Text and Drawing Extraction From Digital Documents

Yashashree U. Borse , Samruddhi V. Aher, Aarti R. Dhikale and Nikita S. Wani

***Abstract*— In modern technology, availability of high resolution camera has lead to a new dimension in digital image processing. As the technology is expanding at a very high speed, various technologies are emerging. The goal of our project is to recognize and extract text and drawing from digital images and once text and image is recognized and separated information about image can be obtain by using user built dataset. We implement two stage framework to detect and separate text from image. Initially we pre-process the image using basic image processing algorithms, using OCR we detect text and apply inpainting method. Using feature extraction and image retrieval algorithms we can use this project to perform various types of searches.**

***Index Terms*— Inpainting, Binarization, Optical Character Recognization, Text Detection**

I. INTRODUCTION

Grouping documents into regions of different content plays a powerful role in human visual perception . For humans it is easy, but for computers reliable and efficient content based segmentation is a great challenge. Page layout analysis is a fundamental step of any document image understanding system. This method separates text from drawings in digital documents. In the first stage we utilize a binarized version of the document to detect and extract text. In the second stage we remove the text from the document and then separate drawings from other classes, e.g., background and noise.

To carry out this task the basic stages for the application of project:

1. Pre-processing: Image is pre-processed using grayscale and binarization algorithms.
2. Text Extraction: To detect proper boundary and regions of text.
3. Text Removal: To remove the texts using inpainting.
4. Feature Extraction: Features are extracted and stored for searching process.
5. Search by Options: We provide options to search similar images and text searching.

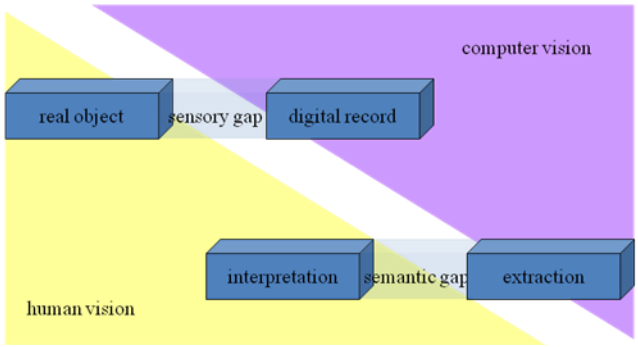

Fig. 1 Goal of our proposed system.

II. PROBLEM STATEMENT

In modern technology, availability of high resolution camera has lead to new dimension in digital image processing. As the technology is being expanding, various
technologies are being developed. The goal of our project is to recognize and extract the image from digital images, and once the text and image is recognized and separated
information about the image can be obtain.

There are many technologies available for searching, but they have following limitations:

1. They are costly.
2. Inappropriate data is given out as output.
3. Requires high number of features and huge dataset.

Focusing on drawbacks and inadequacies of existing process, definitely there is a need of an efficient system.

The proposed system rectifies the demerits and defects of existing process to a greater extend.

III. RELATED WORK

Existing systems had some drawbacks which were that the system model only worked for extraction of text and processing on that extracted text and second that once the text had been extracted the images would get destroyed or degraded.

The existing systems do not process images. The text which is extracted can be saved in document format and used for translation and other processing. To overcome this drawback we propose a new system which extracts features and store them for using in search.

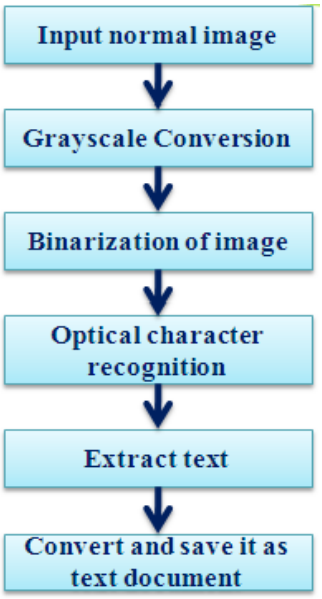

Fig. 2 Existing System Process Flow.

