Regional Processing for Optical Character Recognition of Plain Text on Artistic Background (Image Processing and Pattern Recognition)

***Francis G. Balazon*[a] *and Ace Gregorre C. Bañares*[b]**
Batangas State University-Lipa Campus[a]*, Philippines*
SIMSAC Petro, Canada[b]
Corresponding Author Email: fbalazon@yahoo.com

ABSTRACT

Optical character recognition (OCR) is still capable of reading text from colored or designed images effectively but with the current technological advancements, OCR has been left behind. On the other hand, regional processing makes it more possible to distinguish which pixels belong to the background and which belongs to the text. Regional processing for OCR would single out letters on the artistic image, put them into words and then - words into sentences, thus enabling to access and edit the content of the original document. The objectives of the study were to take a different approach on OCR, further enhance its accuracy and better analyze text on designed images to expand its limits. Optical character recognition for extracting text on designed images needed a varied way of analysis with the existing algorithms because the same problems might arise if the same approach would be used. This focused more on the preprocessing stage which has been the most common cause of mistakes – when images are not prepared enough for character recognition. The algorithms that were used to efficiently read text on designed images are under machine learning and image processing.

The project focused on the proper utilization of the k-nearest neighbors and Tesseract algorithm with regards to the regions of an image. The overall functioning of OCR contained some steps to recognize the text which include: scanning, pre-processing, feature extraction and classification. Here, the input image to OCR is any hand written or printed texts like books, screenshots and photos with text. Such input is given to OCR initially through scanning - where the analog document is digitized. Then, text regions within image are located, symbols are extracted and preprocessed, and features are extracted and recognized.

Findings revealed that the images and texts come in different styles which required different pre-processing methods. There are many factors that affect the result of OCR in an image and a single algorithm is not enough to solve them all. The study has shown the effectiveness of grouping colors by regions in order to extract the text within an image. First, the program analyses the structure of document image. It divides the page into elements such as blocks of texts, tables, images, etc. The lines are divided into words and then - into characters. Once the characters have been singled out, the program compares them with a set of pattern images. The program analyses different variants of breaking of lines into words and words into characters, presenting the recognized text.

KEYWORDS: OCR, Regional Processing, Binarization, Deskew Algorithm

1. INTRODUCTION

Humans have the most complex mind of all living things. They are never contented and usually seek for progress and further development to make everything accessible and easier. The idea of creating something that is not yet invented seemed never-ending.

Ideas are everywhere. When the researchers saw images with text on it, they realized that the messages in the image were just placed by a person and therefore it is reversible. That idea introduced them to optical character recognition (OCR).

OCR does exist but the researchers felt that its function may still be enhanced. The existent OCR can recognize text in a plain background but cannot read text through differently designed backgrounds. The researchers then kept looking for OCR that would fit their need but they could not find one.

The idea bloomed once again when the opportunity came to make an innovation that included the study of a new algorithm. They realized that the way to finding something that does not exist is not to simply stop looking for it. They have to make it. Like others, they have themself and believe that they have done certain development or innovation.

OCR has existed for decades already. Why does its development seem to be so minimal? Sketch recognition, musical OCR, and the like have already been made, but these are not improvements but variations. OCR is also made available online but it works just the same as OCR software. It could be really functional if it would be enhanced efficiently, it can even lead to handwriting recognition in the near future. Nonetheless, what is the extent of the capability of the current OCR and how can it be expanded?

What is Optical Character Recognition (OCR)? This terminology comes from the words *optical* which mean visual appearance; *character* which means number, a letter, or a symbol; and *recognition* which means a process of understanding something. Indeed, OCR is the process of understanding numbers, letters, or symbols based on its visual appearance. It is commonly defined today as the process of extracting a human-readable text on an image into a computer-editable one. Perhaps, it is really how it is used. OCR does exist but the researcher felt that there seemed to be missing in it. It can recognize text in a plain background but cannot read text through differently designed backgrounds[1].

How does OCR work? It follows three basic steps - pre-processing, character recognition, and post-processing. In pre-processing, the image is adjusted in such a way that character recognition becomes easier. Everything unnecessary, such as lines and dusts is removed. The image is also deskewed having text in horizontally straight and vertically upright directions.

