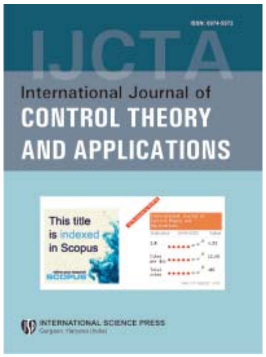


International Journal of Control Theory and Applications

ISSN : 0974–5572

© International Science Press

Volume 9 • Number 40 • 2016

An Automated Text Extraction System for Complex Images

D. Haritha[a] Avirup Guha Neogi[b] and G.N. Balaji[b]

[a]*Professor, Department of Computer Science and Engineering, K L University, Guntur, Andhra Pradesh, India.*

E-mail: haritha_donavalli@kluniversity.in

[b]*Assistant Professor, Department of Computer Science and Engineering K L University, Guntur, Andhra Pradesh, India.*

E-mail: avirupgn@gmail.com, [3]balaji.gnb@gmail.com

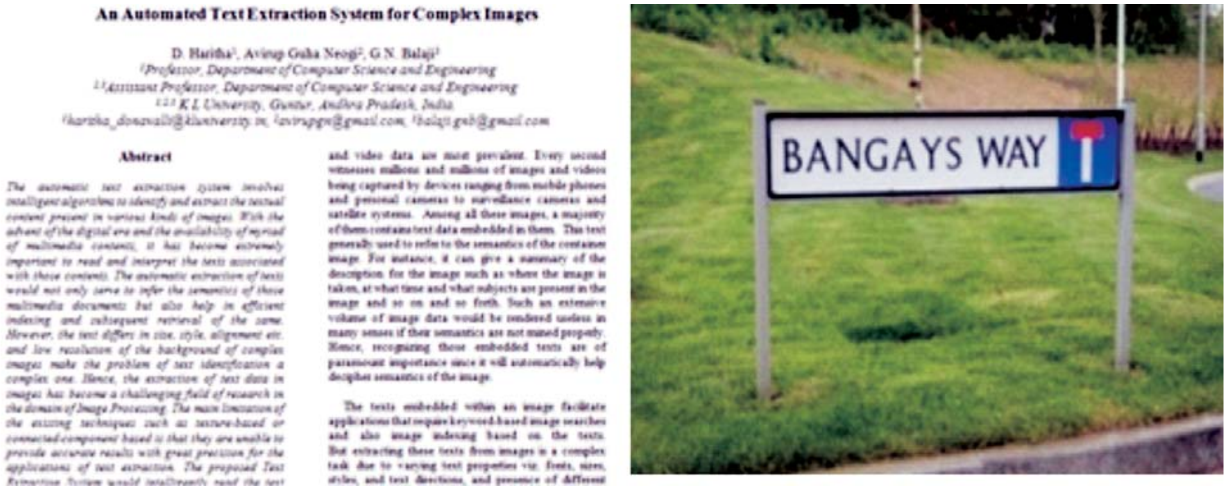
Abstract : The automatic text extraction system involves intelligent algorithms to identify and extract the textual content present in various kinds of images. With the advent of the digital era and the availability of myriad of multimedia contents, it has become extremely important to read and interpret the texts associated with those contents. The automatic extraction of texts would not only serve to infer the semantics of those multimedia documents but also help in efficient indexing and subsequent retrieval of the same. However, the text differs in size, style, alignment etc. and low resolution of the background of complex images make the problem of text identification a complex one. Hence, the extraction of text data in images has become a challenging field of research in the domain of Image Processing. The main limitation of the existing techniques such as texture-based or connected-component based is that they are unable to provide accurate results with great precision for the applications of text extraction. The proposed Text Extraction System would intelligently read the text regions from various complex images. The design includes various stages like localization, segmentation and finally recognition of the textual data in images. For the localization of text, Discrete Wavelength Transform function is used. Then the morphological operations are applied to correctly mark the text regions. After that, the text portion is segmented and recognized by an efficient system. A big advantage of this system is that the output which is a text data can be stored in a .txt file format. Furthermore, modification of the extracted text is also possible. This proposed approach can be used in more advanced and sophisticated applications as it has exhibited better precision rate, efficiency and recall rate.

