Morphology-Based Text Line Extraction

Jui-Chen Wu, Jun-Wei Hsieh* and Yung-Sheng Chen
Department of Electrical Engineering
Yuan Ze University, 135 Yuan-Tung Road, Chung-Li 320, Taiwan, R.O.C.
*shieh@saturn.yzu.edu.tw
Tel: 886-3-4638800 ext 2430

Abstract

This paper presents a morphology-based text line extraction algorithm for extracting text regions from cluttered images. A novel morphology-based method is firstly proposed to extract important contrast features as filters to find all possible text region candidates. The contrast feature is robust to lighting changes and invariant to different image transformations like scaling, translation, and skewing. A text line selection process is then applied to locating all possible text-analogue segments from the analyzed image. An *x*-projection technique is used to extract different geometrical properties of characters from the above text-analogue segments. In addition, a verification process is applied to verifying all extracted text candidates according to the regularities of characters embedding in them. Due to noise, a text region is often fragmented to different pieces of segments. For tackling this problem, a novel recovery algorithm is proposed for locating a complete text line. Since the morphology-based method can significantly reduce the number of candidates, all desired text lines can be efficiently and effectively located. The average accuracy of the proposed system is nearly 94%. Experimental results show that the proposed method improves the state-of-the-art work in terms of effectiveness and robustness for text region detection.

Keywords: Morphological Operations, Text Extraction, Text Verification, Identification.

1. Introduction

Extracting text region from images or videos is an important topic in many applications, like document processing, image database indexing, content-based retrieval and so on. For example, the task can extract important text information to annotate an image or video for increasing the performance of a content-based retrieval system and accuracy of indexing and searching. In usual, texts embedded in an image or video show important information of media contexts like the names of players, title, date, story introduction, etc. In addition, for recognizing the audio in a video, the text can provide extra powerful information for correcting the errors of speech recognition.

However, texts embedded in an image or video have many different visual changes including font, size and style, orientation, alignment, texture and color, low contrast and complex background and so on. All these changes will make the problem of automatic text extraction become very difficult. To tackle the above problems, there are a number of researchers who have devoted themselves to investigating different methods for solving this problem. For example, Zhong et al. [1] used a technique of color reduction to locate texts from complex color images. This approach works well based on the assumption that the characters in a text region usually have similar colors. In addition to color feature, Lienhart et al. [2] applied a motion analyze to locate text regions from a video sequence based on the assumption that text regions have high contrast ness to background. Furthermore, Hasan and Karam [4] used different morphological operations to extract edge features for locating texts in images. In addition to extracting features from spatial domain, text regions can also be detected by extracting their frequency features through an analysis of Fourier spectrum. For example, Sin et al. [7] used frequency features such as the number of edge pixels in horizontal and vertical directions and Fourier spectrum to detect text regions in real scene images. Mao et al. [8] took advantages of wavelet transform to extract high-frequency signals as texture feature for detecting text regions. Furthermore, Kim et al. [9] used a learning algorithm, support vector machines (SVM), to learn important features to locate the described text regions. Most existing methods are designed to detect the regions, which are aligned only with the horizontal directions and cannot be embedded into a cluttered background. Once the text regions have different lighting or orientation changes, all the above methods will fail to work.

In this paper, a novel text extraction scheme is proposed for locating desired text regions from images or videos even with cluttered backgrounds. The proposed method is composed of three main steps. In the first, a morphology-based technique is devised to extract useful highly-contrast features as guides to search the desired text pixels. The highly-contrast feature is not easily affected by different lighting changes and can be well extracted even under cluttered background. At the stages of line localization, a labeling technique is applied to finding all possible text-analogue segments. According to the size and width-height-ratio of text-analogue segments, many impossible cases will be eliminated in advance. Then, all remained segments are fed to the last stage, the text candidate verification. At this stage, each possible text region is verified according to their geometrical properties. Without a training process, the geometrical properties can be easily obtained using a projection technique. To detect text regions in any directions, the project technique first uses a moment-based to estimate the orientation of each selected candidate of text region. Then, through considering the long axis of each region as the *x*-axis, the technique uses an *x*-projection to find the period and frequency of texts appearing in the analyzed region. Then, different text geometrical properties including the variances of character width and height are extracted for candidate verification. Due to noise, a text region cannot avoid being segmented into several fragments. Therefore, a recovery algorithm should be developed to reconstruct an incomplete text region to an intact one. The proposed method has good abilities to detect multiple text regions in any orientations. No matter how cluttered the background is, all desired text regions can be very accurately located. In the experiments, 200 images including different text size, text orientations, and cluttered backgrounds were used to examine the accuracy and effectiveness of the proposed system. Among them, there are totally 876 text lines embedded and 820 lines are correctly located. On average, the accuracy of detection rate is 93.6%. Results have confirmed the superiority of the proposed method in text region detection.

