An Edge Based Text Segmentation From Complex Images

W. Josephin

Dept. of Computer Science,
St. John's College, Palayamkottai,
TamilNadu

Dr. R. K. Selvakumar

Dept. of Computer Science &
Engineering,
Cape Institute of Technologyy,
Levengipuram, Tirunelveli, TamilNadu

Abstract

Text in images is a significant cue for visual content understanding and retrieval. Detection and extraction of text in images have been used in many applications. This paper proposes an edge-based text segmentation algorithm, which is robust with respect to font sizes, styles, color/intensity, orientations and alignment of text. This method can quickly and effectively localize and extract text regions from real scenes .It can be used in a large variety of application fields, such as vehicle license detection and recognition, document retrieving and page segmentation etc.

KeyWords: *Text extraction, Scene TextThresholding,.*

1. Introduction

Text appears in images either in the form of documents such as scanned CD/book covers or video images. Video text can be broadly classified into two categories: overlay text and scene text. Scene text occurs naturally as a part of scene such as text in information boards/signs, nameplates, food containers etc. Automatic detection and extraction of text in images have been used in many applications. Scene text extraction can be used in mobile robot navigation to detect text-based landmarks, vehicle license plate detection / recognition, object identification etc.

Text embedded in images contains large quantities of useful semantic information, which can be used to fully understand images. Among the several textural properties in an image, edge based methods focuss on the 'high contrast between the text and the background'. The edges of the text boundary are identified and merged, and then several heuristics are used to filter out the non-text regions. Usually, an edge filter (eg. a Canny operator) is used for the edge detection and a smoothing operation or a morphological operator is used for the merging stage.

Yassin et al. [1] presented a morphological approach for text extraction. The RGB components of a color input image are combined to give an intensity image Y as follows: Y = 0.299 R + 0.587 G + 0.114 B, where R, G, and B are the red, green and blue components respectively. Although this approach is simple and many researchers have adopted it to deal with color images, it has difficulties dealing with objects that have similar gray scale values, yet different colors in a color space. After the color conversion, the edges are identified using a morphological gradient operator. The resulting edge image is then thresholded to obtain a binary edge image. Adaptive thresholding is performed for each candidate region in the intensity image, which is less sensitive to illumination conditions and reflections. Edges that are spatially close are grouped by dilation to form candidate regions, while small components are removed by erosion. Non-text components are filtered out using size, thickness, aspect ratio, and gray level homogeneity.

Wumo et al. [2] proposed a sparse representation based method for text detection from scene images. Edge information is extracted using Canny operator and then these edge points are grouped into connected components. Each connected component is labeled as text or non-text by a two-level labeling process. The core of the labeling process is a sparsity test using an over-complete dictionary, which is learned from edge segments of isolated character images. Layout analysis is further applied to verify these text candidates.

Wang et al. [3] proposed a connected-component based (CC based) method which combines a color clustering, a black adjacency graph (BAG), an aligning-and-merging-analysis (AMA) scheme and a set of heuristic rules together to detect text in the application of sign recognition such as street indicators and bill boards. As the author mentioned, uneven reflectances result in income plete character segmentation which increases the false alarm rate.

Kim et al. [4] combined a Support Vector Machine (SVM) and continuously adaptive mean shift algorithm (CAMSHIFT) to detect and identify text regions. Gao et al. [9] developed a three layer hierarchical adaptive text detection algorithm for natural scenes. This method was applied in a prototype Chinese sign translation system which mostly has a horizontal and/or vertical alignment.

X. Liu et al. [5] proposed a statistics based method to detect and localize text based features by calculating the spatial intensity variation. This method is very simple and fast. However, in real scenes, due to uneven illumination, reflections and shadows, an image background may contain areas with high spatial intensity variations that do not contain text. X. Liu and J. Samarabandu [6] developed a single scale edge-based text region extraction algorithm for indoor scene images, which is robust with respect to font sizes,styles,colors/intensity, orientations, effects of illumination, reflections, shadows and perspective distortion. X. Liu et al. [7] developed a multiscale edge-based text extraction algorithm, which can localize and extract text from both document and indoor/outdoor scene images.

In this paper we propose a single scale edge-based text segmentation method, which can quickly and effectively localize and extract scene text, especially from outdoor scene images and from object label images.

2. Proposed Method

Edges are a reliable feature of text regardless of color/intensity, layout, orientations etc. Edge strength and density are two distinguishing characteristics of text embedded in images, which can be used as main features for detecting scene text. Characters are made of strokes with different orientations, which can also be interpreted as edges with different orientations. Thus, variance information of the edge orientation is also an important feature of text. The proposed method uses the edge density, strength and orientation variance to extract text from real scenes.

