International Journal of Multidisciplinary Research And Studies

Available online at https://ijmras.com/

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH AND STUDIES

ISSN: 2640 7272

Volume:04; Issue:05 (2021)

GRAPHICAL PROCESSING UNITS OPTICAL CHARACTER

Anugya

M.Phil., Roll No. :150123: Session: 2015-16

University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India.

E-mail-anugya.anu@gmail.com.

ABSTRACT

For decades, OCR was the sole means to transform printouts into computer-process able data, and it is still the preferred method for turning paper invoices into extractable data that can be linked into financial systems, for example. However, electronic document submission now provides organizations with a significantly improved approach to areas like invoicing and sales processing, lowering costs and allowing employees to focus on higher-value activities. All thresholding1 techniques fall under the class of "data-reduction" algorithms. That is, the techniques seek to compress or reduce the information contained within a given image in order to reduce computational complexity. The purpose then, is to remove or reduce the unwanted information or "noise" and leave the "important" information intact, such as removing the background of a bank check so the OCR application can read the signature panel. For OCR applications, data reduction is usually confined to reducing a grayscale or color image to a black and white (binary or bi-tonal) image. This reduction is accomplished by calculating a level of intensity against which individual pixel values are compared.

KEYWORDS: Graphical, Units Optical, OCR applications, information intact,

INTRODUCTION

Text recognition is one of the most prominent applications of computer vision which is being used by several multinational Tech companies such as Apple, Google, etc. Apple recently announced including the "Live Text" feature in iOS15.This functionality is similar to how Google Lens works on Android phones and the Google Search and Photos apps on iOS. So, the basic procedure of how these feature works are, a person has to point the camera at an image or text given on a board sign or a chapter. The Live Text feature recognizes the text present in the image, be it a contact number or an email id. These features work on a service or technology called OCR (Optical Character Recognition). For decades, OCR was the sole means to transform printouts into computer-process able data, and it is still the preferred method for turning paper invoices into extractable data that can be linked into financial systems, for example. However, electronic document submission now provides organizations with a significantly improved approach to areas like invoicing and sales processing, lowering costs and allowing employees to focus on higher-value activities.

OCR PHASES

- **Pre-processing**

The purpose of the pre-processing phase is to prepare a given image for the isolation phase — primarily, to make it easier for the isolation phase to determine where character glyphs begin and end. Section 2.3 describes this phase and its associated problems in detail.

- **Isolation**

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

The isolation phase analyzes the cleaned image data from the pre-processing phase in an effort to locate and isolate pockets of text. These pockets are then further broken down into lines and, finally, into single glyphs. Section 2.5 describes this phase and its associated problems in detail.

- **Identification**

The identification phase examines the isolated glyphs and attempts to classify each of them as a particular character. Section 2.7 describes this phase and its associated problems in detail.

- **Post-processing**

Post-processing attempts to construct text from the output provided by the identification phase. The output might include spacing and formatting. Section 2.9 provides an overview of this phase and gives a brief introduction to its associated problems.

Pre-Processing Phase: Techniques and Solutions

All thresholding1 techniques fall under the class of “data-reduction” algorithms. That is, the techniques seek to compress or reduce the information contained within a given image in order to reduce computational complexity. The purpose then, is to remove or reduce the unwanted information or “noise” and leave the “important” information intact, such as removing the background of a bank check so the OCR application can read the signature panel. For OCR applications, data reduction is usually confined to reducing a grayscale or color image to a black and white (binary or bi-tonal) image. This reduction is accomplished by calculating a level of intensity against which individual pixel values are compared. The values that fall below the threshold are included in the binary image as black pixels; the others are white The threshold value can be calculated on a global or local level, and the techniques implementing each method are referred to as global or adaptive2 thresholding, respectively. Global methods calculate the threshold using the entire image, whereas local or adaptive thresholds readjust the threshold based on the area of the image around which processing is taking place Adaptive techniques typically calculate a “running value” for the threshold and make the black or white decision based on comparisons against this value These thresholding techniques are also appropriate for removing bleed-through text, because such text generally appears on the page at a much lighter contrast than normal text. However, for severely degraded text or in cases where a document exhibits both varying degrees of contrast and bleed-through text, thresholding is usually avoided.