The rest of this paper is organized as follows. In the next section, we will first overview the proposed method. Then, details of feature extraction using morphological operations are described in Section 3. The scheme of text line candidate extraction is described in Section 4. Section 5 illustrates details of the algorithm of text region verification. Section 6 reports our experimental results. A conclusion will be finally given in Section 7.

2. Proposed Method

Fig. 1 shows the flowchart of the proposed detection system. The system consists of three main parts: feature extraction, text line extraction, and text candidate verification. In what follows, details of each part are described.

Feature extraction: Text regions embedded in clutter images or video sequences often have the following visual characteristics including high contrast ness to background, uniform color or intensity, and horizontally aligned. To effectively locate them, we exploit the fact that the relative high contrast ness between text and background can be adopted as a key for extracting described text regions from images. A novel morphology-based scheme is applied for extracting all high contrast regions as possible candidates to locate text regions.

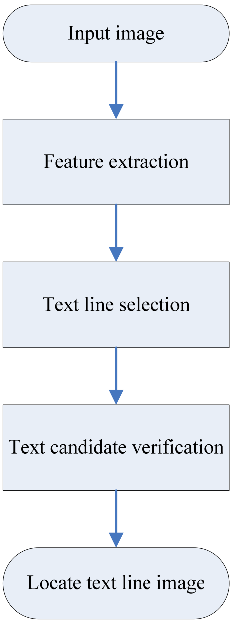


Fig. 1 Flowchart of the proposed system.

Text line selection: Based on the extracted features, a labeling technique is used to locate all possible potential text regions. However, due to noise, many non-text regions may be extracted from cluttered images. In addition, a text region may also be fragmented to different segments. We assume that the text characters located in the same line will have similar heights and width and be aligned along the same direction. Thus a new scheme of text line selection is proposed for locating all possible text regions based on these text geometrical properties.

Text verification: Once all possible text regions have been extracted, a verification procedure should be applied to filtering out all impossible regions. The verification scheme is performed by the measurement of variances of text size and of width-height-ratio. Since each text region has different orientation, a moment-based method is first applied to estimating the orientation of each text region. An x-projection method is then used to find the period frequency of text characters in each region. Thus different text geometrical properties can be verified in this stage. Since a text region is usually fragmented to different pieces of segments due to noise, a text region recovery algorithm is also proposed for recovering a fragmented region to an intact one.

3. Feature Extraction

In practice, text regions are often embedded in a cluttered image or video frame. For well detecting these text regions from cluttered background, this paper presents a novel morphology-based approach to extract high-contrast regions as guide to locate all possible text region candidates. In the past, most approaches are assumed that texts are horizontally aligned along the x-axis. They located possible text regions by projecting the vertical edges of texts on the x-axis to form different response peaks. However, this projection approach is easily affected by noise and fails to work when text lines are not aligned along the x-axis. Since the high-contrast feature is invariant to text orientations and has good resistances to noise. In what follows, details of the morphology-based scheme to locate high constant features are described.

3.1 Morphological Operations

In this stage, we used several morphology operations to extract highly contrast features from cluttered images. Before describing the related morphology operations, we denote $B_{m,n}$ as a strutting element with size $m \times n$, where m and n are odds and larger than zero. Two kinds of different structuring elements are shown in Fig. 2. Let $f(x, y)$ denote a grayscale input image.

