http://dx.doi.org/10.5392/IJoC.2013.9.2.008

Systematic Approach for Detecting Text in Images Using Supervised Learning

Minh Hieu Nguyen
Department of Electronics and Computer Engineering
Chonnam National University, Gwangju, 500-757, South Korea

GueeSang Lee
Department of Electronics and Computer Engineering
Chonnam National University, Gwangju, 500-757, South Korea

ABSTRACT

Locating text data in images automatically has been a challenging task. In this approach, we build a three stage system for text detection purpose. This system utilizes tensor voting and Completed Local Binary Pattern (CLBP) to classify text and non-text regions. While tensor voting generates the text line information, which is very useful for localizing candidate text regions, the Nearest Neighbor classifier trained on discriminative features obtained by the CLBP-based operator is used to refine the results. The whole algorithm is implemented in MATLAB and applied to all images of ICDAR 2011 Robust Reading Competition data set. Experiments show the promising performance of this method.

Key words*: Text localization; Tensor voting; Completed local binary pattern.*

1. INTRODUCTION

Today, people can use various digital devices to produce images at any time and places. Since the text information in image can be embedded in different font styles, sizes, orientations, colours, and against a complex background, the problem of extracting text areas becomes very challenging [1].

Based on basic characteristic of text, a group of region-based approach for the process of detecting text in images have been proposed. In terms of frequency and orientation information, text has some common distinctive properties. Besides, it also has spatial cohesion. Spatial cohesion refers to the fact that text characters of the same string appear close to each other and are of similar height, orientation and spacing [2]. Two of the main region-based methods commonly used to determine spatial cohesion are based on edge and connected component features of text characters. Region-based approach is widely used because it is simple to implement and robust to illumination changes. Nevertheless, this approach generates high false positive rate, which means lots of wrong results, since many non-text regions are similar to text regions.

To avoid the problem with high false positive rate, while efficiently detecting true text areas, in our approach, text alignment information is employed by using 2D tensor voting [3]. Text alignment or text line is a virtual line that characters align on. Information provided by 2D tensor voting is valuable to locate characters and remove noises or to prevent the erroneous detection. Tensor voting generates curve saliency values and normal vectors at characters belonging to a text line. The text line information is used to remove non-text regions when combining with region-based methods [4].

Compare to [4], various improvements have been done: 1) the size is normalized for small input image, then different 2D filters (Sobel and Unsharp) are applied for different resolutions generated from pyramid strategy to emphasize edges; 2) detected regions is refined by using an NN (Nearest Neighbor) classifier, which is trained on highly discriminative features obtained by a Completed Local Binary Pattern operator (CLBP); 3) experimental results are evaluated online by a public methodology of ICDAR 2011. 1690 text and non-text regions, which are manually labeled, are used to train this supervised learning model. These changes have shown significant impact to the results.

The remainder of this paper is structured as follows: In section II the proposed method is described, Section III presents experiment results and the corresponding discussion, and finally, in section IV conclusions are drawn.

2. PROPOSED SYSTEM

2.1 System Overview

As shown in Fig. 1, the proposed text localization system has three main stages: pre-processing, candidate text regions

* *Corresponding author, Email : gslee@chonnam.ac.kr*
Manuscript received Feb. 27, 2013; revised May 30, 2013; accepted Jun 10, 2013

localization and post-processing.

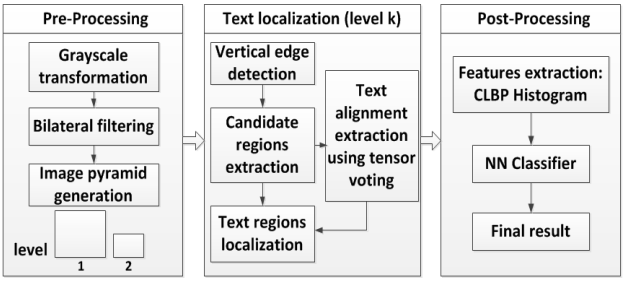


Fig. 1. Flowchart of proposed system

In the first stage, the input image size is normalized, and then it is converted to a grayscale image. Bilateral filtering [5] is then applied to the grayscale image to reduce noises, while preserving edges (Fig. 2b). In the second stage, to detect text with different sizes, the filtered image is stored into two different resolution images with original size and half size. Sobel and Unsharp filter are applied to different sizes for emphasizing and enhancing vertical edges. After that, by using vertical edge detection, connected component analysis and text line information, candidate text regions are extracted separately for each resolution and then they are combined to get collected candidate regions. Finally, in the post-processing stage, the results from previous stages are refined using an NN classifier, which is based on CLBP.