In character recognition, the image is analyzed and grouped into single units to find which corresponds to a number, a letter, or a symbol. It is then recorded as an editable text. On the other hand, in post-processing, the output is given more accuracy by setting a dictionary of possible words based on language. It is like the grammar and spelling correction in a document, though this process is optional.

How is OCR used? It can be used in so many ways, for as long as the purpose is digitizing text. One of the most prominent uses of it is the digitization of analog materials such as books, newspapers, magazines etc. It can also be used to create editable documents from images with text, portable document format (PDF) files, etc.

This study focused on OCR for text on artistic background, which is one of the major difficulties in the current OCR technology. Inaccuracy of results happens when the background is complex rather than plain.

2. OBJECTIVES

The main objective of the project were to take a different approach on optical character recognition, further enhance its accuracy and to better analyze text on designed images, hence to expand its limits.

Optical character recognition for extracting text on designed images needed a varied way of analysis with the existing algorithms because the same problems might arise if the same approach would use. This focused more on the preprocessing stage which has been the most common cause of mistakes – when images are not prepared enough for character recognition. The algorithms that were used to efficiently read text on designed images are under machine learning and image processing.

The following were the specific objectives of this research work:

1.To be able to improve OCR for text on images

2. To efficiently use different algorithms:

    2.1 gray scale filter;
    2.2 feature extraction and
    2.3 regional processing

3. To be able to use systematic methodology, or rapid application development, specifically in program development and testing.

3. SCOPE AND LIMITATIONS

The study only encompassed optical character recognition of text in graphical images from computers. Images were limited to the file types - '.jpg' or '.jpeg'; '.gif'; '.bmp', and '.png'. Primarily, the expected output of this study was software created from the C# programming language. It would be very useful if the idea is applied to a machine like a scanner but the algorithm is the main concern before anything else. The output was merely being an editable text that could be copied through a computer. It might be subjected to copyright issues if not used properly but the responsibility would be solely shouldered by the user since it was not made for such purpose. Fair-use, disclaimer, and mentioning of references can be used for infringement to be avoided.

The researcher did not intend to further increase the accuracy of OCR on plain images which is already around 99% for existing algorithms, but to enhance the accuracy of OCR on colored and designed images with text. Even if there are modifications done, the three major steps in OCR– preprocessing, character recognition, and post processing, where still followed.

Preprocessing were given most attention since the study is about complicated images that are subjected to OCR. Extremely complicated images like overexposed or underexposed ones, overlapping characters, distorted text, and images would not be supported. As previously mentioned, the researcher hoped to enhance the accuracy of OPCR. On colorful images, and minimize errors but not in terms of character recognition, the fonts that are supported by this program are limited to Arial, Tahoma, Verdana, Calibri, Times New Roman, Century Gothic, Comic Sans, and the likes. The input file with image and therefore handwritten texts that resemble the previously mentioned fonts can also be recognized. Designed fonts, like Old English Text or those symbolic fonts are excluded first to avoid confusion. Cursive type of text which commonly has ‘Script’ or ‘Hand’ in their names like Segoe Script, Bradley Hand ITC, Lucida Handwriting, and Monotype Corsiva, are also not supported. Those are subject to future implementation since it is mostly covered by a more complicated technology called Intelligent Character Recognition (ICR).

In post processing, the language that was used was English since it is the universal language. It will be the only language at first so that there will be less complexity and the study can focus more on its main goal.

With regards to algorithm, the researcher use the most of those being used in the current OCR such as deskewing, despeckling, binarization, layout analysis or zoning, line and word detection, character isolation or segmentation, and line removal. Normalizing aspect ratio was not necessary. The essential algorithms – k-nearest neighbor algorithm, and feature extraction or matrix matching were also used without modifications. Regional processing, which was done by grouping areas in an image was the new approached used in this study.

4. DESIGN AND METHODOLOGY

The project focuses on the proper utilization of the algorithms with regards to the regions of an image. An image is composed of pixels and they are the core element of image processing. They can be compared to the atom in a matter which is indivisible but does contain an individual property.

