THINK INDIA JOURNAL

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Detecting Gurmukhi and Hindi Text Object Present In Images

Rahul Malik[1], Nishi Malik[2]
[1]Dept. of CSE, LOVELY PROFESSIONAL UNIVERSITY
[2]Dept. of CSE, DAV UNIVERSITY
Email id.: Rahul.23360@lpu.co.in

ABSTRACT

This paper resolves the trouble needed in detecting Gurmukhi and Hindi text object present in images. As there are lots of differences among the features of script in English and Indian languages (for example line over text string), we cannot directly apply the existing algorithms. Therefore in this paper we propose a new region based bottom up method to extract text from images, where we first identify elementary substructures using connected component and edges, and then merge them successively into huge buildings until all book areas are recognized. We localized text candidates by extracting closed boundaries. As each character counter has high contrast as compared to its neighbors, all character pixels and several non-character pixels which exhibit excessive neighborhood intensity difference are localized as written text within the advantage image. For every localized unit we compute a set of geometric properties (based on height width and area) and filter out non text regions. Although a lot of noise is removed by this step a second pass filter is applied based on stroke width. Stroke width of Hindi and Gurmukhi characters are approximately uniform. The candidates with similar properties are aggregated into chains, according to the observation that the figures of a book type are aimed along a particular path with similar color, stroke, and size width. For performance evaluation we created our own dataset containing 60 images of varied size, resolution and type. The performance is evaluated on this dataset containing caption, document and scene text images using different performance metrics viz., Precision, Recall and F-Score.

Keywords: Gurmukhi, Precision, Recall

1. Introduction

With the increasing number of electronic multimedia libraries, huge quantity of reputation and video information are available today. Consider just about the most popular picture sharing services Flickr being an example, a few enormous amounts of images have been accumulated by it. Yet another example is Youtube, which happens to be a video sharing Site which is hosting enormous amounts of videos. As the biggest picture sharing website, Facebook currently stores thousands of hundreds of enormous amounts of photos. These multimedia entities doesn't appear in isolation but comes with different types of textual info [32]. The requirement to effectively index, browse and also retrieve multimedia info is growing rapidly. A option is using the textual info lodged in the multimedia entities. This provides information that is important for multimedia data understanding plus is an extremely great entity for indexing, querying and retrieving multimedia data. In this particular context, textual content removal has acquired a lot of interest in the previous years in applications that are several. Jung et al. [8] distinguish between four major classes of applications: a)Page segmentation (which include news newspaper statements extraction) b)Address obstruct place c)License plate area d)Content - based image/video indexing.

2. Characteristics of Text

Text in pictures are able to differ with regard to following properties [8].

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

2.1 Content-Based Indexing

Content based image indexing talks about the process of attaching labels and signatures to pictures based on the information of theirs and extracted qualities, whereas content based image retrieval (CBIR) is targeted towards the real retrieval of relevant images from large image databases, based on the attached labels. Attributes will be classified into two main groups: perceptual (low level) and semantic (high level) characteristics. Perceptual features include characteristics like color, severeness, form, feel, and also their temporal modifications, whereas semantic capabilities explain the current objects (for example cars, events, etc.), faces, text, and the relations of theirs [10].

Table 1: Different properties of text.

Property		**Variants and Sub Classes**
Geometry	Size	Regularity in size of text
	Alignment	Horizontal/vertical Straight line with skew (implies vertical direction) Curves
	Strokes	Different Stroke density and statistics
	Inter Character Distance	Aggregation of characters with uniform distance
Color		Gray
		Color (monochrome, polychrome)
Edges		Strong edges (contrast) at text boundaries
Background		The background can be different

As compared to the reduced level functions and also to many other high level characteristics, the content supplies better and much more apparent info about the content of particular multimedia. It's a great supply of info around the information in a picture. For instance synthetic copy in news videos frequently offers info about the title of relevant individuals, location, topic of debate, time and date when an event has occurred. Artificial text usually provides an abstract of the program and it's frequently unavailable in media that are other such as the sound track. Credits and titles shown initially or maybe end of films offer info as names of actors, contributors, producers, and more. Captions in sports activities programs usually have the names on the teams, scores, players, and more. Text info may in addition be discovered in the arena, for e.g. the player 's name and number, the specific staff, brand names, commercials and location. Text about places, heat as well as certificate items are contained by displayed maps, tables and figures in videos. Titles, logos, and labeling of applications shown in videos are important for any annotation with regard to system variations along with models. Furthermore, much many websites choose text photos to enhance the appearance of the net presentations.