2.2 Extracting Candidate Regions

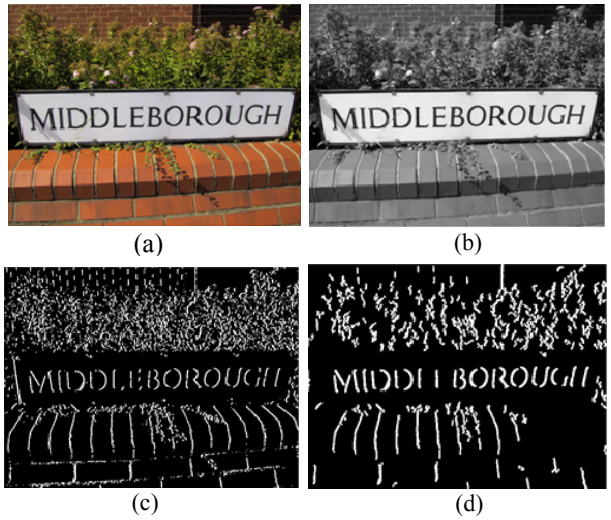


Fig. 2. Extraction of candidate regions, (a): original image, (b): grayscale filtered image, (c): edge connected components, (d): candidate regions

The edge feature is robust against text size and low contrast level. For that reason, vertical edges are used to locate the text regions. The Sobel edge detector is applied to the grayscale image to generate a binary edge map at level *k*. Then, the connected foreground pixels in the binary image are grouped into connected components (Fig. 2c). Some simple heuristic rules are used to remove too big or too small connected components. We assume M and N as the height and width of the input image at the level *k*. The connected component C that satisfies one of the following conditions is removed.

$$\begin{cases} height(C) > \dfrac{M}{2} \\ width(C) > \dfrac{N}{2} \\ \max(height(C), width(C) < MIN\_SIZE \\ density(C) < MIN\_DEN, \end{cases} \quad (1)$$

where *MIN_SIZE* and *MIN_DEN* are threshold values, and *density(C)* of that connected component *C* is calculated by dividing the number of edge pixels by the total number of pixels in the bounding box. The center points of remaining candidate regions (Fig. 2d) are used to extract text line information in next step.

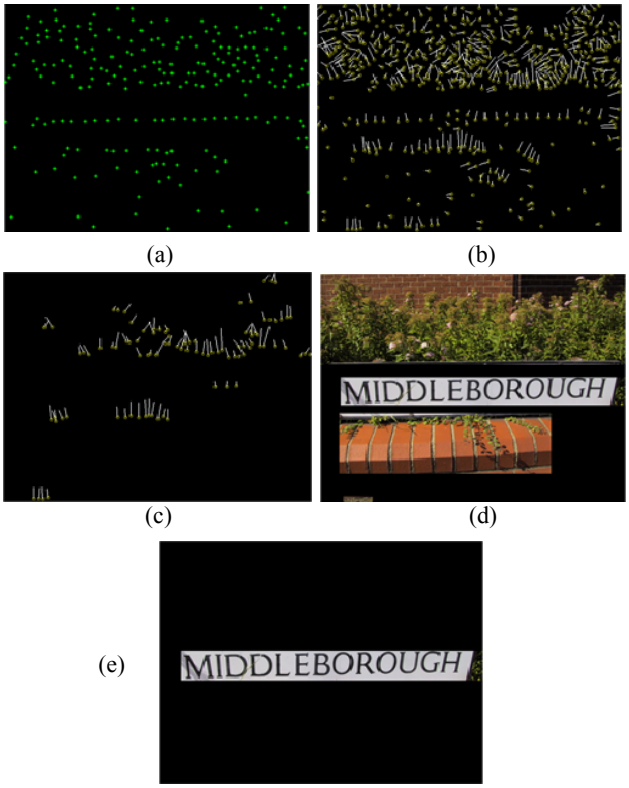


Fig. 3. Candidates verification using tensor voting, (a): candidate center points, (b): resulting tensors, (c): text line information, (d): located regions, (e): final refined result

