Contour-Based Character Extraction from Text Regions of an Image

V.N.Santosh Kumar.Kuppili, Dhanaraj Cheelu, M.Rajasekhara Babu,P.Venkata Krishna
SCSE, VIT University, Vellore, Tamil Nadu, India – 632014

Abstract – Object detection for computer vision is one of the key factors for image understanding. It is still a challenge today to accurately determine an object from an image where similar shaped objects are present in a large number. An object can be defined as a structure within a document that has a particular meaning in the context of the application. An object can be a symbol, character or any disconnected component. The text which usually forms major information provider for the image is one of the crucial events to be made successful. Although there is no generalized algorithm for this problem, this paper helps in understanding the various steps that are taken to recognize the text in an image and the various methods that are implemented in each step to go ahead in the process. Contour based extraction is the crucial step in this process. It also proposes the best method that gives the most appropriate results for a given input set.

Keywords* – *Object Detection, image processing, detection, binarization

I. INTRODUCTION

Vision is the task of "see". It is seeing with understanding other than seeing with camera. When we see things, our eyes (sensing device) capture the image, then pass the information to brain (interpreting device). The brain interprets the image, gives us meanings of what we see. Similarly, in computer vision, camera serves as sensing device, and computer acts as interpreting device to interpret the image the camera captures. Various subjects like Computer vision, Pattern Recognition, Pattern Classification, Machine Learning, and Digital Image Processing help us in these issues at various steps in recognition of text from the images.

Out of the above mentioned domains, Computer Vision is related to many broad areas including biology, psychology, information engineering, physics, maths, computer science and in specific related to areas like pattern recognition, pattern classification, learning machines, digital image processing, object recognition, which is the hardest domain in computer vision research. Defining an object in an image is still an open question in the research field.

There are various steps in the recognition of text from an image. The number of steps differs from process to process. However generally, the binarization of the image is a single first step that is implemented for any process.

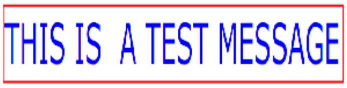

THIS IS A TEST MESSAGE

Fig 1: Original Image

Let us consider the above image to be a test image for this paper and execute various steps to see the output how the output looks like.

II. RELATED WORK

A. BINARIZATION TECHNIQUES

Binarization [1] is an essential module for the preprocessing step in image processing for converting from gray scale image to binary image, which is then used for further processing such as document image analysis. It is a task of converting a gray-scale image to a dual-tone image by using threshold selection techniques to categorize the pixels of an image into either one of the two classes.

Global thresholding techniques [2] can be used to find an adaptive threshold value for the overall image. Otsu's algorithm is generally used for this kind of techniques. The other model, Local thresholding techniques, used to calculate the threshold values locally based on pixel-pixel or region-region basis. Niblack and Sauvola algorithms can be used for this kind of techniques.

Tushar Patnaik [15] et al. presents comparison between three general binarization algorithms – Otsu, Adaptive and Sauvola. Otsu algorithm divides the image into two classes of pixels – black and white based on some threshold. This is applied globally on the image. The threshold is calculated based on the following formula:

$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

Where ω represents the probabilities of the classes separated by a threshold say t, and σ^2 represents the variance.

The second method, adaptive method, extends the former method by dividing the image into N*N blocks and Otsu's method is applied to each block individually. It combines global thresholding with quadratic filter using the local information to fine tune the pixel intensity.

The third method, Sauvola's algorithm is a local algorithm which converts the gray tone document into two tone document based on a local window. A local threshold is calculated for each window and the classification is done based on the threshold of that particular window. This threshold will be an integral value of mean of (1+k/(R-1)) where k represents a constant usually between 0 and 1 and R represents the dynamic range of variance.

SNR and OCR testing is done on an image. That algorithm is called the best which has high SNR ratio and least number of errors in OCR. As inferred from the results in [15], Adaptive algorithm is considered the best among the three tested binarization algorithms.

