OPTICAL CHARACTER RECOGNITION (OCR) FOR TEXT RECOGNITION AND ITS POST-PROCESSING METHOD: A LITERATURE REVIEW

Ridvy Avyodri
Faculty of Informations and Technology
Pelita Harapan University
Jakarta, Indonesia
01679210007@student.uph.edu

Samuel Lukas
Faculty of Informations and Technology
Pelita Harapan University
Jakarta, Indonesia
samuel.lukas@uph.edu

Hendra Tjahyadi
Faculty of Informations and Technology
Pelita Harapan University
Jakarta, Indonesia
hendra.tjahyadi@uph.edu

2022 1st International Conference on Technology Innovation and Its Applications (ICTIIA) | 978-1-6654-8826-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICTIIA54654.2022.9935961

***Abstract*— Most organizations worldwide still rely on paper-based documents. Usage of paper-based documents gives a hard time extracting the data required from those documents. This heavy paper usage also damages the efficiency in cost and time, not to mention the impact on the environment caused by deforestation to produce these papers. These are some reasons that motivate the need to digitalize paper-based documents. To convert the usage of paper-based documents into paperless documents cannot be done in an instant. In its transition, these paper-based documents are usually scanned into image format to reduce the usage of paper. From this comes a need for technology that is able to recognize and extract data in the scanned image of paper-based documents. Optical Character Recognition makes it possible to do text recognition appearing in images. However, despite its long history of development, OCR for text recognition has yet to achieve 100% accuracy. In general, OCR process will be divided into Image Pre-processing, Text Segmentation/Localization, Feature Extraction, Text Recognition, and Post-Processing. Thus, this research will review OCR-related works and the methods used within this framework to support further research.**

Keywords—Optical Character Recognition, Text Recognition, Text Segmentation, Pre-processing, Post-processing

I. INTRODUCTION

Digitalization is currently pursued to be implemented in every sector. Whether in the business sector, educational institution, government, or even medical institution, many of them are still relying on a paper copy of documents. This called for inefficiency in cost or longer processing time. Heavy paper usage also raised an issue regarding environment-friendly business, since heavy paper usage will lead to heavy deforestation. These are some reasons why digitalization become a more urgent topic with every passing day. To convert these worldwide industries with many parties involved from heavy paper consumption into a paperless process is unable to be done in an instant. There may also be some process that must be done on paper-based. Therefore, comes a need for a system that is able to push data in a paper copy of a document into a digitalized system. This is not only mean to convert paper-based document into digital ones, but also to recognize text data appearing in that document so those data can be processed inside digital system.

Optical Character Recognition is part of artificial intelligence with ability to read and understand object(s) appearing in image. This object can be a face as in face recognition tools or alphanumerical text as in text recognition tools. This technology made it possible to extract required data from paper document that has been scanned or captured into image format(s) such as .jpg, .jpeg, .png etc. There are two type of input image that mainly used for text recognition research. One is document image containing text on paper and the other is scene image containing text on environment. Image with text on paper usually have a structured form (in paragraph or text blocks) while the latter mostly containing text in variety of location, alignment and size thus makes them referred as “text in the wild” images[1]. Further, there are also 2 types of text appearing in image. It can be a printed text or handwritten text. Printed text is easier to be recognized than handwritten text. It is because every handwritten text is different depends on the writer. Other than that, handwritten text usually shows inconsistent spacing between words or characters. Thus, make it harder for OCR tools to recognize the characters. Text data that is needed to be digitalized mostly appeared in image containing text on paper type. While whether text appeared as handwritten or printed will depend on each sector’s practical use.

