International Journal of
Advanced Science and Convergence

ISSN : 2619-8150
Volume 1, Number 2, June 2019
https://doi.org/10.22662/IJASC.2019.1.2.001

Image to Text Conversion Technique for Anti-Plagiarism System

Mark B. Batomalaque[1], Chella May R. Camacho[2], Maria Jewella P. Dalida[3] and Jen Aldwayne B. Delmo[4]

[1-4] College of Engineering, Architecture and Fine Arts, Batangas State University, 4225, Philippines

Abstract

Background/Objectives: The IMAGE TO TEXT CONVERSION TECHNIQUE FOR ANTI-PLAGIARISM SYSTEM is a design project on how the Optical Character Recognition will be utilized in order to extract text from images that can be used to increase the accuracy rate of an anti-plagiarism checker. It also highlights the integration of Convolutional Neural Network and its effect in the result of the conversion. **Methods/Statistical analysis:** Optical Character Recognition is a technology that recognizes text within an image. It is commonly used to recognize text in scanned documents, but it serves many other purposes as well. While Convolutional Neural network is a category of neural networks that have been proven very effective in performing image recognition and classification. The main objective of the study is to design a software that will convert images of text into plain editable text. The study aims to use a specific algorithm to extract useful information from the images. **Findings:** It will integrate the two algorithm, convolutional neural network and optical character recognition technology in order to develop a software. The input of the software is a document in .docx format and will generate an output in the same format. **Improvements/Applications:** This software will be an aid to the existing anti-plagiarism checkers to generate a more thorough and better plagiarism check.

Index Terms

Convolutional Neural Network (CNN), Image Processing, Optical Character Recognition (OCR), Plagiarism

Corresponding author : M. B. Batomalaque
markbb0319@gmail.com

• Manuscript received April 17, 2019.
• Revised May 15, 2019; Accepted June 20, 2019.
• Date of publication June 30, 2019.

I. INTRODUCTION

The internet plays a vital role as a major source of information. Generally, students rely on different search engines as resources for their school works, thesis, dissertations and other related works. As a result, the rate of students committing plagiarism is now a growing problem. According to a survey on 43, 000 high school students done by The Josephson Institute Center for Youth Ethics, one out of three high school students have admitted to plagiarizing their assignments through the use of the internet. Another survey conducted by Donald McCabe from the Rutgers University revealed the growing plagiarism rate that is happening in the academe. The survey showed that 36% of undergraduates admit to "paraphrasing/copying few sentences from Internet source without footnoting it." and another 24% of graduate students report doing the same. Also, 38% admitted to "paraphrasing/copying few sentences from written source without footnoting it." and another 25% of graduate students report doing the same. There is a 7% report copying materials "almost word for word from a written source without citation." and 4% of graduate students were reported doing the same (P.org, 2017).

The cases of plagiarism are not only prominent in the United States but also here in the Philippines. University students are aware of the existence of the plagiarism software to check the integrity of their papers. However, students are clever enough to find ways on how to trick these applications. There are videos and blogs surfacing the internet on how to fool plagiarism checkers. Commonly used tricks were paraphrasing the entire sentences copied from the internet, using synonyms in replacing other terms, altering the sequencing of phrases, converting the text into a PDF format and taking a snip of the desired information. Since Turnitin and other plagiarism platforms are unable to recognize whether an image and its content was plagiarized or not which can't be included in the plagiarism report, students resorted to snipping information. Despite that, this trick can be revealed through image processing.

In response to the growing plagiarism rate, innovating these checkers by adding image processing algorithm is of great use. Incorporating optical character recognition and convolutional neural network is a weapon that can help lessen the plagiarism rate. Since images cannot be detected by the existing anti-plagiarism software, the researchers will design a system that converts texts in images into plain editable file. This editable file will then be submitted to the existing plagiarism checkers to be inspected if plagiarism has been performed. The main requirement of this study is to design and develop a software that applies digital image processing to be use as enhancement tool to extract useful information of images, utilize the optical character recognition as the instrument in the conversion of images of text to be able to extract characters as editable file and employ the convolutional neural network for classifying and identifying images to improve the optical character recognition results.