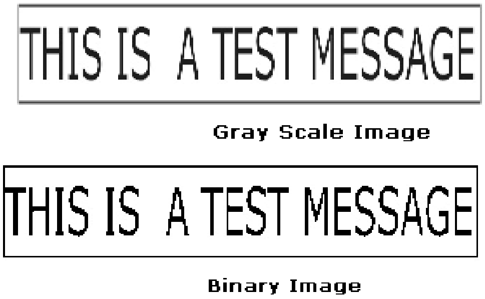
From OCR results, the mean error is displayed in the order as follows:

Otsu > Sauvola > Adaptive

Also based on the various noises being implemented in the SNR performance criteria, the following results are found.

TABLE 1
Best algorithm for a given type of noise

Type of Noise	Best Fit Algorithm
Gaussian	Adaptive
Poisson	Adaptive
Localvar	Adaptive
Speckle	Sauvola



Gray Scale Image

Binary Image

Fig 2: Steps involved in binarizing the image

The above two images display the general procedures for converting the given 3-channel colored input image into a dual tone binary image.

B. STROKING (STROKE WIDTH) FEATURES

Stroking [10] is one of the most important features used to identify the shape of the objects present in the image. Any object, characters in this case, will have certain shape that differentiates it from other objects [3] in the image. This feature can be used to extract the character from the image. This can be done with the help of connectivity of the character with respect to pixels. The stroke width features help to know the curves[1] and edges of each object present in the image and this helps a lot for in case of distractions from the original shape.

C. CLUSTERING AND THRESHOLDING

Any language will contain some basic syntactic rules like words. One of the most basic rules for words is that a word is a combination of characters without any space in between. However, space would exist between the words. Clustering algorithms [5] are used to identify the words from the individual characters that are extracted. The various features based on which the clustering algorithms are used are spacing between the words [7], orientation of the words for a black and white image. Usually thresholding between the characters is considered to cluster the characters within a word. This is one of the most crucial steps in identifying the words that are present in the form of text in an image.

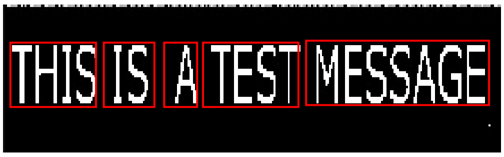


Fig 3: Clustered words in a text image

The above figure shows the example of the clustered image of the given input image.

D. CONNECTED COMPONENT AND REGION BASED METHODS

The Connected Component based methods helps to find the objects present in an image irrespective of the position of the image. Whenever an object is detected in an image, it is taken and processed. There is no specific difference between the parts where the text is present and where it is not. This method will fail whenever the connectivity is lost within the character

Region-based methods[5] distinguish the image in two different divisions. One represents the text region and the other one represents the non-text region. Some behaviors like feature extraction, text contours distinguishes the textual region from the other parts of the image.

E. SEGMENTATION

Segmentation of objects [4] from an image is a very important task in character recognition. The two popular segmentation techniques are Mean shift segmentation and efficient graph based segmentation. Mean shift segmentation is done based on filtration of data followed by the clustering of data. This clustering would result in the segmentation of objects from an image. The latter method, Efficient graph based segmentation works directly on data points without performing filtering step. Adaptive thresholding and single linkage clustering becomes the key to the success of this method.

Caroline et al.[16] in their work compares the afore-mentioned methods with the help Normalized Probabilistic Rand(NPR) Index. The characteristics that were considered to check the efficiency are correctness and stability with respect to parameter and image choice. In all the characteristics, mean

shift algorithm performed significantly better. There is one more algorithm designed, called hybrid algorithm, which is the amalgamation of mean shift filtering followed by graph based segmentation.

The following table demonstrates the results for the performance measure versus various algorithms as follows:

TABLE 2
Comparison of various performance measures with segmentation methods. Smaller the number, the better that algorithm satisfies the performance measure.

	Mean Shift	Graph based	Hybrid
Correctness	2	3	1
Stability [parameter]	2	3	1
Stability [image]	2	3	1

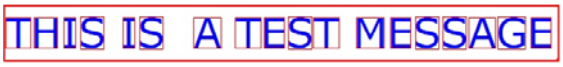


Fig 4: Character extraction from a text image

The above figure displays the individual characters along with the boundaries on each character present in an image.

F. PROJECTION

The concept of projection goes with identifying the individual characters from the given image. Usually, this goes as a supplement to connected component and region based methods for exactly distinguishing one character from the other. Projection [4] helps to break the connected characters to certain extent which helps for the efficiency of the software.

