proportion of successful text conversions. The images to be served must have good image quality so that the accuracy level is higher. therefore, I try to add a filter to get maximum results, the filters I use are Grayscale, Morphology Opening, and Threshold Otsu This is also to prove my project does the filter on the image affect the Tesseract OCR make it easier or harder to tesseract to read text.

Pre-processing Images

image preprocessing is one of the steps used to process datasets so that machine training can run well. This step will later make the data that was originally in the form of an image into data in the form of a collection of arrays so that it can be learned by the machine. and steps to eliminate some of the problems that can interfere with data processing.

Threshold Otsu

Otsu's method, named after its inventor Nobuyuki Otsu, is one of many binarization algorithms. Previous research stated that the threshold technique is one of the segmentation

methods in the image that functions to separate the foreground and background by selecting an adequate threshold value to then be converted into a binary image. The binary image must contain all important information about the position and shape of the object contained in the image [5]. The algorithm iteratively searches for the threshold that minimizes the within-class variance, defined as a weighted sum of variances of the two classes (background and foreground). The colors in grayscale are usually between 0-255 (0-1 in case of float). So, If we choose a threshold of 100, then all the pixels with values less than 100 become the background and all pixels with values greater than or equal to 100 become the foreground of the image. the formula for finding the within-class variance at any threshold t is given by:

$$\sigma^2(t) = Wbg(t)\,\sigma^2_{bg}(t) + Wfg(t)\sigma^2_{fg}(t) \quad (3)$$

In Function 3, $Wb(t)$ and $Wfg(t)$ represent the probability of the number of pixels for each class at threshold T and σ 2 represents the variance of color values. To understand what this probability means

$$Wbg(t) = \frac{P_{bg}(t)}{P_{all}}, Wfg(t) = \frac{P_{fg}(t)}{P_{all}} \quad (4)$$

Function 4, Pall is the total count of pixels in an image, Pbg(t) is the count of background pixels at threshold t, and Pfg(t) is the count of foreground pixels at threshold t.

Erosion

Erosion operation is one of the important morphological operations (morphological transformations) that follows a technique of mathematical morphology for the analysis and processing of geometrical structures. Erosion is one of the image processing morphologies, this operation narrows/minimizes the foreground object. and this erosion operation is defined for binary images. for the formula below:

$$A \ominus \mathrm{B} \quad (5)$$

Function 5, can be described as an input image and b is a Structuring element or kernel the result of the above formula gives an eroded picture.