$B_{1,5}$ =

1	1	1	1	1

(a)

$B_{3,7}$ =

1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1

(b)

Fig. 2 Two kinds of structure elements.

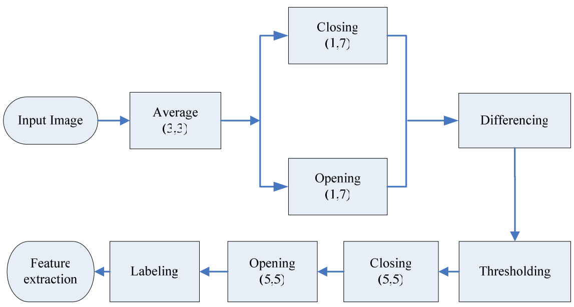


Fig. 3 Flowchart of the proposed method to detect the high contrast features of text regions using morphological operations.

The whole procedure of morphology-based feature extraction is shown in Fig. 3. For eliminating noise, a smoothing operation with a structure element $B_{3,3}$ is first applied. To detect a highly contrast features, the closing and the opening operations with a structure element $B_{1,7}$ are performed respectively on the smoothed image, thus the corresponding images I_c and I_o can be obtained. Here the closing operation is to find the low intensity features, whereas the opening operation is to find high intensity features. To detect the vertical edges, a differencing operation into image I_c and I_o is performed. All possible vertical edges can be extracted by a thresholding operation. Since text lines are aligned horizontally and each character is sufficiently close to its adjacent characters from each text line, a closing operation with a structure element $B_{5,5}$ can be applied to combine vertical edges from the thresholded image, and an opening operation with a structure element $B_{5,5}$ is applied for noise removal. After that, a set of potential text segments can be obtained from a cluttered image. Finally, a connected component analysis [11] is performed on the segmented potential text lines. Fig. 4 illustrates the result of extracted features.

Fig. 4 Result after morphological operation. (a) and (b) original image. (c) and (d) result of feature extraction.

The operations used in the procedure of our morphology-based method include smoothing, dilation, erosion, closing, opening, differencing, and thresholding are defined as follows:

- Smoothing operations:

$$E_{B_{m\times n}}\left(f(x,y)\right)=\frac{1}{mn}\sum_{i=-n/2}^{n/2}\sum_{j=-m/2}^{m/2} f(x+i,y+j)\,B_{m\times n}(i,j),$$

- Dilation operations:

$$f(x,y)\oplus B_{m\times n}=\max_{|i|\le m/2,|j|\le n/2} f(x-i,y-j)\,B_{m,n}(i,j),$$

- Erosion operations:

$$f(x,y)\odot B_{m\times n}=\min_{|i|\le m/2,|j|\le n/2} f(x-i,y-j)\,B_{m,n}(i,j),$$

- Closing operations:

$$f(x,y)\bullet B_{m\times n}=\left(f(x,y)\oplus B_{m,n}\right)\odot B_{m,n},$$

- Opening operations:

$$f(x,y)\circ B_{m\times n}=\left(f(x,y)\odot B_{m,n}\right)\oplus B_{m,n},$$

- Differencing operations:

$$D(f_1,f_2)=\left|f_1(x,y)-f_2(x,y)\right|,$$

- Thresholding operations:

$$T\left(f(x,y)\right)=\begin{cases}255, & \text{if } f(x,y)>T\\ 0, & \text{otherwise.}\end{cases}$$

Fig. 4 shows an example of high contrast region extraction using morphological operations. (a) and (b) are the original images, whereas (c) and (d) are the detection result of high contrast regions using our proposed method. Obviously, all possible text candidates were well detected.

4. Text Line Selection

In Section 3, we have described a novel scheme to extract different high contrast areas from an image using several morphological operations. However, due to noise, many non-text regions will also be extracted. In addition, a text region will be fragmented to different small segments. More importantly, different text regions have different orientations. To tackle the above problems, the orientation of each text region should be first estimated. Then, a merging technique can be applied to linking all fragments coming the same text region together and thus to finding all possible text line candidates. In what follows, details of region orientation estimation are first described. Then, the scheme of text line candidate extraction is illustrated in Section 4.2.