Content-Based reputation retrieval has emerged as a crucial location in computer vision and also multimedia computing. Numerous businesses have large video and image collections (programs, news sections, video games, art) in electronic format, offered for online access. Organizing these libraries into types and also providing good indexing is important for real time browsing and retrieval [4].

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

3. Text Extraction from Images

Text Detection Algorithm

In this section we present the flowchart of proposed framework. The algorithm is based on bottom up approach, where we first identify elementary substructures using connected component and edges, and then merge them successively into huge buildings until all book areas are detected. First we concentrate on the excessive contrast between the background and the text and use it to identify the edges caused from the book contours. Further, connected component analysis is done to find and group the basic pixel elements using the similarity of neighbour pixels in the binary edge image. Next, filtering of non-text region is done based on uniform stroke width and geometric properties of text.

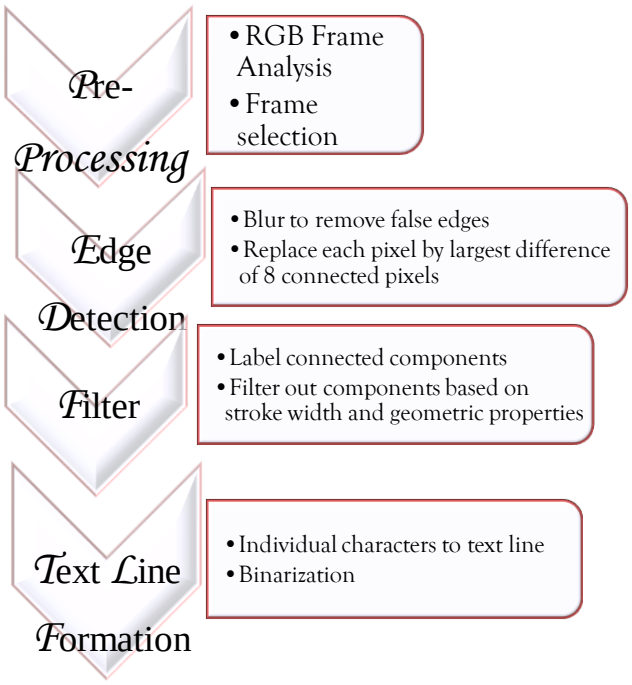


Figure 1: Steps of Text Detection

4. Steps of Text Detection Algorithm

4.1 Preprocessing

In the preprocessing step we use RGB frame analysis to convert color image into grayscale image. If the image is color image, Red, Green and Blue frame are extracted and connected component analysis is done on each frame. If the image is grayscale image we directly go for edge detection (step 2). This technique was discussed by Dr.M.Sundaresan [31] for text extraction from e-comic images. This step removes the unnecessary connected components while keeping the useful textual information.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

The pseudo code is presented below in Algorithm 1. The color image is divided into RGB band for band selection. Based on minimum number of connected regions the appropriate frame is then selected for further processing.

Algorithm 1: Pseudo code for componenet frame analysis
Procedure*: Frame Analysis (Input Image)*
Input*: I/P = n × m input image (I_img)*
Output: *O/P = n × m grayscale image*
Step 1: Check whether the image entered is color or grayscale image.
If (dimension > 3 || dimension<2)
Print: Error!! Input image should be 2D grayscale or 3D color image.
End (**Step 1** if condition)
Step 2: If the image is a grayscale image, directly perform edge detection.
If (dimension = = 2)
Call Edge Detection (I/P)
End (**Step 2** if condition)
Step 3: For color image component frame analysis is done. The image is divided into R, G and B frame and the frame with minimum number of connected components is used for further processing.
If (dimension = = 3)
Divide input image (I/P) into Red, Green and Blue frame.
R=I_img(:,:,1);
G=I_img(:,:,2);
B=I_img(:,:,3);
Frame_Img =Select appropriate frame using minimum number of connected component in each frame
Call Edge Detection (Frame_Img)
End (**Step 3** if condition)

4.2 Edge Detection

Boundaries are characterized by edges and are therefore an issue of essential value in image processing. Image Edge detection greatly lowers the quantity of filter and data out information that is useless, while protecting the necessary structural specifics in an image. In this step we focus our attention to edge detection which is the primary step for any region based text extraction method.