Projection can be done on both axes depending on the orientation of the characters present in the image. This also helps in differentiating each line in the image. This concept depends on parameters like image pixel value, number of channels for the image etc. This would help the connected component analysis in identifying the individual characters to a certain extent by increasing the threshold value.

The below image displays vertical projection of the each word based on the distance between the words in an image.

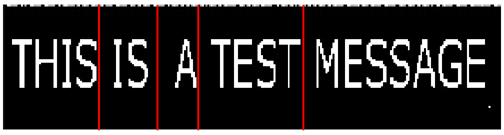


Fig 5: Projection on the input image

G. GRAPHS

Graph based representations are mainly used in image analysis to represent irregular structures. Graphs are used in segmentation, shape matching, fingerprint recognition, etc. Graphs like Attributed neighborhood graphs represent an image as a graph. Attributes are derived from the colors of the pixels. Neighborhood relations are represented by edges. Standard and extended region adjacency graphs (RAG, RAG+) represent the adjacency relations of image regions in any property of the image that we are opting for. In addition to RAGs, RAG+'s keep some self loops and parallel edges to encode inclusion relations (e.g. non-empty self-loops and parallel edges bounding non empty-faces). Sometimes it is not necessary to keep a whole RAG, a spanning tree is enough. From all the spanning trees of a graph, the Minimum Spanning Tree [11] has the minimum sum of edge weights and characterizes the discrepancy of the represented image. Integral features capture important properties of regions. The integral tree [12] is a representation labeling of each vertex of that tree with the integral feature(s) of the sub tree.

Functional Graphical Models [13] (FGM) describe dependence between variables by means of implicit equations. Explicit modeling of functional dependencies by a hyper graph creates a structure well-adapted to information retrieval and processing. The eccentricity transform [14] represents an image by assigning to each point the longest of the shortest paths to any other point of the image. It is robust against Salt & Pepper noise and articulated motion.

H. OPTICAL CHARACTER RECOGNITION

Optical Character Recognition is the oldest data entry technique that involves reading text from paper and translating the images into a form the computer can manipulate. OCR usually reads any image which

contains text and displays the text present in the image. There are two basic methods used for OCR: Matrix matching [6][8] and feature extraction[9]. Of the two ways to recognize characters, matrix matching is the simpler and more common which compares what the OCR scanner sees as a character with a library of character matrices or templates. When an image matches one of these prescribed matrices of dots within a given level of similarity, the computer labels that image as the corresponding ASCII character.

III. BEST PROPOSED METHOD

A.CONTOUR

The extraction of character from an image becomes easy if the boundaries of the character can be found through some algorithm. The boundaries of a character can be found in many ways depending on many factors like the edges of the character, intensity of the text present in the data.

One such method is as follows. This method will be applied on an image after detecting the edges of the character. To start with, the first pixel of a character needs to be identified. This first pixel can be present at any point of the character. This striking of position of first pixel depends on the way we track the input image to extract characters. By default, as the origin of an image lies at the top left corner, the top most pixel of the character will be encountered.

Once the first pixel is identified, the other pixels are identified by following the path along the edge from the first identified pixel. This helps to track along the edge of a character. Based on the fact that the edge detection gives closed path, there is always a chance that we reach the first identified pixel after visiting all other pixels along the edge. This helps as a breaking condition for identifying the boundaries of the tracked character. Once the boundaries of the character are known in the source image, it can be extracted and used for further references.

The below image displays the origin of an image and the arrow in the image denotes the first pixel visited during the scan of the total image. This approach applies for all the characters in the image.

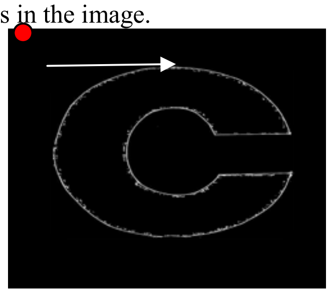
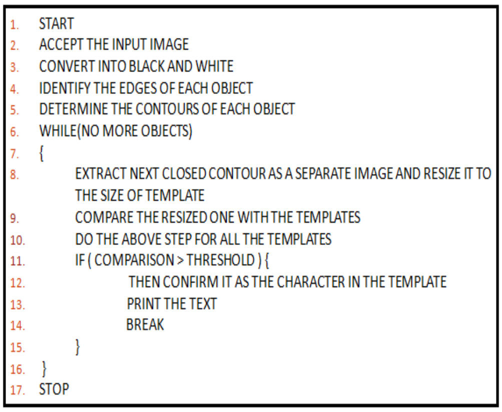


Fig 6: Letter ‘C’ with the origin of the image represented in red circle and arrow indicates the first pixel that is visited.

