Scene Text Detection with Quadtree-based Candidate Text Regions and Convolutional Neural Network

Graham Desmon Simanjuntak[1] and Hertog Nugroho[2]

[1]School of Computing, Telkom University, Bandung, Indonesia
[2]Electric Engineering Department
Bandung State of Polytechnic, Bandung, Indonesia
hertog@polban.ac.id

Abstract: Scene text detection has been developing in recent years. It is due to its numerous practical applications, such as automatic text translation, assist visually impaired people, content-based image retrieval and so on. In this work, we propose a quadtree-based candidate text regions extraction (CTR) to localize texts in the scene image. The CTR takes the advantage of color consistency of text. The result of CTR is a CTR map, extracted by dividing the image into homogeneous regions with quadtree and performing a dilation operation. To perform text localization and classification, a convolutional neural network (CNN) is adopted. The final result is taken from the bounding box of CNN which is compared with the CTR map. The experiments on MSRA-TD500 and ICDAR 2013 show that the proposed method outperforms the previous ones, and it achieves a competitive result on ICDAR 2015. The proposed method achieves F-score of 78.69%, 91.59%, 82.15% on MSRA-TD500, ICDAR 2013, and ICDAR 2015 datasets respectively.

Keywords: Scene text detection; Quadtree; Convolutional neural network

1. Introduction

Scene text detection is a system that aims to find and localize text regions from a natural scene image. Scene text detection has been developing in recent years. It is due to its numerous practical applications in textual image understanding. An effective scene text detection can improve the performance of several multimedia applications such as automatic language translation, content-based image retrieval, robot navigation, autonomous driving, assisting visually impaired people, etc.

Detecting text in natural scene images are highly challenging and more complex than text in documents. In the past years, researchers have proposed numerous methods to perform scene text detection. Epshtein et al. [1] proposed a method to detect character candidates by finding the value of stroke width for each image pixel. However, it may produce false positives when the object resembles letters. Turki et al. [2] proposed a method to detect character candidates by exploiting Maximally Stable Extremal Regions (MSER) [3]. However, MSER may sensitive to blur and strong illumination. Wang et al. [4] proposed superpixel segmentation and hierarchical clustering to detect character candidates. The proposed superpixel segmentation exploits color and edge information, and then the single-link clustering extracts the character candidates. Zhang et al. [5] proposed a method to detect character candidates by exploiting symmetry (with color and structure) and appearance (using Local Binary Pattern) feature of character groups. However, it may fail to detect text with low contrast or strong illumination, single character, or large space between characters. Zhang et al. [6] proposed a method to detect multi-oriented text by utilizing Fully Convolutional Network (FCN) [7]. Zhou et al. [8] proposed a network called Efficient and Accuracy Scene Text detection (EAST) to perform scene text detection. This was the first work that gained significant balanced within efficiency and accuracy. Liu et al. [9] proposed an end-to-end network called Fast Oriented Text Spotting (FOTS) which covered both detection and recognition task. The text detector adopted FCN model and NMS to generate detection results. This model outperforms previous state-of-the-art methods. However, the model may result false prediction against text-like pattern or failed to detect text with very similar color with its background.

Received: February 22nd, 2020. Accepted: March 22nd, 2021
DOI: 10.15676/ijeei.2021.13.1.8

Recently, most methods of scene text detection are built upon deep learning models [10]. The methods have obtained promising results, and able to deal with various challenging scenarios like long text, multi-oriented text, multilingual text, and multi-scale text. However, the performance of deep learning model or machine learning model mainly depends on the training data. The model could achieve a perfect prediction when the input is the same or similar with the data which its trained on, yet output false prediction (either false positive or false negative) when the input is completely new which never trained on. The false positive prediction can be reduced by filtering the prediction with a non-learning method, such as connected component based methods. The most popular methods for this approach are Maximally Stable Extremal Regions (MSER) [11] and Stroke Width transform (SWT) [1]. However, these methods are sensitive to low contrast and small text.