Development of Optical Character Recognition has been underway for a long time. Dated back to 1929 when first patent for OCR was obtained by Tausheck in Germany and followed by Handel in USA on 1933[2]. What was used in Tausheck method was optical and mechanical template matching. A mechanical mask is passed by light which then captured by a photodetector and scanned mechanically. This will make light fails to reach the photodetector when an exact match happens. Hence, the machine recognizes the characters appeared on paper[2]. Research continues until ground-breaking implementation of commercial OCR machine first attempted by IBM in 1965. Named as ‘IBM 1287’ this machine capable of reading handwritten numbers[2]. Since then, many methods have been explored in order to improve the accuracy of OCR. From the earliest method using binarization and histogram projection[3] until recently OCR has implemented Machine learning algorithm such as K-Nearest Neighbor, Decision Tree, Support Vector Machine etc. [4]–[6]. Alongside with the development that improve computational performance, Deep Learning method also implemented in OCR technique such as Long Short-Term Memory and Convolutional Neural Network [7][8].

Despite its long history of research and development, an OCR system that reach 100% in accuracy has yet to be found. This is due to challenges in OCR implementation. Abdulwahhab *et al.* in [9] summarize challenges in OCR implementation as scene complexity, condition of uneven lighting, skewness, blurring and degradation, aspect ratios, tilting, fonts, multilingual environments and image warping. To overcome these challenges researchers have experimented various method in pre-processing the input image. Other than that, some methods of post-processing the OCR output also researched in order to increase its accuracy.

II. REVIEW METHOD

This research presents a literature review by identifying and comparing previous studies that relevant to the OCR pipeline for text recognition. Objective of this research is to summarize current techniques and results that has been implemented in previous research regarding OCR for text recognition.

According to Snyder H [10] there are 3 different approaches to conduct a literature review. They are systematic literature review, semi-systematic literature review and integrative review. Systematic literature review defined as a research method to collect, identify, analyze and appraise relevant research that fit with the pre-specified inclusion criteria for fulfilling the research's objective. Systematic literature review considered as the most rigorous and accurate method to collect articles. Thus, systematic literature review is a suitable research method to reach the objective of this article which is to summarize techniques and results of previous research.

In order to find relevant resources for this research, A search strategy is implemented based on inclusion/exclusion criteria as below:

1. The papers provide a clear knowledge regarding OCR process, specifically OCR for text recognition
2. The papers research one or more OCR steps (image pre-processing, text segmentation, feature extraction, text recognition, post-processing)

These inclusion criteria were chosen to specifically find research regarding OCR for text recognition and its process. Search keywords are in accordance with this framework. Resources found through online searching on IEEE, Elsevier and Google Scholar. Other than that, resources also found from citation of previously selected research. This is to understand the concept of those researches better. Selection of resources based on the topic that must be related to any step of OCR pipeline. Selection of resources also prioritizes newer research in 10-year period. However, older research can also be referenced if the topic is strongly relevant to the OCR pipeline. From 100+ papers found, 119 papers selected from their title and abstract were partially read for relevance checking and 29 papers were fully read and referenced in this research. Distribution of reference's resource and related topic that has been selected as this research's reference can be seen in Table 1. Difference in total references caused by some reference discuss more than 1 OCR steps. Other related topic in table 1 are supporting papers that build understanding of this research related to research method, history and steps of OCR.

III. IMAGE PRE-PROCESSING

OCR for text processing has some challenges affecting its accuracy. These challenges described by Abdulwahhab *et al.* in [9] can be mitigated mostly by pre-processing the image. A scanned document may not show the text in perfect horizontal line, thus rotating the image is required. Archana *et al.* in [3] pre-process the image for de-skewing input image. De-skewing is done based on image spectrum which shows original text line orientation and rotated to form a horizontal line. Further pre-process is text segmentation using histogram projection. Histogram projection will separate lines horizontally and separate words or characters vertically.

Hough Transform Method is also one popular method for skewness and tilting correction. Hough Transform was first introduced in 1962 by Paul Hough, and first invented by R. Duda and P. E. Hart on 1972[11]. This method basically acts as a feature extraction to detect the line and shapes of objects in input image. Kumar Shukla in [11] research skew and tilting correction method using Hough Transform. Research is done with skew rotation less than 90 degrees and resulting in correct skew angle. However, total datasets used are not specified in this research.