The well knownCanny edge detection algorithm [1] is a multistage algorithm developed by John F. Canny in 1986. It finds the image gradient to highlight regions with high spatial derivatives. Despite its good results the algorithm has few drawbacks [21] [22]. The canny edge detection algorithm depends on a number of adjustable parameters (standard deviation for Gaussian air filter and two threshold values T1 as well as T2). The larger the importance for standard deviation (sigma) the bigger the dimensions of the Gaussian air filter becomes. This means additional blurring, needed for noisy pictures, along with detecting bigger edges. As normal, nonetheless, the bigger the machine of the Gaussian, the much less correct will be the localization of the advantage. A scaled-down Gaussian filtration system that restricts the quantity of blurring, maintaining finer tips in the picture is implied by smaller values of sigma. The person is able to customize the algorithm by setting these variables to conform to various environments [21].

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Since it is a gradient based approach, large intensity gradients are much more apt to match to tips than tiny intensity gradients. It's most certainly not possible to establish a threshold at what a certain intensity gradient changes from corresponding experience to an advantage. So canny uses two thresholds high and low (T1 nd T2). High threshold allows discarding few noisy pixels that do not constitutes an edge while low threshold allows having faints sections of edges. Despite having two thresholds the general problem of threshold approach is still there. A threshold set way too high can miss information that is important. On another hand, a threshold started absurdly small will incorrectly identify info that's irrelevant (such as noise) as important. It is hard to make a generic threshold that is effective on many images. No tried as well as tested method of this particular issue yet exists [22].

As discussed above the canny algorithm require few adjustable details, that could affect the computation time and usefulness of the algorithm. We show an easy algorithm to transform the grayscale picture into an advantage image. The algorithm is based on pixel connectivity (the edges are nothing but set of connected pixels) and character contours having huge contrast to their neighborhood friends. As an outcome, most character pixels that show high nearby intensity contrast are authorized in the advantage picture.

The pseudo code of our method is described below in algorithm 2. The value of each pixel in the frame image (Frame_Img generated in previous step) is replaced by the largest difference of the intensity values of given pixel with its 8 connected neighbors (North-East, North, North-West and West). Based on connected component we localized connected regions in the grayscale image. For every component we compute a set of geometric properties (based on height width and area) and stroke width to filter out non text regions.

Algorithm 2: Pseudo code for edge detection.

Procedure: *Edge Detection (Frame_Img)*
Input: *n × m frame image*
Output: *Image_Edge = n × m binary edge image*
Step 1: Initialization of local variables.
X ← 0
Y ← 0
Left ← 0
Upper ← 0
Rightupper ← 0
Leftupper ← 0
End (**Step 1**)

Step 2: Given pixel is replaced with the largest difference of North-East, North, North-West and West neighbor pixels.
For all pixels (x, y) € Frame_Img
Left = (bit(x, y) - bit(x, y-1))
Upper = (bit(x, y) - bit(x-1, y))
Rightupper = (bit(x, y) - bit(x-1, y+1))
Leftupper = (bit(x, y) - bit(x-1, y-1))
Image_Edge = max (Left, Upper, Rightupper, Leftupper)
End (**Step 2** for loop)

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

4.3 Filter

Many components generated by edge detection are not part of text. This stage identifies and filters out these components. Before moving to next stage we perform the close operation with a 3 by 3 structuring element to reduce the number of connected components. After the close operations, all connected components are screened with their size (height / width), and area information. We perform two pass filtering based on geometric properties and stroke width transform. Based on tesseract [11] we choose to remove bounding box with less than 20 pixels. In addition we use following geometric constraint to filter out non text regions.

1) Width of text region < 0.5 image width
2) Height of text region < 0.5 image height
3) Width of box < 0.5 height of box
4) Filter out height < 30 pixel
5) Filter out area < 20 pixels

As discussed in previous, an important property of text is that they have uniform (constant) stroke width (as shown in figure below). In second pass we utilize this important property to filter out remaining non text regions.

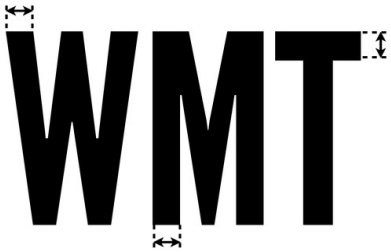


Figure 2: Stroke width of characters.