The block diagram of the proposed text segmentation algorithm is given in figure 2.1. As can be seen in the block diagram, a color image is entered to the system as the input data and the segmented text on a clear black background is the output.

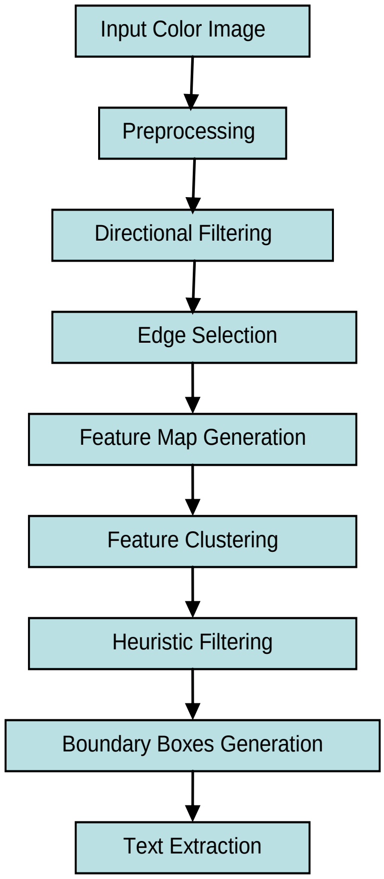

Figure 2.1: Block Diagram of the proposed Text Segmentation Algorithm

2.1 Preprocessing

If the input image is a color image, its RGB components are combined to give the intensity image Y as follows:

$$Y = 0.299R + 0.587G + 0.114B$$

2.2 Candidate Text Region Detection

In this stage a feature map is built by using three important properties of edges: edge strength, density and variance of orientations. The feature map is a gray scale image with the same size of the input image where the pixel intensity represents the possibility of text.

2.2.1 Directional Filtering

The magnitude of second order derivative of intensity is used as a measurement of edge strength as this allows better detection of intensity peaks that normally characterize text in images. The edge density is calculated based on the average edge strength within a window. Four orientations 0°, 45°, 90°,135° are used to evaluate the variance of orientations. 0° denotes horizontal orientation, 90° denotes vertical orientations and 45° and 135° denote the two diagonal directions respectively. After convolving the image with a compass operator [8] as shown in Fig.2.2, we get four oriented edge intensity images $E_{\theta = 0,45,90,135}$, which contain all the properties of edges required in our method.

-1	-1	-1
2	2	2
-1	-1	-1

0° Kernel

-1	-1	2
-1	2	-1
2	-1	-1

45° Kernel

-1	2	-1
-1	2	-1
-1	2	-1

90° Kernel

2	-1	-1
-1	2	-1
-1	-1	2

135° Kernel

Figure 2.2: Compass Operator

2.2.2 Edge Selection

Vertical edges form the most important stroke of characters and their lengths also reflect the heights of corresponding characters. By extracting and grouping these strokes, we can locate text with different heights (sizes). Some non-character objects in the real scene also produce strong vertical edges and they have very large lengths. By grouping vertical edges into long and short edges, we can eliminate the vertical edges with extremely long lengths and retain short edges are retained for further processing.

After thresholding, the long vertical edges may become broken short edges which may cause false alarms. In order to eliminate the false grouping, we use a two stage edge generation method.

The first stage is used to obtain strong vertical edges as follows.

$$Edge_{90bw}^{strong} = |E_{90}|_z \quad \text{-------} \quad (1)$$

Where E_{90} is the 90° intensity edge image, which is the 2D convolution result of the original image with the 90° kernel. $|\mathbf{.}|_z$ is a thresholding operator to get the binary result of the vertical edges.

The second stage is used to obtain weak vertical edges, $Edge_{90bw}^{weak}$ which are obtained as follows.

A morphological dilation operation is performed on the strong vertical edges obtained in the first stage. Then a closing operator with appropriate structuring element is employed on the resultant vertical edges. The difference of the vertical edges obtained in the previous two steps is found. After that the following operation is performed, which results in weak vertical edges.

$$Edge_{90bw}^{weak} = |E_{90} \times (closed - dilated)|_z$$

The resultant vertical edges of the two stage edge generation method are the combination of the strong and weak edges as described below.

$$Edge_{90bw} = Edge_{90bw}^{strong} + Edge_{90bw}^{weak}$$

A morphological thinning operator followed by a Connected Component Labeling and Analysis algorithms are then applied on the resultant vertical edges as described below.

$$Thinned = Thinning(Edge_{90bw})$$
$$Labeled = BWlabel(thinned,4)$$

After the connected component labeling, each edge is uniquely labeled as a single connected component with its unique component number. The labeled edge image is processed with a length labeling process as described in [6]. As a result, all the pixels belonging to the same edge are labeled with the same number which is proportional to its length. A high value in the length labeled image represents a long edge. Therefore a simple thresholding is used to separate the short edges.