Many research has been done in implementation of OCR on low quality image. Brisinello *et al.* in [12] proposed a set of pre-processing steps to increase the accuracy of OCR. OCR engine used in this experiment is Tesseract OCR v.3.5 and v.4.0. Pre-processing steps that have been done in this research are image resizing, image sharpening, image blurring and K-means clustering. Image resizing done due to limitation of Tesseract OCR engine that will not detect text with height under 20 pixels. Image resizing only done to characters that is lower than 100 pixels and resized to 100 pixels. Image sharpening done to enhance contrast between edges using gaussian low pass filter. This will also help to segment the text from the background. Image blurring done using low pass filter. Blurring method done here not to cancel the sharpening method from previous steps but using low filter to remove noise that often recognized as diacritic character. Last, K-means clustering is applied for image with colorful background. This research resulting in increase of accuracy for 33.3% for Tesseract 3.5 and 22.6% for Tesseract 4.0.

TABLE 1. DISTRIBUTION OF REFERENCE'S RESOURCE AND TOPIC

	IEEE	ELSEVIER	GSCHOLAR
PRE-PROCESS	2	3	3
TEXT SEGMENTATION	3	5	2

FEATURE EXTRACTION	0	3	1
TEXT RECOGNITION	3	4	1
POST PROCESSING	2	0	3
OTHER	1	1	1
TOTAL PAPER	**10**	**10**	**9**

Accuracy of text recognition also influenced by appearance of tables and boxes in document. Therefore, Gatos *et al.* in [13] proposed a method to detect tables and boxes in document without the need to train the model for table detection. Method used by author consist of pre-processing the image, line detection using horizontal and vertical black runs processing, line enhancement and table detection based on line intersections.

IV. TEXT LOCALIZATION/ SEGMENTATION

Text in image may appear in brief or long texts, also in variety of sizes. This is the example of variety in aspect ratios. To overcome the challenge of this scenario best text segmentation and localization method is required. Tesseract OCR as an OCR engine include pre-processing of image in its module which consist of adaptive thresholding to return the output as binarize image, then connected component analysis to extract character outlines[14]. Further details of this process are line finding, baseline fitting, fixed pitch detection and chopping and proportional word finding[15]. Sporici *et al.* in [16] researched the flaws in Tesseract OCR engine and how to improve them. It is then concluded that the weakness of Tesseract OCR engine is not within its character classifier itself, but on the segmentation procedure that caused Tesseract OCR to ignore some sections of text[16]. Thus, additional text segmentation may be required in Tesseract OCR with pre-installed image pre-processing and segmentation procedure.

Kumar A. in [17] presented text extraction method based on edge detection using Gabor filter and text localization using heuristic filtering. First, using two-dimensional Gabor filter as proposed by Daugman [18] to provide the optimal resolution for both the orientation and the spatial frequency of a local image region. This will help to find out edges in images. Then the contrast of conversion result from Gabor filtering is increased by sharpening filter and further passed through Otsu's threshold to obtain binary image[17]. Binary image obtained then dilated to find the non-text long edge which will be removed by connected component labelling operator. Finally heuristic filtering supports to find out non-text region by using major to minor axis ratio[17].

Bukhari *et al.* [19] proposed a classification method between text and non-text region in image using discriminative learning of connected component. Classification process start with feature extraction. Instead of extracting complex feature of connected components, raw shape of connected components is extracted along with the surrounding area of connected component. Text component size in images mostly smaller than non-text component and structure of non-text component is irregular compare to text component. Based on these characteristics, shape-based component feature is extracted. Text component mostly aligned horizontally. Based on this characteristic, context-based feature is extracted containing connected component and its surrounding image downscaled to 40x40 pixel window size. Then, these features further processed through Multilayer Perceptron classifier to classify between text and non-text region.