1. INTRODUCTION

With the information age advancing at a lightning pace, the multimedia content in the digital world is growing at an exponential rate. Out of which, the image and video data are most prevalent. Every second witnesses millions and millions of images and videos being captured by devices ranging from mobile phones and personal cameras to surveillance cameras and satellite systems. Among all these images, a majority of them contains text data embedded in them. This text generally used to refer to the semantics of the container image. For instance, it can give a summary of the description for the image such as where the image is taken, at what time and what subjects are present in the image and so on and so forth. Such an extensive volume of image data would be rendered useless in many senses if their semantics are not mined properly. Hence, recognizing those embedded texts are of paramount importance since it will automatically help decipher semantics of the image.

The texts embedded within an image facilitate applications that require keyword-based image searches and also image indexing based on the texts. But extracting these texts from images is a complex task due to varying text properties viz. fonts, sizes, styles, and text directions, and presence of different light conditions and complex backgrounds. Broadly, images can be divided into three categories viz. document images, born-digital images and scene images. Document images refer to the standard document file present in image format. The document may be a pdf or note or word document. The textual data present in document image is known as document text. The document image sample is given in Figure 1(a). Born-digital images refer to those images which are generated in-silico by the software. The textual data present in such types of images is known as overlay text or caption text. Sometimes overlay text is also known as artificial text or superimposed text.

Scene images refer to the images which are captured by cameras and have lots of textual contents as in hoardings, banners, etc. Example of such an image is shown in Figure 1(*b*).



An Automated Text Extraction System for Complex Images

D. Haritha[1], Avirup Guha Neogi[2], G.N. Balaji[3]
[1]*Professor, Department of Computer Science and Engineering*
[2,3]*Assistant Professor, Department of Computer Science and Engineering*
[1,2,3] *K L University, Guntur, Andhra Pradesh, India*
[1]*haritha_donavalli@kluniversity.in,* [2]*avirupgn@gmail.com,* [3]*balaji.gnb@gmail.com*

Abstract

The automatic text extraction system involves intelligent algorithms to identify and extract the textual content present in various kinds of images. With the advent of the digital era and the availability of myriad of multimedia contents, it has become extremely important to read and interpret the texts associated with these contents. The automatic extraction of texts would not only serve to infer the semantics of these multimedia documents but also help in efficient indexing and subsequent retrieval of the same. However, the text differs in size, style, alignment etc. and low resolution of the background of complex images make the problem of text identification a complex one. Hence, the extraction of text data in images has become a challenging field of research in the domain of Image Processing. The main limitation of the existing techniques such as texture-based or connected-component based is that they are unable to provide accurate results with great precision for the applications of text extraction. The proposed Text Extraction System would intelligently read the text

and video data are most prevalent. Every second witnesses millions and millions of images and videos being captured by devices ranging from mobile phones and personal cameras to surveillance cameras and satellite systems. Among all these images, a majority of them contains text data embedded in them. This text generally used to refer to the semantics of the container image. For instance, it can give a summary of the description for the image such as where the image is taken, at what time and what subjects are present in the image and so on and so forth. Such an extensive volume of image data would be rendered useless in many senses if their semantics are not mined properly. Hence, recognizing these embedded texts are of paramount importance since it will automatically help decipher semantics of the image.

The texts embedded within an image facilitate applications that require keyword-based image searches and also image indexing based on the texts. But extracting these texts from images is a complex task due to varying text properties viz. fonts, sizes, styles, and text directions, and presence of different

(*a*)

(*b*)

Figure 1: (*a*) Document image (*b*) Scene text image

These types of images are more complex with less resolution complicated foreground/background and severe edge softness. This gives rise to difficulty in detecting the textual content from the scene. The textual data present in images exhibit the following properties which are enumerated below:

1. **Size:** The textual data size may vary from small to medium to large depending on the domain of application.

2. **Alignment:** This refers to the geometric arrangement of the text from the planar or non-planar aspect. For document and born-digital images, they usually lie in horizontal fashion with sometimes special non-planar effects. However, scene texts have geometric distortions and can be aligned in any direction.

3. **Inter-character distance:** This refers to the distance between the characters in a line of text. Usually, the inter-character distance is uniform in a line of text.

4. **Colour:** Colour is an important property to identify texts based on connected-component approach. Usually, characters in a line of text are monochromatic. However, some complex documents can have polychromatic words for giving visual effects.