II. METHODOLOGY

A. Digital Image Processing

Image processing is a method to convert an image into digital form and perform some operations on it, in order to get an enhanced image or to extract some useful information from it. It is a type of signal dispensation in which input is image, like video frame or photograph and output may be image or characteristics associated with that image. Usually image processing system includes treating images as two-dimensional signals while applying already set signal processing methods to them. There are different types of tasks in digital image processing that includes image acquisition, storage and transmission, image enhancement and restoration, image understanding and image recognition, and pre-processing stage of computer vision (Kang, 2007).

This technology was used in the study, as the main task of this project that includes the steps and process involving processing, analysis and classifications of documents images. The input of this project is documents images that undergo to the process of image acquisition, preprocessing, segmentation, feature extraction, classification. The technology is essential in order to use the image as input in the project. Enhancing the image provides a better output that was be easier to recognize and decode.

B. Optical Character Recognition

Optical Character Recognition was a process used for interpretation of scanned or printed images into machine-encoded text or editable. An OCR system is composed of many components. The one component used was image acquisition. This was the first step of OCR which involves obtaining a digital image from an external source. Optical character recognition is the process of producing a digital format from a digital image representation of a machine-printed document or handwritten sheet. The purposed of the transformation is that a computer can read and process the text automatically, and hence the productivity of a human increases. The interdisciplinary nature of an OCR may involve many several disciplines of computer science topics including, but not limited to, image processing, pattern recognition, natural language processing,

artificial intelligence, and database systems (Jana, et.al, 2014).

The developers used optical character recognition for it is the technology that be utilized in achieving the desired conversion of images to digital text. The engine used in this project was the optical character technology to obtained the text from images in the documents.

C. Machine Learning

Machine Learning is a form of Artificial Intelligence based on the idea to automatically learn from data, identify patterns, and predicting outcomes rather through explicit programming. There are different approaches for a machine to learn. Machine Learning can be separated into three types; supervised learning, unsupervised learning and reinforcement learning. Supervised learning analyze pattern using both input data and output data. Unsupervised learning finds pattern based only on input data. Reinforcement learning trains the algorithm using a system of trial and error to improve its performance (Esposito,t.al., 2017).

The developers used machine learning that enables the software to train using the model. It needs a dataset that will pass through the specific Neural Network that the developers used. The datasets be fed into the Neural Network model in order to analyze the dataset's pattern. The aim of the project was to classify the characters on the image of any documents. With the help of Machine Learning, it will be easier to identify, recognize, and classify the pattern of the characters.

D. Neural Networks

Neural Network is a computing system made up of layers which models itself like a human brain. The layers consist of numerous amounts of interconnected 'nodes' which connect the layers on either side. Some of the 'input layers' which communicates several forms of information from one or more layers. Others are in the opposite of the network which is called 'outside layers. This layer is responsible for computation and how it responds to the information from the network to the outside world. And the 'hidden layers' perform the transferring of communication between the input and output layers (Woodford, 2019)..

Character recognition is one of the real-world applications of neural network. Taking the Neural Network approach, the system fed training to recognized the English alphabet. Once the image of the character is fed to the system, it recognizes the input character which the output is given in image. The idea of Neural Network was used to the project as it recognizes the different set of characters. It has the ability to extract and detect pattern that other techniques can't.

E. Convolutional Neural Network

Convolutional Neural Network (CNN) is successfully applied on different problems such as image processing, script identification, and character recognition. Convolutional Neural Network is used for image classification. CNN can learn a hierarchy of feature detectors and train a nonlinear classifier to identify complex document layouts. By using CNN, it can down sample the image and then the normalized image is fed to the Convolutional Neural Network to predict the class label (Bhandare, et.al, 2016).

The convolutional neural network was used in this project as an aid in the optical character recognition technology to produce more accurate results in the conversion of images to text file. It classifies and identifies the images that contains text that will be helpful in the conversion.

F. Python Programming Language

Python is a powerful and high level, object-oriented programming language. It has user-friendly data structure and has very simple syntax. Also, it is very simple and easy to use since it requires unique syntax that focuses on readability. An interpreted language, python has a design philosophy which emphasizes code readability (notably using whitespace indention to delimit code blocks rather than curly brackets or keywords), and a syntax which allows programmers to express concepts in fewer lines of codes than possible in languages like C++ or Java (Hari, 2019).

Python is used to the project as the programming language that the developers will used for the development of the project. It was chosen because it is easy to use and because its powerful environment for machine learning and computer vision. It has varieties of machine learning framework that makes it easier for the developers to save time in the development phase.