Inspired by Bukhari *et al.* works in [19], research done by Aguilar *et al.* [20] to compare methods used in classification of text and non-text in image. Started with Otsu's threshold binarizing the input image, then extracting features as in [19] obtaining shape-based and context-based feature. Difference made by Aguilar *et al.* in [20] that these 2 features processed through Convolutional Neural Network while adding 19 additional geometric features that is processed through Multilayer Perceptron separately to decide the best model. It is then resulting in Convolutional Neural Network even when trained using only component images alone presented better result than Multilayer Perceptron using 19 manually extracted features. The result from Bukhari method in [19] was compared with Leptonica page segmentation algorithm. Results was Bukhari method provide slightly better segmentation result for text-region but provide a significant better result for non-text region such as drawing components.

Lukas *et al.* in [21] implement adaptive thresholding and region labeling to detect text area in Vehicle Registration Plate (VRP) recognition. In adaptive threshold method, threshold for each pixel is calculated based on value mean of that pixel region. This threshold will separate foreground and background value. Author implements integral image for adaptive thresholding to reduce computational cost of this method. Region labeling is connected component-based labeling. This method label areas on image with similar value of connected pixel. In this paper, connected pixel checked is in North and West direction of current pixel because labeling process start from left to right and top to bottom. Results of this labeling process then feed through vertical and horizontal image projection to determine candidate VRP area and to segment each character. Experiments with images taken from 3 and 4.5 meters distance to VRP resulted in more than 95% of accuracy in segmentation of VRP area.

Matas *et al.* [22] presented a robust method of image segmentation known as maximally stable extremal region (MSER). Instead of defining the average threshold to binarize image, MSER process all threshold value from 0 to 255 and inverted its value. When t is current threshold, then value below t will be converted to white (0) and value above t will be converted to black (1). Inverted value is when white is 1 and black is 0. MSER will process the image with t value from 0 to 255, respectively. This will result in some area with stable value throughout the process. Therefore, the output of this method will not be a binarized image, but a system of nested subsets of stable threshold area.

Aladhadh *et al.* in [23] combine MSER method as proposed by Matas *et al.* [22] with stroke width transformation to eliminate the non-text characters. Further using Recurrent

Convolutional Neural Network and Long Short-Term Memory for text recognition in payable documents. Result of the experiment was an accuracy of 95% on a test set of 20,000 payable images when used without lexicon, and 99% accuracy while using the lexicon-based approach[23].

V. FEATURE EXTRACTION

Phangtriastu *et al.* in [6], combined and compared 3 methods of feature extraction. Zoning, Projection profile and Histogram of Oriented Gradient (HOG). Zoning is one of popular method for feature extraction in image. Each image will be divided into certain zone, in that paper images are divided to 16 zones with 4x4 pixel size. These zones then calculated for its average pixel density value. Results are 16 values as a feature. Khan *et al.* in [5] also implement this feature extraction method. Projection profile counts the number of black pixels in each row of image for horizontal projection and each column for vertical projection[3]. Result from these then extracted as a feature. Histogram of Oriented Gradient is another popular method in image segmentation. It works by dividing image to several cells then detecting the changes of gradient in a certain direction of these cells. In this research image used is 128x128 pixels, divided into 4 forming 64x64 pixels size and 9 orientation bins totaling 36 features extracted. This paper also experimented on combination of these 3 methods totaling 7 types of feature extraction methods. They are Zoning, Projection Profile, HOG, Zoning + Projection Profile, Zoning + HOG, Projection profile + HOG and Zoning + Projection Profile + HOG.

Aladhadh *et al.* in [23] proposed MSER-Recurrent Convolutional Neural Network (MRCNN) for image pre-processing and feature extraction. MSER is used as pre-processing and character segmentation. Then convolutional layers will extract a feature sequence which will be used as a construction of recurrent layers producing probabilistic prediction. Then Final layer translates the prediction into sequence labels.