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

1. Histogram-based thresholding techniques build a histogram of grayscale values and then analyze the histogram topology. The analysis finds groups of peaks that are then combined by averaging the grayscale values within the group, thereby reducing the total number of grayscale values. Figure 1.1 shows an example of this method. Figure 1.1a contains 6 grayscale values. The histogram groups the peaks. Averaging the groupings results in the image Figure 1.1c, which gives a reduction in grayscale values of 50% .

2. The clustering technique (or Otsu’s method) attempts to find the threshold that separates the grayscale values into classes that maximize the between-class variance. In simpler terms, the technique attempts to find the histogram grouping that maximizes the

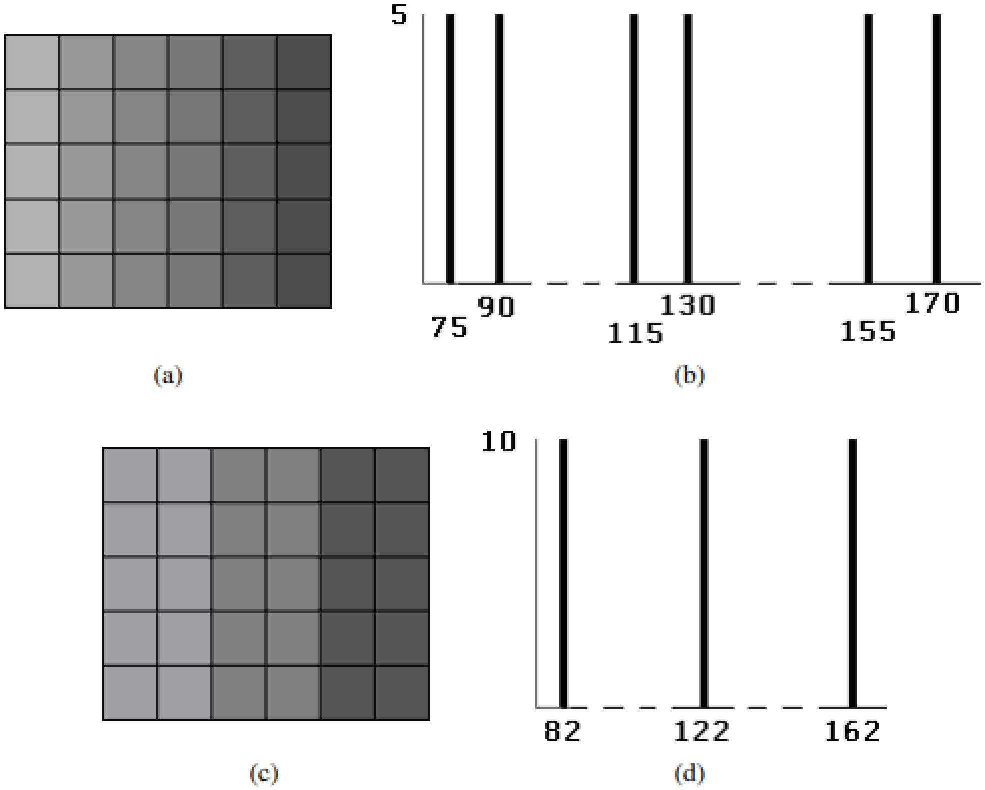

Fig 1. 1: The original image with 6 grayscale values (a) and its histogram (b). The reduced image with 3 grayscale values (c) and its histogram (d).

APPLICATIONS OF OCR

Optical character recognition has been applied to a number of applications. Some of the literatures covering these are languages other than English, namely, Latin, Cyrillic, Arabic, Hebrew, Indic, Bengali (Bangla), Devanagari, Tamil, Chinese, Japanese, Korean, etc.