Algorithm 3: Pseudo code for stroke width filtering.

Procedure*: Stroke Width Filter*
Input: *Image_Edge = binary edge image*
Output: *Filter_Img = after stroke width filtering*
Step 1: Assign each pixel a number that is the distance (using Euclidean distance transform) between that pixel and its nearest non zero pixel.
Dist: =bwdist(Image_Edge)
For pixel(x, y) € foreground
p: = Dist(x, y)
LookUp(x, y) = 8 neighbor of (x, y) having value < p
End (**Step 1** for loop)

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Step 2: Label similar stroke width value and assign the same stroke width value to the 8 neighbors of that pixels having distance less than that particular stroke value.
MaxStroke: = max(Dist)
For Stroke = MaxStroke to 1
StrokeIndex: = find(Dist == Stroke)
For s = 1: numel(strokeIndex)
NeighborIndex: = Lookup(StrokeIndex(s))
While NeighborIndex not empty do
Dist(NeighborIndex : = Stroke
NeighborIndex: = Lookup(NeighborIndex)
End (**Step 2** while loop*)*
End (**Step 2** for loop)
End (**Step 2** for loop)
Filter_Img := Dist

Step 3: Discard objects where standard deviation of stroke width is more than half of its mean.
R: = find(std(Filter_Img) / mean(Filter_Img) > 0.5)
Filter_Img:= Reject(R)
End (**Step 3**)

The pseudo code is discussed above in algorithm 3. The second pass filter is applied based on stroke width transform as discussed by Epshtine et.al [19]. Using Euclidean distance transform, we generate the stroke width photograph of the identical verdict as the original picture with the stroke width great characterize for every pixel. Next we removed the connected components whose standard deviation to mean stroke width ratio was greater than 0.5 (std/mean > 0.5) as discussed in [27].

4.4 Text Line Formation

The remaining elements are considered as character candidates. Text lines are important cues because of the existence of text, as copy often show up in the form of slight curves or straight lines. In order to identify these lines, in this particular phase we attempt to aggregate persona applicants into chains. The initial step of the book line formation phase is linking the persona applicants into pairs. 2 adjacent applicants are classified into a pair in case they've similar geometric color and properties. Among the crucial qualities of text is it seems like in linear form. If 2 candidates have very similar stroke widths (ratio between the hostile stroke widths is under 2.0), identical sizes of the shoes, and are in close proximity enough (characters within a word often a consistent distance between them), they are labeled as a pair.

This point also can serve as a filtering stage since the candidate characters that can't be connected into chains are considered as elements casually created by noises or maybe background clutters, and therefore are thus abandoned.

5. Text Extraction Results

Most of the image processing applications and techniques require scanning a large number of pixels. Storage of this type of data is also one of the most important concerns. These values are stored in the

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

form of two dimension and three dimension matrices for grayscale and color image respectively. The proposed algorithm is implemented in MATLAB 2012 using MATLAB programming language (as it provides some predefined image processing functions) and run on a system with Intel Core 2 duo CPU at 2.83GHz and 2 GB memory under Microsoft Windows 7. The decapitation time for an image can vary from 19 to 40 seconds based on the dimensions of the picture as well as number of text string in the image.

We did comprehensive experiments using Gurmukhi and Hindi text image datasets in order to assess the recommended text detection algorithm. In this chapter we first introduce the two (Gurmukhi and Hindi) datasets and some sample images used to evaluate the proposed algorithm.

5.1 Dataset of Gurmukhi and Hindi Text Images

In this particular area, two datasets are introduced by us and even several of the sample pictures in the datasets. In order to assess the functionality of the method of ours for removing texts from scene pictures, 60 images were used by us (thirty containing Gurmukhi and various other thirty with Hindi text) in the experiment of ours. All of the pictures had been grabbed from web. Graycolor and scale cover pictures of books/journals/magazines, color chart, billboard are utilized in our experimentation to exhibit the effectiveness of the approach of ours. The text present is in different languages (Hindi and Gurmukhi), fonts, sizes, colors and orientations. We focused on these types of images because they have many variations.