4.1 Orientation Estimation by Moments

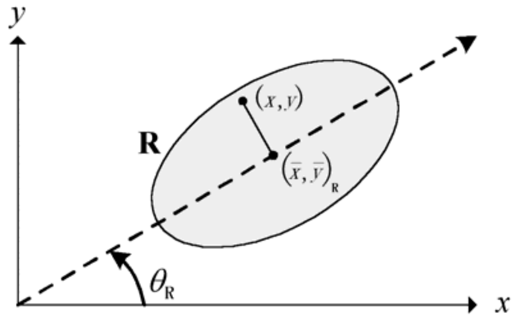


Fig. 5 The gravity center $(\bar{x},\bar{y})_R$ and orientation θ_R of the object R.

Given a two-dimensional binary region $R(x,y)$, the central moments of R can be defined as

$$(\mu_{p,q})_R=\sum_{(x,y)\in R}(x-\bar{x})^p\,(y-\bar{y})^q,$$

where $(\bar{x},\bar{y})=(\frac{1}{|R|}\sum_{(x,y)\in R}x,\frac{1}{|R|}\sum_{(x,y)\in R}y)$ and $|R|$ is the area of R. Here, if a pixel (x, y) belongs to R, the value of $R(x,y)$ is one; otherwise, its value is zero.

Then, as shown in Fig. 5, the orientation θ_R of R can be obtained as follows:

$$\theta_R = arg\ \min_{\theta} \sum_{(x,y)\in R} \left[(x-\bar{x})\sin\theta - (y-\bar{y})\cos\theta \right]^2 . \quad (1)$$

Setting the term $\frac{1}{\partial\theta}\sum_{(x,y)\in R}\left[\left(x-\bar{x}\right)\sin\theta\right]^2$ and $\frac{1}{\partial\theta}\sum_{(x,y)\in R}\left[(y-\bar{y})\cos\theta\right]^2$ to zero, we can get

$$\theta_R = \frac{1}{2}\tan^{-1}\left[\frac{2\mu_{1,1}}{\mu_{2,0}-\mu_{0,2}}\right]. \quad (2)$$

Then, according to Eq. (2), the orientation of R can be obtained.

4.2 Text Line Candidate Extraction

Let R be a potential text region extracted using our morphology-based scheme. Due to noise, many impossible non-text regions are also extracted. Therefore, a rule-based method is first used to eliminate impossible text regions. Assume that R^{θ_R} is the rotated version of R with the angle θ_R, where θ_R is the major orientation of R. Then, we can estimate the width $w_{R^{\theta_R}}$ and height $h_{R^{\theta_R}}$ of R^{θ_R}, respectively. Since a text region usually has a longer width than height, the first rule requires the ratio between $w_{R^{\theta_R}}$ and $h_{R^{\theta_R}}$ being larger than 1.5. In addition, the density of R should be not smaller, i.e., $den = \frac{Area\ of\ R}{w_{R^{\theta_R}} \times h_{R^{\theta_R}}} > 0.1$. The final rule requires the area of R being larger enough. According to the above requirements, R is said to be a candidate of text region if it satisfies the following three rules:

Rule 1: *den* should be larger than 0.1;

Rule 2: the ratio between $w_{R^{\theta_R}}$ and $h_{R^{\theta_R}}$ should be larger than 1.5;

Rule 3: the area of R should be larger than a threshold.

After filtering out some impossible text regions using the rule-based approach, we propose a novel merging scheme to link all fragmented text regions together if they come from the same text region. Given two regions R_i and R_j, four similarity measures are defined in this paper for determining whether they belong to the same text region. In practice, if R_i and R_j come from the same text region, their heights and orientations should be similar. In addition, their centroids and major axes are close to each other. Therefore, if R_i and R_j will be merged, we first require the orientation difference between them being less than 10°, i.e.,

$$d_\theta(R_i,R_j) = \min(|\theta_{R_i} - \theta_{R_j}|, |\theta_{R_i} - \theta_{R_j} + 360^\circ|, |\theta_{R_i} - \theta_{R_j} - 360^\circ|) < 10^\circ, \quad (3)$$

where θ_{R_i} and θ_{R_j} are the major orientations of R_i and R_j, respectively. In addition, the distance between the heights of R_i and R_j should be smaller enough, i.e.,

$$d_h(R_i,R_j) = \frac{2\,|h_{R_i^{\theta_{R_i}}} - h_{R_j^{\theta_{R_j}}}|}{h_{R_i^{\theta_{R_i}}} + h_{R_j^{\theta_{R_j}}}} < 0.15, \quad (4)$$

