Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

(Special Issue for ITECE 2016)

Comparative Analysis of Optimized Character Recognition Algoritms

Asst. Prof. Ravi Khimani[1], Asst. Prof. Nishant Sanghani[2], Asst. Prof. Pooja Vasani[3]

[1]*Computer Science & Engineering, SLTIET*
[2]*Computer Science & Engineering, SLTIET*
[3]*Computer Science & Engineering, AITS*

Abstract — Character Recognition, specially Optical Character Recognition is the mechanism of converting soft copy of typed, handwritten or printed text into system encoded text. It is widely used in the area of data reading from printed papers, whether invoices, passport, personal documents, bank statements, machine generated receipts, business cards, e-mail, printouts or any suitable documents. It is a approach to extract out printed texts so that it can be edited on machine and maintained comprehensively in machine either on any kind of storage media, displayed on-line, useful in processes such as machine translation, text-to-speech, key data and text mining. CR is a field of research in password matching, handwriting matching and computer vision.

Keywords: Character, Optical Character Recognition, Text reading, Extract Text from image, Filtering process

I. INTRODUCTION

The traditional way of entering data into a computer is through the keyboard. But this method is not efficient enough due to typing mistakes. In that case we have variety of ways available. number of technologies are exist for automatic detection of text, and they cover requirements for different portions of application. Different areas like Document Verification, Data Entry, Speech Recognition, Barcode, Mark Reading are required Character Recognition [1].

Optical Character Recognition used to recognize processed character optically. Optical recognition is performed in two ways, in off-line after the printing of text in machine has been over, while in on-line recognition is done as the characters are drawn on the machine. Both hand printed and printed characters may be recognized, but quality of inputted document defines the performance of recognition [2].

The more constrained the input is, the better will the performance of the OCR system be. However, when it comes to totally unconstrained handwriting, OCR machines are still a long way from reading as well as humans. However, the computer reads fast and technical advances are continually bringing the technology closer to its ideal.

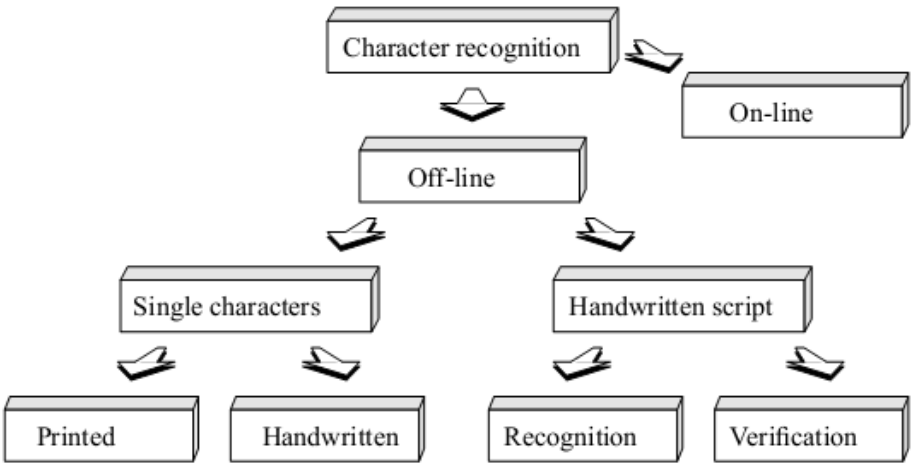

Figure 1. Types of character recognition

II. COMPONENTS IN OCR

A typical OCR system consists of several components. In figure 2 a common setup is illustrated. The first step in the process is to digitize the analog document using an optical scanner. When the regions containing text are located, each symbol is extracted through a segmentation process. The extracted symbols may then be pre-processed, eliminating noise, to facilitate the extraction of features in the next step.

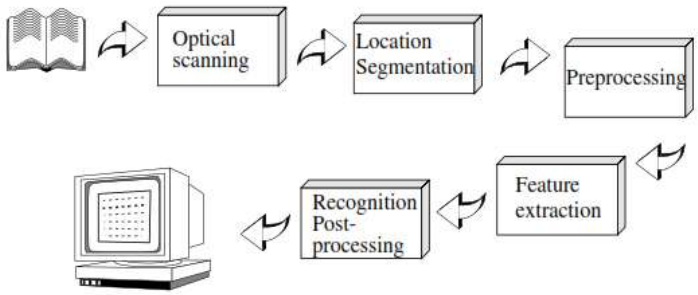

Figure 2. Components of OCR

The identity of each symbol is found by comparing the extracted features with descriptions of the symbol classes obtained through a previous learning phase. Finally contextual information is used to reconstruct the words and numbers of the original text. In the next sections these steps and some of the methods involved are described in more detail [3].