We divide both groups of images (containing Hindi and Gurmukhi text) into three sets, each containing 10 images. Each set contains a set of JPEG and PNG images. The resolution of the images can vary from 194 × 263 to 1276 × 1600. Some typical photograph from this dataset are shown below in Figure 3.

Figure 3: Photograph from dataset

5.2 Performance Evaluation

The huge discrepancy between the normal copy output and the specific text output of the service are being measured by the goal of performance evaluation for a book extraction device. Good performance evaluation techniques are able to provide info that's valuable not only for method calculation and correlation, but in addition for technique choice and improvement. Recall as well as precision are familiar evaluation metrics in information retrieval community. Generally, precision is labeled as the level of suitable estimates split through the total amount of estimates as well as recall is referred to as the quantity of proper estimates divided by total volume of surfaces truths.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

We calculate *Precision Recall* and *F-Score* [9] to analyze our algorithm. These metrics are defined as.

$$Precision = \frac{Correctly\ Detected\ Character}{Correctly\ Detected\ Character\ +\ False\ Positives}$$

$$Recall = \frac{Correctly\ Detected\ Character}{Correctly\ Detected\ Character\ +\ False\ Negatives}$$

$$F - Score\ = \frac{Precision\ \times\ Recall}{Precision\ +\ Recall} \times 2$$

*False Positives*Although have already been recognized by the algorithm as text, are those areas within the picture that are really not heroes of a book.

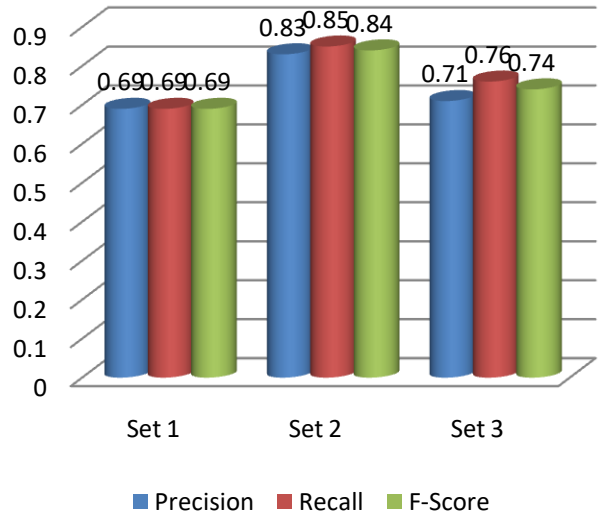
*False Negatives*Although haven't been recognized by the algorithm, are those areas within the picture that are really text characters.

6. Experimental Results

When applied to Hindi text images following results are obtained.

Table 2: Different measures evaluated on Hindi text images.

Measures	Total Chracters	Correctly Detected Characters	False Positives	False Negatives	Accuracy
Set I	29	20	9	9	0.69
Set II	83	71	14	12	0.85
Set III	58	44	18	14	0.80
Total	170	135	41	35	0.79



ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Figure 4: Graph showing *Precision Recall and F-Score* for three sets of Hindi text images.

When applied to Gurmukhi text images following results are obtained.

Table 3: Different measures evaluated on Gurmukhi text images.

Measures	Total Chracters	Correctly Detected Characters	False Positives	False Negatives	Accuracy
Set I	110	83	34	27	0.75
Set II	93	57	46	36	0.61
Set III	89	76	13	13	0.85
Total	292	216	93	76	0.73

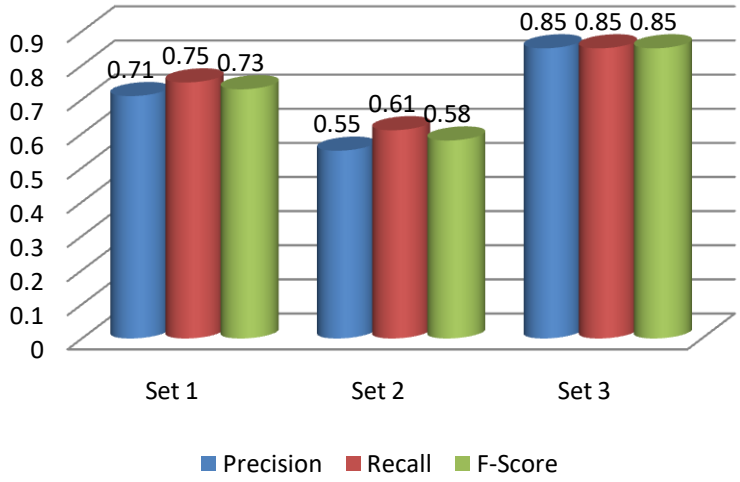


Figure 5: Graph showing *Precision Recall and F-Score* for three sets of Gurmukhi text images.