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

Highlight in 1950’s [99], applied throughout the spectrum of industries resulting into revolutionizing the document management process. Optical Character Recognition or OCR has enabled scanned documents to become more than just image files, turning into fully searchable documents with text content recognized by computers. Optical Character Recognition extracts the relevant information and automatically enters it into electronic database instead of the conventional way of manually retyping the text. Optical Character Recognition is a vast field with a number of varied applications such as Practical applications, Invoice imaging, Legal industry, Banking, Healthcare, etc. OCR is also widely used in many other fields like Captcha, Institutional repositories and digital libraries, Optical Music Recognition without any human correction or human effort, Automatic number plate recognition, Handwritten Recognition and Other Industries. Some of them have been explained below:

Practical Applications

In recent years, OCR (Optical Character Recognition) technology has been applied throughout the entire spectrum of industries, revolutionizing the document management process. OCR has enabled scanned documents to become more than just image files, turning into fully searchable documents with text content that is recognized by computers. With the help of OCR, people no longer need to manually retype important documents when entering them into electronic databases. Instead, OCR extracts relevant information and enters it automatically. The result is accurate, efficient information processing in less time.

Invoice Imaging

It is widely used in many businesses applications to keep track of financial records and prevent a backlog of payments from piling up. In government agencies and independent organizations, OCR simplifies data collection and analysis, among other processes. As the technology continues to develop, more and more applications are found for OCR technology, including increased use of handwriting recognition. Furthermore, other technologies related to OCR, such as barcode recognition, are used daily in retail and other industries.

Legal Industry

In the legal industry, there has also been a significant movement to digitize paper documents. In order to save space and eliminate the need to sift through boxes of paper files, documents

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

are being scanned and entered into computer databases. OCR further simplifies the process by making documents text-searchable, so that they are easier to locate and work with once in the database. Legal professionals now have fast, easy access to a huge library of documents in electronic format, which they can find simply by typing in a few keywords.

Banking

The uses of OCR vary across different fields. One widely known application is in banking, where OCR is used to process checks without human involvement. A check can be inserted into a machine, the writing on it is scanned instantly, and the correct amount of money is transferred. This technology has nearly been perfected for printed checks, and is fairly accurate for handwritten checks as well, though it occasionally requires manual confirmation. Overall, this reduces wait times in many banks.

Healthcare

It Healthcare has also seen an increase in the use of OCR technology to process paperwork. Healthcare professionals always have to deal with large volumes of forms for each patient, including insurance forms as well as general health forms. To keep up with all of this information, it is useful to input relevant data into an electronic database that can be accessed as necessary. Form processing tools, powered by OCR, are able to extract information from forms and put it into databases, so that every patient's data is promptly recorded. As a result, healthcare providers can focus on delivering the best possible service to every patient.

Captcha

CAPTCHA is a program that can generate and grade tests that human can pass but current computers programmers’ cannot. Hacking is a serious threat to internet usage. Now a day’s most of the human activities like economic transactions, admission for education, registrations, travel bookings etc are carried out through internet and all this requires a password which is misused by hackers. They create programs to like dictionary attacks and automatic false enrolments which lead to waste of memory and resources of website. Dictionary attack is attack against password authenticated systems where a hacker [102] writes a program to repeatedly try different passwords like from a dictionary of most common passwords. In CAPTCHA, an image consisting of series of letters of number is generated which is obscured by image

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

distortion techniques, size and font variation, distracting backgrounds, random segments, highlights and noise in the image. This system can be used to remove this noise and segment the image to make the image tractable for the OCR (Optical Character Recognition) systems.

RESEARCH METHODOLOGY

VIRTUAL AND PHYSICAL ARCHITECTURE

In modern NVIDIA hardware, a graphics processing unit, or GPU, is a device that is connected to a computer's central processing unit (CPU) by means of a high-speed bus and that possesses onboard memory as well as up to three thousand processing elements that function according to a single instruction, multiple thread (SIMT) architecture. The single instruction, multiple data (SIMD) architecture is very similar to the SIMT design, with the exception that instructions are sent to groups of threads (which are comparable to CPU processes) called warps. Each individual warp has 32 threads and is a component of a block, which is itself a component of a grid. The virtual architecture for GPU programming is comprised of a hierarchy that is referred to as thread-warp-block-grid. See Figure 1.2.