VI. TEXT RECOGNITION

Text segmentation/localization in the previous step only differentiate text and non-text area in images. After text area has been determined then it is processed through text recognition process to convert the image to text data.

Continuing the research by Lukas *et al.* in [21], Yugopuspito *et al.* [24] proposed recognition method using Discrete Cosine Transform (DCT) and Radial Basis Function (RBF) Network. After segmentation process as in [21] then character region image with 35 x 15 pixels in size is obtained. This means there are 525 pixels with each DCT Coefficient as a feature of a character. RBF Network that is used for training and text recognition is based on Multilayer perceptron. In that paper the authors used Gaussian function for hidden nodes and linear function for output nodes. System designed by 36 hidden nodes the same as the output nodes (26 alphabets and 10 Numeric). The result of this experiment was 96% accuracy from 490 character in 64 VRP images.

Phangtriastu *et al.* in [6] compare implementation between Neural Network and Support Vector Machine on OCR and several feature extraction methods. Research done without pre-processing of image due to the datasets that already in high quality images. For feature extraction this research use zoning, projection, and histogram of oriented gradients (HOG) and the combination of those methods (zoning + projection, projection + HOG, zoning + HOG, projection + zoning + HOG). Further the classifier of Neural Network and Support Vector Machine used and compared to achieve the best result. Total images that used in this experiment is 26416 from Chars74K dataset and limited to capitalized characters (A-Z) with each of characters contains 1016 different style of images. The highest accuracy of this experiment came from the utilization of projection profile algorithm and combination of projection and HOG using SVM classifier with the 94.43% of accuracy.

K-Nearest Neighbor mostly used for text recognition in handwritten form. Khan *et al.* in [5] experimented text recognition for handwritten Pashto (Pakistani Letters) using K-Nearest Neighbor and Neural Network as classifier. The authors develop their own dataset that consist of medium size handwritten character of 4488 samples (contains 102 samples for each letter) by collecting handwritten samples from different individuals. Zoning technique is used as a feature extraction method. Results of this experiment are accuracy of 70.07% for KNN and accuracy of 72.00% for NN.

Breuel [7] researched the application of deep learning on OCR in the form of Hybrid Convolutional-LSTM Network. Dataset used are text lines from the University of Washington Database III, split randomly into 95190 text lines representing 4.5 million samples for training, and 1253 text lines representing 60.000 samples for testing. In this research author compare deep convolutional layers and 1D LSTMs without normalization and 1D LSTMs with normalization, resulting both models provide similar performance. However, using convolutional layer with 1D LSTM and additional normalization boost the performance significantly from 0.43% to 0.25% error rates.

VII. POST-PROCESSING

Basic post-processing methods to improve OCR accuracy are spelling check and lexicon-based word. Spelling checks on OCR result using Google's Online Spelling Suggestion was researched by Bassil *et al.* [25]. By making use of Google's massive data, post correction is done through "did you mean?" feature in Google search. Experiment used only using two low quality images each in English and Arabic languages. Result for English images is error word reduced from 27 words out of 126 words in total to only 4 words remain error. As for Arabic images error word reduce from 8 words out of 64 words to only 2 words remain error.

Kazuhiro [26] is one of many who research lexicon-based OCR post processing. Dictionary used contain about 60.000 words and documents to be recognized are in Japanese containing about 1300 samples. Post-processing done through 2 steps where first by extracting keywords from documents and second by using keywords extracted character string then corrected. Strohmaier *et al.* [27] researched lexical-based post-processing using static dictionary and dynamic dictionary in English and German languages. Dynamic dictionary is a web-based dictionary taken from 1000 top scored websites searched using 25 terminological expressions in topics researched. Result is dynamic dictionary perform better and English lexical-based correction shows better results than German language due to more complex composition of words in German language.