Overall performance measures of proposed algorithm are formulated below:

Table 4: Table showing overall *Precision, Recall and F-Score*.

Measures	Image Type	Containing Hindi Text	Containing Gurmukhi Text

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Precision	0.73	0.70
Recall	0.76	0.73
F-Score	0.74	0.71

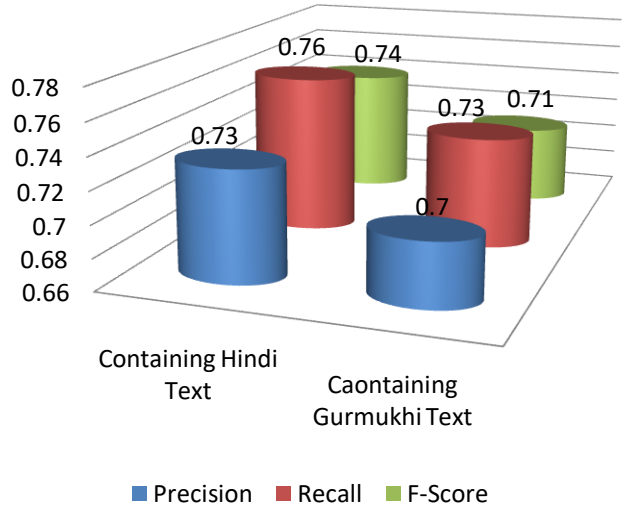


Figure 6: Graph showing overall *precision Recall and F-Score.*

We also compare our algorithm with L. Neumann and J. Mattas real time scene text detection algorithm [34]. The algorithm is designed for identification of English text from images. Results below show that the work [34] cannot be applied as such to Indian script.

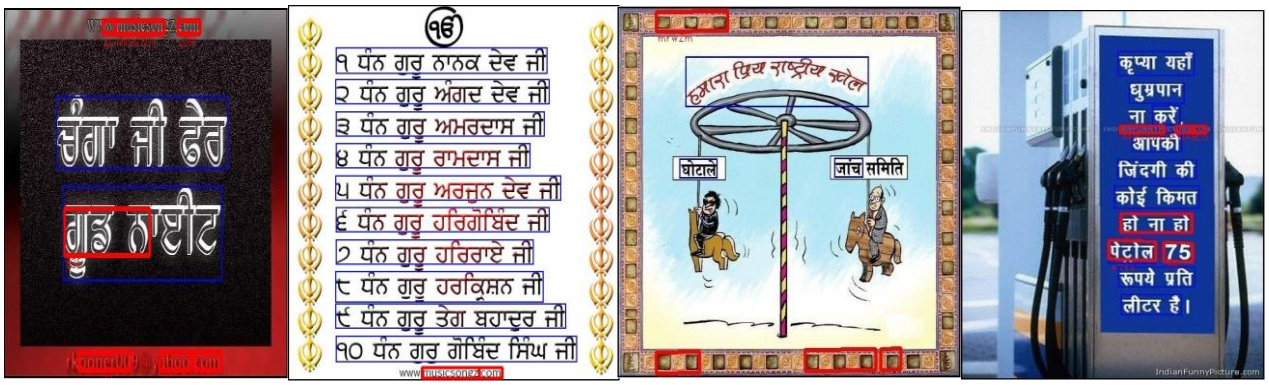


Figure 7: Comparisons with Neumann [34] method. Red boxes show the text detected by [34] while the blue boxes are detected by our algorithm.

The graph below shows the *Precision* and *Recall* comparison of our algorithm with Neumann's method. From the graph in Figure 4.6 it is evident that the performance of Neumann's method degrades when applied to our dataset (containing Gurmukhi& Hindi text). Neumann's method gives the *recall* of 34% and *precision* of 27% when applied to our dataset in comparison to the *recall* of

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

64.7 % and the *precision* of 73.1% when applied on the ICDAR dataset. The results clearly depict the degraded performance of this method.