5. **Motion:** For caption texts, the motion is uniform either in horizontal or vertical manner. Scene texts usually have arbitrary non-uniform motion due to object or camera movement

6. **Edge:** In majority of caption and scene images, there is a strong edge between the text and the background.

7. **Compression:** Digital images are stored, processed and transferred in compressed form. It is therefore fast and convenient to extract text from the compressed image without decompression.

The remaining contents of this paper are structured as follows. Section 2 discusses the related works carried out in this field of research. Section 3 gives the proposed methodology and the experiments and results are explained and analyzed in section 4. Finally, Section 5 concludes the paper by discussing the advantages of the proposed system.

2. RELATED WORKS

The various approaches pertaining to text extraction mainly include region-based methods and texture-based methods. The region-based methods are basically divided into two categories, viz. edge-based [1] and connected-component based methods.

A connected-component based approach for text extraction is proposed by J. Gllavata et al. [3] which is based on colour reduction technique and which uses OCR for character recognition. The primary limitation of this approach is that it is capable of detecting texts only with horizontal arrangement. Also, it is incapable of processing low quality images with a good accuracy. A connection-component based strategy is proposed by Zhong et al. [4] which also utilizes colour reduction technique. In this approach, the quantization of the colour space is done by analyzing the colour histogram generated in the RGB colour space. The main assumption behind this approach is that the text region occupies a significant portion of the image and that they tend to cluster together in this colour space. Every text component present will undergo filtering stages using a number of heuristics like spatial alignment, area, diameter, etc. The execution of this framework is assessed using CD and other book cover images. Kim et al. [5] uses transition map to recognize overlay text. They proposed a technique for overlay text identification and extraction from complex videos. The identification strategy is based on the perception with respect to the presence of transient colour between embedded text and its adjacent background. The transition map is produced first which is based on its logarithmical change of intensity and modified saturation. Connected components are created for each candidate region by the generated linked mask and each of these connected components is reshaped to have smooth boundaries. Findings based on Caption text identification are proposed in [9]. Xiaoqing Liu, et al. [6] proposed a technique based on the properties of edges. Despite the fact that it can handle both printed and document images effectively, this approach is not sensitive to image intensity or colour. One of the major limitations of this system is that it analyzes texts in the form of blocks and hence, small image regions and strokes are misinterpreted as texts in areas containing large characters. Classifiers like Support Vector Machine (SVM), back-propagation neural network can also be used in identifying texts [14].

The proposed system uses wavelet transform and various morphological operations in a different way as compared to the available methodologies for the identification of textual data. A better efficiency and accuracy is obtained in our approach in the extraction of text.

3. PROPOSED SYSTEM

The working of the proposed is based on the fact that the texts present in images have some unique features which include the properties of edges. The block diagram of the proposed method is shown in Figure 2. The proposed method is mainly divided into three modules viz. Edge map generation module, Text area segmentation module and Text recognition module. The input is an image to the system and the output obtained is in the form of a text file.

3.1. Edge Map Generation Module

The input image to this module can be in gray scale or coloured, compressed or uncompressed form.

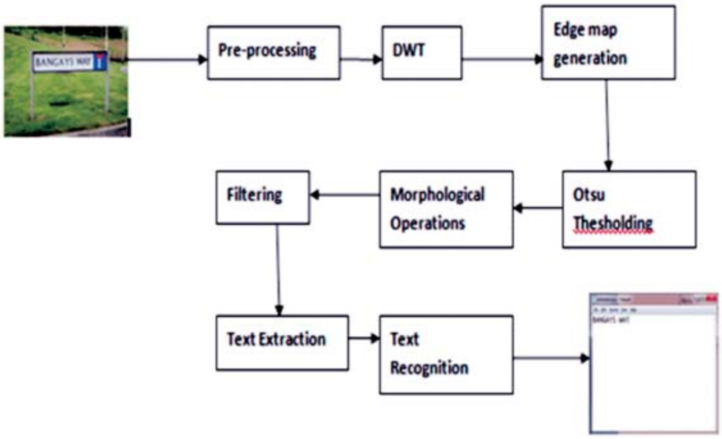


Figure 2: Architecture of the proposed system

Various steps in this module are elaborated below. The first step is the pre-processing step.