Figure 1 illustrates how an image is formed by set of pixels (squares).

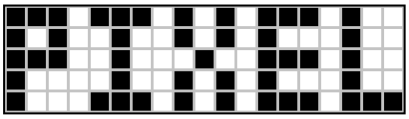


Figure 1. Illustration of pixel

What appears to be meaningful to a person’s eyes is meaningless to a computer. But if the pixels can be grouped into meaningful regions, then the idea would be created and the computer would be able to understand it.

OCR is a complex technology that converts images with text into editable format[2]. OCR allows to process scanned books, screenshots and photos with text and gets editable documents like TXT, DOC or PDF files. This technology is widely used in many areas and the most advanced OCR systems can handle almost all types of images, even such complex as scanned magazine pages with images and columns or photos from a mobile phone.

How does modern OCR work? The process of converting an image to editable document is separated to several steps; every step is a set of related algorithms which do a piece of OCR job. There are general steps in OCR process.

When loading an image as bitmap from a given source, source can be a file or a pointer to a memory block. Also, a good OCR system must understand a lot of image formats: BMP, TIFF (both one-page and multi-pages images), JPEG, PNG and so on. PDF files must be supported as well. Many documents are stored as images in PDF format and the only way to extract text from such files is to perform OCR.

Detecting the most important image features like resolution and inversion is essential. Many OCR algorithms expect some predefined range of font sizes and foreground/background colors so the image must be rescaled and inverted before processing when necessary.

Image can be skewed or it can have a lot of noise **s**o deskew and denoising algorithms are applied to improve the image quality as shown in Figure 2.

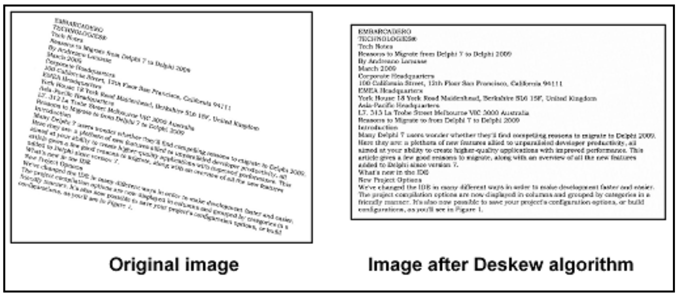

Original image
Image after Deskew algorithm

Figure2. Deskew Algorithm

Many OCR algorithms require bi-tonal image; therefore color or gray image must be converted to black-white image. This process is called "binarization". It is a very important step because incorrect binarization would cause a lot of problems. Figure 3 shows a binarization process.

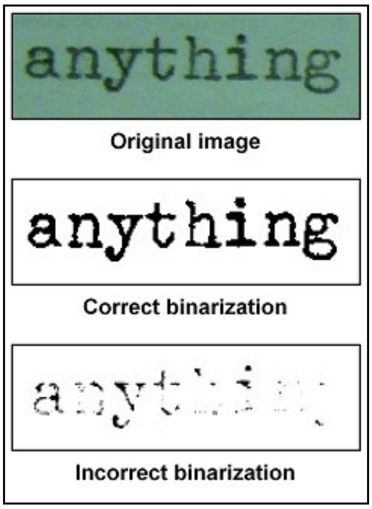


Figure3. Binarization

Lines detection and removing is the step required to improve page layout analysis, to achieve better recognition quality for underlined text, to detect tables, etc., as shown in Figure 4.

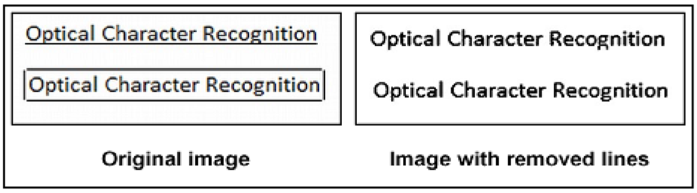

Optical Character Recognition
Optical Character Recognition
Optical Character Recognition
Optical Character Recognition
Original image
Image with removed lines

Figure4. Line Removal

Page layout analysis is the step also called "zoning". At this stage OCR system must detect positions and types of all important areas on the image. Detection of text lines and words is sometimes not an easy task because of different font sizes and small spaces between words. Combined-broken characters analysis is usually needed. It is very common situation when some characters are broken to several parts, or when a few characters touch each one; it is necessary to detect such cases and find correct position of every character as shown in Figure 5.