2.1. Optical scanning.

Through the scanning process a digital image of the original document is captured. In OCR optical scanners are used, which generally consist of a transport mechanism plus a sensing device that converts light intensity into gray-levels. Printed documents usually consist of black print on a white background. Hence, when performing OCR, it is common practice to convert the multilevel image into a bi-level image of black and white. Often this process, known as thresholding, is performed on the scanner to save memory space and computational effort. The thresholding process is important as the results of the following recognition are totally dependent of the quality of the bi-level image. Still, the thresholding performed on the scanner is usually very simple. A fixed threshold is used, where gray-levels below this threshold is said to be black and levels above are said to be white. For a high-contrast document with uniform background, a pre-chosen fixed threshold can be sufficient. However, a lot of documents encountered in practice have a rather large range in contrast. In these cases more sophisticated methods for thresholding are required to obtain a good result.

The best methods for thresholding are usually those which are able to vary the threshold over the document adapting to the local properties as contrast and brightness. However, such methods usually depend upon a multilevel scanning of the document which requires more memory and computational capacity. Therefore such techniques are seldom used in connection with OCR systems, although they result in better images [1].

2.2. Location and segmentation.

Segmentation is a process that determines the constituents of an image. It is necessary to locate the regions of the document where data have been printed and distinguish them from figures and graphics. For instance, when performing automatic mail-sorting, the address must be located and separated from other print on the envelope like stamps and company logos, prior to recognition. Applied to text, segmentation is the isolation of characters or words. The majority of optical character recognition algorithms segment the words into isolated characters which are recognized individually. Usually this segmentation is performed by isolating each connected component, that is each connected black area. This technique is easy to implement, but problems occur if characters touch or if characters are fragmented and consist of several parts. The main problems in segmentation may be divided into four groups:

- Extraction of touching and fragmented characters: Such distortions may lead to several joint characters being interpreted as one single character, or that a piece of a character is believed to be an entire symbol. Joints will occur if the document is a dark photocopy or if it is scanned at a low threshold. Also joints are common if the fonts are serif. The characters may be split if the document stems from a light photocopy or is scanned at a high threshold.
- Distinguishing noise from text: Dots and accents may be mistaken for noise, and vice versa.
- Mistaking graphics or geometry for text: This leads to non-text being sent to recognition.

- Mistaking text for graphics or geometry: In this case the text will not be passed to the recognition stage. This often happens if characters are connected to graphics.

2.3. Pre-processing

The image resulting from the scanning process may contain a certain amount of noise. Depending on the resolution on the scanner and the success of the applied technique for thresholding, the characters may be smeared or broken. Some of these defects, which may later cause poor recognition rates, can be eliminated by using a pre-processor to smooth the digitized characters. The smoothing implies both filling and thinning. Filling eliminates small breaks, gaps and holes in the digitized characters, while thinning reduces the width of the line. The most common techniques for smoothing move a window across the binary image of the character, applying certain rules to the contents of the window.

In addition to smoothing, pre-processing usually includes normalization. The normalization is applied to obtain characters of uniform size, slant and rotation. To be able to correct for rotation, the angle of rotation must be found. For rotated pages and lines of text, variants of Hough transform are commonly used for detecting skew [1].

2.4. Feature extraction

In these methods, significant measurements are calculated and extracted from a character and compared to descriptions of the character classes obtained during a training phase. The description that matches most closely provides recognition. The features are given as numbers in a feature vector, and this feature vector is used to represent the symbol.

2.4.1. Distribution of points

This category covers techniques that extract features based on the statistical distribution of points. These features are usually tolerant to distortions and style variations. Some of the typical techniques within this area are listed below.

- Zoning: The rectangle circumscribing the character is divided into several overlapping, or non-overlapping, regions and the densities of black points within these regions are computed and used as features.
- Moments: The moments of black points about a chosen centre, for example the centre of gravity, or a chosen coordinate system, are used as features.
- Crossings and distances: In the crossing technique features are found from the number of times the character shape is crossed by vectors along certain directions. This technique is often used by commercial systems because it can be performed at high speed and requires low complexity. When using the distance technique certain lengths along the vectors crossing the character shape are measured.
- n-Tuples: The relative joint occurrence of black and white points (foreground and background) in certain specified orderings, are used as features.
- Characteristic loci: For each point in the background of the character, vertical and horizontal vectors are generated. The numbers of times the line segments describing the character are intersected by these vectors are used as features.