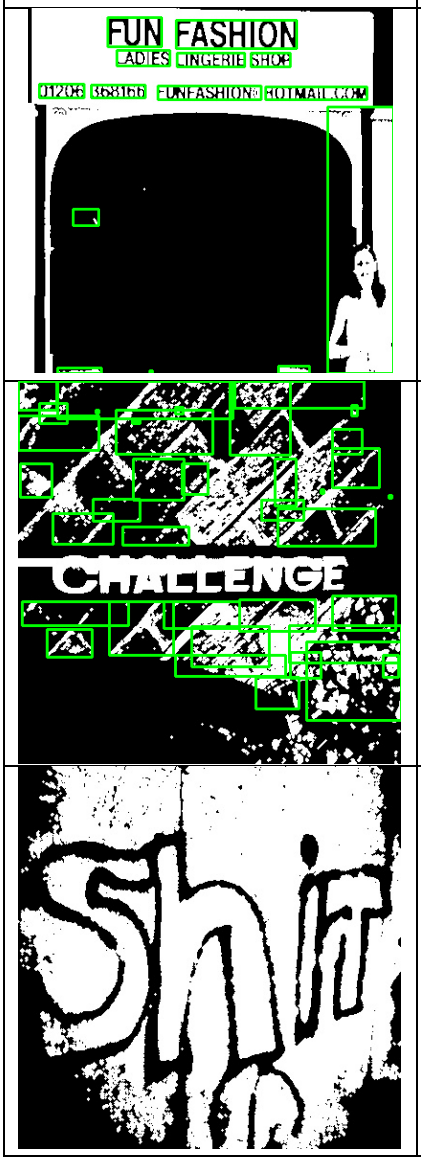
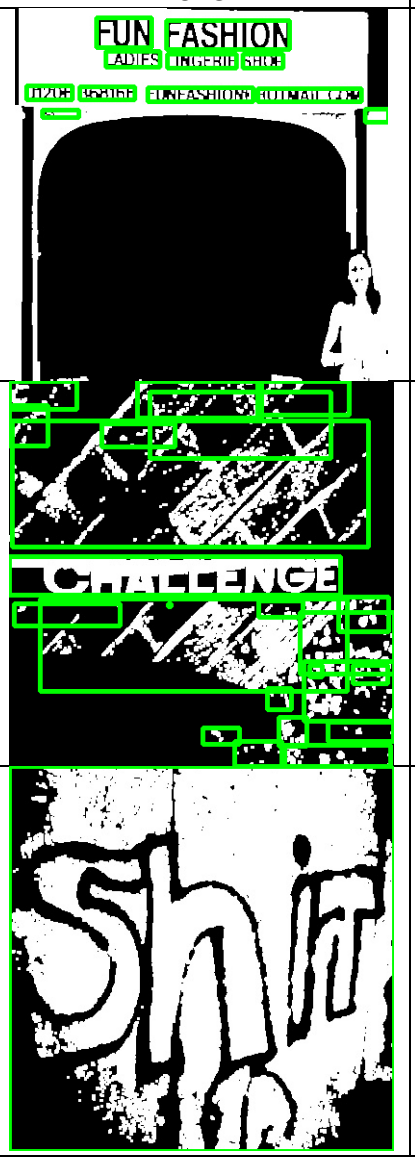
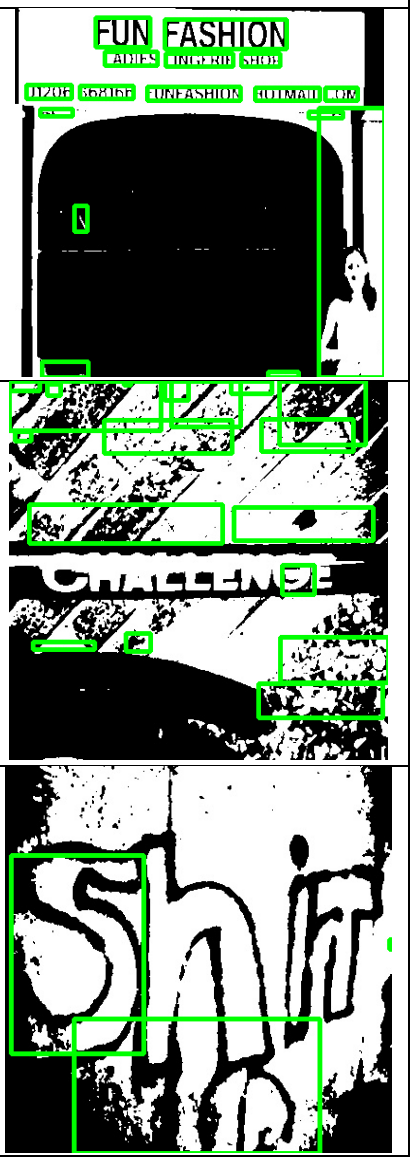
Dilation

Almost the same as the Erosion method, only if Erosion narrows the reverse Dilation, namely expanding the input image. Dilation is also one of the Morphological Operations used to add or expand the shape contained in the input image. still the same as Erosion, the Dilation process also requires a structure element or kernel, and this Dilation operation is defined for binary images for the equation (6). A is the image while B is the structuring element.

$$A \oplus \mathrm{B} \quad (6)$$

Function 6, can be described as an input image and b is a Structuring element or kernel the result of the above formula gives a dilated picture.

Table 1. Image Text Detection

Opening kernel-cross	Opening kernel-square all one	Thresholding Otsu
		

Implementation Project

Using the image dataset from ICDAR 2003, because the size of the ICDAR 2003 dataset varies so I have to change the size to all the same, which is 400 x 400pixel, after changing the image size, then preprocessing the image, entering the morphology opening and thresholding otsu operators. after the image is preprocessed then it enters the image recognition process with tesseract, after the image text is successfully detected by tesseract then compares the final result

of text detection it is better to use tesseract only or use images that have been preprocessed using the Morphology Opening & Thresholding Otsu.