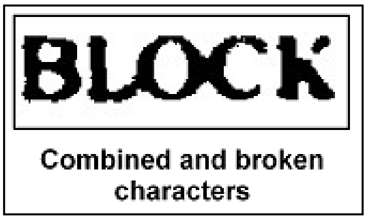


Figure 5. Overlapping and disconnected parts in characters

Recognition of characters is the main algorithm of OCR; an image of every character must be converted to appropriate character code. Sometimes this algorithm produces several character codes for uncertain images. For instance, recognition of the image of "I" character can produce "I", "|" "1", "l" codes and the final character code will be selected later.

Dictionary support is the step that can improve recognition quality, some characters like "1" and "I", "C" and "G" can look very similar and the dictionary can help to make the decision.

Saving results to selected output format, for instance, searchable PDF, DOC, RTF, and TXT is important to save original page layout: columns, fonts, colors, pictures, background and so on.

It is not a complete list, a lot of other minor algorithms must also be implemented to achieve good recognition on various image types, but they are not principal in most cases and can vary in different OCR systems.

Every OCR step is very important; the whole OCR process will fail if one of its steps cannot handle given image correctly. Every algorithm must work correctly on the highest range of images that is why only a few good universal OCR systems are available. On the other hand, if some features of given images are known, the task becomes much easier.

It is possible to get better recognition quality if only one kind of image must be processed. To achieve the best results when some features of images are known, good OCR system must have the ability to adjust the most important parameters of every algorithm; sometimes this is the only way to improve recognition quality. Unfortunately, nowadays there are no OCR systems that can be comparable with human eyes and it seems they will not be created in the near future.

5. RESULTS AND DISCUSSIONS

After all the efforts made by different groups and enthusiasts throughout the years, OCR nowadays are completely astonishing. Just imagine how tons of books are being made available in digital copy. It seems to be miraculous that people can somehow touch a text in a paper by converting it into a computer -readable text. In that case, they can search a certain word in a book even if it has hundreds of pages and find all of their instances. That would take hours or even days if done manually, but OCR could help lessen the time and effort in the process. Despite that, OCR still faced problems that are not yet solved; it is not a matter of perfection but rather a matter of perspective. It is almost impossible for OCR not to have errors; the current OCR is good enough but not in the case of OCR for text on images.

The capability of such technology must not be limited to reading simple text in simple backgrounds. If development has stopped on black and white television, colored TVs would not have been invented. Thus, further development and innovations should be done to bring OCR to the next level.

Preprocessing of image is the key point to enhance the accuracy of OCR on images. AForge Imaging, which is not so popular though it contains a package of useful image processing algorithms, becomes very handy. AForge.Net Framework described as a C# framework was designed for developers and researchers in the fields of Computer Vision and Artificial Intelligence which include image processing, neural networks, genetic algorithms, machine learning, robotics, etc. They have the so-called Image Processing Lab, which is an image processing application written in C#. This includes different image processing filters and tools available in AForge.NET framework to analyze images[3].

Functionality was the main concern of the program and not the design itself which emphasized substance over matter.

Figure 6 shows the main screen of the program.

Figure 6. Main program interface

5.1 Gray Scale Filter

This study focused on text with artistic background. It simply means that the only difference here compared to a normal OCR is that the background is not plain. To make it simple, the foremost question that is needed to be answered is how can plain background to be made artistic? As a human, it is quite simple for us to identify a text inside a colorful and designed background. But how can a computer do the same?

People need to find the logic of what they do to be able to isolate the text in an image. Taking it into an experiment, a set of different artistic and colored pictures with different messages inside was observed, from simple to complex. A conclusion is set that there is a certain amount of difference in color between the pixel of a text and the pixel of a background because the text would not be readable if its color is exactly the same as the background. It is easier to compare a green from a red and harder to compare a dark green to a darker green but for a computer, it is the other way around.