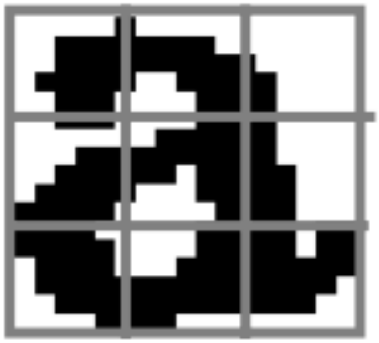

Figure 3. Zoning

2.4.2. Transformations and series expansions

These techniques help to reduce the dimensionality of the feature vector and the extracted features can be made invariant to global deformations like translation and rotation. The transformations used may be Fourier, Walsh, Haar, Hadamard, Karhunen-Loeve, Hough, principal axis transform etc.

2.4.3. Structural analysis

During structural analysis, features that describe the geometric and topological structures of a symbol are extracted. By these features one attempt to describe the physical makeup of the character, and some of the commonly used features are strokes, bays, end-points, intersections between lines and loops. Compared to other techniques the structural analysis gives features with high tolerance to noise and style variations. However, the features are only moderately tolerant to rotation and translation [9].

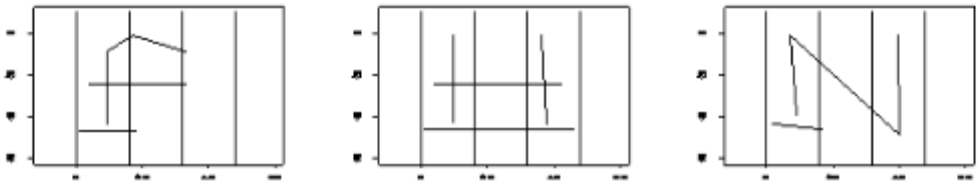

Figure 4. Strokes extracted from capital letter F, H and N

2.5. Classification

The classification is the process of identifying each character and assigning to it the correct character class. In the following sections two different approaches for classification in character recognition are discussed. First decision-theoretic recognition is treated. These methods are used when the description of the character can be numerically represented in a feature vector. We may also have pattern characteristics derived from the physical structure of the character which are not as easily quantized. In these cases the relationship between the characteristics may be of importance when deciding on class membership. For instance, if we know that a character consists of one vertical and one horizontal stroke, it may be either an "L" or a "T", and the relationship between the two strokes is needed to distinguish the characters. A structural approach is then needed.

2.6. Post Processing

2.6.1. Grouping

The result of plain symbol recognition on a document is a set of individual symbols. However, these symbols in themselves do usually not contain enough information. Instead we would like to associate the individual symbols that belong to the same string with each other, making up words and numbers. The process of performing this association of symbols into strings, is commonly referred to as grouping. The grouping of the symbols into strings is based on the symbols' location in the document. Symbols that are found to be sufficiently close are grouped together. For fonts with fixed pitch the process of grouping is fairly easy as the position of each character is known. For typeset characters the distance between characters are variable. However, distance between words is usually significant larger than the distance between characters, and grouping is therefore still possible. The real problems occur for handwritten characters or when the text is skewed.

2.6.2. Error detection and correction

Up until the grouping each character has been treated separately, and the context in which each character appears has usually not been exploited. However, in advanced optical text recognition problems, a system consisting only of single-character recognition will not be sufficient. Even the best recognition systems will not give 100% percent correct identification of all characters, but some of these errors may be detected or even corrected by the use of context. There are two main approaches, where the first utilizes the possibility of sequences of characters appearing together. This may be done by the use of rules defining the syntax of the word, by saying for instance that after a period there should usually be a capital letter. Also, for different languages the probabilities of two or more characters appearing together in a sequence can be computed and may be utilized to detect errors. For instance, in the English language the probability of a "k" appearing after an "h" in a word is zero and if such a combination is detected an error is assumed.

Another approach is the use of dictionaries, which has proven to be the most efficient method for error detection and correction. Given a word, in which an error may be present, the word is looked up in the dictionary. If the word is not in the dictionary, an error has been detected, and may be corrected by changing the word into the most similar word. Probabilities obtained from the classification, may help to identify the character which has been erroneously classified. If the word is present in the dictionary, this does unfortunately not prove that no error occurred. An error may have

transformed the word from one legal word to another, and such errors are undetectable by this procedure. The disadvantage of the dictionary methods is that the searches and comparisons implied are time-consuming.