In this work, we propose a method to extract the text candidates. To address the low contrast problem, we adopted a contrast enhancement technique which can improve the image contrast. Furthermore, to address the small text problem, we observe that text has nearly similar color. Since text has nearly similar color, despite the text size variation we can distinguish text and the background by its color. Taking advantage of this definition, we propose a novel method for extracting text candidates.

The pipeline of the proposed scene text detection is as follows. Firstly, a contrast enhancement is performed to adjust the contrast. Then, candidate text regions are extracted by dividing the image into homogeneous regions with quadtree and perform a dilation operation. This process is called candidate text regions (CTR) extraction. The result of CTR extraction is a CTR map. To perform a localization and classification, a Convolutional Neural Network (CNN) is adopted. The model prediction is a bounding box which covered the text location. Finally, the bounding box is evaluated agains the CTR map to reduce false positive prediction.

In summary, our main contribution is the CTR extraction process. Based on experiments, we show that the proposed method improves the detection performance by reducing the false positive prediction. The rest of the paper is organized as follows. Section 2 describes the proposed method. Section 3 presents the experimental results and discussions. Finally, section 4 offers the conclusion remarks.

2. Research Method

The proposed design method consisted of 4 main steps: preprocessing, candidate text regions extraction (CTR), detection network, and bounding box filtering. The proposed scene text detection method is illustrated in Figure 1.

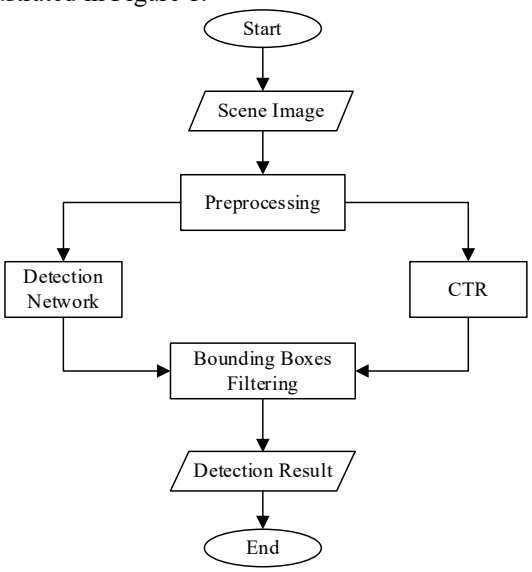


Figure 1. Flowchart of the proposed scene text detection.

A. Preprocessing

The scene image is resized to a fixed size to accelerate the performance of extraction process. After the image is resized to a new size, the image is converted to L*a*b* color space. Then, a contrast enhancement is performed by employing Contrast Limited Adaptive Histogram Equalization (CLAHE). The color conversion process (RGB to L*a*b*) is performed to satisfy the CLAHE, since CLAHE involves the image intensity and not the color components. Furthermore, L*a*b* is more perceptually-uniform than other color spaces. Therefore, the CLAHE is performed on L* (lightness) space instead of each space of RGB. Finally, the contrast enhanced image is converted back to RGB color space as the result of the preprocessing and will be used for CTR and detection network.

B. Candidate Text Regions Extraction (CTR)

We propose a candidate text regions extraction by implementing quadtree and dilation process. The quadtree is employed for segmenting image by dividing it into homogeneous regions. If the image region is homogeneous, then a single leaf node is created. Otherwise, the image is divided into four quadrant regions: (North West (NW), North East (NE), South West (SW), and South East (SE). This process is repeatedly performed until the homogeneous region is obtained. A region is labeled as homogeneous if all pixels inside the region satisfy a threshold value of standard deviation for each color space. Then, after the homogeneous region is obtained, a dilation operation is performed, and finally the output is a CTR map. The algorithm is summarized in Algorithm 1.

