

# Apply Adaptive Threshold Operation and Conditional Connected-component to Image Text Recognition

Chuen-Min Huang\*, Yu-Kai Lin, Rih-Wei Chang

Department of Information Management, National Yunlin University of Science & Technology, 123, University Rd., Sec. 3, Douliou, Yunlin 64002, Taiwan

\*Corresponding Author: [huangcm@yuntech.edu.tw](mailto:huangcm@yuntech.edu.tw)

Copyright © 2014 Horizon Research Publishing All rights reserved.

**Abstract** How to effectively extract text from an image is a critical issue in the text recognition domain. Due to the variety of background components, for example, different kind of colors, texture, or brightness in an image will deteriorate the problem of text recognition. In this research, we applied "adaptive threshold operation" and "conditional connected-component" to deal with non-uniform lightness and complicated background images. Different from the general procedure of using the whole image to separate the background from the objects, our research adopted the divide and merge strategy to tackle this problem. Instead of segregating the grayscale image into many regions, our approach partitioned an image into three equal-sized horizontal segments to identify the local threshold value of each segment efficiently. With this approach, we successfully identified and recognized texts from an image. The result shows that the rates of object identification and recognition achieve 81.17% and 91.30%, respectively.

**Keywords** Text recognition, Image pre-processing, Adaptive Threshold, Conditional connected-component, OCR

---

## 1. Introduction

There is a large amount of text in images in our daily life. Those texts often contain important or useful information, such as title, location, indication, or even advertisement. With the progress of handheld devices, a convenient photographing function has been embedded in the device and can be used in multiple applications, such as codes scanning or dangerous warning notification. Based on the reason, the need of decoding, text extraction and image recognition is obvious[1]. Optical Character Recognition(OCR) is frequently used to identify the text on books or in documents[2]. With the explosion of handheld devices, the need of text recognition in an image, such as signs or posters is continuously rising[3].

Many terms have been used to describe images, including

pictures, graphics, drawings, photographs, digitized data, and visual resources. Here, an image is defined as a pictorial representation that conveys visual or abstract properties. An image usually consists of background context and foreground texts. Due to the variety of background components, for example, different kind of color, texture, or brightness in an image, text visibility is hence seriously affected. Notwithstanding the prominent improvement of the OCR technique at the present stage, there are still problems that would affect the result of recognition including the complicated background and non-uniform lightness situations[4-6]. Some studies utilized machine-learning based method or edge detection approach to solve the problems, while the effects are limited[7-9].

To tackle these problems, we focused on the image pre-processing part to deal with the stains and unwanted objects taken from the shooting and the computation complexity of the images. Different from the general procedure of image pre-processing, our research used the concept of divide and merge strategy to facilitate the process. We partitioned an image into three horizontal segments and merge them afterwards. An adaptive threshold operation was proposed to deal with the problem of non-uniform lightness. Further, we applied the conditional connected-component to discard the unimportant or redundant background via condition setting. The result showed that the accuracy rate of recognition was raised and our treatment outperforms other studies.

The remaining part of the paper is outlined as follows. Section 2 describes the related work and techniques applied in this study. Section 3 addresses our research architecture and the processes. Experimental results are explained in Section 4. Finally, we conclude this study and suggest future work in Section 5.

## 2. Applied Technology

### 2.1. Image Binarization

The purpose of image pre-processing is to execute the

signal operation of any input image. It is believed that the performance of image pre-processing has significant influence on image recognition and final results[10].

To conduct image pre-processing, a color image is suggested to convert into grayscale for the reduction of the computational complexity as well as memory requirements. After that, a series of steps including smoothing and contrast enhancement shall be taken into consideration.

In the grayscale-level histogram, there are two wave crests representing the areas of foreground and background of an image. Figure 1 shows the distribution of grayscale value. In the diagram, there are two wave crests represented the class of foreground and background. Between the two wave crests, there exists the best threshold for separating foreground and background appropriately.

Otsu's experiment is one of the most used methods to perform clustering-based image thresholding [11], in which an optimum threshold will be identified to separate the two classes of the histogram so that their intra-class variance is minimal and the inter-class variance is maximal.