III. COMAPIRITIVE ANALYSIS

3.1. Character Recognition using Feature Extraction

Feature extraction method is applied on handwritten recognition based on the candidate search and elimination technique. The initial candidates for recognition are found by applying by zoning method on input glyphs. it's propose cavities as a structural approach suited specifically for Telugu script, where cavity vectors are used to prune the candidates by zoning. It gives the 100% features and cavity features of the input dataset. it's propose an improved and robust recognition strategy which first uses the pixel distributions of the script and later exploits the structural information of Telugu orthography. It uses three techniques.

First is Zoning, For a candidate search this method uses pixel density measurement distribution in different zones of the input glyph as a feature vector. First the input glyph is broken into zones by super-imposing a grid and then the percentage of the number of foreground pixels. A codebook of this feature vector is pre-computed from the training set. The feature vector of the input glyph is computed and searched in the codebook to obtain k nearest neighbors. The distance measure is Euclidean Distance between the feature vectors. The search concludes if a unique match is produced after pruning. If the search does not select an unique candidate, then the remaining candidates are passed to the next stage. This method is invariant under linear scaling as the percentage of pixels is unaffected by scaling. The present candidate search technique promises low computationally complexity. The main component of time complexity

Second is Cavity Based Structural Analysis, Cavities are used as structural features in our recognition. The existence and position of these cavities is a structurally distinguishing feature. They use cavities since they provide discrimination between glyphs which are could be very confusing for recognition. Cavities are detected by generating a contour of the glyph and performing connected component detection on the contour image, since cavities get disconnected from outer boundary in a contour image.

Third one is Normalization and Template Matching. Template matching only if the previous stage doesn't conclude the search. This stage has two stages internally. The first stage is the nonlinear. Normalization stage where image scaling is performed based on the image features such as projection profiles or crossing counts [5].

3.2. CR using Statistical and Background Directional Distribution Features

In this method of Background Directional Distribution Feature, a novel approach is used some statistical features like zonal density, projection histograms (horizontal, vertical and both diagonal), distance profiles (from left, right, top and bottom sides). In addition with above features, background directional distribution (BDD) features are also considered to recognize character pattern with depth. It must have sample database that is enough to identify specific characters according to the feature specified. With near about 200 samples for each character in alphabet with digits must be collected from different writers with different features. These samples are pre-processed and normalized to specific size like 32*32 sizes. SVM, K-NN and PNN classifiers are used for classification. The performance comparison of features used in different combination with different classifiers is presented and analyzed. It should take more time, because every character are compared feature by feature with the samples that were collected previously. During comparison it maintains information about the best match from the sample and at the end it gives the best sample match with the inputted text. It generally used in Handwritten data matching [6].

3.3. CR using Template Matching

Optical Character Recognition by using Template Matching is a system prototype that useful to recognize the character or alphabet by comparing two images of the alphabet. The objectives of this system prototype are to develop a prototype for the Optical Character Recognition (OCR) system and to implement the Template Matching algorithm in developing the system prototype. This system prototype has its own scopes which are using Template Matching as the algorithm that applied to recognize the characters, characters to be tested are alphabet (A – Z),grey-scale and bit-map (bmp 2bit depth) images were used with Times New Roman font type, using bitmap image format with 240 x 240 image size and recognizing the alphabet by comparing between two images. The purpose of this system prototype is to solve the problem in recognizing the character which is before that it is difficult to recognize the character without using any techniques and Template Matching is as one of the solution to overcome the problem.

Matlab R2006a is the software tool that was used in developing the system prototype. There are a few processes that were involved in this system prototype. The processes are starting from the acquisition process, filtering process, threshold the image, clustering the image of alphabet and lastly recognize the alphabet. All of these processes are very important to get the result of recognition after comparing the two character images. The value of the data that was entered will be extracted from the images, comprising letters. Each character was automatically selected and threshold using methods previously described. This process involves the use of a database of characters or templates. There exists a template for all possible input characters. For recognition to occur, the current input character is compared to each template to find either an exact match, or the template with the closest representation of the input character. They use the filtering image and thresholding to filter the image [7].

It has disadvantage that, we must have alphabets as sample images to compare each font type, that makes our database lengthy and complicated. And also it increases time to find out exact match of alphabet. Along with that it can't recognize if text is drawn as image with shading and designs in the image. we must need to use different methods to remove such shades and designs from the image to perform Template matching process.