B.PROPOSED METHOD

For any given image, first convert into binary image using the threshold. Once the binary image is obtained, then obtain the edges for the given image. The edges of each object are found along with the inner data. Now that the inner data increases along with the thickness of the object, the rate of finding the data decreases along with the increase of thickness of the data. Hence, contours are more useful in these types of applications which are always closed and curved and help in finding the boundaries of each and every object present in the given character. The contours help in identifying only the boundary pixels of each and every object present in the image. Now the scanning of the whole image takes place and the boundaries of each and every pixel is found. These boundaries are used to extract the characters from the image. The extracted characters can be used to compare with the templates after resizing the extracted image to the size of the template. The comparison is done usually after certain training which helps to find the nearest data which is available. If the given threshold matches, then the extracted image is said to be in correlation with a specified character.

The general algorithm is as follows:



```
START
ACCEPT THE INPUT IMAGE
CONVERT INTO BLACK AND WHITE
IDENTIFY THE EDGES OF EACH OBJECT
DETERMINE THE CONTOURS OF EACH OBJECT
WHILE(NO MORE OBJECTS)
{
    EXTRACT NEXT CLOSED CONTOUR AS A SEPARATE IMAGE AND RESIZE IT TO
    THE SIZE OF TEMPLATE
    COMPARE THE RESIZED ONE WITH THE TEMPLATES
    DO THE ABOVE STEP FOR ALL THE TEMPLATES
    IF ( COMPARISON > THRESHOLD ) {
            THEN CONFIRM IT AS THE CHARACTER IN THE TEMPLATE
            PRINT THE TEXT
            BREAK
    }
}
STOP
```

Fig 7: Our proposed algorithm that describes the basic steps

IV. IMPLEMENTATION ISSUES

From the Figure 8, it is clear that the while loop would take the order of the number of variables. But the identification of contours would make us travel along the width and height of the image making the time complexity of the algorithm to go as high as the square of the input to the algorithm.

Hence, the time complexity of the algorithm is **O(n^2)**, where n is the input size given to the algorithm in the form of width and height of the image. If they are different, the time complexity is considered as **O(width*height)**.

V. PERFORMANCE ISSUES

The comparison between the old and our newly proposed method is as follows:

TABLE 3

Comparison of existing methods with our proposed method

OLD METHOD	OUR METHOD
Whole solid character needs to be visited	Boundary of the character is visited
Recursive neighborhood check even if visited once	One visit to a pixel is enough
Complete character is visited	Boundaries are only visited
Backtracking needs to be done	No need of backtracking

The boundary pixels in the character is visited in the following order.

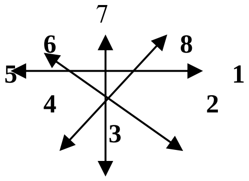


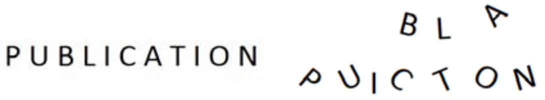
Fig 8: Order of Orientation of consideration for next adjacent pixel from the current pixel if more than one adjacency exists

The above mentioned orientation helps to identify the only the edges of the characters that are present in an image. A pre-defined orientation also helps to find the boundaries and hence the contours of each and every character of an image.

VI. RESULTS

Our contour based character extraction from text regions in an image provides an edge over the previous techniques in terms of time and space complexities. Our algorithm not only completes faster by traversing only through the edges but also completes in O(n^2) time.

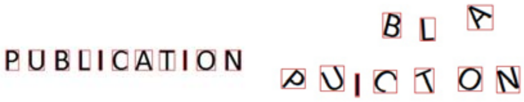
The results for our algorithm are presented here. Let us consider the following two images – one which contains the regular order in the text style and the other irregular style.