G. TensorFlow

Created by the Google Brain team, TensorFlow is an open source library for numerical computation and large-scale machine learning. It bundles together a slew of machine learning and deep learning (aka neural networking) models and algorithms and makes them useful by way of a common metaphor. It uses Python to provide a convenient front-end API for building applications with the framework, while executing those applications in high-performance C++ (Yegulalp, 2018).

This technology was used in the project as the library to be used in implementing the convolutional neural network. The study needs the assistance of the TensorFlow because of its features neural network that is essential in the software to be developed.

H. Related Studies

A study about font classification presents a simple framework based on Convolutional Neural Network. The study aims to classify pages or text lines into font categories. The CNN is trained to classify the text into font classes. The study shows that CNN can perform well at classifying fonts. Although the study also shows that fonts Times New Roman and Arial can be very similar when distinguishing between those two fonts (Tensmeyer, et.al, 2017).

The study proved that the CNN can be used in order to classify fonts that was essential in the development of the software. The project aims to do the similar process of identifying the typefaces used in a certain document that was to be converted into editable text accurately.

In an article about document image classification using CNN, it was discussed that document images classes are defined by the structural similarity. The study presents an approached for classifying the documents image by using Convolutional Neural Network. The approached was down sampling and pixel value normalization that predicts the class label by feeding the image to the CNN. The study used rectified linear units and trained the datasets to enhance the performance of the CNN (Kang, et.al, 2014).

In this study, the developers were inspired to used Convolutional Neural Network to identify and classify the label of the images. By training the datasets in the Convolutional Neural Network, the project aims to enhanced the results of the output of the image to texts conversion.

In a study conducted about image enhancement methods that can be used in improving image quality of documents, different techniques were used to remove any unwanted interferences in the text in document images. It also discussed how adaptive image enhancements (AIE) differs in other methods (Chiang, et.al, 2017)

The developers want to apply this method as it focused on enhancement of the document images and it was used to help the developers to remove unnecessary interferences in the document images.

A study for text recognition using image processing was conducted to recognized the text from any hardcopy documents into an editable text. The study stated that text recognition contains several steps that include pre-processing, segmentation, feature extraction, classification, and post processing. They proposed an algorithm which solve offline character recognition. They trained the algorithm by using an image as input that are in the database and then they used pre-processing, segmentation, and detect the line (Mizan, et.al, 2017).

In this study, the developers considered the used of these steps as they were very similar to the project that extracts the texts from documents. This provides the developers an overview to the optical character recognition and how to efficiently extract important information in all kinds of images in the documents. This also served as a guide to the developers for the development of the project.

According to a study of detecting and recognizing different charts and graphics that are present in a business document, using computer vision techniques involves the understanding and recognition of document images. The research proposed a system that classifies chart images. The process includes segmentation, primitive detection and chart detection. The proposed algorithm extracts information that the charts can be re-created. The strategy could be a syntactic approach which employments numerical linguistic uses to recognize and classify each component of a chart (Syendsen, 2015).

The study involves classifying the components of chart images which relates to the project as the input documents include charts. The aim of the study was also related to the project, which was the recognition of documents' images. Though the study is focused only on the recognition of charts in the documents, it also helps the developers to extracts important information in the documents.

HawkEye is an innovative mobile system used to detect code cloning. It uses Multi-Language OCR-Compiler that convert screenshots into texts. Then, the OCR extracts the relevant program keywords from the converted texts. A compiler was then used to compile the extracted program keywords based on the identified program language. Karp Rabin & Greedy String Tiling Algorithm plagiarism detection algorithms is used to perform plagiarism detection on these extracted keywords (Puri, et.al, 2016).

The present study was similar in a sense that it has the same concepts of detection of plagiarism. The main focused of the proposed system was to avoid cloning of programming codes. This served as a helpful guide for the development of the project.

The Optical Capture Recognition aims to develop an Optical Character Recognition (OCR) for android mobile devices. The project used OCR to convert different types of documents or images captured by a camera. The project has three stages which is scan document, capture and recognize the data to save to desired format (Zdadou, 2015).

This study was pertinent to the present study since the aim of the developers was to developed a software that convert documents images. In line with

the present study, the developers used the study for the development of the project as it uses similar approached like iConvert. The purposed of the application was to recognized text in scanned text documents, text images, and any picture taken by an Android based device in order to reused it later, which was similar to the objectives of the project.