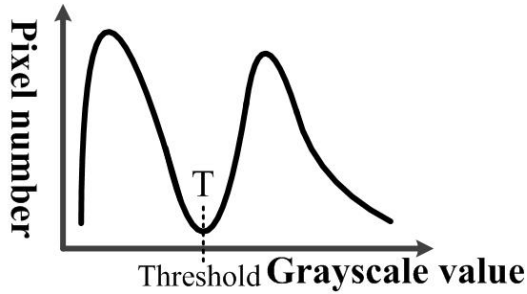


Figure 1. Grayscale-level histogram

Assume  $C_0$  and  $C_1$  stand for the class of foreground and background, respectively. The grayscale value distribution of  $C_0$  is  $\{0, 1, 2, 3, \dots, T\}$ , and the grayscale value distribution of  $C_1$  is  $\{T+1, T+2, T+3, \dots, L\}$ .  $T$  represents the threshold point;  $L$  means 255, the maximum value of grayscale. Equations (1) to (5) show the corresponding calculation.

$$P_i = \frac{n_i}{N} \quad (1)$$

$$W_0 = \sum_{i=0}^T P_i, \quad W_1 = \sum_{i=T+1}^L P_i \quad (2)$$

$$m_0 = \frac{\sum_{i=0}^T i * P_i}{W_0}, \quad m_1 = \frac{\sum_{i=T+1}^L i * P_i}{W_1} \quad (3)$$

$$\sigma_0^2 = \frac{\sum_{i=0}^T (i-m_0)^2 * P_i}{W_0}, \quad \sigma_1^2 = \frac{\sum_{i=T+1}^L (i-m_1)^2 * P_i}{W_1} \quad (4)$$

$$\sigma_w^2 = W_0 \sigma_0^2 + W_1 \sigma_1^2 \quad (5)$$

Where  $P_i$  denotes the probability of the  $i$ th shade of gray in an image.  $W_0$  and  $W_1$  indicate the probabilities of  $C_0$  and  $C_1$  separated by a threshold  $T$ .  $N$  is the total number of pixels of the image;  $n_i$  represents the counts of the  $i$ th shade of gray. The mean ( $m_0$ ,  $m_1$ ) and variance ( $\sigma_0^2$ ,  $\sigma_1^2$ ) of  $C_0$  and  $C_1$  will be calculated in order to acquire the total sum

of variance  $\sigma_w^2$ .  $\sigma_0^2$  indicates the variance of the shades of gray in the background (below threshold);  $\sigma_1^2$  represents the variance of the shades of gray in the foreground (above threshold). The purpose of Otsu's method is to find a  $T$  value for minimizing  $\sigma_w^2$  and then it can be the best threshold.

## 2.2. Filter

In signal processing, it is often desirable to be able to perform some kind of noise reduction on an image or signal. To reduce noise in an image, a couple of studies have applied in the smoothing filter to solve the geometric distortion [12]. There are two common types of filter: one is "smoothing filter" and the other is "sharpening filter". The principle of smoothing filter is to average the grayscale values of mask operation. Then, the average value will be a substitute for all the corresponding pixels to eliminate noise. Sharpening filter functions as image sharpening and contrast enhancement. The major purpose of image sharpening is to magnify edges or regions, consequently the contour of an image can be more significant after sharpening.

There are many kinds of filters, such as, low-pass, median filter, or high-pass. Low-pass filters and median filters are used most often for noise suppression or smoothing, while high-pass filters are typically used for image enhancement[13]. It is assumed that if we enhance the image contrast, the feature area will be easily represented. However, after several trials, we found that not all images are suitable to perform contrast enhancement. In high brightness contrast situation, the inner details of an object in an image will totally disappear after performing contrast enhancement. Due to this reason, we will not perform contrast enhancement in the stage of image preprocessing.

Median filter is a nonlinear digital filtering technique, under certain conditions, it preserves edges while removing noise. By far, the majority of the computational effort and time is spent on calculating the median of each window. Because the filter must process every entry in the signal, for large signals such as images, the efficiency of this median calculation is a critical factor in determining how fast the algorithm can run. For this reason we adopted the low-pass filter in the experiment.