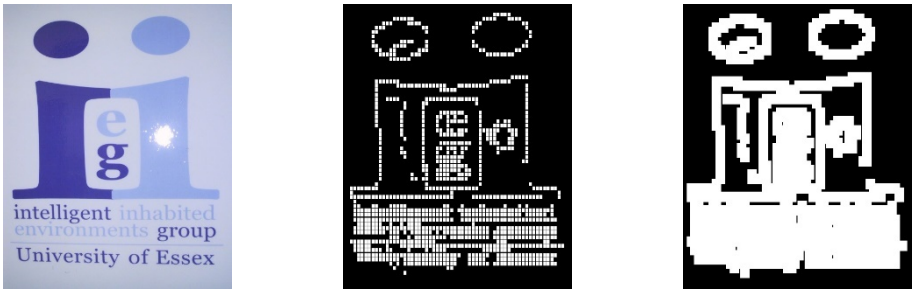
Algorithm 1: Candidate text regions extraction algorithm

Input: Image (RGB) *I*, homogeneous threshold *t*, dilation kernel *k*, minimum region size threshold *ts*

Output: CTR map *C*

1: Initialize list *homogen* and *heterogen*
2: Insert *I* to *heterogen*
3: **while** *heterogen is not empty* do
4: Let the first element of *heterogen* as *node*
5: Calculate standard deviation of *node* for each color space *stdr*, *stdg*, *stdb*
6: **if** standard deviation > *t* & *node* region size > *ts* **then**
7: Divide node into four quadrants *NW*, *NE*, *SW*, *SE*
8: Insert *NW*, *NE*, *SW*, *SE* to *heterogen*
9: **else**
10: Insert *node* to *homogen*
12: **end**
13: **end**
14: Draw a binary image of regions in *homogen* node(s) as *B*
15: Run dilation operation on *B* with kernel *k* as *C*

Example illustration of the CTR is depicted in Figure 2. The CTR map is represented in binary image, 1 (white) for text region and 0 (black) for non-text region.



(a) Input Image (b) Homogeneous Regions (c) CTR Map

Figure 2. Example illustration of the CTR.

C. Detection Network

Detection network is designed to perform classification and localization. In this work the network architecture is adopted from an Efficient and Accurate Scene Text Detector (EAST) [8]. The network architecture consists of three parts: feature extractor, feature-merging, and output layer. The feature extractor is a VGG16-BN. The feature-merging gradually merge the feature maps and produce the final feature map and fed it to the output layer. The final output layer produce score map and quadrangle (QUAD) geometry map. To calculate the loss, EAST loss function is adopted. Finally, the result of the network (QUAD geometry) merged with a Non-Maximum Suppression (NMS) which is also adopted from EAST. The result of the network is bounding boxes (quadrilateral) which represent the text location. The network is depicted in Figure 3.

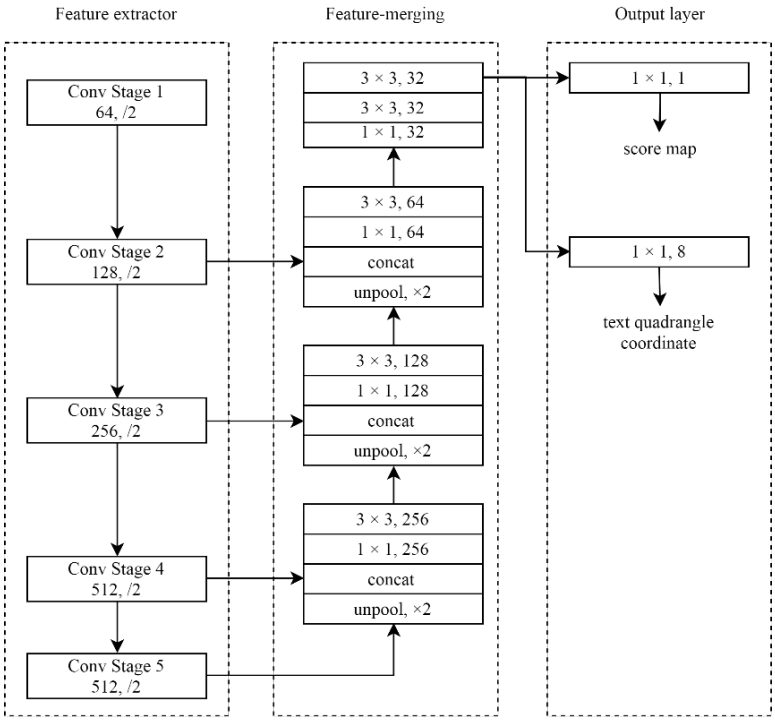
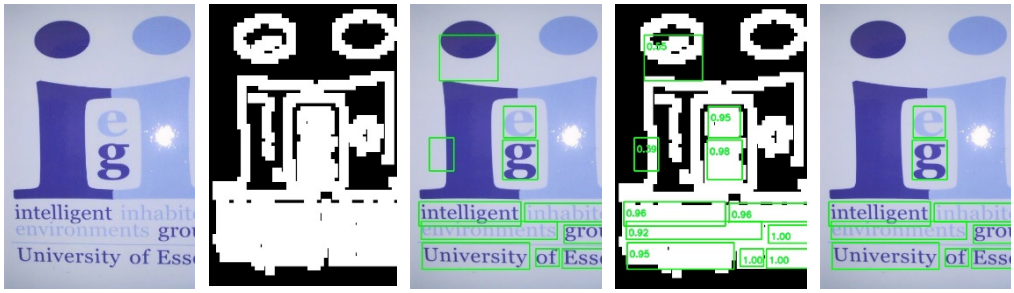