(a) (b)

Fig 9: Two input images - (a) with regular structure and (b) with irregular structure

When our algorithm is applied to the above inputs, all the characters get extracted individually and saved as a separate image. In order to show the output here, we have consolidated the outputs and enclosed each identified character in a red colored rectangle. Each rectangle resembles one character extracted from an image.



(a) (b)

Figure 10: Two output images – (a) with regular structure and (b) with irregular structure

VII. CONCLUSION

The number of methods for each step and the number of steps for the recognition of text from images provided in this paper are only a means of selecting the procedure from the many available. Still, a generalized method for the above mentioned process doesn't exist. Although the above process is limited to certain fonts, the same is the case for the handwritten formats also. It needs a lot of work to be done in this area for developing better algorithms and training the systems in a better and much efficient way.

REFERENCES

[1]Huy Phat Le et al. "Text Detection in Binarized Text Images of Korean Signboard by Stroke Width Feature",10th IEEE International conference on Computer and Informational Technology, 2010, Page(s): 1588 – 1592

[2]Anh-Nga Lai et al. "Binarization by local K-means clustering for Korean Text Extraction", IEEE International Symposium on Signal Processing and Information Technology, 2008. ISSPIT 2008.Page(s): 117 - 122

[3]Yangxing LIU et al. "A Robust Algorithm for Text Detection in Color Images", Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition, 2005 IEEE.

[4]Rodolfo P.dos Santos et al "Text Line Segmentation Based on Morphology and Histogram Projection", Tenth International Conference on Document Analysis and Recognition, 2009 IEEE, Page(s): 651 – 655

[5]Manoj Kumar et al. "Automatic Text Location from Complex Natural Scene Images", Second International Conference on Computer and Automation Engineering, 2010,Page(s): 594 – 597

[6]Xiaojun Li et al. "Fast and Effective Text Detection", Fifteenth IEEE International Conference on Image Processing, 2008,Page(s): 969 – 972

[7]Yi-Feng Pan et al. "A Robust System to Detect and Localize Texts in Natural Scene Images", Eight IAPR Workshop on Document Analysis Systems, 2008 IEEE,Page(s): 35 – 41

[8]Tabassam Nawaz et al. "Optical Character Recognition System for Urdu using Pattern Matching Technique", International Journal of Image Processing, voulme 3, issue 3, 2008, Page(s): 92 – 104

[9]Kirill Safronov et al. "Optical Character Recognition using Optimization Algorithms", Proceedings of the Ninth International Workshop on Computer Science and Information Technologies, Russia, 2007, Page(s): 1 – 5

[10]Sung-Hyuk Cha et al. "Approximate Stroke Sequence String Matching Algorithm for Character Recognition and Analysis", Proceedings of the Fifth International Conference on Document Analysis and Recognition, 1999. ICDAR '99. Page(s): 53 – 56

[11] Y. Haxhimusa and W. G. Kropatsch. Segmentation graph hierarchies. In Proceedings of Joint International Workshops on Structural, Syntactic, and Statistical Pattern Recognition S+SSPR 2004, pages 343 – 351, 2004.

[12] W. G. Kropatsch, Y. Haxhimusa, and Z. Pizlo. Integral trees: Subtree depth and diameter. In Proceedings of the IWCIA 2004, pages 77 – 87, 2004.

[13] J. Marchadier and W. G. Kropatsch. Functional modeling of structures images. In 4th IAPR-TC15 Workshop on Graph-Based Representation in Pattern Recognition, page 12, 2003.

[14] W. G. Kropatsch, A. Ion, Y. Haxhimusa, and T. Flanitzer. The eccentricity transform (of a digital shape). In Discrete Geometry for Computer Imagery, pages 437 – 448, Szeged, Ungarn, 2006.

[15] Tushar Patnail, Shalu Gupta, Deepak Arya, “Comparison of Binarization algorithm in Indian Language OCR”, Proceedings of ASCNT, CDAC, Noida, India, pp-61-69, 2010.

[16] Caroline Pantofaru, Martial Hebert, “A Comparison of image segmentation algorithms”, The Robotics Institute, Carnegie Mellon University, 2005.