2.2.3. Feature Map Generation

Regions with text in them will have significantly higher values of average edge density, strength and variance of orientations than those of non-text regions. We generate a feature map which suppresses the false regions and enhances the true candidate text regions. The feature map is generated using the following procedure.

$$DilCandidate = Dilation(short_{90bw})_{m\times m}$$

$$refined = DilCandidate \times \sum E_{\theta\ (\theta = 0,45,90,135)}$$

$$fmap(i,j)=N\{\sum_{m=-c}^{c}\sum_{m=-c}^{c}[refined(i+m,j+n)]\times weight(i,j)\}$$

N is the normalization operation that normalizes the intensity of the feature map into a range of [0,255]. Weight(i,j) is a weight function, which determines the weight of a pixel (i,j) based on the number of orientations of edges within a window.

2.3. Text Region Localization

This stage involves three steps: feature clustering, heuristic filtering and boundary boxes generation.

2.3.1 Feature Clustering

Normally, text embedded in an image appears in clusters. Thus, characteristics of clustering can be used to localize text regions. The intensity of the feature map represents the possibility of text. Therefore a simple global thresholding is employed to highlight those with high text possibility regions resulting in a binary image. A morphological dilation operation is performed on the binary image obtained from the previous step, to get the text blobs.

2.3.2 Heuristic Filtering

A connected component labeling algorithm is employed on the text blobs obtained from the previous step. Then two constraints are used to filter out those blobs which do not contain text. The first constraint is used to filter out all the very small isolated blobs as described below.

$$Area_{region} >= (1/15) \times Area_{max}$$

The second constraint is used to filter out those blobs whose widths are much smaller than corresponding heights.

$$\text{Ratio}_{w/h} = \frac{\text{Width}_{region}}{\text{Height}_{region}} >= 0.2$$

2.3.3 Boundary Boxes Generation

The retaining blobs are enclosed in boundary boxes. The coordinates of the boxes are obtained from the maximum and minimum coordinates of the top left and bottom right of the corresponding blobs.

2.4 Text Extraction:

The corresponding regions (blobs) in the original grayscale image are taken. Finally, an adaptive threshold is applied on these regions, which results in the segmentation of the real text regions from the image.

3. Experimental Results and Discussion

Currently, our algorithm has been implemented in IDL language under Windows XP. Experiments were carried out on two types of images, namely the outdoor scene images and object label images. Each image is in BMP or JPEG format. Text in the images is of different font sizes, colors, orientations, alignments, perspective projection under different lighting conditions.

In order to evaluate the performance of the proposed method, we use 45 test images. The results of the proposed algorithm when run on some typical images are shown below.