## 2.3. Connected-Component Labeling

Considering the regions of adjacent pixels share a similar set of intensity values, therefore connected-component labeling is often used to detect the connected regions in an image [14] and the connectivity can be 4-connected or 8-connected shown as Figure 2. The full utility of connected-component labeling can be realized in an image analysis scenario wherein images are pre-processed via some segmentation or classification scheme. Following the labeling stage, the image may be partitioned into subsets, after which the original information can be recovered and processed.

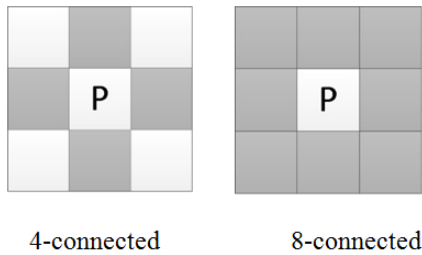


Figure 2. Connected component

Connected-component labeling could not only be used to identify objects, but also describe their location, height, width, or density information. Figure 3 displays the connected-component labeling. For a binary image, we can label the white pixels with a unique symbol to find interesting objects in an image.

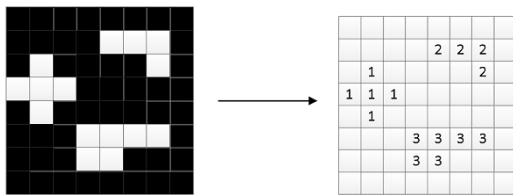


Figure 3. Connected-component labeling

### 2.4. Morphology

Morphology[15] is a principle to describe space structure with set theory. Mathematical morphology proposed by G. Matheron and J. Serra in 1989[16] is used to extract the shape feature of a graph and reduce noise. It includes four operators: dilation, erosion, open, and close. Dilation and open operators are usually used to launch the edge enhancement and fill the broken pixel. On the other hand, erosion and closure operators are frequently used to carry out the noise reduction and erase the weak edge. In the operation of morphology, it will firstly define a structure element. The structure element is a small set or a sub-image, and its size is often set as a 3\*3 mask. If an input image is dilated by the structure element, it will be denoted as  $A \oplus B$ . And if an input image is eroded by the structure element, it will be denoted as  $A \ominus B$ [15]. Figure 4(a) describes the process of dilation. If the center pixel is white, the neighbors will change into white. Figure 4(b) describes the process of erosion. If the center pixel is white, the neighbors will change into black. During the operation, if it is open it will perform the dilation first and then erosion. On the other hand, if it is close, it will firstly perform the erosion and then dilation.

### 2.5. Optical Character Recognition (OCR)

Traditional OCR is mainly used to identify the text on books or in documents, and the transformed content can be shared on the Internet[17] and applied in text recognition, computer vision, and machine learning. Notwithstanding the progress of OCR technique at the present stage, there are still

problems that would affect the result of recognition such as the complicated background or non-uniform lightness situations[4-6]. Some studies utilize machine-learning based method or edge detection approach to solve this problem, while the effects are limited[7-9].

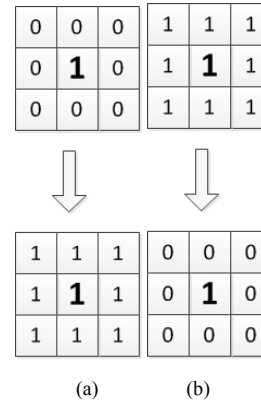


Figure 4. Dilation and Erosion of morphology

## 3. Methodology

The processing stage can be divided into three sub-stages: pre-processing, text extraction and OCR. Our study focuses on the pre-processing and text extraction. Figure 5 depicts the steps involved in the whole process.