3.1.1. Pre-processing

The three basic characteristics of an embedded text in a complex image are edge, strength and density. These serve as an important feature for extracting text. In the proposed algorithm, the input can be either a colour or a gray-scale image. If it is a colour image, it is passed through the pre-processing stage. In this stage, the colour to gray-scale conversion of image takes place i.e. the RGB image is transformed to Hue-Saturation-Value (HSV) colour space using the following equation.

$$Y = 0.299R + 0.587G + 0.114B \qquad (1)$$

Here, 'Y' refers to the value component of the Hue-Saturation-Value (HSV) colour space. Alternatively, we could have also pre-processed the image by converting it to YCbCr colour space and taking the Y, the luminance part for processing. After the gray-scale image is obtained, noise filtering is done by applying median filter which is the best known order statistics filter. It computes the value of a particular pixel by taking the median of the gray-scale level values in the neighbourhood of the target pixel. Median filtering is very effective in removing unipolar and bipolar impulsive noises and has great noise filtering capabilities with less blurring as compared to linear smoothing filters of similar size. After this noise filtering step, most of the noise gets filtered out but the edges in the image still remains present. The image Y then undergoes 2-D discrete wavelet transformation.

3.1.2. Discrete Wavelength Transform

In this system, Haar Discrete Wavelet Transform (DWT) is used in the proposed algorithm. The Haar DWT is powerful enough to model most of the image characteristics. The textured images are generally well characterized by their edges. After applying DWT, it is decomposed into the components of frequency domain [8]. The input image is decomposed into four components, i.e., one average component and three detail components. To obtain the components, it has to deal with row and column directions separately. At first, High Pass Filter (H.P.F) G and Low Pass Filter (L.P.F) H are applied to each row data and then are down sampled by 2 to get high and low frequency components of the row. Next, the high and low pass filters are again applied to each column data and then down sampled by 2 to get high and low frequency components of the column. In this way, the four sub-band images are generated, namely: HH1, HL1, LH1 and LL1.

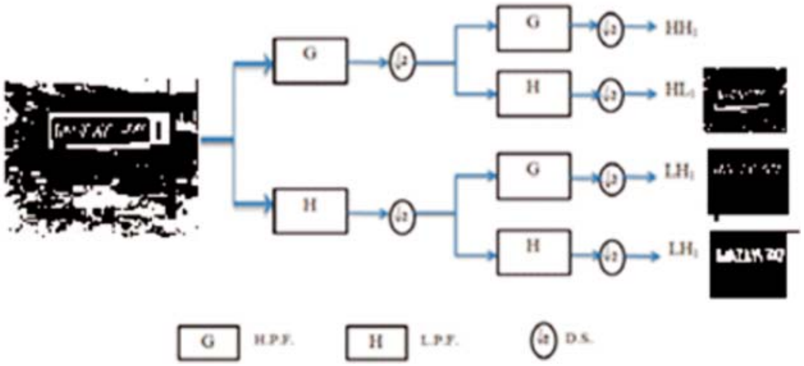


Figure 3: Block diagram of 2-D DWT

The detail component sub-bands containing the vertical details LH1, horizontal details HL1 and diagonal details LL1 are used to detect text edges present in the original image. The process of applying DWT to image is shown in Figure 3. The D.S. represents the down sampling of the image by 2.

Since filtering is done before applying DWT, the effect of noise in the components can be reduced. Now edge detection method is applied to each component. The wavelet function and the scaling function of haar wave-lets are defined below.

$$\varphi(t) = \left\{ \begin{cases} 1,0 \leq t < 1/2 \\ -\dfrac{1,1}{2} \leq t < 1 \\ 0, otherwise \end{cases} \right. \qquad (2)$$

$$\phi(t) = \begin{cases} 1,0 \leq t < 1 \\ 0, otherwise \end{cases} \qquad (2)$$

The Haar is comparatively simpler than other wavelets since it reduces the complexity of the algorithm. The advantage of using Haar wavelet is that it is the only wavelet that allows perfect localization in the transform domain. Its coefficients are either 1 or -1 and are real symmetric and orthogonal. On applying this transform, the portions with higher edge strength in identical directions can be identified. After that, the threshold is calculated which helps filter out the infeasible edges. The second derivative of intensity is utilized in the estimation of edge strength as it gives enhanced detection of intensity spots which generally leads to characterization of text in images. The traditional edge detection filters are also capable of providing similar results but they will not be able to detect three kinds of edges simultaneously. Hence, the processing time for the conventional edge detection filters is more as compared to 2-D DWT.