Technically, the computer identifies only red, green, and blue or commonly termed as RGB. To show an example, let it be that Red-Green-Blue or RGB (0,0,0) determines the amount of red, green, and blue respectively in a pixel. The previous entry shows that red, green, and blue are all set to zero and therefore it is pure black. On the other hand, RGB(255,255,255) is pure white by setting the color range as 0 to 255. RGB (255,0,0) is the purest plain red. RGB(0,255,0) is the purest plain green.

If the average of red, green, and blue were taken there would still be a conflict. It would be that the purest green, [(0+255+0)/3] = (255/3), and purest red, [(255+0+0)/3)] = (255/3), would have the same result. It means that the computer would still be blinded to a red text in a green background or vice versa. Nonetheless, knowing that a color goes darker as it approaches zero, if we take the pure green, RGB (0,255,0) and compare it to a dark green, RGB(0,200,0), or to a darker green, RGB(0,150,0), or even to the slightest difference of 1, RGB(0,254,0), the computer will be able to clearly see it even if our bare eyes could not because a single variable is changed.

Therefore, the conclusion that color differentiates the background from the text is false. It is rather the color tone. To make it easier, the image can be converted into gray scale. Turning an image to gray scale can be done by equalizing the levels of red, green, and blue.

5.2 Feature Extraction

It would be really hard to start developing new version of OCR from scratch and it is quite impractical because it is like going back to ages and doing what others have already done to come up with the same result. With that being said, Tesseract-OCR, maybe the most common OCR with all-in-one algorithm for recognizing text was used. It was OCR Engine that was developed at HP Laboratories between 1985 and 1995 and now at Google[4]. Tesseract is probably the most accurate open source OCR engine available.

Combined with the Leptonica Image Processing Library it can read a wide variety of image formats and convert them to text in over 60 languages. It was one of the top 3 engines in the 1995 UNLV Accuracy test. Between 1995 and 2006 it had little work done on it, but since then it has been improved extensively by Google. It is released under the Apache License 2.0. Feature extraction was be done primarily by the Tesseract OCR Engine (see Figure 7). It is something that is not modified or is used as it is because it was the fixed variable in the study. It determined whether the proposed algorithm has reached its goal by comparing it to existing software that uses the same engine.

5.3 Regional Processing

Upon having the image turned to gray scale, it was easier for the computer to recognize and compare one pixel to another. The slightest difference between each pixel could be noticed. It was where regional processing takes place. Assuming that the text is oriented horizontally, the image was virtually divided into horizontal regions as shown in Figure 7.

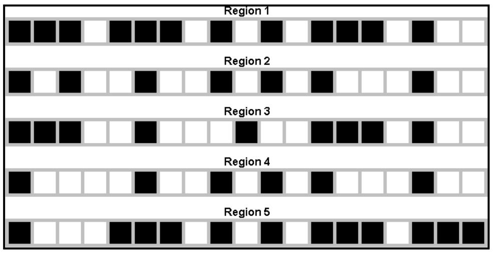
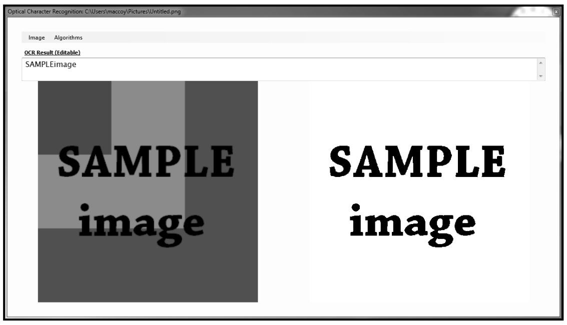


Figure7. "PIXEL" image divided into regions

In one way or another, there would always be the darkest and the lightest tone in a line. It is not unless all the pixels in the set are of the same tone, but that simply means that no part of it belongs to a text. The extremes could be compared to and decide if the difference is huge enough to consider that the line contains a pixel from a text. If there is, taking the mean of the set, or the middle value between the extremes can determine which of the pixels belong to the background and which belong to the text.