RESULTS AND EVALUATION

Result Image Text Detection

The method was tested with a dataset from ICDAR 2003 which has 113 datasets, this dataset has various forms of text images and various kinds of detailed text images, ranging from various text shapes, text sizes on large and small images, dominant images with other backgrounds in addition to text, there are also images whose text is close to looking very clear and there are also images whose text looks far/unclear. Table 1 are the results of text detection using tesseract and image preprocessing using thresholding otsu and morphology opening kernel size 3x3 cross and square all one.

Evaluation

The evaluation for this project is to compare the results of text detection on images that have been processed by preprocessing opening with kernel cross, opening with kernel all one, and thresholding Otsu. After that, the three model images will each detect the text image dataset from ICDAR 2003 to measure the character accuracy of the test image.

Table 2. Detail Explanation TP, TN, FP, FN

TP	TN	FP	FN
			

Table 2, is a grouping of images resulting from text detection, which will be used to evaluate the confusion matrix, which images are truly positive, then negative, false positive, and false negative.

Table 3. Table Result

Evaluation	Opening(Cross)	Opening(Square One)	Thresholding Otsu
True Positive (TP)	17	16	23
True Negative (TN)	38	32	44
False Positive (FP)	25	25	18
False Negative (FN)	33	40	28

Table 3, is the result of the evaluation grouping on the icdar2003 dataset using the confusion matrix. Table 5.5 shows that the results of the evaluation of the confusion matrix in the Otsu

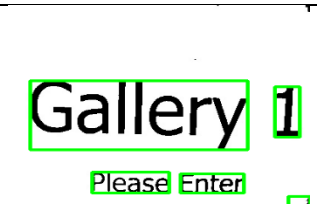
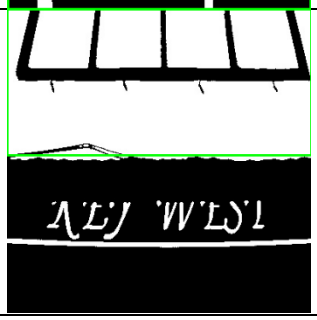
thresholding method are higher than the results of the opening kernel size 3x3 cross and kernel size 3x3 square in all methods.

Table 4. Results of Several Preprocessing Techniques

Pre-processing	Accuracy	Precision	Recall
Opening(Cross)	49%	40%	34%
Opening(All One)	42%	39%	29%
Thresholding Otsu	59%	55%	45%

Table 4, shows the results of the percentage of accuracy, precision, and recall using the evaluation of the confusion matrix. It can be seen that the accuracy of the Otsu Thresholding method is higher than that of the Opening kernel size 3x3 cross and Opening kernel size 3x3 square. Then the distance to the image text are calculated using Levenshtein Distance.

Table 5. Result Levenshtein Distance

Image	Image Text	Text Detection	Levenshtein Distance
	Gallery 1 Please Enter	Gallery 1 Please Enter	0
	Private Cark Park Lion Walk Church	Cark Park Lion Walk Church	11
	Key West		7

In table 5, the results of the evaluation of the ICDAR2003 datasheet using the Levenshtein Distance. In the case of the first image, it shows an image large text with clear separation between text and non-text section. The result shows it has Levenshtein Distance of 0 which means all of the text in the image are detected perfectly. In the case of the second image, the text are smaller than the first image. The detection system fails to detect some parts of the text thus it has non-zero

distance. The last case is a partially shown text. In this case, the detection system fails to detect any of the text. This comparison is then calculated with all of the image available in the dataset. To compare the performance of each method, the total characters that exists in the text will be used to calculate the proportion of misdetection.

Table 6. Levenshtein Distance Comparison Results

Pre-Processing	Total Text	Levenshtein Distance	Average
Opening(Cross)	2642	1373	51.97
Opening(All One)	2642	1514	57.51
Thresholding Otsu	2642	1295	49.02

Table 6, Evaluation results using Levenshtein Distance shows that the Thresholding Otsu value is the least in number than Morphology Opening with a 3x3 kernel-cross and kernel-all one, this proves that Thresholding Otsu is better at detecting each character. can be seen in table 6 the average value of Thresholding Otsu is lower than the average value of Morphology Opening Kernel 3x3 cross and all one, this shows that Thresholding Otsu is better than Morphology Opening Kernel 3x3 cross and all one for text detection in our case.