IV.PROPOSED SYSTEM

In the proposed system, it is a two stage approach. The suggested method separates text from drawings. The two stages are as follows:

1) In the first stage, we utilize a binarized version of the document to detect and extract text.

2) In the second stage, we remove the text from the digital document using OCR and inpainting method.

3) Feature Extraction is done and features are saved in a separate file. These features are used for searching using text.

4) Search by image can be also done.

Grayscale:

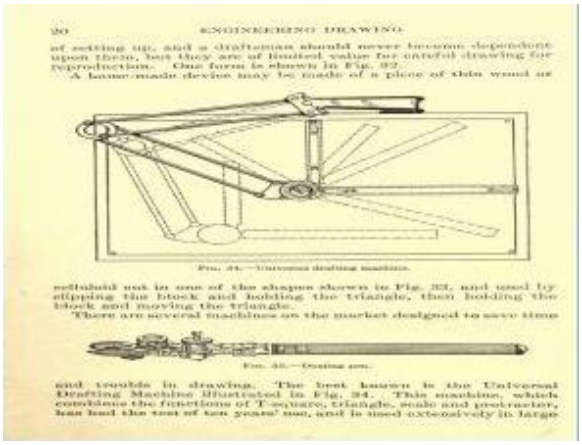

Fig. 3. Original Image

Average method: Average method is the most simple one. You just have to take the average of three colors. Since its an RGB image, so it means that you have add r with g with b and then divide it by 3 to get your desired grayscale image.

Grayscale = (R + G + B / 3)

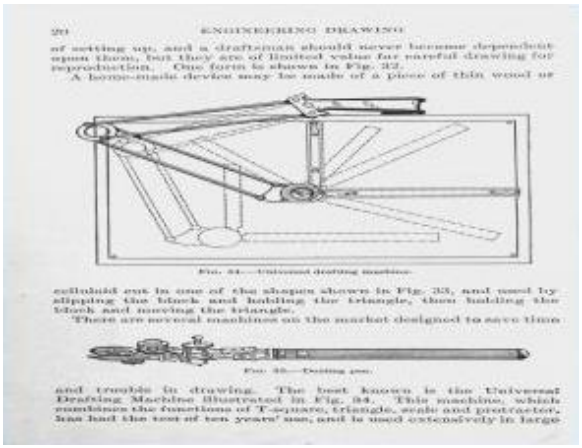

Fig. 4. Grayscale Image

Binarization:

In image processing, **Otsu's method**, named after Nobuyuki Otsu is used to automatically perform clustering-based image thresholding, i.e the reduction of a graylevel image to a binary image. The algorithm assumes that the image contains two classes of pixels following bi-modal histogram (foreground pixels and background pixels), it then calculates the optimum threshold separating the two classes so that their combined spread (intra-class variance) is minimal, or equivalently (because the sum of pair-wise squared distances is constant), so that their inter-class variance is maximal.

In Otsu's method we exhaustively search for the threshold that minimizes the intra-class variance (the variance within the class), defined as a weighted sum of variances of the two classes:

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t)$$

Weights $\omega_{0,1}$ are the probabilities of the two classes separated by a threshold t and $\sigma_{0,1}^2$ are variances of these two classes.

The class probability $\omega_{0,1}(t)$ is computed from the L histograms:

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i)$$

$$\omega_1(t) = \sum_{i=t}^{L-1} p(i)$$

Otsu shows that minimizing the intra-class variance is the same as maximizing inter-class variance:

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1$$

$$= \omega_0(t)\omega_1(t)\left[\mu_0(t) - \mu_1(t)\right]^2$$

which is expressed in terms of class probabilities ω and class means μ.

while the class mean $\mu_{0,1,T}(t)$ is:

$$\mu_0(t) = \sum_{i=0}^{t-1} ip(i)/\omega_0$$

$$\mu_1(t) = \sum_{i=t}^{L-1} ip(i)/\omega_1$$

$$\mu_T = \sum_{i=0}^{L-1} ip(i)$$

The following relations can be easily verified:

$$\omega_0\mu_0 + \omega_1\mu_1 = \mu_T$$

$$\omega_0 + \omega_1 = 1$$

The class probabilities and class means can be computed iteratively. This idea yields an effective algorithm.