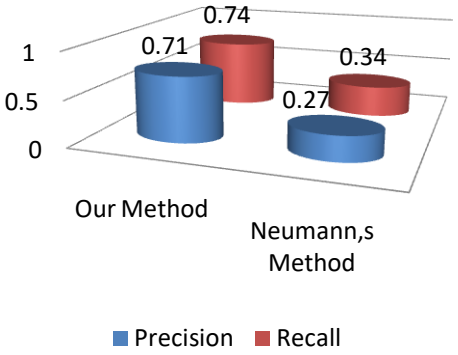


Figure 8: Graph showing Comparison with Neumann's method.

Further, as discussed in literature review only a few methods for text detection from Indian languages have been developed. Table 4.4 below shows the comparison of algorithm [28] designed for Gurmukhi text extraction with our proposed algorithm.

Table 5: Comparison with [28].

Methods / Measure	Hindi Text Extraction using Our Method	Gurmukhi Text Extraction using Our Method	Gurmukhi Text Extraction using SVM [28]
Accuracy	0.79	0.73	0.70

6.1 Text Extraction Examples

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Figure 9: Text detection results on several images from the test set.

7. Conclusion

Thus, many techniques for the extraction along with analysis of textual info in pictures plus movies have been proposed. In this dissertation we proposed a brand new textual content removal algorithm for the book mixed document and scene text images. This particular algorithm is insensitive to text direction the result on the book eradication algorithm could be given to an OCR phone structure to understand the encompass info. The results obtained on mixed set of pictures are as opposed with respect to accuracy and recall rates. Assumption is made that pictures are of quality that is good and also free of noise. If no such presumption is created then many pre-processing methods are offered which may be utilized.

REFERENCES

[1] Canny, John, "A Computational Approach to Edge Detection," Pattern Analysis and Machine Intelligence, *IEEE Transactions on*, vol.PAMI-8, no.6, pp.679, 698, Nov. 1986

[2] Sato. T, Kanade. T, Hughes, E. K. Smith, M.A "Video OCR for digital news archive," Content-Based Access of Image and Video Database, Proceedings, *IEEE International Workshop on*, vol., no., pp.52,60, 3 Jan 1998

[3] D. Chen, K. Shearer, and H. Bourlard. "Text Enhancement with Asymmetric Filter for Video OCR," poceedings of the International Conference on Image Analysis and Processing, pages192–197. *IEEE Computer Society*, 2001

[4] Vailaya. A, Figueiredo. M. A. T, Jain. A. K, Hong-Jiang Zhang, "Image classification for content-based indexing," *Image Processing, IEEE Transactions on*, vol.10, no.1, pp.117, 130, Jan 2001

[5] Ying Zhang, Bing Zhao, JieYang and Alex Waibel"Automatic Sign Translation," *Proceedings of ICSLP2002*, Denver, September, 2002.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

[6] Digital Image Processing Using Matlab by Rafael C. Gonzalez, Richard Richard Eugene Woods, Steven L. Eddins, published by *Pearson Education*, 2004

[7] Xiangrong Chen, Alan L. Yuille, "Detecting and Reading Text in Natural Scenes," *CVPR* (2), : 366-373, 2004

[8] K. Jung, K.I. Kim, and A.K. Jain. "Text Information Extraction in Images and Videos: A Survey," *Pattern Recognition Letters* 37:977–997, 2004.

[9] Cyril Goutte, Eric Gaussier "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation,"*Proceedings of the 27th European Conference on Information Retrieval Springer* 2005.

[10] Michael S. Lew NicuSebe , ChabaneDjeraba , Ramesh Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Transactions on Multimedia Computing, Communications, and Applications* (TOMCCAP), v.2 n.1, p.1-19, February 2006

[11] Smith, R., "An Overview of the Tesseract OCR Engine," *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on* , vol.2, no., pp.629,633, 23-26 Sept. 2007

[12] Datta. R., Joshi. D, Li. J, and Wang, J. Z. "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Survey.* 40, 2, Article 5 (April 2008), 60 pages

[13] Srivastav, A, Kumar. J, "Text detection in scene images using stroke width and nearest-neighbor constraints", *TENCON 2008 - 2008 IEEE Region* 10 Conference, 2009.