3.1.3. Text Region Detection

The text region detection process comprises two steps viz. detection and localization. The detection step is responsible for classifying the frame into text and non-text regions. For different algorithmic techniques, the size and shape vary. For instance, 8x8 pixel blocks are classified by some algorithms, while individual scan lines   are classified by others. In the localization step, the results of detection are grouped together to form one or more instances of the text. This is represented as a bounding box around each text instance. For the detection of the text region, the three detailed sub-components (which are obtained by applying Haar DWT) are used. To these each detailed sub-components, Sobel edge detection algorithm is applied which is more efficient to locate the presence of strong edges. Let 'hc', 'vc' and 'dc' are images obtained after edge detection as horizontal, vertical and diagonal components respectively; where 'hc' contains all the edges in horizontal direction, 'vc' contains all the edges in vertical direction and 'dc' contains all the edges in diagonal direction. Next, an edge map is generated by 'hc', 'vc' and 'dc' using weighted OR operation as given by the following equation.

$$I = (40 * hc + 70 * vc + 30 * dc)$$

Here, ‘I’ is the image obtained after applying the edge map. Figure 4 shows a sample edge map with all possible text regions.

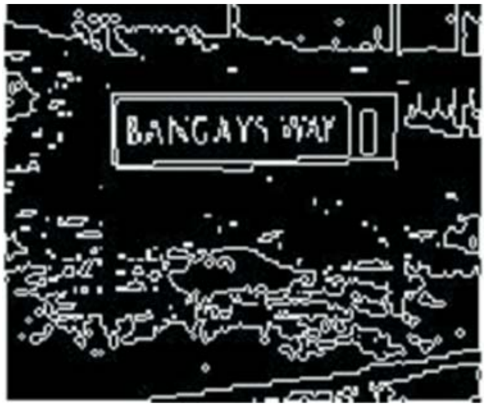


Figure 4: Edge map of the image obtained

3.2. Text Area Segmentation

After obtaining the edge map, the binarization of the image is done. The thresholding operation is used to obtain the binary form of the edge map. Otsu algorithm is used for this binarization process. The thresholding operation removes the non-text regions identified so far in this technique. The steps to compute Otsu threshold are enumerated below.

1. The image is reshaped to 1D.
2. Histogram and values are computed at each intensity level.
3. A matrix is initialized with values from 0 to 255.
4. Step through all possible thresholds maximum intensity
5. The mean, weight and the variances for the foreground and background are computed.
6. (Weight of the foreground) * (variance of the foreground) + (weight of the background) * (variance of the background) are calculated.
7. The minimum value is calculated.

The calculated minimum value is considered as the threshold for the process of binarization. The text localization process makes further refinements in the text regions by deleting non-text regions from the image. One of the important properties that the text exhibits is that all the characters appear in close proximity to one other in the image thereby forming a cluster. Hence, morphological operations are used taking into account the above property. Clustering can be done with the possible text pixels using the dilation operation, thereby eliminating pixels that are far away from the candidate text. The dilation process involves large structuring element so that the regions which lie adjacent to each other can be enhanced. The morphological operations are used to localize the text part clearly. The morphological operations include erosion and dilation. The segmentation of the identified text portions are done in this segmentation phase. For this purpose, the labelling of the connected components is done and the connectivity used is 8. The set of properties, shape and measurements of the connected components are computed. The area and bounding box shape measurements are considered as per the requirements only. The area of the connected component is considered as a scalar entity as it represents the number of pixels in that region. Bounding box is put to the connected component recognized and it should be the smallest containing the required text region [11]. The height, width, and the upper left corner positions are marked. A new value can be calculated by taking the product of width and height of bounding box. Then, the ratio of this new value and area is computed. If the ratio is below 1.5, then the regions acquired are considered as text regions. The resultant image obtained after dilation operation may even comprise some non-text regions or unwanted noises. To wipe out these noise blobs exhibited in the image an area filtering is performed. After that, only those regions in the last image whose area is more than or equal to 1/15 of the maximum area region identified are kept.