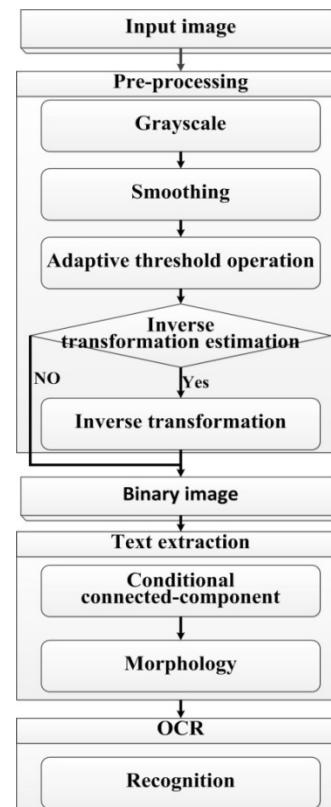


Figure 5. Flow chart of image processing

### 3.1. Pre-processing

### 3.1.1. Grayscale Transformation

In order to reduce the computational complexity as well as memory requirements, the acquired color images were converted into gray scale. To convert a grayscale image into binary, the original image will be applied the grayscale threshold  $T$ , the cutting point, then change the pixels into black or white based on the grayscale value of each pixel as (6). If the pixel value exceeds  $T$ , it will be reset as value 1 indicating a white pixel. In contrast, if the pixel value is lower than or equals to  $T$ , it will be reset as value 0 indicating a black pixel. Based on National Television System Committee (NTSC), this equation shown as (7) combines the RGB values in relevance to the eye sensitivity. Where  $R, G, B$  represents the color of red, green, and blue, respectively.

$$\text{Pixel color} = \begin{cases} \text{grayscale value} > T, & \text{white} \\ \text{grayscale value} \leq T, & \text{black} \end{cases} \quad (6)$$

$$Y = 0.299R + 0.587G + 0.114B \quad (7)$$

### 3.1.2. Smoothing

The smoothing filters deliberately making an image less sharp has been commonplace on most digital cameras but only recently have they become the focus of mainstream user attention. When the image has fine details, a number of ugly digital artifacts such as thin lines and edges in a blocky, stair-step fashion can occur from the shooting. Due to impaired words usually share the same texture with its background context, therefore we conducted the optical low-pass filter (OLPF) to reduce the noise. LPF will blur the broken characters in the image, consequently it enlarges the contrast between background context and foreground text. The results showed a promising effect by applying OLPF shown as Figure 6.



Figure 6. Image smoothing

### 3.1.3. Adaptive Threshold Operation

In the binarization stage, a grayscale image will be assigned a threshold value to be binary. Nevertheless, this traditional approach will not be suitable for some scenarios, such as non-uniform lightness and complicated background which may affect the effect. To remedy the problem, we applied the "adaptive threshold operation" to the stage of binarization. Other application, for instance, the face recognition research [18] has also used this approach to improve the preprocessing performance. Knowing that English texts often present in a horizontal yet inconsistent

way of display in an image, we proposed a novel approach to better analyze the text. Different from the general procedure of using the whole image to separate the background from the objects, our research adopted the divide and merge strategy to tackle this problem. Instead of segregating the gray image into many regions [19], our approach only partitioned an image into three equal-sized horizontal segments to identify the local threshold value of each segment and then restored the three segments back to the original state shown in Figure 7.

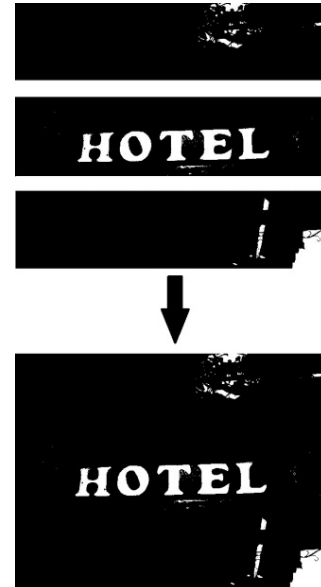


Figure 7. Divide and Merge treatment

Experimental results showed that the accuracy rate of text recognition with the divide and merge treatment enhanced significantly compared to that considering only the global threshold. The result is displayed in section 4.

### 3.1.4. Inverse transformation estimation

At this stage, we expect to have a black background and white foreground as the standard template for streamlining the binarization process. If the result happens to be opposite, we will need to conduct an inverse transformation (IT) estimation. The transformation estimation is listed in (8), where  $T_w$  denotes the total number of pixels in white and  $T_b$  means the total number of pixels in black. If the binary image needs to perform IT, we will conduct the process displayed in (9), where  $P'_i$  indicates the value of pixels after processing IT, and  $P_i$  is the original value.

$$IT = \begin{cases} \text{Yes}, & T_w > T_b \\ \text{No}, & T_b > T_w \end{cases} \quad (8)$$

$$P'_i = \begin{cases} 0, & \text{if } P_i = 255 \\ 255, & \text{if } P_i = 0 \end{cases} \quad (9)$$

## 3.2. Text Extraction

### 3.2.1. Conditional Connected-Component

In addition to the non-uniform lightness problem, the complicated background context is another challenge for OCR. To better identify foreground texts, it is necessary to eliminate the background noise as complete as possible. In this stage, we will engage in determining a threshold value to undergo an optimal reduction of the amount of connected-components for better identifying text objects.