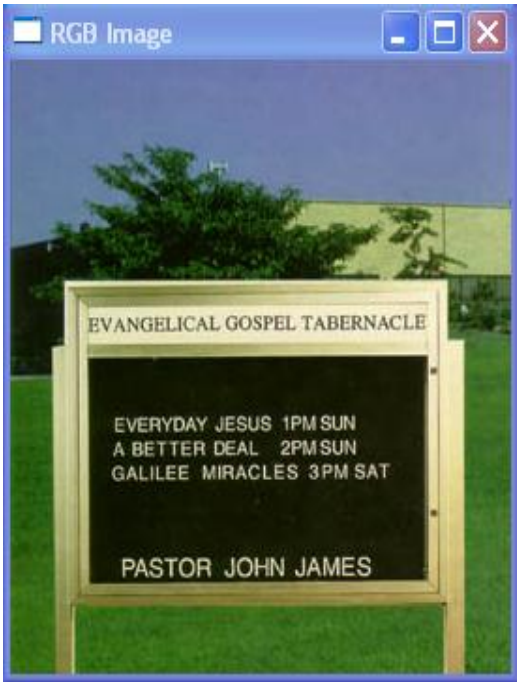

Figure 3.1: *OutDoor Scene Image*

a) Original Image

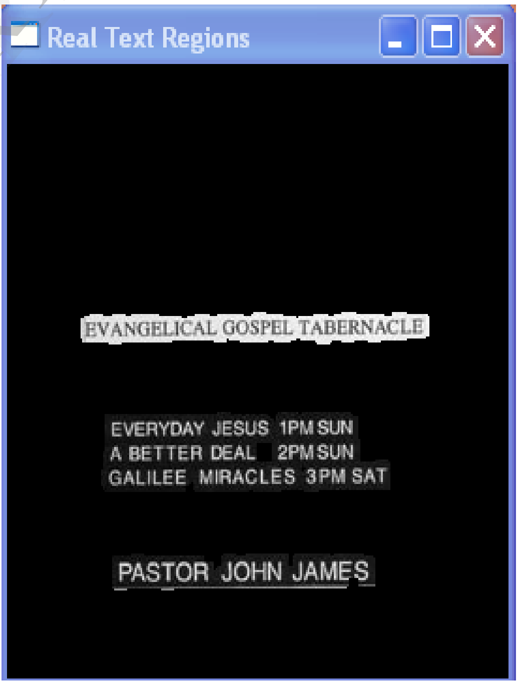

Figure 3.1: b) Real Text Regions

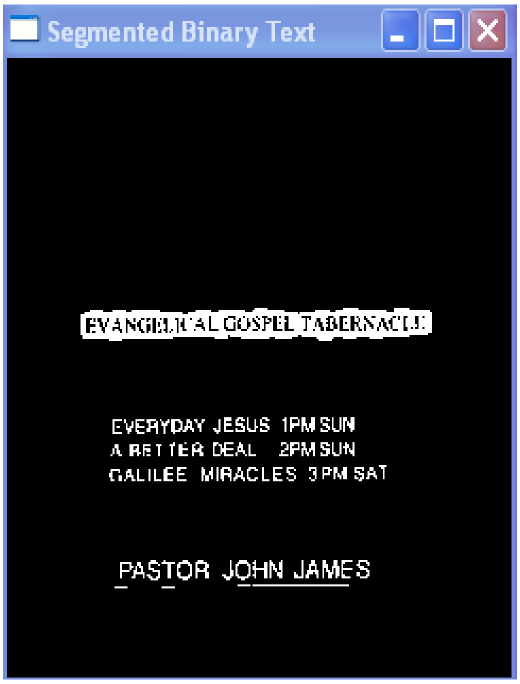

Figure 3.1: c) Segmented Text

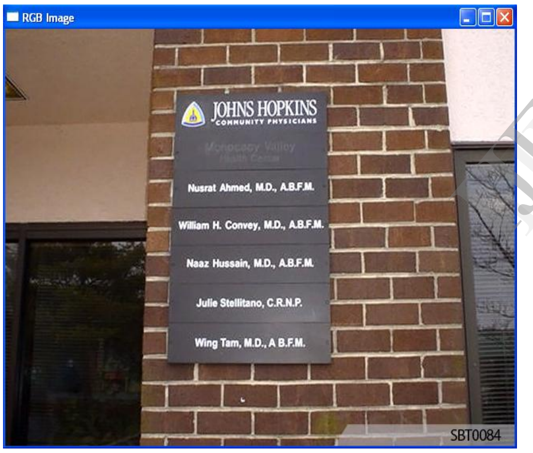
RGB Image
JOHNS HOPKINS
COMMUNITY PHYSICIANS
Monocacy Valley
Health Center
Nusrat Ahmed, M.D., A.B.F.M.
William H. Convey, M.D., A.B.F.M.
Naaz Hussain, M.D., A.B.F.M.
Julie Stellitano, C.R.N.P.
Wing Tam, M.D., A B.F.M.
SBT0084

Figure 3.2: *OutDoor Scene Image*

a) Original Image

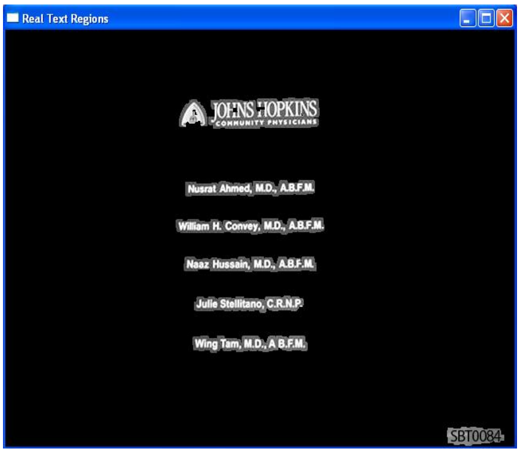
Real Text Regions
JOHNS HOPKINS
COMMUNITY PHYSICIANS
Nusrat Ahmed, M.D., A.B.F.M.
William H. Convey, M.D., A.B.F.M.
Naaz Hussain, M.D., A.B.F.M.
Julie Stellitano, C.R.N.P.
Wing Tam, M.D., A B.F.M.
SBT0084

Figure 3.2: b) Real Text Regions

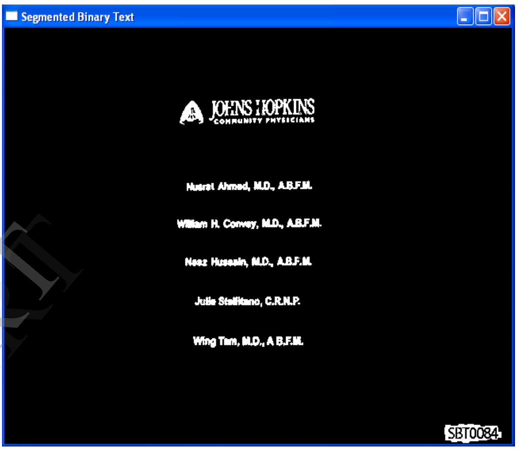
Segmented Binary Text
JOHNS HOPKINS
COMMUNITY PHYSICIANS
Nusrat Ahmed, M.D., A.B.F.M.
William H. Convey, M.D., A.B.F.M.
Naaz Hussain, M.D., A.B.F.M.
Julie Stellitano, C.R.N.P.
Wing Tam, M.D., A B.F.M.
SBT0084