3.3. Text Recognition

The input to this phase is the image obtained from the previous stage. Tesseract OCR is used for text recognition. The binary image with polygonal text regions defined is given as input to the tesseract. The processing of the system follows a traditional pipeline. As the first step, the analysis of the connected components is performed in which outlines of the components are stored. Although this is a computationally expensive plan, it has many significant advantages like detection of inverse text and recognizing it as easily as black-on-white text. In this stage, outlines are assembled by nesting into Blobs. These Blobs which are obtained are organized into lines of text. The lines and regions are examined for proportional text or fixed pitch. Proportional text is divided into words using fuzzy and definite spaces, while fixed pitched text is chopped immediately by character cells. Here, recognition process undergoes a two-pass process. In the first pass, an endeavour is made to recognize each word. Each word that is agreeably recognized is then passed to an adaptive classifier as training data. This makes the adaptive classifier to try to recognize the text more accurately in the page. After the first pass, a second pass is executed. This is to ensure that the words which are learnt late (and were not recognized before) do not go unrecognized in the second pass. A final phase resolves fuzzy spaces, and locates the small-cap text by checking the alternative hypotheses for the x-height. After successfully recognition of the characters in the text, the UTF-8 code of each character is returned. This UTF-8 code can be easily converted to its corresponding character and displayed as output in the form of a text file. The results of morphological operation and segmentation is shown in Figure 5(a) and Figure 5(b).

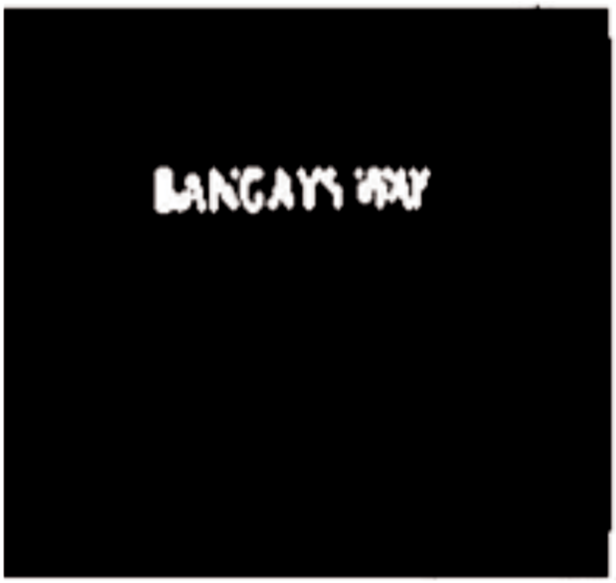


Figure 5 :(*a*) Image after morphological operations

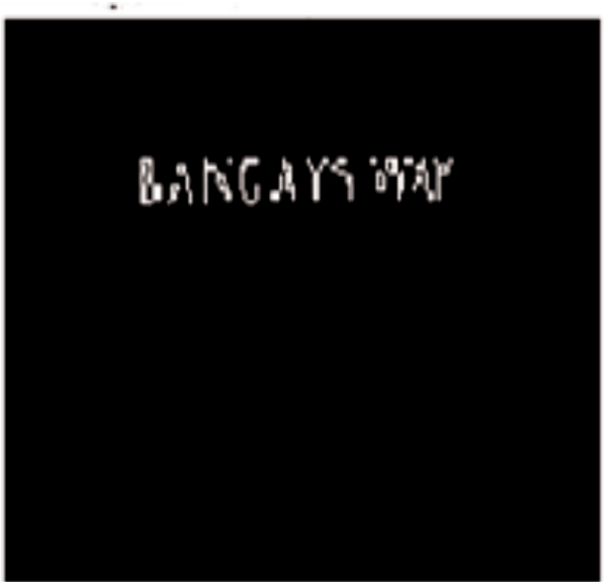


Figure 5(b): Output image after segmentation

4. EXPERIMENT AND RESULTS

The proposed system is tested in MATLAB [13] with data sets having large number of images and the results obtained are enumerated below.

Table 1. gives a comparative analysis of the proposed approach with some existing techniques. The method proposed by Samarabandhu et al. shows 91.8 % of precision rate and 96.6% of recall rate whereas the method given by J.Yang et al. have lesser rates of about 84.9% and 90.0% precision and recall respectively. The other existing methods proposed by Kim et al. and J.Gllavata et al. exhibit even smaller rates. But our method shows 97.2% of precision rate and 98.3% of recall rate. It is clear from the study that the proposed method outperforms the existing methods with respect to precision rate and recall rate.