Algorithm

1. Compute histogram and probabilities of each intensity level
2. Set up initial $\omega_i(0)$ and $\mu_i(0)$
3. Step through all possible thresholds $t = 1 \ldots$ maximum intensity
    1. Update ω_i and μ_i
    2. Compute $\sigma_b^2(t)$
4. Desired threshold corresponds to the maximum $\sigma_b^2(t)$

Binarized image:

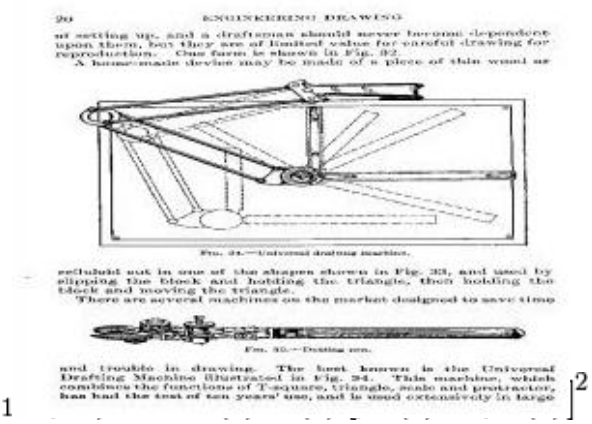

Fig.5. Binarized Image

Optical Character Recognition and Inpainting:

Optical Character Recognition is the mechanical or electronic conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image.

After applying OCR, we calculate the length the of the characters for Inpainting the character detected. After this calculation a patch of white colour is inpainted at character's place. Binary image and this inpainted image goes under XOR process in which the extracted image is given as output . The remaining text part is saved as another image file.

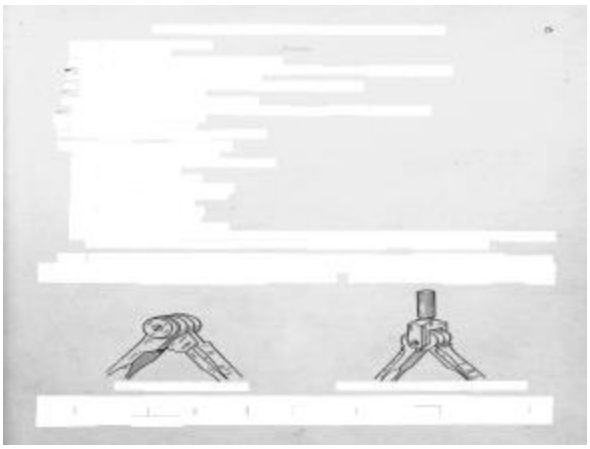

Fig.5. Image after application of OCR

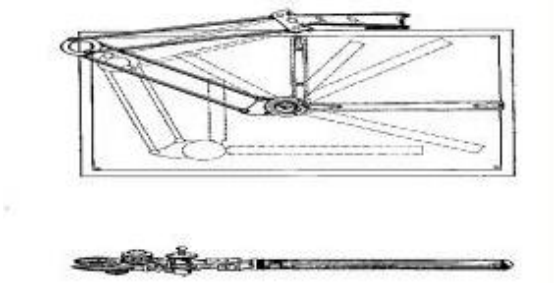

Fig.7. Extracted Image after the whole process

Fig.8. Extracted Image (Text)

Feature Extraction:

In image processing, feature extraction starts from an initial set of measured data and builds derived values intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Here we will using it to extract features and store it in separate files for efficient matching and searching techniques.