5.4 Testing


Optical Character Recognition: C:\Users\maccoy\Pictures\Untitled.png

OCR Result (Editable)

SAMPLEimage

The program can be used simply in its executable form. Test data would be of various images with distinct characteristics that differs one from another. Figure 8 demonstrates a sample image that was processed.

The output text does not add up with previous results. The scroll bars adjust the value of the button before them. Invert option decides for Regional Processing option whether to invert the color of the image to be processed or not. Original Image does not change unless another image is loaded. The Processed Image does not retain previous image processing. Both images are of the same size and are converted to 300 dots per inch before OCR to provide better result. The image size adjusts together with the main form. The variability of image size, scrollbar values, invert check box, and image processing algorithms is primarily for testing of project efficiency.

6. CONCLUSIONS

The current OCR technology has almost perfected reading text on plain background. However, it doesn't end there. When it comes to advancement, it is not about cherishing what is already done well but rather filling up what is still missing. Though OCR makes amazingly accurate results, it still fails when it comes to graphical images. This project study worked on one small step in making it possible for OCR to better analyze text on designed images.

Regional processing is a way image preprocessing whereas the image is turned into black and white by removing the background image and leaving the text intact before it is processed by the Tesseract Engine. With the Tesseract specializing in reading text from plain background, the result will be as if the designed image is seen by the Tesseract as plainly black and white and therefore would bring optimal results. Color tone helps a lot for regional processing algorithm to completely remove the background image but it is still not enough to be able to manage each and every picture. Color invert is needed to be used when the text have lighter color than the background. Variation in the intensity of comparison between background and text is also essential.

The OCR technology still needs to improve reading text on designed images because it is one missing part on the equation. Computer technology evolves so quick that answers should better be prepared before questions are asked. What if decades or centuries from now, plain papers no longer exist and every paper and printing materials is already designed? Then the OCR technology that has been worked upon would have not encountered any mistake on plain background. It is quite an exaggeration, yet nobody knows what would happen in the future. Perhaps, a programmer's mind works in such a way that every hole should be patched to make a program applicable to anything within its purpose.

Some errors are still unsolved with the regional processing algorithm. Since color tone is the core factor that affects the result of regional processing, a picture with inconsistent and wide variety of colors makes it hard for the algorithm to perform its function, therefore leading some errors.

Image size is also quite important and sometimes manual adjustment is still required for OCR to have a correct result.

7. RECOMMENDATIONS

In order to read text from designed images properly, it is needed to literally remove the background image. To make it happen, the key difference between the background and the text should be found, and an algorithm to distinguish such difference should be formulated. Color tone is one key item, but there is still a lot more to be found.

The possibility of having a light colored text or a dark colored text should meet at one point that it would not matter and there is no longer need for any intervention to make it possible for the OCR to work on both aspects. Additionally, a part of the background having the same color with that of the text should also be classified as a part of the background and not part of the text.

Turning an image to gray scale is very efficient but it might work better if a colored image is taken into a process of color equalization so that variety of colors will be more minimized. The scaling of image and detection of the dimensions of a single character will also help a lot in making better results in the OCR process.

After all is done well, it would be really great if texts can be read either horizontally or vertically inclined, together or not in the same image and in such case preserving their orientation and at the same time preserving their position and sizes and spacing too. It sounds quite complex but it is very awesome and yet very feasible. It would be a great breakthrough in the OCR technology.

REFERENCES

[1] G. M. Design and Construction of an Opaque Optical Contour Tracer for Character Recognition Research. Available from https://circle.ubc.ca/bitstream/handle/2429/36013/UBC_1968_A7%20A88.pdf (1964); accessed 10 April 2013.

[2] Nicom. Optical Character Recognition (OCR) – How it works. Accessed from http://www.nicomsoft.com/optical-character-recognition-ocr-how-it-works (5 February 2012); accessed 9 April 2014.

[3] W. Philpot. Digital Image Processing. Cornell University

[4] Berlin, J. Improve OCR Accuracy on Color Documents. Available from http://learn.accusoft.com/white-paper/improve-ocr-accuracy-on-color-documents.html; accessed April 2014.