3.4. Character recognition without segmentation

A new method for Character Recognition as a part of knowledge based word interpretation model. This new method is based on the recognition of sub-graphs homeomorphism to previously defined prototypes of characters. Gaps in between characters are identified as potential parts of characters by implementing a variant of the notion of relative neighborhood used in computational perception. In the system, each sub-graph of strokes that matches a previously defined character prototype is recognized anywhere in the word even if it corresponds to a broken character or to a character touching another one. The characters are detected in the order defined by the matching quality. Each sub-graph that is recognized is introduced as a node in a directed net that compiles different alternatives of interpretation of the features in the feature graph. A path in the net represents a consistent succession of characters in the word. The method allows the recognition of characters that overlap or that is underlined. A final search for the optimal path under certain criteria gives the best interpretation of the word features. The character recognizer uses a flexible matching between the features and a flexible groping of the individual features to be matched. Broken characters are recognized by looking for gaps between features that may be interpreted as part of a character. Touching characters are recognized because the matching allows non-matched adjacent strokes [10].

This method has advantage for finding out text from the images which has broken characters or characters along with shades or designs, because without segmentation, we need to create sub-graph for each character according to matching quality. And after creating best qualitative sub-graph, character has been identified by model.

3.5. Character recognition using optimization algorithm

A new method of optical character recognition using hierarchical optimization algorithms. it this technique, a new algorithm is described, which is based on the pattern character recognition algorithms and uses hierarchical optimization. The better recognition results obtained using the proposed algorithm give us a confirmation of a better aptitude of the approach for the industrial environment. The main problem with text recognition is distorted and noisy characters in image. To solve this problem, they use two approaches; one is remove distortion of recognized character. And another one is consistent change in pre-processing filters setting and analyze the result with previous results. Both increase recognition time. So, hierarchical probabilistic matching is used. In which, first is only some part of possible template position compared with character image. Second is resolution of templates and search fields turn by turn changed. This method helps to decrease the search area dynamically and increase the velocity. It uses the translation, scaling, rotation operations on character and template. The main idea of this method is, the image recognized using pre-processing using filters with different parameters. One important step is quality criterion calculation which represents quality of template. The distortion and noise carried out using the quality criterion calculation procedure. To increase recognition accuracy by comparing template with recognized character, the steps are modified, like, change in resolution, search area modification, quality criterion modification. This method uses optimization methods based on patterns with different resolution [11].

In this, pre-processing takes too much time to compare or extract characters from images. Then comparison is done portion by portion of character image. To read text from defective documents, this method has best approach compared.

VI. COCLUSION

After analyzing performance of each methods for different-different type of documents, time and quality are the parameters are needed to identify which method is better. For non-readable documents optimizing algorithm method

gives best qualitative output but others are fails and it also doesn't need samples to compare characters. But you must have character scripts to use such method. While others having advantage on time, if we use template matching or no-segmentation method, it requires less time. so, at the end we can say, according to document / image type and condition, which algorithm should be comprehensively match criteria of time and quality to extract or compare characters.

REFERENCES

[1] Line Eikvil, "Optical Character Recognition", 1993.
[2] Shunji Mori, Ching Y. Suen, Kazuhiko Yamamoto, "Historical Review of OCR Research and Development, Proceeding of IEEE, Vol. 80(7), pp. 1029-1058, 1992.
[3] Quin Chen, "Evaluation of OCR Algorithms for Images with Different Spatial Resolutions and Noises", University of Ottawa, 2003.
[4] Thomas Natschlager, "Optical Character Recognition", Institute of Theoretical Computer Science.
[5] Aparna Vara Lakshmi Vemuri, T.V.Sai Krishna, Atul Negi, "Dataset Generation for OCR" [OCR_Datasheet_Generation.pdf].
[6] Kartar Singh Siddharth, Mahesh Jangid, Renu Dhir, Rajneesh Rani, "Handwritten Gurmukhi Character Recognition Using Statistical and Background Directional Distribution Features", International Journal on Computer Science and Engineering, Vol. 3(6), pp. 2332-2345, 2011.
[7] Nadira Muda, Nik Kamariah Nik Ismail, Siti Azami Abu Bakar, Jasni Mohamad Zain "Optical Character Recognition By Using Template Matching (Alphabet)".
[8] Jesse Hansen, "A Matlab Project in Optical Character Recognition (OCR)".
[9] Oivind Due Trier, Anil K. Jain, Troffin Taxt, "Feature Extraction Methods for Character Recognition – A Survey", Pattern Recognition, Vol. 29(4), pp. 641-662, 1996.
[10] Jairo Rocha, Theo Pavlidis, "Character Recognition Without Segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17(9), pp. 903-909, 1995.
[11] Kirill Safronov, Dr.-Ing. Igor Tchouchenkov, Dr.-Ing Heniz Worn, "Optical Character Recognition Using Optimization Algorithm", Workshop on Computer Science and Information Technology, pp. 1-5, 2007.