According to a conducted study about text extraction from natural scene, the demand of application and techniques in image/video-based analytics are increasing. However, text extraction is challenging problem to be solved because it cannot be assumed that the image contains only character. The researcher proposed a framework of scene text extraction to solve these problems. The framework is composed of two components. Scene text detection, to identify areas that contains text in an image. Scene text recognition, to convert the detected and located areas containing text into readable text data (Yi, 2014).

In this study, the developers were interested on the application and techniques used in the proposed framework of extracting texts from an image. There were methods in text extraction used which was similar to the project. The said study served as a very helpful guide on the development of the iConvert.

According to most of the plagiarism detection systems such as Turnitin, it forgets to evaluate the figures and charts present in a document before checking for plagiarism. This means that people can take advantage of discarding the figures and charts. People can easily plagiarize any figures and charts. Thus, the researchers come up to develop a system that will solve or detect in plagiarism of figure and charts. The paper presents a method to prevent the plagiarism of figures and charts using a shape-based processing and multimedia retrieval.

The developed system retrieves figures with almost the same characteristics based on the query. The system retrieved figure with the similarity value of 1 with an exact match. Similarity value of less than one was given when there is only partially match in the query. And when there is mismatch in the query, the system returns similarity value close to zero (Arrish, et.al, 2014).

This study connects to the present study in terms of its goal which was to create a plagiarism system. The primary focused on detection of plagiarized flowcharts. The methods used greatly help the developers such as the pre-processing to reduced errors and the training of datasets.

In a thesis paper authored by Ohlsson, it stated that there are several steps that the Tesseract OCR algorithms has used. The first step is connected-component labeling in which Tesseract searches the images and outlines of the components is stored. It classifies the foreground pixels and the outlines is gathered together as ‘Blobs’ or potential characters (Ohlsson, 2016).

Second, the Blobs were used in line finding algorithm. Detection and correction of skew in the image of any page is an important part in a document recognition system. The line finding algorithm is designed so that it can be identify the skewed page to lessen the loss of quality of an image without having to de-skew.

Blobs are organized into line of texts by analyzing the adjacent space of the image to the potential characters. To find the locations of the binary image, the algorithm does a Y projection and count the pixels less than the threshold. It will undergo more analyzing to confirm. The third step is the baseline fitting algorithm which it is to find the baselines for each of the lines. When the Tesseract OCR engine has detected the texts lines, it will examine the texts lines to be able to find the proper height across the lines. This algorithm is the first stage on how to recognize the characters.

In the next step of the Tesseract algorithm it uses the fixed pitch algorithm. In this algorithm, Tesseracts analyzes whether the texts lines are fixed pitched. This allows to find the appropriate character width. When it recognizes that the lines are fixed pitched, Tesseracts split the words into single character using the pitch. The fifth step of the Tesseract algorithm used in this paper is the non-fixed pitch spacing delimiting or proportional word finding. In this algorithm the characters that does not have uniform in width or there are problems in the width in surrounding neighborhood.

To solve this problems Tesseract then reclassified these characters. It measures the gap in a partial vertical range between the baseline and mean line. Spaces that are close to the threshold at this stage are made fuzzy, so that a final decision can be made after word recognition. The final step in the Tesseract algorithm is the word recognition. When Tesseracts done finding all possible characters in the document. It recognized word by word and line by lines and then the words are will be processed through a contextual and syntactical analyzer which ensures accurate recognition.

In the word recognition, there are recognition process to be able to produce better character recognitions, the first one is chopping joined characters. While the result from a word is unsatisfactory, Tesseract attempts to improve the result by chopping the blob with worst confidence from the character classifier. Candidate chop points are found from concave vertices of a polygonal approximation of the outline, and may have either another concave vertex opposite, or a line segment. It may take up 3 pairs of chop points to successfully separate joined characters from the ASCII set.

Chops are executed in priority order. Any chop that fails to improve the confidence of the result is undone, but not completely discarded so that the chop can be re-used later by the associator if needed. Associating chopped characters is when the potential chops have been exhausted, if the word is still not good enough, it is given to the associator. The associator makes an A* (best first) search of the segmentation graph of possible combinations of the maximally chopped blobs into candidate characters.