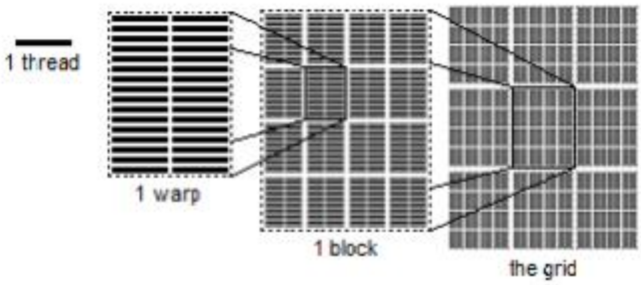

Fig 1. 2: The thread-warp-block-grid hierarchy. From left to right: one thread, 32 threads inside a warp, which is inside a block of 16 warps, which is inside a grid of 9 blocks.

DATA ANALYSIS

The first thing that the algorithm does on the picture is execute a horizontal pixel count, which involves counting the number of black pixels present in each row of horizontal pixels. It saves the data in an array that is accessed using the Y value of the line as the index. White space makes up the lines that have no black pixels at all; these lines are either immediately before or

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

immediately after a line. Therefore, the bottom of a line is shown by the first pixel count in ascending Y values that is not zero, and after that, the top of a line is indicated by a value of zero (assuming there are no markings outside of the bounding box for the line). The line is contained inside a bounding box that is formed by the top and bottom limits, as well as the edge of the page. This phase generates a list of bounding boxes as its output, which corresponds to the lines of text that are present in an image. The next step of the algorithm counts the number of black pixels present in each column and inside the bounding boxes of each line. The counts are saved in Version 1 in an array that is indexed by the value of X. As was the case in the earlier stage, values of zero indicate blank space. Therefore, beginning at the left, a number that is not zero denotes the left edge of a glyph, and the value that immediately follows it that is zero denotes the right edge of the glyph. In conjunction with the borders along the top and bottom lines, the edge boundaries form the