2.3 Using Text Line Information To Verify Regions

Tensors are geometric entities introduced into mathematics and physics to extend the notion of scalars and vector, it can be 0th-order (a scalar), 1th-order (a vector) or 2th-order (a square matrix) in 1-D, 2-D or 3-D dimensions. In tensor voting, tensor is some information with direction, and it checks if a set of points (called tokens) can be a curve, an ellipse or isolated point in 2-D dimension. A second order, symmetric, non-negative definite tensor T is equivalent to a 2 by 2 matrix or an ellipse (Fig.4). It can be decomposed into stick and ball components as the following equation:

$$T = [\hat{e}_1 \hat{e}_2] \begin{bmatrix} \lambda_1 0 \\ 0 \lambda_2 \end{bmatrix} \begin{bmatrix} \hat{e}_1^T \\ \hat{e}_1^T \end{bmatrix} = (\lambda_1 - \lambda_2)\hat{e}_1 \hat{e}_1^T + \lambda_2(\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T)$$
$$= T_S + T_B \quad (2)$$

where T_S, T_B are the respective stick and ball components. λ_1, λ_2 are the eigenvalues and $\hat{e}_1$, $\hat{e}_2$ are the corresponding eigenvectors. $\lambda_1 - \lambda_2$ is the size of the stick component indicated curve saliency, in other words, how certain it is that corresponding center point belongs to a curve.

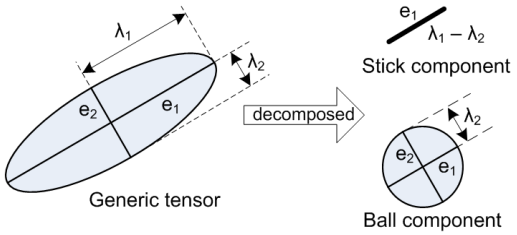


Fig. 4. Tensor decomposition

Center points are encoded with tensors, and then these tensors cast their information to nearby neighbors via a predefined voting field or kernel. This step is similar to the convolution operation but instead of using scalar matrix as a kernel in convolution operation, a voting field is used in tensor voting. The voting kernel defines the normal vector by selecting the most likely continuation curve between two point O and P in Fig. 5. The strength of the vote at P is defined by Eq. (3), called decay function, in spherical coordinates.$s = (l\theta)/\sin(\theta)$ and $k = 2\sin(\theta)/l$. The parameter s is the arc length OP, k is the curvature, c is a constant, and σ is the scale of voting field controlling the size of the voting neighborhood and the strength of votes.

$$DK(s,k,\sigma) = e^{\left(\frac{s^2+ck^2}{\sigma^2}\right)} \quad (3)$$

For this implementation, we use second order vote to infer the curve saliency value and normal vector of token at each center point. By casting votes to all neighbors by the same voting fields and voting algorithm, tensors are encoded into new tensors. At each center point, the curve saliency value is the magnitude of stick component of the resulting tensor and the normal vector is the main axis $\hat{e}_1$.

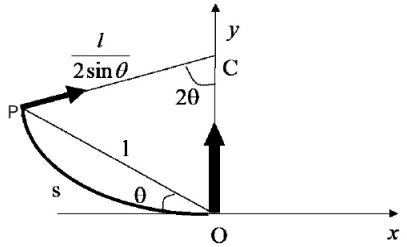


Fig.5. 2D stick voting kernel.

A text line is a virtual line that characters align on. The center points of the connected components in a text region are usually close together and are mostly aligned on a line or smooth curve. Therefore, the curve saliency of a token corresponding to a center point in a text region has higher curve saliency than that of a token in a non-text region. Additionally, the curve normal of a token in a text region indicates the normal vector of the text line. Since text is assumed to be horizontally aligned on a smooth curve or line, those candidates with low curve saliency values or nearly horizontal normal vectors are removed. The remaining connected components are used as detected text regions. Fig. 3 shows how to use text line information to remove non-text regions: Fig. 3b shows the resulting tensors generated from center points extracted from Fig. 3a. Figs. 3c and 3d show remaining after verification and corresponding located regions.

2.4 Nearest Neighbor Classifier based on CLBP

Tensor voting is truly useful to remove wrong detected regions. However, this technique also fails in many cases. To overcome this problem, we apply a supervised learning classifier model for post-processing phase.

Text can be referred as texture having some special characteristics [6]. Local Binary Pattern (LBP) is highly discriminative for texture classification and it has achieved impressive classification results [7]. LBP is also invariant to monotonic gray level changes and computational efficiency. These reasons have motivated us to use a modified LBP, which is CLBP (Completed Local Binary Pattern), for text detection.