It does this without actually building the segmentation graph, but instead maintains a hash table of visited states. The A* search proceeds by pulling candidate new states from a priority queue and evaluating them by classifying unclassified combinations of fragments. It may be argued that this fully-chop-then-associate approach is at best inefficient, at worst liable to miss important chops, and that may well be the case. The advantage is that the chop-then-associate scheme simplifies the data structures that would be required to maintain the full segmentation graph.

Inspired by the advancement of new technology and the problem regarding plagiarizing any kind of sources such as books, journals, magazines, theses, and dissertation without proper citation. The developers come up with a plagiarism detection tool named “Image to Text Conversion Technique for Anti-Plagiarism Systems”.

The researchers gathered necessary data in line with the development of the project for the success of the study.

The software development followed a five-phase of SDLC. The first process was requirement analysis. In this stage, the developers contributed ideas on how to execute the project. Developers did a lot of brainstorming until it jumps on the idea and concept of the operation and on what the design of the project be and conducted consultation of the project. Data gathering was included in this stage. Developers researched for the data that supports and needed to meet the requirements of the project. All of the needed requirements are collected accordingly.

After the requirements analysis, the developers come up with the software design that was suitable for the execution of the project. The design was prepared based on the given requirements.

Next phase was the development of the design. The proposed design was implemented. In this phase, the actual coding starts. This phase takes the longest time of the software development life cycle because it aims to achieved the design and the agreed requirements based on the implementation and it took a lot of hard-work and patience to developed the system.

The fourth phase was the testing of the design. Testing verified the results according to the requirements needed of the implemented design. When the design failed, the developers reviewed the third phase which was the development stage, to alter the codes and replaced it with the working codes that can achieved the requirements needed.

The final phase was the evolution. In this phase, the system was delivered to the target customers. Feedbacks were collected to gathered the user experience and the problems encountered upon trying the software. These problems are then solved until it satisfies the user requirements.

Developers used Python as their programming language that features Open CV (Open computer vision) which is an open source library of Python that facilitates real time image processing, machine learning, and computer vision. TensorFlow is a deep learning framework supported by Open CV. This was an open source machine learning library used for Convolutional Neural Networks. Convolutional Neural Networks perform ahead of the existing computer vision that generates modern output. The application of this neural network is image classification and recognition. This feature was used as the image detector that recognizes the text in the images. The text that was recognized by the CNN was converted into digital text by the used of the Optical Character Recognition. OCR is a technology that converts text in images into editable text.

The software that was developed by the developers was named iConvert which highlights the conversion of images into digital text.

iConvert interface has a height of 3.16 inches by 5.14 inches of width. It has element including two buttons which were the convert button and the download button. The interface has a status bar that guide the users which was located at the bottom part of the interface. The iConvert’s logo was located at center of the interface with a background of a gif. When the user clicked the convert button, a file-browser window appears that locates the input document files. This window was an open filename dialog box that the users can browse their local disk to select the input Microsoft Word Document. After selecting the input file, conversion automatically starts.

The module python-docx scan the whole input document and print the present text in the document. While the module docx2txt was responsible for the detected images in the document. All images were saved in a folder. These images undergo OCR by using the module pytesseract. After converting the images into text, these texts and the result texts of module python-docx combines with iConvert’s template by using docxcompose.composer from the module Composer. During conversion, iConvert user interface appears. Users can identify if the file was already converted and ready to download by taking a look on its status bar.

When the status bar displays "Your file is ready to be downloaded.", click the download button to download the file and save it with the desired filename and location. A file-browser automatically pop-up and the converted file is ready to be browsed. Output file should be open with the MS Word. The output file contains the text from the input file and the converted images into text. Text presents in the input file are printed first followed by the converted text from the screenshots of the text. The output file has a template that the developers developed. It has a unique watermark of the developers. The output file restricts to editing to assure that the contents were not be altered or edited again.

This output file is now ready to be uploaded into the plagiarism checker.

This software was supported by python-docx module that only supports Microsoft Word Document (.docx).

III. Results and Discussion

The first tests done by the developers is by using their OCR with CNN software. They run tests on different images printed in different fonts. The typefaces used by the developers were archivo-regular, calibri, comic neue, concertone-regular, fjallaone-regular, helvetica neue, karla-regular, montserrat-regular, open sans and times new roman.