Figure 3.2: c) Segmented Text

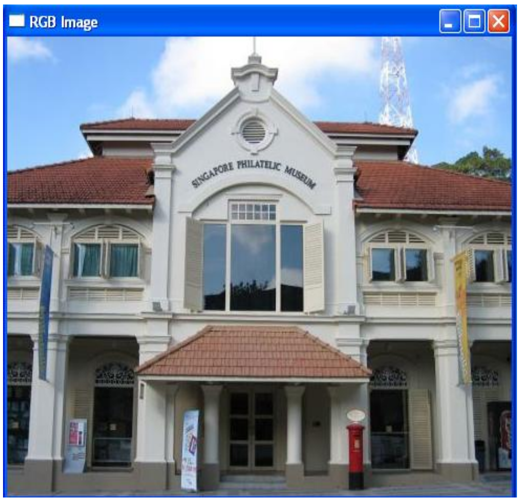

Figure 3.3: *OutDoor Scene Image*

a) Original Image

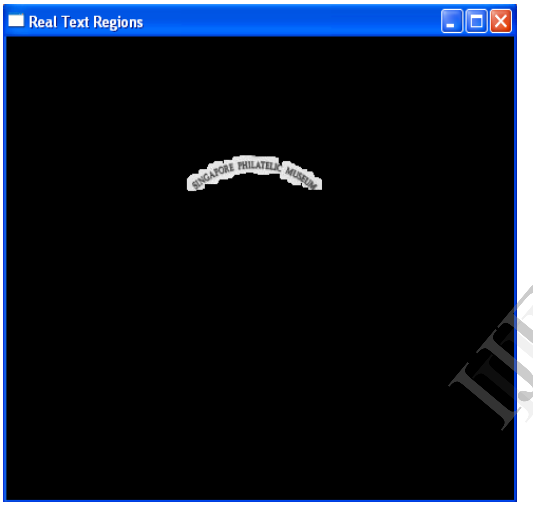

Figure 3.3: b) Real Text Regions

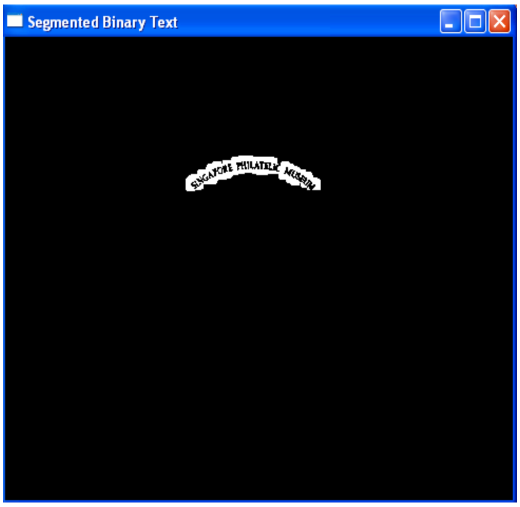

Figure 3.3: c) Segmented Text

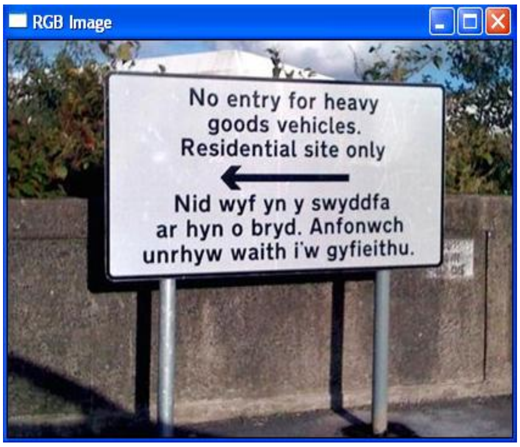

Figure 3.4: *OutDoor Scene Image*

a) Original Image

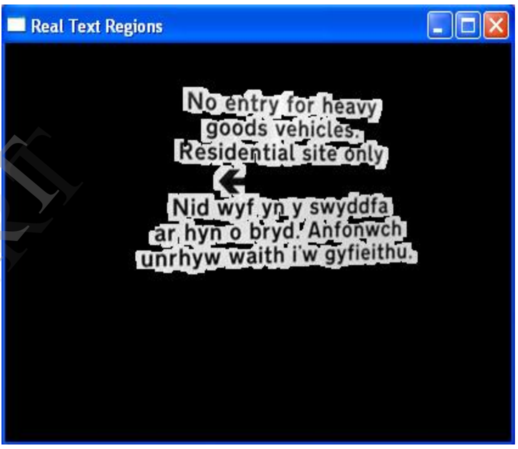

Figure 3.4: b) Real Text Regions

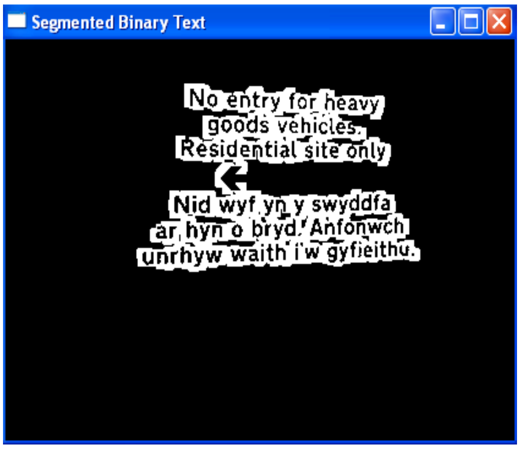

Figure 3.4: c) Segmented Text

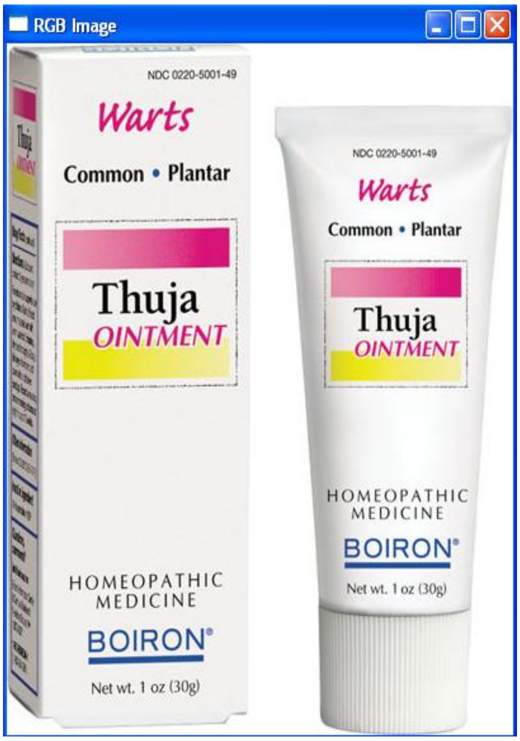

Figure 3.5: *ObjectLabel Image*

a) Original Image

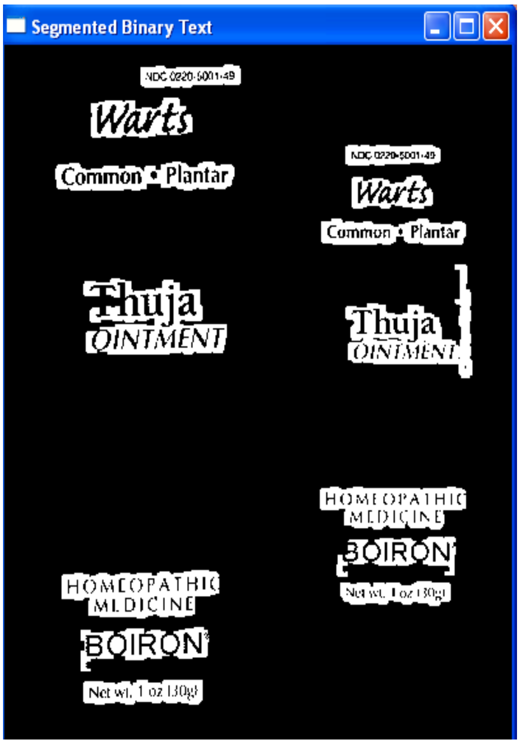

Figure 3.5: c) Segmented Text