Let Cen_{R_i} and Cen_{R_j} be the centroids of R_i and R_j, respectively. Then, the distance between Cen_{R_i} and Cen_{R_j} is defined by their Euclidian distance and should be less than the double summation of the heights of R_i and R_j, i.e.,

$$d_{\mathrm{Cen}}(R_i,R_j) = |\,\mathrm{Cen}_{R_i} - \mathrm{Cen}_{R_j}\,| < 2(h_{R_i^{\theta_{R_i}}} + h_{R_j^{\theta_{R_j}}}), \quad (5)$$

Let L_R be the longest axis of R and represented by this equation: $y = m_R x + b_R$. Then, the distance between the major axes L_{R_i} and L_{R_j} of R_i and R_j can be defined by:

$$d_L(R_i,R_j) = \frac{|\,y_{Cen_{R_j}} - m_{R_i}x_{Cen_{R_j}} - b_{R_i}\,|}{2\sqrt{1+m_{R_i}^2+b_{R_i}^2}} + \frac{|\,y_{Cen_{R_i}} - m_{R_j}x_{Cen_{R_i}} - b_{R_j}\,|}{2\sqrt{1+m_{R_j}^2+b_{R_j}^2}}, \quad (6)$$

where x_{Cen_R} and y_{Cen_R} are the coordinates of the centroid Cen_R in the *x* and *y* directions, respectively. Based on Eq. (6), the fourth requirement is to require the distance $d_L(R_i,R_j)$ being less than 5 pixels, i.e.,

$$d_L(R_i,R_j) < 5. \quad (7)$$

According to the requirements defined in Eqs. (3), (4), (5) and (7), whether two region candidates R_i and R_j should be merged can be determined. Thus, different text line candidates can be well extracted for further verification. Fig. 6 shows one example of text line detection after region merging. Clearly, all possible text regions are well extracted.

(a) (b)

Fig. 6 Results of text line selection.

5. Text Line Verification

After extracting all possible text lines from the analyzed image, a verification process should be applied for ensuring whether they are real text lines. In this section, an *x*-projection scheme is presented to estimate different text geometrical properties from each text line candidate. With the help of the text properties, all desired text regions can then be identified correctly. In what follows, details of the verification scheme are described.

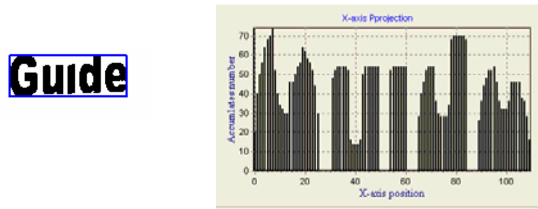
5.1 X-Projection

Before analyzing, a "*minimum within-group variance*" dynamic thresholding method [10] is applied to binarizing each input region. Since each processed text region is different orientation, the verification scheme takes its longest axis as the *x*-axis to perform an *x*-projection for finding all desired text geometrical properties. The *x*-projection accumulates the numbers of text pixels along *x*-axis column by column. Then, according to the result of *x*-projection, different text characters can be well separated and used for text verification. In particle, for all characters in a text line, they should have the following properties:

A1: their widths should be similar.

A2: their heights should be similar.

A3: their center should be aligned a straight line.



(a) (b)

Fig. 7 Characters in a text line. (a) Original image; (b) Result of *X*-axis project.