Spelling check and lexical-based method sometime unable to perform well in words with certain topic. Zhu *et al.* in [28] gives an example of this word "For instance, given a candidate word of two possible values in "cat" and "car", the classifier will choose "cat" ahead of "car" if the term "mouse" appears within its 5-gram neighbor because mouse is considered highly relevant to cat. However, this may not work in cases such as "how to catch a mouse in the car". To solve this problem the authors then proposed post-processing of OCR using the connection word not only in its text block but also with its neighboring text blocks.

More current research in post-processing of OCR result already implements deep learning mechanism. Karthikeyan *et al.* in [29] proposed a method for medical text data post-correction named as Robustly Optimized Bidirectional Encoder Representations from Transformers (RoBERTa). First, domain-specific words which are not included in English vocabulary are masked. Then RoBERTa used to predict these words. This experiment resulting in good reductions of Word Error Rate (WER) and Character Error Rate (CER) even without training the model with domain specific data.

TABLE 2. METHOD AND RESULT SUMMARY

REF	OCR STEPS AND RESULT		
	DATA	***METHOD***	***RESULT***
[3]	several printed document images	Pre-process: Image spectrum and histogram projection in MATLAB	Able to find no of lines and words
[11]	Not specified	Pre-process: Hough Transform	Able to de-skew image with <90-degree rotation
[12]	84 Images Black and White and Multicolor	Pre-process: Resize, Sharpen, Blurring, K-Means in MATLAB	+33.3% for Tesseract 3.5 +22.6% for Tesseract 4.0
[17]	32 complex images and video frames	Text Segmentation: Gabor Filter and Heuristic Filtering	94.67% precision rate
[19]	95 UW-III images and 8 ICDAR 2009 images	Text Segmentation: Connected component and MLP classifier	97.25% Accuracy
[20]	53 ICDAR 2009 Images	Text Segmentation: Connected component and CNN classifier	98.68% Accuracy
[21]	53 Camera Captured VRP Images	Text Segmentation: Adaptive Thresholding and Region Labeling	95% Accuracy
[6]	Chars74K dataset containing 26416 images	Feature Extraction: Zoning, Projection, HOG and combinations of them Text Recognition: SVM and Neural Network	Highest Accuracy is 94.43% using Projection + HOG and SVM
[5]	4.488 Pashto Handwritten Characters	Feature Extraction: Zoning Text Recognition: KNN and ANN	70.07% for KNN 72% for ANN
[7]	60.000 characters from University of Washington Database III	Pre-processing: Gaussian blurring Text Recognition: 1D LSTM + Convolutional Layer with and without normalization	Most minimum error rate is 0.25% with normalization
[23]	20.000 images of payable docs	Pre-processing and Text segmentation: MSER Feature Extraction: CNN Text Recognition: LSTM Post-processing: Lexicon-based	95% w/o lexicon 99% w/ lexicon
[24]	64 Camera Captured VRP Images consists of 490 characters	Text Segmentation: Adaptive Thresholding and Region Labeling Feature Extraction: DCT Text Recognition: RBF Network	96% Accuracy in character recognition
[25]	126 mispelled English words & 64 mispelled Arabic words	OCR software: Omnipage 17 Post-processing: Google's "did you mean" feature	Leaving only 4 misspelled English words & 8 misspelled Arabic words
[26]	20 printed datasheet in Japanese language containing 13.000 characters	Post-processing: Lexicon-based with keyword information	Only 5 highest doc recognition rates shown with 99.8% being the highest
[28]	patFT, full text patent information containing 500 text documents	Post Processing: Text block classification	+28.37% average improvement
[29]	100 NHS medical report and 211 pages from MiBio datasets	OCR software: Tesseract OCR Post-processing: RoBERTa word prediction	WER and CER reduction for MiBio: 5.709 & 2.07 NHS: 1.39 & 0.83

VIII. DISCUSSION

Along with its long history of development, OCR already implement many methods to improve its accuracy. Pre and post-processing in OCR will be crucial in its accuracy. Pre-processing will help to minimize noise in images while post-processing will help to do correction in OCR results. However, correction in post-processing step may not be able to be implemented in some cases. For example, when there is a need to recognize text along with every typing error shown on the image. Post-processing method will correct this typing error in its result. Some methods also perform better only for specific tasks. [19], [20] showing results that is significantly better to recognize non-text region, but for the text region recognition the improvement is insignificant.