Figure 3. Schematic diagram of detection network [8].

D. Bounding Box Filtering

The result of detection network is bounding boxes. We propose bounding box filtering by evaluating the result obtained from detection network against the CTR map (binary image) information. Each bounding box is evaluated by the corresponding score on the CTR map

(illustrated in Figure 4). A bounding box is labeled as detection result when its score s is more than a specified threshold value.



(a) (b) (c) (d) (e)

Figure 4. Example illustration of Bounding Box Filtering: (a) input image; (b) CTR map; (c) Detection network result; (e) Bounding box score; (e) Detection result

The score is defined as

$$s = \frac{\sum_{i=1}^{n} P_i}{n} \quad (1)$$

where, n is the number of pixels in part of the CTR map which locations are within the area of corresponding bounding box, and p_i is the pixel value.

The detailed process of bounding box filtering is defined in Algorithm 2.

Algorithm 2: Bounding box filtering algorithm

Input: Bounding boxes *B*, CTR map *M*

Output: Detection Result *final*

```
Initialize list final
Let i the first position of B
while i is not the last position of B do
    Let b the bounding box at position i
    Calculate the score s of b based on M
    if s > 0.9 then
        Insert b to final
    end
end
```

E. Bounding Box Merging

The detection network is designed and trained on word based text. The ICDAR 2013 and ICDAR 2015 datasets ground truth is based on words which is suitable for the network. However, the MSRA-TD500 ground truth is a sequence of words. Therefore, for the MSRA-TD500 a post-processing is performed to merge the detection network result as a long text. The merge process is performed by measuring $\boldsymbol{L^2}$ distance (euclidean distance) and angle difference of the rectangle to its neighbors. The angle difference α_{PN} is defined as

$$\alpha_{PN} = |a_P - a_N| \quad (2)$$

where, P is the current rectangle, N is the neighbor, a_P is the angle value of P and a_N is the angle value of N.

To measure the distance, let $C = \{x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4\}$ as the coordinates of the rectangle, where $\{x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4\}$ is the top left, top right, bottom right, and bottom left coordinate of the rectangle respectively. Supposed P is the current rectangle and N is the neighbor, then the distance D of a P to N is defined as

$$l_1 = \frac{x_2+x_3}{2} - \frac{x_1+x_4}{2}$$
$$l_2 = \frac{y_3+y_4}{2} - \frac{y_1+y_2}{2} \quad (3)$$

$$P_{C1} = \begin{cases} (\frac{x_1+x_4}{2}, \frac{y_1+y_4}{2}), & \text{if } l_1 > l_2 \\ (\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}), & \text{otherwise} \end{cases}$$
$$P_{C2} = \begin{cases} (\frac{x_2+x_3}{2}, \frac{y_2+y_3}{2}), & \text{if } l_1 > l_2 \\ (\frac{x_3+x_4}{2}, \frac{y_3+y_4}{2}), & \text{otherwise} \end{cases} \quad (4)$$