The first rule implies that the period of a text region obtained from the result of *x*-projection should correspond to the width of character. The period can be obtained by tracing the minimum valleys of the result of *x*-projection. Like Fig. 7, (b) is the result of *x*-projection obtained from (a). The widths between any two adjacent minimum valleys will correspond to different character widths like 'g', 'u', 'i', 'd', and 'e'. Assume that $XP_R[i]$ is an array to record the result of *x*-projection gotten from the text region R. Then, the minimum valleys in XP can be detected by seeking the local minimum point whose value in XP is less than $h_{R^{R\theta}}/8$. According to the position of each minimum valleys, different vertical lines are be drawn such that R can be separated to different characters C_i. Assume that $\overline{W_R}$ and $\overline{h_R}$ are the average width and height of characters embedded in R. Then, the variances of character width in R can be calculated as followings

$$\sigma^2_{W,R} = \frac{1}{N_R^C}\sum_{C_i \in R}\left(W_{Ci} - \overline{W}\right)^2. \tag{8}$$

Similarly, the variance of character height in R is

$$\sigma^2_{n,R} = \frac{1}{N_R^C}\left(h_{Ci} - \overline{h}\right)^2. \tag{9}$$

In addition, all centers of characters C_i in R form a line. Assume that this line is formulated by:

$$y = m_R^C x + b_R^C, \tag{10}$$

where the value of m_R^C and b_R^C can be obtained by a line fitting model. Then, the linearity of this line can be measured by:

$$L_{inearity(R)} = \frac{1}{\left|N_R^C\right|}\sum_{C_i \in R}\frac{\left|m_R^C x_{Ci} + b_R^C y\right|}{\sqrt{\left(m_R^C\right)^2 + \left(b_R^C\right)^2}}. \tag{11}$$

Thus, if R is a text line region, it should satisfy

$$\sigma_{R,w}^2 < th_w,\ \sigma_{R,L}^2 < th_h, \text{and } linearity(R) < th_l.$$

Then, different text lines can be correctly verified from R for further images and video analysis.

5.2 Text Line Recovery

In the previous section, we have used an *x*-projection method to extract different text line regions from images or video frames. However, due to noise, the extracted candidates may be fragments of a complete text region. Therefore, in this section, a text line recovery algorithm is proposed for recovering the whole text line region from these fragments using character information. As described in the previous section, characters are often embedded in a text line with the similar size, font, and intensity. According to this observation, a whole text region will be recovered from its pieces.

Assume that *R* is the processed incomplete text region and R^{θ_R} is its rotated version with the angle θ_R, where θ_R is the major orientation of R. The whole recovery algorithm is performed on R^{θ_R} by considering the major axis of *R* as the *x* direction and the center of R^{θ_R} as the original. With the transformation, all the text characters in R^{θ_R} will be horizontally aligned. Then, a horizontally extending technique can be used for recovering *R* to a complete one. Let l_{R^θ}, r_{R^θ}, t_{R^θ}, and b_{R^θ} denote the most left, right, top, and bottom coordinates of R^{θ_R} in the *x* and *y* directions, respectively. In addition, th_R is the threshold used to binarize R to a binary map. The recovery algorithm is iteratively performed. At each iteration, we create two new regions extended from both sides of R^{θ_R} with the same height of R^{θ_R} and a fixed width. The fixed width is proportion to the average width of characters embedded in R^{θ_R}. Then, the threshold th_{R^θ} is used to binarize these two new regions from which different isolated characters can be extracted. An isolated character is a component extracted from the extended regions using a connected component analysis and having similar width and height to $\bar{w}_{R,C}$ and $\bar{h}_{R,C}$. If no characters is further isolated, the iterative process will be terminated. The retrieved region will be the final desired text line. In what follows, the details of text line recovery algorithm are described.

Procedure Text Line Recovery Algorithm

Input: a text region *R*, the average character width $\bar{w}_{R,C}$ and height $\bar{h}_{R,C}$ in *R*, the threshold th_R to binary *R*, and the width W_I and height H_I of input image;

Output: a new recovered region $\bar{R}$.

Step 1: According to the major orientation of *R*, obtain its rotated version R^{θ_R}.