Though 100% accuracy is yet to be achieved, OCR method has produced good results. With some research resulting in over then 90% accuracy[6], [12], [23], [25] OCR can prove useful for supporting digitalization of paper documents. More detailed results and method used can be seen in Table 2. However, data used in every research may vary. That also the case with language used. Therefore, there may be a need to train the model for specific language or specialization to achieve an even better result. Further research can be done with focus on efficiency of time and computational cost of the OCR model without damaging the accuracy. Another opportunity is to research OCR for text recognition with general purpose, i.e. covering many languages and domain-specific criteria.

REFERENCES

[1] Y. Li, W. Jia, C. Shen, and A. van den Hengel, "Characterness: An indicator of text in the wild," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1666–1677, 2014, doi: 10.1109/TIP.2014.2302896.

[2] S. Mori, C. Y. Suen, and K. Yamamoto, "Historical Review of OCR Research and Development," 1992.

[3] A. A. S. #1 and D. G. Chougule, "Text Pre-processing and Text Segmentation for OCR," 2012.

[4] Tapan Kumar Hazra, Dhirendra pratap singh, and Nikunj Daga, *Optical Character Recognition using KNN on Custom Image Dataset*. 2017.

[5] S. Khan, H. Ali, Z. Ullah, N. Minallah, S. Maqsood, and A. Hafeez, "KNN and ANN-based Recognition of Handwritten Pashto Letters using Zoning Features," Apr. 2019, doi: 10.14569/IJACSA.2018.091070.

[6] M. R. Phangtriastu, J. Harefa, and D. F. Tanoto, "Comparison between Neural Network and Support Vector Machine in Optical Character Recognition," in *Procedia Computer Science*, 2017, vol. 116, pp. 351–357. doi: 10.1016/j.procs.2017.10.061.

[7] T. M. Breuel, "High Performance Text Recognition Using a Hybrid Convolutional-LSTM Implementation," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, Jul. 2017, vol. 1, pp. 11–16. doi: 10.1109/ICDAR.2017.12.

[8] T. M. Breuel, A. Ul-Hasan, M. A. Al-Azawi, and F. Shafait, "High-performance OCR for printed english and fraktur using lstm networks," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2013, pp. 683–687. doi: 10.1109/ICDAR.2013.140.

[9] K. Abdulwahhab Hamad and M. Kaya, "A Detailed Analysis of Optical Character Recognition Technology," 2013. [Online]. Available: http://ijamec.atscience.org

[10] H. Snyder, "Literature review as a research methodology: An overview and guidelines," *Journal of Business Research*, vol. 104, pp. 333–339, Nov. 2019, doi: 10.1016/j.jbusres.2019.07.039.

[11] B. Kumar Shukla, G. Kumar, and A. Kumar, "An approach for Skew Detection using Hough Transform," 2016.

[12] M. Brisinello, R. Grbic, M. Pul, and T. Ancelic, "Improving Optical Character Recognition performance for low quality images," *59th International Symposium ELMAR-2017*, 2017.

[13] B. Gatos, D. Danatsas, I. Pratikakis, and S. J. Perantonis, "Automatic Table Detection in Document Images," *International Conference on Pattern Recognition and Image Analysis*, 2005.