Take an image of 640 \* 480 pixels for example, after the process of the adaptive threshold operation and IT estimation; the thresholds of three-partition segments (top, middle, and bottom) are identified. We then calculated the total number of pixels in white of each segment. During the experiment, we observed that except for some cases, the text region usually appears in the segment with the least number of pixels in white and this region may span two or three segments. Considering varied situations may lead to different conclusions, our experiment was carried out by a series of exhaustive testings based on each respective condition. The following is the procedure to determine the potential text region in an image.

- Step1: Select the segment with the least number of pixels in white as  $S_1$ .
- Step2: Compute the difference of the number of pixels in white between  $S_1$  and the other two segments  $S_2$  and  $S_3$ , respectively.
- Step 3: Execute the conditional connected-component processing.

To determine whether the identified text regions need to be merged and in what situation, we conducted a series of tests to observe the change. We used different thresholds applied in 10, 20, 30, and 40 images, respectively. The result is presented in Table 1. The digits in the table express the number of connected components before and after the merging. Tire presents the total number of pixels of image  $i$ . When the threshold set as  $< 5\% * \frac{1}{3} * T_i$ , it shows the least connected-component after segment combination whereas it often neglects valid text region. When the threshold set as  $< 45\% * \frac{1}{3} * T_i$ , it makes no difference of the amount of components before and after the segment combination. Finally we identified  $15\% * \frac{1}{3} * T_i$  as the optimal threshold to tackle this problem.

There are three kinds of component merging circumstances:

1. Circumstance1:  $S_1$  is the top segment, the result will

- be (1) top segment only (2) the merger of top and middle segments(3) the merger of all three segments
2. Circumstance2:  $S_1$  is the bottom segment, the result will be (1) bottom segment only (2) the merger of the bottom and middle segments (3) the merger of all three segments
3. Circumstance 3:  $S_1$  is the middle segment, the result will be (1) middle segment only (2) the merger of top and middle segments (4) the merger of the middle and bottom segments (3) the merger of all three segments

After the above processing, we successfully captured the text region of an image. To eliminate invalid objects (area too large or too small, width is far longer than the height, etc.) which may lead to recognition error, we configured three filtering rules as(10), (11), (12) based on the shape of fonts. Any object satisfying any one of the following criteria is eliminated.

$$\text{Font height} \times \text{Font width} > 250 \tag{10}$$

$$0.5 < \frac{\text{Font height}}{\text{Font width}} < 1.8 \tag{11}$$

$$0.5 < \frac{\text{Font height}}{\text{Font height}} < 1.5 \tag{12}$$

### 3.2.2. Morphological Enhancements

In this part, we executed the erosion and dilation to cope with the non-textual noise. Erosion was conducted to reduce image noise appeared in the connected components and magnified the difference between foreground texts and background context. Due to the side effect of erosion may shrink the size of the connected components, were summed it by the way of dilation. After executing morphological enhancement, we successfully erased the unnecessary stains and made the texts clearer as Figure 8.

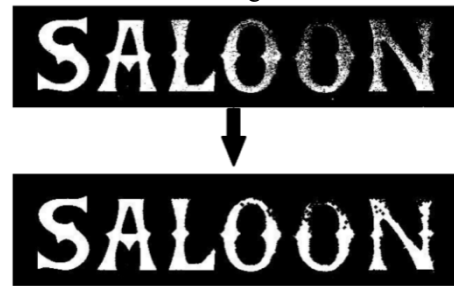


Figure 8. Morphology execution

Table 1. Different thresholds of segment combination

Threshold	Total images 10	20	30	40
$< 5\% * \frac{1}{3} * T_i$	1130→145	2838→313	3618→573	4360→835
$< 15\% * \frac{1}{3} * T_i$	1130→243	2838→455	3618→826	4360→1311
$< 25\% * \frac{1}{3} * T_i$	1130→311	2838→672	3618→1287	4360→2531
$< 35\% * \frac{1}{3} * T_i$	1130→693	2838→1532	3618→2785	4360→3732
$< 45\% * \frac{1}{3} * T_i$	1130→988	2838→2356	3618→3206	4360→4085