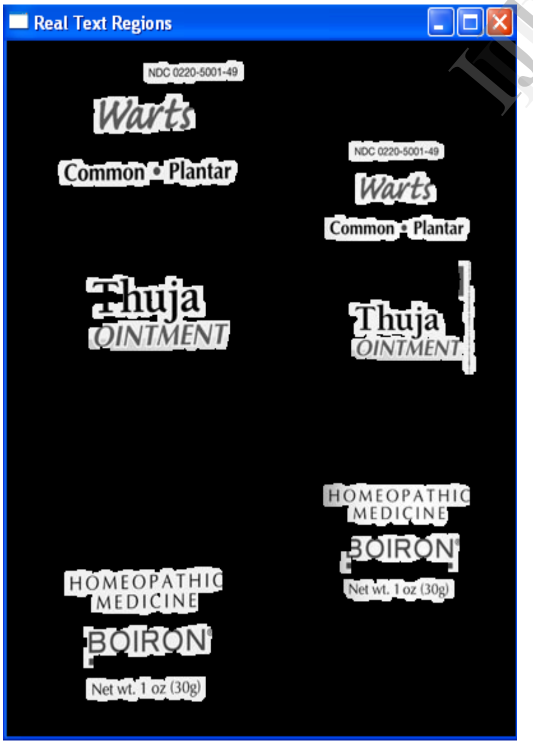

Figure 3.5: b) Real Text Regions

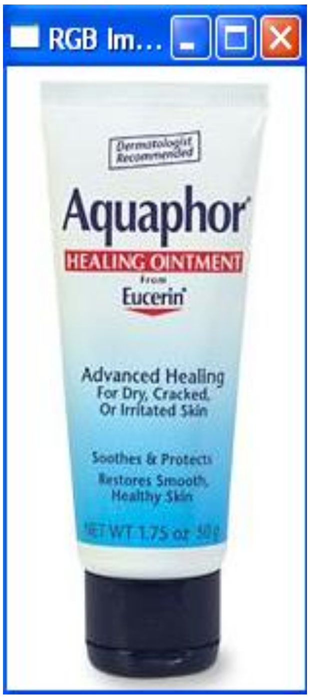

Figure 3.6: *ObjectLabel Image*

a) Original Image

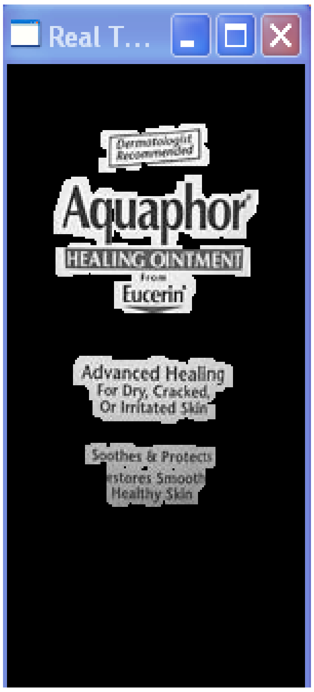

Figure 3.6: b) Real Text Regions

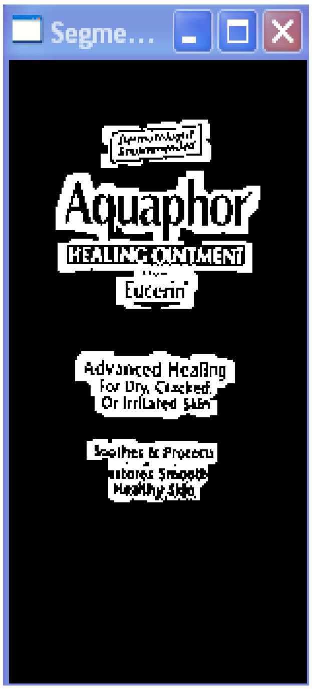

Figure 3.6: c) Segmented Text

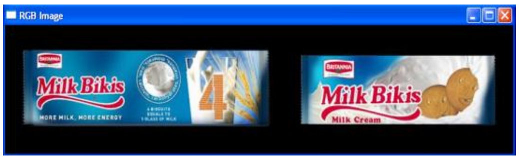

Figure 3.7: *ObjectLabel Image*

a) Original Image

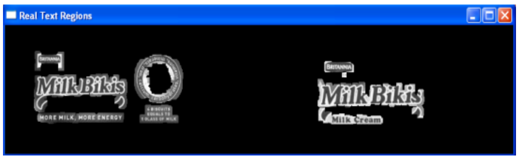

Figure 3.7: b) Real Text Regions

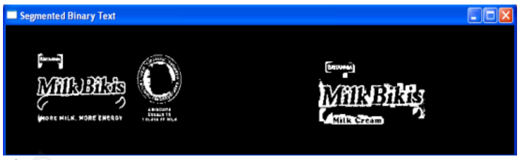

Figure 3.7: c) Segmented Text

In our proposed method, the accuracy of the algorithm output is computed by manually counting the number of correctly located text blocks, which are regarded as ground-truth. Precision Rate and Recall Rate are quantified to evaluate the performance.

$$\text{Precision} = \frac{\text{Correctly Located}}{\text{Correctly Located} + \text{FalsePositive}} \times 100$$

$$\text{Recall} = \frac{\text{Correctly Located}}{\text{Correctly Located} + \text{FalseNegative}} \times 100$$

In Table I, we see the precision rate and recall rate of the proposed method and that of the other existing methods. The performance of our proposed method is excellent overall. Therefore the proposed method is proved to be efficient for extracting the text from the outdoor scene images and from the object label images.