Searching:

Searching by image and text keywords can be performed.
Here we provide search by images in which images can be given as “key-images” to find similar patterns. We use Edge of Histogram to perform this. The basic idea in this step is to build a histogram with the directions of the gradients of the edges (borders or contours).Text recognition is done in which we can search using “keywords”. When we fire the “keyword” query as an input, the output we get will be images which contain that “keyword”. This is an exceptional feature of our proposed system.

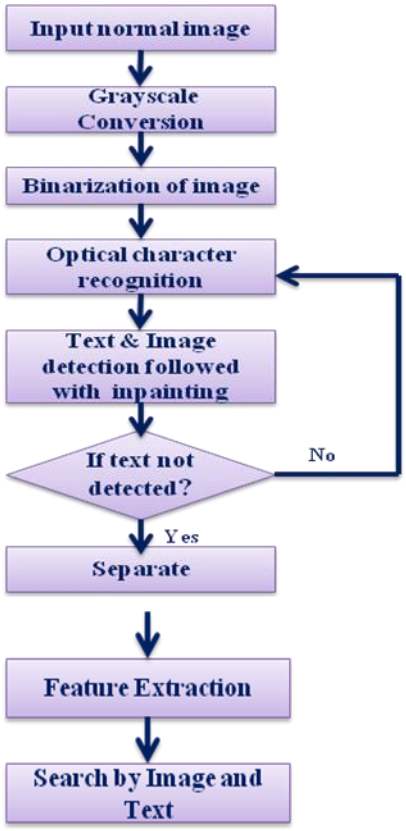

Fig.9. Proposed system process flow

VI. CONCLUSION

This proposed system separates text from drawings in documents. A binarized version of the document is used to detect and extract text. We remove the text from the document and use a classifier to separate drawings from other classes, e.g., background and noise. This paper gives the ability for computer machine to search an image with reference to text and images. Any image or text in an image or a figure name is given as input and the image is searched and retrieved. This paper gives accurate results without deteriorating the image pixels.

REFERENCES

[1] Rafi Cohen, Abedelkadir Asi, Klara Kedem, Jihad {rafico,abedas,} Itshak Dinstein, "*Robust Text and Drawing Segmentation Algorithm for Historical Documents*". Ben-Gurion University of the Negev 2012

[2] Priyanka Deelip Wagh Dept. of Computer Engineering SES's R. C. Patel Institute of Technology Shirpur (MH), India; D. R. Patil Dept. of Computer Engineering SES's R. C. Patel Institute of Technology Shirpur(MH), India; 2015 International Conference on Pervasive ,"Text Detection and Removal from Image using Inpainting with Smoothing." Computing (ICPC) 978-1-4799-6272-3/15/$31.00(c)2015.

[3] S. S. Bukhari, F. Shafait, and T. M. Breuel. Improved document image segmentation algorithm using multiresolution morphology. In Proceedings of SPIE. International Society for Optics and Photonics, 2011.

[4] M. Lettner and R. Sablatnig. Spatial and spectral based segmentation of text in multispectral images of ancient documents. In 10th International Conference on Document Analysis and Recognition, pages 813–817, 2009.

[5] L. Likforman-Sulem, A. Zahour, and B. Taconet. Text line segmentation of historical documents: a survey. International Journal on Document Analysis and Recognition, 9:123–138, 2007. [21]

[6] R. Manmatha, C. Han, and E. M. Riseman. Word spotting: New approach to indexing handwriting. In Proceeding of Computer Vision and Pattern Recognition, pages 631–637, 1996. [22]

[7] N. Otsu. A threshold selection method from gray-level histograms. Automatica, 11(285-296):23–27, 1975.

[8] N. Otsu. A threshold selection method from gray-level histograms. Automatica, 11(285-296):23–27, 1975.]

[9] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. International Journal of ComputerVision,40(2):99–121,2000

[10] S. Bukhari, M. A. Azawi, F. Shafait, and T. Breuel. Document image segmentation using discriminative learning over connected components. In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, pages 183–190. ACM, 2010