## 4. Experiment and Result

### 4.1. Software and data set

In this research, our software development tools were MATLAB and Tesseract OCR engine. MATLAB is a widely-used image processing software with high reputation worldwide. We used MATLAB to conduct image pre-processing, including grayscale transformation, smoothing, binarization, inverse transformation estimation, and text extraction. To recognize the texts of binary images, we selected Tesseract OCR engine as the tool. The Tesseract OCR engine was one of the top 3 engines in the 1995UNLV Accuracy test. Since then it has been improved extensively by Google and is probably one of the most accurate open source OCR engines. Besides, we utilized the data set of ICDAR2003 Robust Reading Competition and selected 90 images with the size of 640 \* 480. The dataset including the images with several kinds of image conditions, such as different font-size, non-uniform illumination, low contrast, or complicated background is suitable for our experiment.

### 4.2. Result discussion

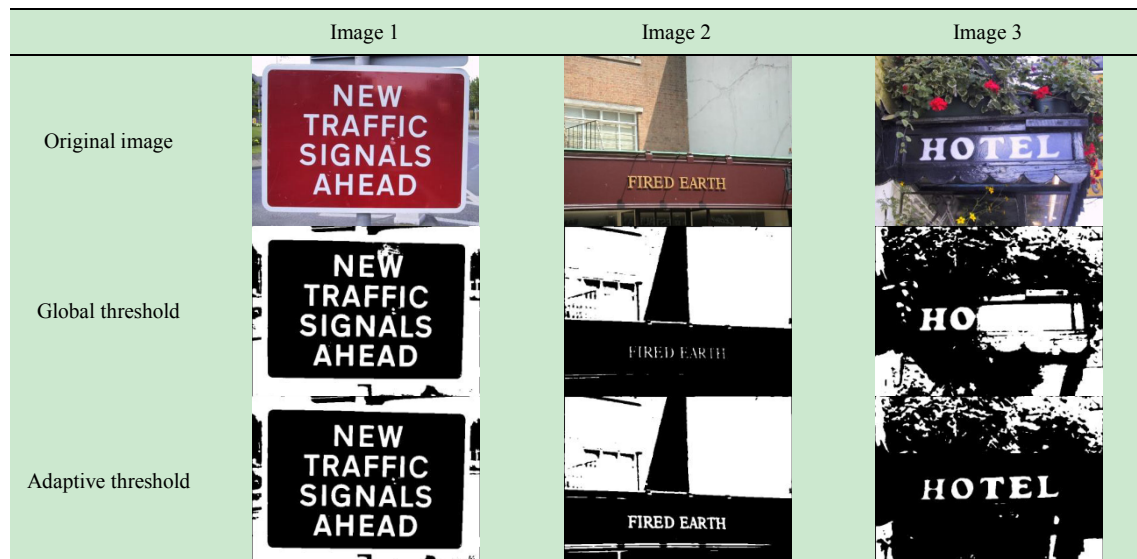
Table 2 displays the result of image pre-processing for three sample images by global threshold and adaptive threshold, respectively. We list them side by side to have an easy comparison through the perception of naked human eye. Apparently, the result of the adaptive threshold operation performed much better than that of the global threshold, particularly for the images with non-uniform lightness such as images 1 and 2 or the texts share the similar color with the background such as image 3. With the satisfactory result, it provided a better ground to process the next task of text recognition.

Next, we will show a couple of unsatisfactory results of text recognition after image pre-processing and analyze the reason that may cause the failure. Table 3 presents three sample images with complicated background such as image 1, non-uniform lightness like image 2, and text skewness as image 3. The identified text region of each image expresses the binarization result of image pre-processing by means of adaptive threshold and conditional connected-component proposed in this study. The result showed that our method successfully identified the text area of each image while the text recognition task of images 2 and 3 didn't perform satisfactorily. The reason for the unsatisfactory result of image 2 could be the text-like object mistakenly identified as the letter A. As for the image 3, we assume the text area with a skew orientation makes texts being mistakenly measured its external shape of fonts and causes part of texts incorrectly recognized..