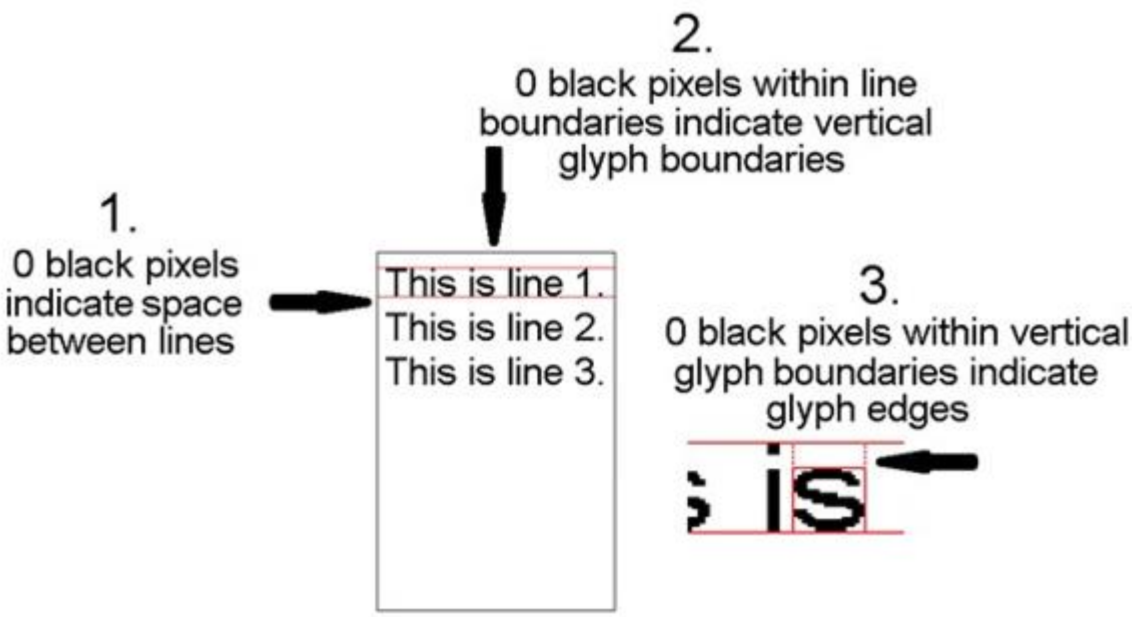

Fig 1.3 AgraphicalrepresentationofVersion1oftheSegRecalgorithm.

Create a box around the glyph using the given parameters. The result of completing this stage is a list of bounding boxes that match to the glyphs found along each line.

In the subsequent step, rows that are entirely white are removed from each bounding box. This causes the box to become smaller and more condensed. At this step, we count the pixels on each row that falls inside the bounding box of the glyph, and we get rid of any rows that have

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

0 black pixels. The result of this phase is a collection of bounding boxes that match to the glyphs in each line and are the best fit for that glyphs. The subsequent step separates each bounding box into a five-by-five grid, counts the amount of black pixels in each location inside the grid, and computes a 25-dimensional vector containing the counts. This is done in order to identify each individual glyph.

The concept is the same, despite the fact that Figure 4.2 depicts a 4 3 grid. When you count the number of black pixels in each location, beginning in the bottom left corner, you get the following nine-dimensional vector: (13, 1, 15, 7, 9, 11, 3, 18, 7)

The approach computes the Global Density Vector (GDV) for the glyph by dividing each element of the vector by the entire area of the glyph. This results in the following formula: (.034, .002, .039, .018, .024, .029, .008, .047, .018)

Version 1 locates the GDV within the training data that has the closest match in Euclidean 25-space; the character that is linked with that match is the recognised character. The search is performed by doing a direct comparison of the search vector to all of the vectors included in the training data in order to determine the closest match. The k-d tree nearest-neighbor search is a technique that is more effective than others; nevertheless, this search involves recursion, which GPUs only provide limited support for.

The most significant shortcoming of this method is that it is only able to function properly with characters that are clearly separated by white space. Because of the manner in which the method generates bounding boxes, there must be a minimum of one pixel of empty space around each glyph in order for the algorithm to successfully isolate it. Because of this limitation, the algorithm is unable to handle kerning, character ligatures, or typefaces with characters that are spaced too tightly together.

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

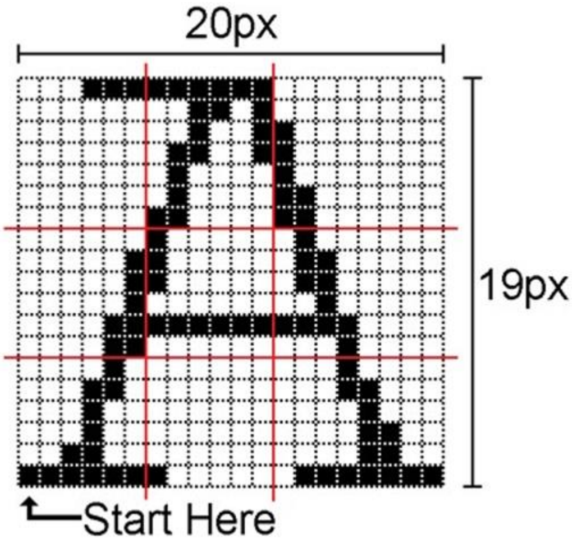

Fig 1.4 Aglyphbrokenintoa3×3grid

Additionally, the approach has a few minor flaws, such as the fact that it does not deal with marks, that it does not deal with even the most fundamental pixel noise, and that it does not have a mechanism for addressing rotation. The inclusion of extra pre- or post-processing processes, on the other hand, may help alleviate the effects of these defects and perhaps potentially rectify them. Examples of how these flaws can be mitigated include identifying marks and bases separately and then connecting them after they have been identified, rejecting "isolated" glyphs that are smaller than a certain size, gathering training data for rotated characters, or pre-processing to correct rotation by employing a shear transform.