Step 2: Obtain the most left, right, top, and bottom coordinates of R^{θ_R}, i.e., l_{R^θ}, r_{R^θ}, t_{R^θ}, and b_{R^θ}, respectively by considering the major axis of R^{θ_R} as the *x* direction.

Step 3: Create a new region for left extension

- S3.1: Create a new region R_{left}^{New} with the boundary coordinates: l = max (0, l_{R^θ} - 5 $\bar{w}_{R,C}$), r = l_{R^θ} , t = max(0, t_{R^θ} - $\bar{h}_{R,C}$/5), and b = min ($b_{R^\theta} + \bar{h}_{R,C}$/5, H_I).
- S3.2: Binarize R_{left}^{New} using the threshold th_R.
- S3.3: Check whether there are isolated characters in R_{left}^{New} using a connected component analysis.

S3.4: If any isolated characters exist, l_{R^θ} = the most left x-coordinate of pixels in them and go to Step 3; otherwise, go to Step 4.

Step 4: Create a new region for right extension

S4.1: Create a new region R^{New}_{right} with the boundary coordinates: l = r_{R^θ} , r = $\min(r_{R^\theta} + 5\bar{w}_{R,C}, W_I)$, $t = \min(t_{R^\theta} - \bar{h}_{R,C}/5,$ $0)$, and $b = \max(b_{R^\theta} + \bar{h}_{R,C}/5,)$.

S4.2: Binarize R^{New}_{right} using the threshold th_R .

S4.3: Check whether there are isolated characters in R^{New}_{right} using a connected component analysis.

S4.4: If any isolated characters exist, r_{R^θ} = the most right x-coordinate of pixels in them and go to Step 4; otherwise, go to Step 5.

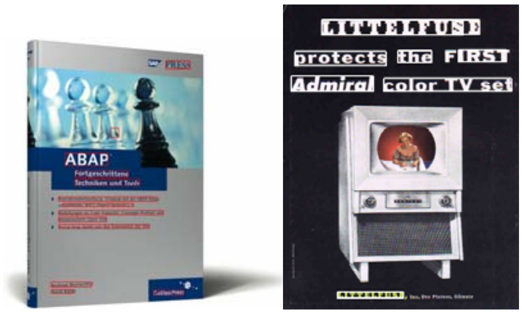
Step 5: Obtain $\bar{R}$ with the new boundary coordinates: l_{R^θ} , r_{R^θ} , t_{R^θ} , and b_{R^θ} .

6. Experimental Results

To analyze the performance of the proposed approach, 200 images were used for testing. For demonstrating the superiority of our proposed method, these images include various visual changes like different contrast, complex backgrounds, and different fonts and sizes.

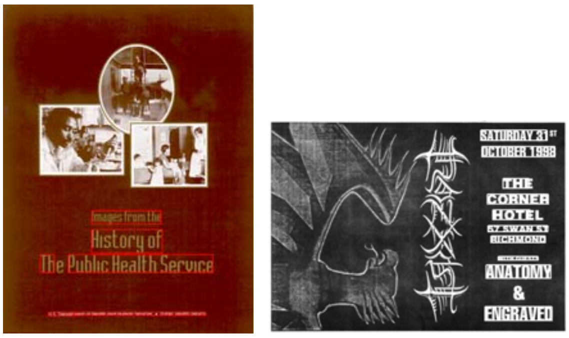
Fig. 8 shows the result of text line extraction from normal images using our proposed method. Fig. 9 shows another result of images with different visual changes. In (a), the contrast between text regions and background is very low. However, our method still performed good to detect all desired text regions. (b) is the detection result of texts embedded in a cluttered background. Even with a very complicated background, our method still worked very successfully to detect all expected text regions.

For comparison, the method proposed by Hasan and Karan [4] was implemented. Fig. 10 shows the comparison result. (a) is the result of text detection using the method of Hasan and Karan. Their method is very sensitive to the existence of noise. (b) is the one obtained by our method. Clearly, our method performs much better than their one [4].



(a) (b)

Fig. 8. Results of text detection from normal images using our morphology-based method.