Table I : Performance Comparison

Method	Test Images No.	Text Blocks No.	Correctly Located No.	Precision Rate (%)	Recall Rate (%)
Proposed Method	45	184	175	94.1	97.2
X. Liu et al. [7]	75	75	-	91.8	96.6
J.Samarabandu et al. [6]	25	208	201	91.8	96.6
Wang et al. [3]	325	3597	3314	89.8	92.1
Kim et al. [4]	-	839	645	63.7	82.8
Xi et al. [10]	90	244	231	88.5	94.7
Wolf et al. [12]	60	371	-	-	93.5
Gao et al.[9]	-	823	-	89.9	93.3
Gllavata et al. [11]	326	1104	979	83.9	88.7

In Table II, we provide another evaluation using “False Positive Rate”. It is defined as follows.

$$\text{FalsePositive} = \frac{\text{FalsePositive}}{\text{Correctly Located} + \text{FalseNegative}} \times 100$$

Table II: False Positive Comparison

Method	**Image Type**	**False Positive Rate (%)**
Proposed Method	Outdoor Scene and Object Labels Text	6.1
J.Samarabandu et al. [6]	Indoor Scene Text	5.0
Wang et al.[3]	Outdoor Scene Text	10.5
Xi et al. [10]	Text Captions	12.3
Gllavata et al. [11]	Overlay Text	17.0

4. Conclusions

In this paper, a single scale edge based segmentation algorithm which can automatically detect and extract text from outdoor scene images and object label images is proposed. This method is robust with respect to font sizes, styles, color/intensity, orientations, and alignment of text. According to the experimental results, the proposed method is proved to be effective and efficient for extracting the text regions from the complex images.

5. References

[1] Yassin M.Y., Hasan and Lina J. Karam, “Morphological Text Extraction from Images”, *IEEE Transactions on Image Processing*, Vol. 9, No. 11. November 2000.

[2] Wumo Pan Bui, T.D. Suen, C.Y. , **“**Text detection from scene images using sparse representation ”, *Pattern Recognition,* 2008. ICPR 2008. 19th International Conference on. Dec. 2008, pp. 1-5

[3] Kongqiao Wang and Jari A. Kangas, “Character Location in Scene Images from Digital Camera”, *Pattern Recognition*, Vol. 36, no. 10, pp. 2287 – 2299, 2003.

[4] K.C.Kim, H.R.Byun, Y.J.Song, Y.M. Choi, S.Y. Chi, K.K. Kim and Y.K.Chung, “Scene Text Extraction in natural scene images using hierarchical feature combining and verification”, *Pattern Recognition, 2004, Aug 2004, vol.2 of ICPR 2004. Proceedings of the 17th International Conference on*, pp. 679 – 682.

[5] X. Liu and J.Samarabandu, “A Simple and fast text localization algorithm for indoor mobile robot navigation”, in *Proc. Of the SPIE – IS & T Electronic Imaging*, 2005,San Jose, California, USA, Jan 2005, Vol. 5672, pp.139 – 150.

[6] X. Liu and J. Samarabandu, “An edge-based text region extraction algorithm for indoor mobile robot navigation”, *in Proc. of the IEEE International Conference on Mechatronics and Automation* (ICMA 2005), Niagara Falls, Canada, July 2005, pp. 701 – 706.

[7] X. Liu et al., “Multiscale Edge-based Text extraction from complex images”, 2006 *IEEE, ICME* 2006.

[8] A.K. Jain, *Fundamentals of Digital Image Processing*, Englewood Cliff, NJ; Prentice Hall, 1989, ch.9, pp. 356 – 357.

[9] Jiang Gao and Jie Yang, “An adaptive algorithm for text detection from natural scenes”, in *Computer Vision and Pattern Recognition*, 2001, CVPR 2001, *Proceedings of the 2001 IEEE Computer Society Conference on*, pp. II-84-II-89.

[10] Jie Xi, Xian Sheng Hua, Xiang Rong Chen, Liu Wenyin and Hong Jiang Zhang, “A video text detection and recognition system”, in *Multimedia and Expo*, 2001, ICME 2001, 2001 *IEEE International Con ference on*, pp. 873-876.

[11] J. Gllavata, R. Ewerth and B. Freisleben, “A robust algorithm for text detection in images”, in *Image and Signal Processing and Analysis*, 2003, ISPA 2003, 2003 Proceedings of the 3rd International Symposium on, pp. l611-616.

[12] C. Wolf, J.M. Jolion and F. Chassaing, “Text Localization, enhancement and binarization in multimedia documents”, in *Pattern Recognition*, 2002, Aug 2002, vol.2 of Proceedings. 16th International Conference on. Pp.1037 – 1040.