$$N_{C1} = \begin{cases} (\frac{x_1+x_4}{2}, \frac{y_1+y_4}{2}), & \text{if } l_1 > l_2 \\ (\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}), & \text{otherwise} \end{cases}$$
$$N_{C2} = \begin{cases} (\frac{x_2+x_3}{2}, \frac{y_2+y_3}{2}), & \text{if } l_1 > l_2 \\ (\frac{x_3+x_4}{2}, \frac{y_3+y_4}{2}), & \text{otherwise} \end{cases} \quad (5)$$

$$D_1 = \sqrt{(N_{C2_x} - P_{C1_x})^2 + (N_{C2_y} - P_{C1_y})^2}$$
$$D_2 = \sqrt{(P_{C2_x} - N_{C1_x})^2 + (P_{C2_y} - N_{C1_y})^2} \quad (6)$$

$$D = \begin{cases} D_1, & \text{if } P_{C1_x} > N_{C2_x} \\ D_2, & \text{otherwise} \end{cases} \quad (7)$$

where, P_{C1_x} is the x coordinate value of P_{C1}, P_{C1_y} is the y coordinate value of P_{C1}, P_{C2_x} is the x coordinate value of P_{C2}, P_{C2_y} is the y coordinate value of P_{C2}, N_{C1_x} is the x coordinate value of N_{C1}, N_{C1_y} is the y coordinate value of N_{C1}, N_{C2_x} is the x coordinate value of N_{C2}, and N_{C2_y} is the y coordinate value of N_{C2}.

A rectangle will be merged with its neighbor if the distance D is less than a specified threshold value and the angle difference α is less than a specified threshold value.

3. Results and Discussion

We evaluate the proposed method on three public datasets: MSRA-TD500 ICDAR 2013, and ICDAR 2015 using the standard evaluation protocol of each dataset, and compare with the previous works.

A. Datasets

MSRA-TD500 [12] was released in 2012 for scene text detection task. The dataset contains 300 training images and 200 testing images with complex background and large variation of text pattern (fonts, colors, sizes, orientations, languages). ICDAR 2013 [13] was released in 2013 for scene text detection and recognition task. The dataset contains 229 training images and 233 testing images in different resolutions, and mostly horizontal text (some text slightly oriented). ICDAR 2015 [14] was released in 2015 for scene text detection and recognition task. The dataset contains 1000 training images and 500 testing images with varying orientation, scale, and resolution.

B. Implementation Details

Our method is implemented using OpenCV [16] for basic image processing library and PyTorch [17] for developing the model. The experiments are conducted on a workstation (GPU: GeForce GTX 1080 Ti; RAM: 12 GB; CPU: Intel Core i7-37770 CPU @ 3.40GHz×8) and a regular laptop (CPU: AMD Ryzen 5 2500u; RAM: 12GB). The proposed method is performed with following parameters:

- **Preprocessing**. If the input image width > 512, we resize the width to 512 and the height following the aspect ratio.
- **CTR**. We set the homogeneous threshold to 10, minimum region size threshold to 49, and dilation kernel size to 15×15.
- **Bounding boxes merging**. We perform bounding boxes merging only on MSRA-TD500 dataset. We set the distance threshold to 80.
- **Detection Network Training**. The model is pre-trained on SynthText, then finetuned on MSRA-TD500, ICDAR 2013, and ICDAR 2015 datasets until convergence. The model is optimized using ADAM [15] with the learning rate set to $1e^{-5}$. The image is resized to 512×512 and batch size set to 24.

C. Evaluation of Bounding Box Merging

The performance of bounding boxes merging is shown in Table 1.

Table 1. Comparison with and without proposed bounding box merging on MSRA TD-500 with percentage scores.