### 4.3. Evaluation






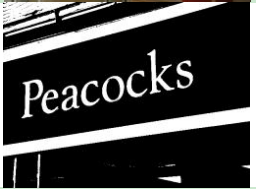



In the final stage, we utilized object identification rate (OIR), false alarm rate (FAR) and recognition rate (RR) to evaluate the performance of our approach. The formulas are listed as (13), (14), and (15). OIR is the rate of correctly identified text objects, the FAR is the rate of falsely identified text objects and the RR is the rate of correctly recognized text objects.  $O_i$  denotes the total number of text objects in the image  $i$ ,  $E_T$  and  $E_F$  denote the number of correctly located text objects and the number of falsely identified text objects,  $R_i$  indicates the number of correctly recognized images. Table 4 shows how the treatment affects the accuracy rate in all aspects of object identification and recognition.

**Table 2.** Comparison between global and adaptive thresholds





**Table 3.** Result presentation

	Image1	Image2	Image3
Original image			
Identified text region			
Text extraction			
Text recognition	HOTEL	A NatWest	P€2ICOCKa

$$OIR = \frac{E_T}{O_i} \tag{13}$$

$$FAR = \frac{E_F}{O_i} \tag{14}$$

$$RR = \frac{R_I}{O_I} \tag{15}$$

$$P = \frac{T_C}{T_C + T_F} \tag{16}$$

$$R = \frac{T_C}{T_C + T_N} \tag{17}$$

objects;  $T_F$  denotes the number of falsely recognized objects;  $T_N$  denote the number of correct and not recognized objects. Table 5 displays the performance comparison with other approach. Our treatment outperforms most of the other studies and the performance achieves 74.6% in precision and 80.2% in the recall.

### 5. Conclusion and Future Work

Although OCR is an active research area, most of efforts concern the actual recognition, not the pre-processing phase. A common problem in the early stage of OCR is to identify the text areas of the acquired image. In this study we have proposed a text extraction approach by utilizing adaptive threshold operation and conditional connected-component in solving the problems of complicated background and non-uniform lightness situation to correctly identify the text area. Different from the general procedure of using the whole image to separate the background from the objects, our research adopted the divide and merge strategy to tackle this problem. Instead of segregating the gray image into many regions, our approach partitioned an image into three equal-sized horizontal segments to identify the local threshold value of each segment and then restored the three segments back to the original state shown in Figure 6 in a more efficient way.

The proposed approach can reduce unnecessary objects when executing the connected-component and thereby enhance the accuracy. The result shows that our approach achieves 81.17% object identification rate and 91.30 % recognition rate. To compare with other ICDAR 2003 result, we conducted another evaluation to examine the validity of our approach. Experiment shows that our treatment outperforms most of the other studies and the performance achieves 74.6% in precision and 80.2% in the recall.

Taking into account these accomplishments, our future

**Table 4.** Performance: with and without our approach

	OIR	FAR	RR
With	81.17%	14.49%	91.30%
Without	64.71%	36.43%	72.72%

**Table 5.** A comparison with other study

	P	R
Ashida	0.55	0.46
HW David	0.44	0.46
Wolf	0.3	0.44
Chan	0.58	0.59
Kim	0.64	0.83
Our approach	0.75	0.80

\* the digits are the rounded value.

To compare with other ICDAR2003 result[20][21], we conducted another evaluation to examine the validity of our approach. In the second evaluation, we adopted precision and recall as the measure criteria. There were totally 895 text objects in the 90 images. Precision, P measures the proportion of recognized text objects actually correct, whereas recall, R measures the proportion of correct objects actually recognized. The equations are in(16)and (17). Where  $T_C$  indicatesthe number of correctly recognized

work will focus on optimizing the current recognition results by exploiting new approaches for better noise attenuation and correction of text skew orientation.

## Acknowledgments

This research is supported by National Science Council, Taiwan, R.O.C. under grant number NSC 101-2221-E-224-056.