The software can identify and print the letters, characters, and numbers. However, the output generated by the software were all inaccurate. It can be deduced to have a less than 50% accuracy based on the results generated as shown in fig. 1.

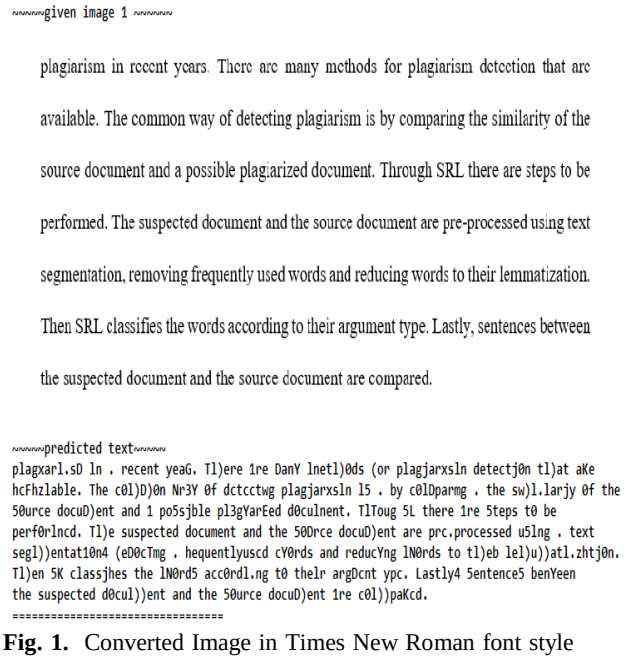

Fig. 1. Converted Image in Times New Roman font style using CNN

The second software designed by the developers was a pure OCR technology execution. Instead of incorporating OCR with CNN, the developers decided to exclude the CNN codes to compare the result of the two software. The same input used in the first testing was used in the second execution.

Series of testing was done to identify the accuracy of the conversion. The output generated were almost accurate for most of the typefaces used. All the characters, letters and numbers were recognized by the software and were converted into text. Although some of the words were not converted as to how it should be, still there is a minimal error detected in the image conversion to text. The typefaces that generated a full accuracy conversion were karla-regular and times new roman as shown in fig. 2.

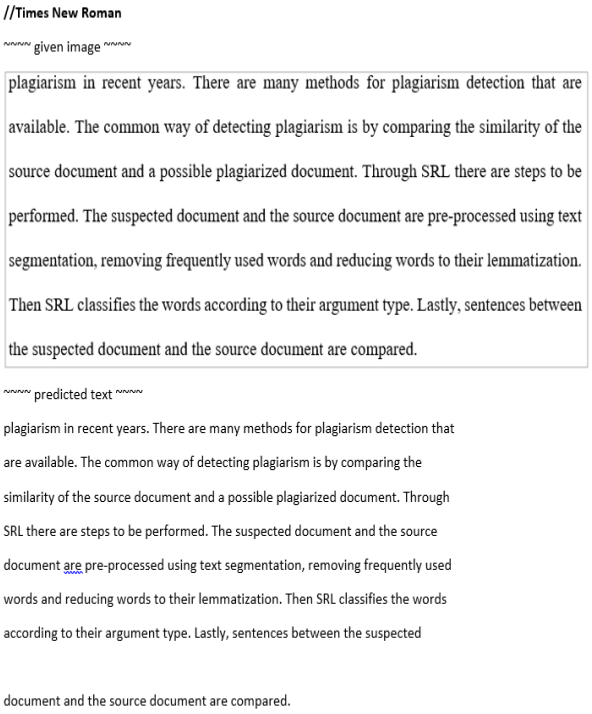

Fig. 2. Converted Image in Times New Roman font style using OCR

IV. Conclusion

After the software development and several tests conducted, the developers were able to conclude the following:

1. Developers implemented two digital image processing algorithms which were the Optical Character Recognition with CNN and Optical Character Recognition alone that was used to extract useful information of images. It turns out in the result that OCR alone was more effective

than using with CNN.As a result, developers chose the OCR alone.
2. Optical Character Recognition was used by the developers as a tool for extraction of characters that can be used in the software to convert the characters in the image to editable files.
3. Based on the comparison of both algorithms, the developers chose OCR alone as the implementation of converting of the images into machine-readable text because this produces the higher accuracy than with CNN.