Method	Precision	Recall	F-score
without merging	58.06	76.66	66.07
with merging	**75.54**	**82.11**	**78.69**

Table 1 demonstrates that the proposed bounding box merging improve the system performance on MSRA-TD 500. The system achieves 78.69% in F-Score, which 19.1% higher than without using the merging process.

(a) without merging (b) with merging

Figure 5. Example result of bounding box result.

Figure 5 illustrates example of bounding boxes merging on MSRA-TD500 dataset. It demonstrates that the proposed bounding box merging are able to merge text in horizontal or vertical position. However, the proposed bounding boxes merging process is limited to curved

text, due to its angle difference similarity. Figure 6 illustrates example of bounding boxes merging on curved text.

(a) without merging (b) with merging (c) Ground truth

Figure 6. Example illustration of the proposed bounding boxes merging limitation on curved text.

D. Evaluation of Candidate Text Regions Extraction (CTR)

The performance of scene text detection with and without the proposed preprocessing and CTR are shown in Table 2. The proposed preprocessing and CTR improve the precision which improved the F-score. The proposed preprocessing and CTR improve the precision which improved the F-score. The improvement of precision infers that the proposed preprocessing and CTR can reduce false positive prediction.

Table 2. Comparison with and without preprocessing and CTR with percentage scores. "PCTR", "P", "R", "F" stand for "Preprocessing+CTR", "Precision", "Recall", "F-score" respectively.

Method	MSRA TD-500			ICDAR 2013			ICDAR 2015		
	P	R	F	P	R	F	P	R	F
Detection Network	73.71	82.11	77.73	92.77	88.73	90.74	80.15	79.44	79.79
Detection Network + PCTR	**75.54**	82.11	**78.69**	**94.56**	88.73	**91.59**	**85.05**	79.44	**82.15**

Table 2 demonstrates that the proposed preprocessing and CTR improve the system performance on MSRA-TD500, ICDAR 2013, and ICDAR 15 datasets. On MSRA-TD500 the proposed achieves 78.69% in F-score, which is 1.24% higher than only using the detection network. Then, on I ICDAR 2013 the proposed achieves 91.59% in F-score, which is 0.94% higher than only using the detection network. Finally, on ICDAR 2015 the proposed achieves 82.15% in F-score, which is 2.96% higher than only using the detection network. Figure 6 illustrates example of detection network output bounding box and the bounding boxes filtering result.

E. Evaluation of Detection Performance

The comparison performance of the proposed scene text detection system and the previous works is shown in Table 3.

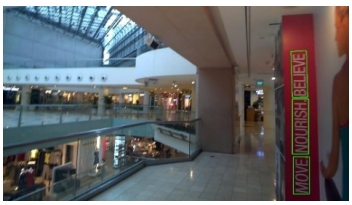
Table 3. Comparison with other results with percentage scores "P", "R", "F" stand for "Precision", "Recall", "F-score" respectively.

Method	MSRA TD-500			ICDAR 2013			ICDAR 2015		
	P	R	F	P	R	F	P	R	F
EAST [8]	**87.28**	67.43	76.08	-	-	-	83.27	78.33	80.72
FOTS [9]	-	-	-	-	-	87.32	**88.84**	**82.04**	**85.31**
EAST [8]	**87.28**	67.43	76.08	-	-	-	83.27	78.33	80.72
Ch'ng et al. [18]	-	-	-	90	83	88	80	66	73
Turki et al. [2]	72	79	75.33	87.25	84.15	85.87	52	49	50.45
Wang et al. [4]	-	-	-	81.86	87.43	84.56	-	-	-
Zhang et al. [6]	83	67	74	88	78	83	71	43	54
Zhang et al. [5]	-	-	-	88	74	80	-	-	-
Proposed	76.64	**82.11**	**78.69**	**94.65**	**88.73**	**91.59**	85.05	79.44	82.15

As shown in Table 3, on MSRA-TD500 the proposed method achieves competitive performance. The proposed method achieves 78.69% in F-score, which is 3.43% higher than the second best. Then, on ICDAR 2013 the proposed method outperforms the previous works performance. The proposed method achieves 91.59% in F-score, which is 4.89% higher than the second best. Finally, on ICDAR 2015 the proposed method achieves competitive performance. FOTS achieves the best F-score, which is 3.85% higher than the second best (the proposed method). Figure 7 illustrates example result of the proposed scene text detection system on MSRA-TD500, ICDAR 2013, and ICDAR 2015 datasets. It also demonstrates that the proposed method able to detect text with small size, low contrast, multi-oriented, multilingual, and multi-scale.