[14] C. Patel Smt Chandaben Mohanbhai, A. Patel, S. Chandaben Mohanbhai, and D. Patel Smt Chandaben Mohanbhai, "Optical Character Recognition by Open Source OCR Tool Tesseract: A Case Study," 2012.

[15] R. Smith, "An Overview of the Tesseract OCR Engine," *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, 2007, [Online]. Available: http://code.google.com/p/tesseract-ocr.

[16] D. Sporici, E. Cuşnir, and C. A. Boiangiu, "Improving the accuracy of Tesseract 4.0 OCR engine using convolution-based preprocessing," *Symmetry (Basel)*, vol. 12, no. 5, May 2020, doi: 10.3390/SYM12050715.

[17] A. Kumar, "An Efficient Approach for Text Extraction in Images and Video Frames Using Gabor Filter," *International Journal of Computer and Electrical Engineering*, vol. 6, no. 4, pp. 316–320, 2014, doi: 10.7763/ijcee.2014.v6.845.

[18] John G. Daugman, "Efficient Approach for Text Extraction in Images and Video Frames Using Gabor Filter," *International Journal of Computer and Electrical Engineering*, vol. 6, 2014.

[19] Syed Saqib Bukhari, Mayce Ibrahim Ali Al Azawi, Faisal Shafait, and Thomas M. Breuel, "Document Image Segmentation using DiscriminativeLearning over Connected Components," 2010, p. 490.

[20] F. D. Julca-Aguilar, A. L. L. M. Maia, and N. S. T. Hirata, "Text/Non-Text Classification of Connected Components in Document Images," in *Proceedings - 30th Conference on Graphics, Patterns and Images, SIBGRAPI 2017*, Nov. 2017, pp. 450–455. doi: 10.1109/SIBGRAPI.2017.66.

[21] Samuel Lukas, Pujianto Yugopustito, and Dion Krisnadi, "Identification of Indonesian Vehicle Registration Plate by Adaptive Thresholding and Region Labeling Algorithm," IEEE, 2013.

[22] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust Wide Baseline Stereo from Maximally Stable Extremal Regions," 2004.

[23] S. Aladhadh, H. U. Rehman, A. M. Qamar, and R. U. Khan, "Recurrent convolutional neural network MSER-based approach for payable document processing," *Computers, Materials and Continua*, vol. 69, no. 3, pp. 3399–3410, 2021, doi: 10.32604/cmc.2021.018724.

[24] P. Yugopuspito, "Recognition of Indonesian Vehicle Registration Plate by Discrete Cosine Transform and Radial Basis Function Network," *International Journal of Information and Electronics Engineering*, 2015, doi: 10.7763/ijiee.2015.v5.544.

[25] Y. Bassil and M. Alwani, "OCR POST-PROCESSING ERROR CORRECTION ALGORITHM USING GOOGLE'S ONLINE SPELLING SUGGESTION," 2012. [Online]. Available: http://www.cisjournal.org/journalofcomputing/archive/vol3no1/vol3no1_7.pdf

[26] Kazuhiro Kayashima and Hisao Niwa, "POSTPROCESSING FOR CHARACTER RECOGNITION USING KEYWORD INFORMATION," 1992.

[27] C. M. Strohmaier, C. Ringlstetter, K. U. Schulz, and S. Mihov, "Lexical Postcorrection of OCR-Results: The Web as a Dynamic Secondary Dictionary?," 2003.

[28] W. Zhu, Y. Liu, and L. Hao, "A novel OCR approach based on document layout analysis and text block classification," in *Proceedings - 12th International Conference on Computational Intelligence and Security, CIS 2016*, Jan. 2017, pp. 91–94. doi: 10.1109/CIS.2016.28.

[29] S. Karthikeyan, A. G. Seco de Herrera, F. Doctor, and A. Mirza, "An OCR Post-correction Approach using Deep Learning for Processing Medical Reports," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021, doi: 10.1109/TCSVT.2021.3087641.