Conclusion

Optical character recognition is a necessary first step for all applications that consider image as input Recognition of printed text gives good results. Almost all the data read was correct. Only few recognized fields contained mistakes, but they have been unreadable or damaged during the scanning process. Our evaluation shows that LBP with SVM gives optimal results with accuracy of 96.5%. Our survey has shown that data manually rewritten from the form by an experienced user contains less mistakes than the data recognized by OCR/system. The "Type-reading Optophone," which was first developed in 1914, was a device that turned the light that was reflected off of typewritten text into sounds, with each glyph producing its own distinct pitch. A person might learn to "read" the text and distinguish between the tones with the help of training The "Statistical Machine," which was a subsequent innovation that was patented in 1931, was supposed to assist in accelerating the process of searching for text within

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

microfilm archives A light source that is placed below the microfilm and "search plates," which are solid, opaque plates with the search text carved out, are placed above the microfilm provide a variable light source to a light detector that is placed above the search plate. The search plates are positioned so that they are placed directly on top of the microfilm. When the text on the microfilm perfectly matches the text on the search plate, very little light is allowed to travel through the film and reach the detector.

Reference

1. Dibco 2014. http://users.iit.demokritos.gr/~kntir/HDIBCO2014/ index.html. Accessed: 2015-08-03.

2. ABBYY. About ABBYY. https://www.abbyy.com/en-us/company/key-facts/, October 2016.

3. A. S. Abutableb. Automatic thresholding of gray-level pictures using twodimensional entropy. Computer Vision, Graphics, and Image Processing, 47:22–32, 1989.

4. Adnan Amin and Sue Wu. A robust system for thresholding and skew detection in mixed text/graphics documents. International Journal of Image and Graphics, 5(2):247–265, Apr 2005.

5. Itay Bar-Yosef, AlikMokeichev, Klara Kedem, ItshakDinstein, and Uri Ehrlich. Adaptive shape prior for recognition and variational segmentation of degraded historical characters. Pattern Recognition, 42(12):3348–3354, Dec 2009.

6. Junior Barrera, Marcel Brun, Routo Terada, and Edward R. Dougherty. Boosting OCR classifier by optimal edge noise filtering. Computational Imaging and Vision, 18(9):371–380, 2002.

7. Subhadip Basu, Nibaran Das, Ram Sarkar, Mahantapas Kundu, MitaNasipuri, and Dipak Kumar Basu. A hierarchical approach to recognition of handwritten bangla characters. Pattern Recognition, 42(7):1467–1484, July 2009.

8. C. I. Chang, Y. Du, J. Wang, S. M. Guo, and P. D. Thouin. Survey and comparative analysis of entropy and relative entropy thresholding techniques. IEEE Proceedings Vision, Image and Signal Processing, 153(6), Dec 2006.

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*

9. Xin Chen, Yuefang Gao, and Zhonghong Huang. Cuda-accelerated fast sauvola’s method on kepler architecture. Multimedia Tools and Applications, pages 1–12, 2014.

10. John Cheng, Max Grossman, and Ty McKercher. Professional CUDA C Programming. John Wiley and Sons, Inc., 2014.

11. E.E. Fournier D’Albe. On a type-reading optophone. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 90(619):373–375, May 1914.

12. A. K. Das and B. Chanda. A fast algorithm for skew detection of document images using morphology. International Journal on Document Analysis and Recognition, 4(2):109–114, 2001.

Anugya*, *University Department of COMPUTER SCIENCE, B.R.A. Bihar University, Muzaffarpur, India. E-mail: anugya.anu@gmail.com.*