The accuracy of the proposed algorithm is calculated by counting the occurrences of the correctly located characters, which is considered as the ground truth. The precision and recall rates are computed by the following equations:

Table 1
Comparisons with existing methods

Method	*Recall rate (%)*	*Precision rate (%)*
Proposed method	97.2	98.3
Samarabandhu et. al[6]	96.6	91.8
J. yang et. al[11]	90.0	84.9
K.C.kim et al[10]	82.8	63.7
J. Gllavata et.al[3]	88.7	83.9

$$Precision\ rate = \frac{Correctly\ located}{Correctly\ located + falsenegative} * 100$$

$$Recall\ rate = \frac{Correctly\ located}{Correctly\ located - falsenegative} * 100$$

5. CONCLUSION

Although a handful of text extraction algorithms are developed in this area, there is no single unified approach that is suitable for all the applications. In this paper, a comparatively simpler and intelligent method for text identification and extraction is proposed. Discrete Wavelength Transform is used in this method for increasing the efficiency of the algorithm. The limitation of the existing systems is that they are unable to perform under the circumstances when the characters are not in proper alignment or when they are very small and have poor contrast with respect to the background image. But our proposed method is tolerant to the variations in colours or intensities of the images and also to its reflection effects and uneven illumination. The process will find great utility in real-time applications as it requires less processing time and also exhibit high precision rates. It finds real-world applications in industrial automation, intelligent transport system, robotics, etc. The proposed approach works effectively both with document images and scene text images.

REFERENCES

[1] Xin Zhang, Fuchun Sun, Lei Gu, “A Combined Algorithm for Video Text Extraction,” Seventh International Conference on Fuzzy Systems and Knowledge Discovery,China, 2010.

[2] J. Ohya A. Shio, and S. Akamatsu, “Recognizing Characters in Scene Images,” IEEE Transactions on Pat-tern Analysis and Machine Intelligence, 1994, pp 214-224.

[3] Gllavata, R. Ewerth, and B. Freisleben, “A robust algorithm for text detection in images,” Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis, Italy, 2003, pp.611– 616.

[4] Zhong Yu., KaruK., and Jain, A.K, ” Locating text in complex color images,” Proceedings of the Third International Conference on Document Analysis and Recognition, Canada, 1995, pp. 146-149.

[5] WonjunKim,Changick Kim, “A New Approach for Overlay Text Detection and Extraction From Complex Video Scene,” IEEE Transactions on Image Processing, 2009, V.18 , No.2, pp. 401 – 411.

[6] Xiaoqing Liu, JagathSamarabandu, “Multiscale Edge-Based Text Extraction from Complex Images,” International Conference on Multimedia and Expo, Canada,2006, pp.1721-1724.

[7] Xiao-Wei Zhang, Xiong-Bo Zheng, Zhi-Juan Weng, “Text Localization Algorithm Under Back-ground Image Using Wavelet Transforms,” Proceedings of the 2008 International Conference on Wavelet Analysis and Pattern Recognition, Hong Kong, Aug. 2008.pp.30-31.

[8] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms,” IEEE Transactions on Systems, man and Cybernet, 1979.

[9] Debapratim Sarkar, Raghunath Ghosh,”A Bottom-Up Approach of Line Segmentation from Handwritten Text, ”2009.

[10] K.C. Kim, H.R. Byun, Y.J. Song, Y.W. Choi, S.Y. Chi,K.K. Kim and Y.K Chung, “Scene Text Extraction inNatural Scene Images using Hierarchical Feature Combining and verification,” Proceedings of the 17thInternational Conference on Pattern Recognition (ICPR’04),IEEE, 2004

[11] J. Yang, J. Gao, Y. Zhang, X. Chen and A. Waibel, “An Automatic Sign Recognition and Translation System,” Proceedings of the Workshop on Perceptive User Inter-faces(PUI’01), 2001, pp. 1-8.

[12] A.K Jain,”Fundamentals of Digital Image Processing,” Englewood cliff, NJ: Prentice Hall, 1989,

[13] R C Gonzalez,”Digital Image Processing Using MATLAB,” Tata McGraw Hill Education Private Limited, 2010

[14] Balaji, G. N., T. S. Subashini, and N. Chidambaram. “Detection of heart muscle damage from automated analysis of echocardiogram video.” IETE Journal of Research 61.3 (2015): 236-243.