[14] Jui-Chen, WuJun-Wei, Hsieh, Yung- ShengChen, "Morphology-basedtext line extraction" *Machine Vision and Applications, 2008.*

[15] Digital Image Processing, by Gonzalez Rafael C: Woods Richards E, published by *Pearson Education*, 2008

[16] Leon.M,Vilaplana. V,Gasull,A.and Marques. F, "Caption text extraction for indexingpurposes usinga hierarchical region-based imagemodel," IEEE *ICIP 2009,El Cairo, Egypt*, 2009.

[17] J. Fabrizio, M. Cord, And B. Marcotegui(2009), "Text Extraction From Street Level Images,",*CMRT*, Vol. Xxxviii, Part 3/W4 , pp. 199–204, 2009.

[18] Chu Duc Nguyen, Mohsen Ardabilian and Liming Chen,"Robust Car License Plate Localization using a Novel Texture Descriptor", *Advanced Video and Signal Based Surveillance, 2009.*

[19] Epshtein, B.; Ofek, E.; Wexler, Y., "Detecting text in natural scenes with stroke width transform," Computer Vision and Pattern Recognition (CVPR), *2010 IEEE Conference on* , vol., no., pp.2963,2970, 13-18 June 2010

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

[20] XinZhang, FuchunSun, LeiGu,"ACombined Algorithm forVideo Text Extraction",*2010SeventhInternational Conferenceon Fuzzy Systems and Knowledge Discovery*,2010.

[21] R Maini, H Aggarwal - "Study and comparison of various image edge detection techniques" , International Journal of **Image** Processing, 2009.

[22] Canny Edge Detection - http://en.wikipedia.org/wiki/Canny_edge_detector.

[23] H. Chen, S. Tsai, G. Schroth, D. Chen, R. Grzeszczuk, and B. Girod. "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," *IEEE International Conference on Image Processing (ICIP*), September 2011.

[24] DavodZaravi, HabibRostami, AlirezaMalahzaheh, S.S Mortazavi," Journals Subheadlines Text Extraction Using Wavelet Thresholding And New Projection Profile", *WorldAcademy Of Science, Engineering And Technology* .Issue 73,2011.

[25] ChucaiYi ,YingLiTian, "Text String Detection From Natural Scenes by Structure-Based Partition and Grouping," *IEEE Transactions on Image Processing*, v.20 n.9, p.2594-2605, September 2011.

[26] Petter, M.; Fragoso, V.; Turk, M.; Baur, Charles "Automatic text detection for mobile augmented reality translation", Computer Vision Workshops (ICCV Workshops), 2011 *IEEE International Conference on*,On page(s): 48 – 55.

[27] Chen, H.; Tsai, S.S.; Schroth, G.; Chen, D.M.; Grzeszczuk, R.; Girod, B. "Robust text detection in natural images with edge-enhanced Maximally Stable Extremal Regions", Image Processing (ICIP), 2011 18th *IEEE International Conference* on, On page(s): 2609 – 2612.

[28] Sukhwinderkaur and Gurpreetsinghjosan,"Gurmukhi text extraction From image using Support vector machine (2011)" *International Journal of Engineering Science and Technology* (IJEST).

[29] Cong Yao; Xiang Bai; Wenyu Liu; Yi Ma; ZhuowenTu, "Detecting texts of arbitrary orientations in natural images," Computer Vision and Pattern Recognition (CVPR), 2012 *IEEE Conference on* , vol., no., pp.1083,1090, 16-21 June 2012.

[30] Uddin, M.S.; Sultana, M.; Rahman, T.; Busra, U.S., "Extraction of texts from a scene Image using morphology based approach," *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on* , vol., no., pp.876,880, 18-19 May 2012.

[31] Sundaresan, M.; Ranjini, S. "Text extraction from digital English comic image using two blobs extraction method", *Pattern Recognition, Informatics and Medical Engineering* (PRIME), 2012.

[32] Zha, Zheng-JunWang, MengShen, JialieChua, Tat-Seng "Text Mining in Multimedia" Multimedia Surrounding Text Tagging Social Network," *Springer US*, 2012.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

[33] ParagKulkarni, AmitKhandebharad, DattatrayKhope, Pramod U. Chavan "License Plate Recognition: A Review" *IEEE- Fourth International Conference on Advanced Computing*, ICoAC 2012 MIT, Anna University, Chennai. December 13-15, 2012.

[34] Neumann, L.; Matas, J., "Real-time scene text localization and recognition," *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* , vol., no., pp.3538,3545, 16-21 June 2012.