(a) (b)

Fig. 9 Results of text detection from images with various visual changes. (a) Result obtained from low-contrast image. (b) Result obtained from a cluttered image.

(a) (b)

Fig. 10 Comparison result of text detection between the method of Hasan et al. [4] and our method. (a) Result obtained by the method of Hasan and Karam. (b) Result of text detection using our method.

In order to evaluate performance, two performances were used in this paper, i.e., recall and precision. Recall is the ratio of the total number $Num_{Correct}$ of characters correctly detected by the algorithm to the total number Num_{actual} of actual characters appearing in the test image, i.e.,

$$Recall = \frac{Num_{Correct}}{Num_{actual}}.$$

Precision is the ratio of the total number of characters correctly detected by the algorithm to the total number $Num_{Detected}$ of detected characters, that is,

$$Precision = \frac{Num_{Correct}}{Num_{Detected}}.$$

Table 1 shows the performance comparisons of recall and precision between the method in [4] and our one. From this table, it is clear that our algorithm performed better than the one of Hasan and Karan. The average accuracy of our proposed method is nearly 94%. All the experimental results have confirmed the feasibility and superiority of the proposed algorithm.

Table 1 Performance comparison.

Method	Recall	Precision
Our method	94.1%	93.6%
Hasan and Karan	91.4%	84.5%

7. Conclusions

In this paper, we have proposed an automatic text extraction algorithm to extract text regions in cluttered images. In the first stage, morphological operations were performed to extract the text-analogue segments. Based on their geometries features and horizontal aligning, possible text lines were located from text-analogue segments. Finally, possible text lines were verified according to their aligned direction and profile histogram. Since all true text lines were identified, their corresponding aspects are extended based on their textual information in order to completely and correctly extract text line. The morphology-based text-analogue segmentation process can effectively locate potential text-analogue regions, which cannot easily be affected with different lighting conditions. Therefore, no matter how text of different fonts, sizes and color appear in the same image, the proposed method still works well.

Acknowledgments

This work was partially supported by National Science Council of Republic of China under grant NSC 94-2213-E-155-043.

References

[1] Y. Zhong, K. Karu, and A.K. Jain, "Locating text in complex color images," *Pattern Recognition*, Vol. 28, No. 10, pp. 1523-1536,Oct. 1995.

[2] R. Lienhart, F. Stuber, "Automatic text recognition in digital videos," *Proceeding of SPIE*, pp.180-188, 1996.

[3] R. Lienhart and W. Effelsberg, "Automatic text segmentation and text recognition for video indexing," *Technical Report TR-98-009, Praktische Informatik IV*, University of Mannheim, 1998.

[4] Y.M.Y. Hasan and L.J. Karam, "Morphological text extraction from images," *IEEE Trans. On Image Processing*, Vol. 9, pp. 1978-1983, 2000.

[5] A. K. Jain and B. Yu, "Automatic text location in images and video frames," *Pattern Recognition*, Vol. 31, No.12, pp-2055-2076, 1998.

[6] M.A. Smith and T. Kanado, "Video skimming

for quick browsing based on audio and image characterization," *Technical Report CMU-CS-95-186*, Carnegei Mellon University, July. 1995.

[7] B. Sin, S. Kim, B. Cho, "Locating characters in scene images using frequency features," *Proceeding of International Conference on Pattern Recognition*, Vol. 3, Quebec, Canada, pp. 489–492, 2002.

[8] W. Mao, F. Chung, K. Lanm, W. Siu, "Hybrid Chinese/English text detection in images and video frames," *Proceeding of International Conference on Pattern Recognition*, Vol. 3, Quebec, Canada, pp. 1015–1018, 2002.

[9] K.I. Kim, K. Jung, S.H. Park, and H.J. Kim, "Support vector machine-based text detection in digital video," *Pattern Recognition*, pp.527–529, 2001.

[10] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision,* London, U. K.: Chapman & Hall, 1993.

[11] L.D. Stefano and A. Bulgarelli, "A simple and efficient connected components labeling algorithm," *IEEE Proc. Of the 10th International Conference on Image Analysis and Processing*, pp.322-327, 1999.