The Image to Text Conversion Technique for Anti- Plagiarism System shows that it was possible to use convolutional neural network and optical character recognition technology as an aid in plagiarism detection. Developers were able to achieved the objectives of the project. However, the software developed was a new researched that still need further study and has its own limitations.

For future works, some recommendations have been listed:

1. Use a dataset that contains more fonts that are usually used in writing researches and school works.
2. Training and testing the software in different computers with different specifications to know the full capability of the software.
3. Optimize the software to support other word file formats like Microsoft Word 97 - 2003 Document (.doc), Microsoft Word Macro-Enabled Document (.docm), Rich Text Format (.rtf), Text Document (.txt), and XML Document (.xml) and could scan documents with or without images.

ACKNOWLEDGMENT

The developers would like to express their gratitude and appreciation to the following people for their support and inspiration to accomplish this research work:

To God Almighty, the Greatest Engineer of all, for giving life, knowledge and strength to the researchers to overcome all the complexities that came their way during the stages of completing the project;

To their family for their unconditional love and support, and for their never-ending patience and prayers;

To all their friends and loved ones who have been their inspirations at all times, and for giving them some tips in providing a good research project;

To their project adviser, chairman and panel members and other professors who have helped to make this project possible and for giving them some tips in providing a good research project

Appreciation is also extended to countless others, who have helped them in one way or another to finish this project. To each member of the group, for being cooperative all throughout.

REFERENCES

[1] Arrish, S., F. N. Afif, A. Maidorawa & N. Salim. (2014). Shape-Based Plagiarism Detection for Flowchart Figures in Texts, *International Journal of Computer Science & Information Technology,* 113-124.

[2] Bhandare, A., M. Bhide, P. Gokhale & R. Chandavarkar. (2016). Applications of Convolutional Neural Networks, *International Journal of Computer Science and Information Technologies,* 2206-2215.

[3] Esposito, M., K. Bheemaiah & T. Tse. (2017). What is machine learning?, 04 May 2017. [Online]. Available: http://theconversation.com/what-is-machine-learning-76759.

[4] Chiang, J., C. Hsia, H. Tu, H. Giang & T. Lin. (2017). Adaptive image enhancement method for document, IEEE, 2017.

[5] Hari, S. (2019). Python Programming Language – A Gentle Introduction, 12 April 2019. [Online]. Available: https://hackr.io/blog/python-programming-language.

[6] Jana, R., R. Chowdhury & M. Islam. (2014). Optical Character Recognition from Text Image, *International Journal of Computer Applications Technology and Research,* 239-243.

[7] Kang, B.H. (2007). A Review on Image and Video processing, *International Journal of Multimedia and Ubiquitous Engineering,* p. 49.

[8] Kang, L, J. Kuma, P. Ye, Y. Li & D. Doermann. (2014). Convolutional Neural Networks for Document Image Classification, *2014 22nd International Conference on Pattern Recognition,* 3168-3172.

[9] Mizan, C., T. Chakraborty & S. Karmakar. (2017). Text Recognition using Image Processing, *International Journal of Advanced Research in Computer Science,* 765-768.

[10] Ohlsson, V. (2016). Optical Character and Symbol Recognition using Tesseract, 2016.

[11] P.org. (2017). Plagiarism: Facts & Stats, 7 June 2017. [Online]. Available: https://www.plagiarism.org/article/plagiarism-facts-and-stats.

[12] Puri, K. & P. Mulay. (2016). Hawk Eye: A Plagiarism Detection System, 2016.

[13] Svendsen, J. (2015). Chart Detection and Recognition in Graphics Intensive Business

Documents, 2015.
[14] Tensmeyer, C., D. Saunders & T. Martinez. (2017). "Convolutional Neural Networks for Font Classification, 2017.
[15] Woodford, C. (2019) How neural networks work - A simple introduction," 04 April 2019. [Online]. Available: https://www.explainthatstuff.com/introduction-to-neural-networks.html.
[16] Yegulalp, S. (2018). What is TensorFlow? The machine learning library explained, 6 June 2018. [Online]. Available: https://www.infoworld.com/article/3278008/what-is-tensorflow-the-machine-learning-library-explained.html.
[17] Yi, C. (2014). Text Extraction From Natural Scene: Methodology And Application, CUNY Academic Works, 2014.
[18] Zdadou, F. (2015). Project: The Optical Capture Recognition, 2015.