(a) Non-latin and diagonal orientation

(b) Latin, small size, low contrast, and horizontal orientation



(c) latin and vertical orientation

Figure 7. Example of detection result.

4. Conclusion

In this work, we propose a scene text detection system which is designed to cover small text, low contrast, multi-oriented, multilingual, and multi-scale problems in scene text detection topic. The pipeline consists of preprocessing, candidate text regions extraction (CTR), detection network, and bounding boxes filtering. The experiments on MSRA-TD500, ICDAR 2013, and ICDAR 2015 demonstrate that the proposed method outperforms the previous work on MSRA-TD500 and ICDAR 2013, and achieves competitive result on ICDAR 2015.

5. References

[1]. B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp.2963–2970.

[2]. H. Turki, M. Ben Halima, and A. M. Alimi, "Text detection based onmser and cnn features," in *201714th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 01, Nov 2017, pp.949–954.

[3]. J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and Vision Computing*, vol. 22, no. 10, pp. 761–767, 2004.

[4]. C. Wang, F. Yin, and C. Liu, "Scene text detection with novel superpixel based character candidate extraction," in 2017 *14th IAPR International Conference on Document Analysis and Recognition (ICDAR),* vol. 01, Nov 2017, pp. 929–934.

[5]. Z. Zhang, W. Shen, C. Yao, and X. Bai, "Symmetry-based text line detection in natural scenes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[6]. Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4159–4167.

[7]. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[8]. X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: An efficient and accurate scene text detector," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[9]. X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan, "Fots: Fastoriented text spotting with a unified network," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[10]. S. Long, X. He, and C. Ya, "Scene text detection and recognition: The deep learning era," *arXiv preprint arXiv:1811.04256*, 2018.

[11]. L. Neumann and J. Matas, "A method for text localization and recognition in real-world images," *in Proceedings of the 10th Asian Conference on Computer Vision - Volume Part III*, ser. ACCV'10.Berlin, Heidelberg: Springer-Verlag, 2011, pp. 770–783.

[12]. C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 1083–1090.

[13]. D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i. Bigorda, S. R.Mestre, J. Mas, D. F. Mota, J. A. Almaz`an, and L. P. de las Heras, "Icdar 2013 robust reading competition," in *2013 12th International Conference on Document Analysis and Recognition*, Aug 2013, pp. 1484–1493.

[14]. D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny, "Icdar 2015 competition on robust reading," in *Proceedings of the 2015 13th International Conferenceon Document Analysis and Recognition (ICDAR)*, ser. ICDAR '15.Washington, DC, USA: IEEE Computer Society, 2015, pp. 1156–1160.

[15]. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[16]. G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[17]. A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS Autodiff Workshop*, 2017.

[18]. C.-K. Ch'ng, C. S. Chan, and C.-L. Liu, "Total-text: toward orientation robustness in scene text detection," International Journal on Document Analysis and Recognition (IJDAR), vol. 23, pp. 31–52, 2019.

Graham Desmon Simanjuntak received B.S. degree in informatics engineering from Telkom University, Indonesia, in 2018, and M.S. degree in informatics from Telkom University, Indonesia, in 2020. His research interests include computer vision, pattern recognition, and image processing.

Hertog Nugroho received B.S. degree in Electrical Engineering from Bandung Institute of Technology, Indonesia, in 1984, M.Sc degree and Ph.D degree in Electrical Engineering from Keio University, Yokohama. Japan, in 1995 and 1999 respectively. He is now working as Associate Professor in Bandung State of Polytechnic. His research interests include computer vision, pattern recognition, and signal processing.