CONCLUSION

The conclusion that can be drawn is that tesseract can indeed be used directly without preprocessing but the results produced are still lacking, therefore in my opinion based on research what I do pre-processing before the tesseract is very important because it can be seen from the results of several images that I have tested, the final results of detection on the original image are not good compared to the results of image detection that have been preprocessed the image first, and after doing the preprocessing test with 3 ways Morphology Opening kernel 3x3 cross, Morphology Opening kernel 3x3 all one, and Thresholding Otsu, I can conclude that image preprocessing using Thresholding Otsu is better than Morphology Opening kernel 3x3 cross or all one, to detect the 2003 ICDAR Dataset. If you want to detect text using tesseract for further research, you should use a text image dataset whose writing is clear, not handwritten or scribbled, because tesseract is bad at detecting handwriting or unclear text images.

REFERENCES

[1] J. H. Kim, A. Canedo-Rodríguez, J. H. Kim, and J. Kelly, "Simple and Efficient Text Localization for Compressed Image in Mobile Phone," *JSIP*, vol. 05, no. 04, pp. 208–228, 2014, doi: 10.4236/jsip.2014.54022.

[2] R. Gunawan, S. Suwarno, and W. Hapsari, "PENERAPAN OPTICAL CHARACTER RECOGNITION (OCR) UNTUK PEMBACAAN METERAN LISTRIK PLN," *Jurnal Informatika*, vol. 10, no. 2, Art. no. 2, Jan. 2015, doi: 10.21460/inf.2014.102.331.

[3] I. Sarief, H. Y. Biu, F. Harismana, and S. I. Chandra, "Detection of Vehicles Number Plate Using Image Processing with Template matching Method," *Jurnal Ilmiah Telekomunikasi,*

Kendali dan Elektronika Terapan, vol. 7, no. 1, pp. 14–24, Apr. 2019, doi: 10.34010/telekontran.v7i1.1634.

[4] R. Siregar, "Implementasi OTSU Thresholding pada Optical Character Recognition Menggunakan Engine Tesseract," *Jurnal Ilmiah Core IT : Community Research Information Technology*, vol. 7, no. 1, Art. no. 1, Apr. 2019, Accessed: Aug. 01, 2023. [Online]. Available: https://ijcoreit.org/index.php/coreit/article/view/97

[5] K. Ibnutama and M. G. Suryanata, "Ekstraksi Karakter Citra Menggunakan Optical Character Recognition Untuk Pencetakan Nomor Kendaraan Pada Struk Parkir," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 4, no. 4, Art. no. 4, Oct. 2020, doi: 10.30865/mib.v4i4.2432.

[6] N. D. Hoang, "Detection of Surface Crack in Building Structures Using Image Processing Technique with an Improved Otsu Method for Image Thresholding," *Advances in Civil Engineering*, vol. 2018, 2018, doi: 10.1155/2018/3924120.

[7] O. Nina, B. Morse, and W. Barrett, "A recursive otsu thresholding method for scanned document binarization," *2011 IEEE Workshop on Applications of Computer Vision, WACV 2011*, pp. 307–314, 2011, doi: 10.1109/WACV.2011.5711519.

[8] K. A. M. Said, A. B. Jambek, and N. Sulaiman, "A study of image processing using morphological opening and closing processes," *International Journal of Control Theory and Applications*, vol. 9, no. 31, pp. 15–21, 2016.

[9] S. Tangwannawit and W. Saetang, "Recognition of Lottery Digits Using OCR Technology," *Proceedings - 12th International Conference on Signal Image Technology and Internet-Based Systems, SITIS 2016*, pp. 632–636, 2017, doi: 10.1109/SITIS.2016.105.

[10] X. Bai, "Morphological infrared image enhancement based on multi-scale sequential toggle operator using opening and closing as primitives," *Infrared Physics and Technology*, vol. 68, pp. 143–151, 2015, doi: 10.1016/j.infrared.2014.11.015.