## REFERENCES

- [1] H. Nakajima, Y. Matsuo, M. Nagata, and K. Saito, "Portable translator capable of recognizing characters on signboard and menu captured by built-in camera," in Proceedings of the ACL 2005 on Interactive Poster and Demonstration Sessions, University of Michigan, USA, 2005, pp. 61-64.
- [2] S. Mori, C. Y. Suen, and K. Yamamoto, "Historical review of OCR research and development," Proceedings of the IEEE, vol. 80, pp. 1029-1058, 1992.
- [3] R. Minetto, N. Thome, M. Cord, J. Stolfi, F. Precioso, J. Guyomard, et al., "Text detection and recognition in urban scenes," in 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 2011, pp. 227-234.
- [4] X. Liu and J. Samarabandu, "Multiscale edge-based text extraction from complex images," in Proceedings of the 2006 IEEE International Conference on Multimedia and Expo, Toronto, Ontario, Canada, 2006, pp. 1721-1724.
- [5] S. P. Chowdhury, S. Dhar, A. K. Das, B. Chanda, and K. McMenemy, "Robust Extraction of Text from Camera Images," presented at the Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, Barcelona, Spain, 2009.
- [6] E. Kim, S. Lee, and J. Kim, "Scene text extraction using focus of mobile camera," in 10th International Conference on Document Analysis and Recognition, 2009, pp. 166-170.
- [7] M. U. Raza, A. Ullah, K. M. Ghorri, and S. Haider, "Text extraction using artificial neural networks," in Proceedings of the International Conference on Networked Computing and Advanced Information Management, Gyeongju, Gyeongsangbuk-do, South Korea, 2001, pp. 134-137.
- [8] L. Shutao and J. T. Kwok, "Text extraction using edge detection and morphological dilation," in Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004, pp. 330-333.
- [9] Q. Ye, Q. Huang, W. Gao, and D. Zhao, "Fast and robust text detection in images and video frames," Image Vision Comput, vol. 23, pp. 565-576, 2005.
- [10] F. King-Sun and A. Rosenfeld, "Pattern recognition and image processing," IEEE Transactions on Computers, vol. C-25, pp. 1336-1346, 1976.
- [11] N. Otsu, "A threshold selection method from gray-level histograms," IEEE Transactions on Systems, Man and Cybernetics, vol. 9, pp. 62-66, 1979.
- [12] L. Pei-Jun and Effendi, "Adaptive edge-oriented depth image smoothing approach for depth image based rendering," presented at the 2010 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), 2010.
- [13] L. Taisheng, Z. Xuan, and L. Chongrong, "An improved adaptive image filter for edge and detail information preservation," presented at the 2012 International Conference on Systems and Informatics (ICSAI), 2012.
- [14] F. Marqués and V. Vilaplana, "Face segmentation and tracking based on connected operators and partition projection," Pattern Recognition, vol. 35, pp. 601-614, 2002.
- [15] Z. Shuqing and Y. Qiaoning, "Microarray images processing based on mathematical morphology," in Proceedings of the 2006 8th International Conference on Signal Processing, 2006.
- [16] J. Besag, "Digital Image Processing," Journal of Applied Statistics, vol. 16, pp. 395-407, 1989.
- [17] V. Yakobov, L. Mash, and N. Thirer, "On chip implementation of an OCR algorithm for musical notation," in Proceedings of the 2010 IEEE 26th Convention of Electrical and Electronics Engineers in Israel (IEEEI), 2010, pp. 581-584.
- [18] Y. Rongmu, L. Zongyue, and C. Junming, "Face detection system using adaptive threshold," in Proceedings of National Conference on Modern Electrical Engineering Technologies 2006.
- [19] Y. Jian, Z. Yi, T. Kok-Kiong, and L. Tong-Heng, "Text extraction from images captured via mobile and digital devices," in IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2009. AIM 2009. , 2009, pp. 566-571.
- [20] C. Wai-Lin and P. Chi-Man, "Robust Character Recognition Using Connected-Component Extraction," in 2011 Seventh International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2011, pp. 310-313.
- [21] K. C. Kim, H. R. Byun, Y. J. Song, Y. W. Choi, S. Y. Chi, K. K. Kim, et al., "Scene text extraction in natural scene images using hierarchical feature combining and verification," in Proceedings of the 17th International Conference on Pattern Recognition, 2004, pp. 679-682.