Given a central pixel g_c (gray value of the central pixel) in the image, Fig 6, an LBP code is computed by comparing it with its neighbors g_p, p = 0, 1,…,7 (gray value of neighbors). The difference between g_c and g_p is computed as $d_p = g_p - g_c$. The local difference vector $[d_0,\ldots,d_7]$ characterizes the image local structure at g_c. d_p can be further decomposed into two components:

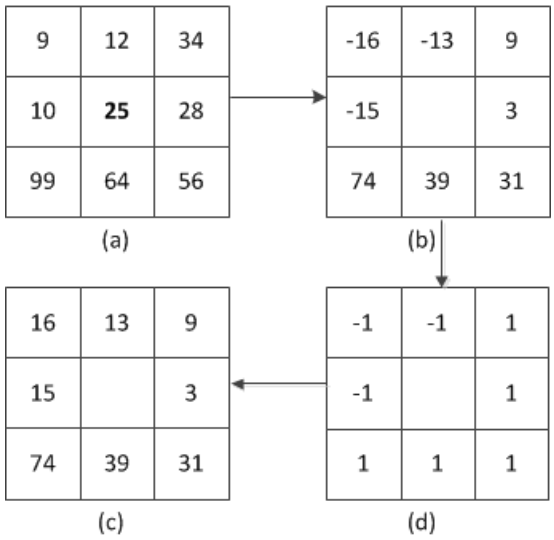


Fig. 6. (a) 3x3 sample block; (b) the local differences; (c) the sign; and (d) magnitude components

$$d_p = s_p * m_p \; and \; \begin{cases} s_p = sign(d_p) \\ m_p = |d_p| \end{cases} \qquad (4)$$

where $s_p = \begin{cases} 1, d_p \geq 0 \\ -1, d_p < 0 \end{cases}$ is the sign of d_p and m_p is the magnitude of d_p. With (5), $[d_0,\ldots,d_7]$ is transformed into a sign vector $[s_0,\ldots,s_7]$ and a magnitude vector $[m_0,\ldots,m_7]$. (4) is called the local difference sign-magnitude transform (LDSMT).

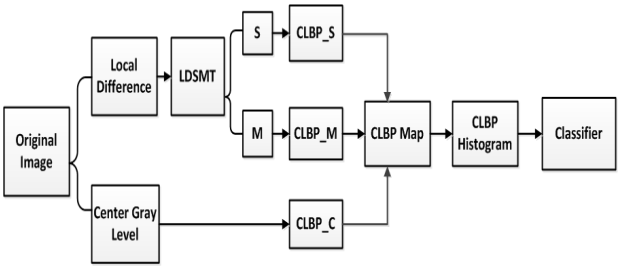


Fig. 7. Framework of CLBP

The CLBP framework is illustrated in Fig 7. The original image is represented as its center gray level (C) and the local difference. The local difference is then decomposed into the sign (S) and magnitude (M) components by the LDSMT defined in (5). Thus, three operators, namely CLBP_C, CLBP_S and CLBP_M, are proposed to code the C, S, and M features, respectively. Then, the CLBP_C, CLBP_S, and CLBP_M codes are combined to form the CLBP feature map of the original image [7]. Finally, a CLBP histogram can be built, and the Nearest Neighbor classifier is used to classify candidate regions of images as text and non-text.

The chi-square distance is used with the classifier to measure the dissimilarity between two histograms: the test sample and training data.

$$d(X,Y) = \sum_{i=1}^{n} \frac{(X_i - Y_i)^2}{X_i + Y_i} \qquad (5)$$

where n is the number of bins in the histogram, X_i and Y_i are the values of the training samples and test sample..

Fig. 3e shows the final result after using classifier to refine.

3. EXPERIMENTAL RESULTS

The proposed method is implemented and tested on ICDAR 2011 Reading Competition dataset [8], which contains 420 images (with size larger than 100x100 pixels), containing 3583 words of more than 3 characters for the training set and 102 images, containing 918 words, for the test set. Sample results are shown in *Table 1,* evaluation in *Table 2* below.

For evaluating the performance of Text Localization methods, ICDAR 2011 implemented the methodology proposed by Wolf [9]. Assessment in *Table 2* is produced by performance evaluation online tool of ICDAR 2011 competition.

Table 1. Example of segmented results

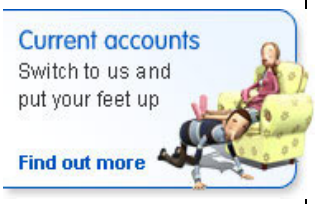
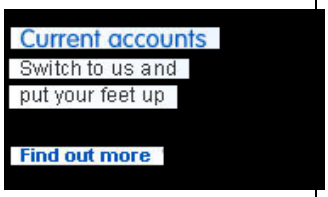
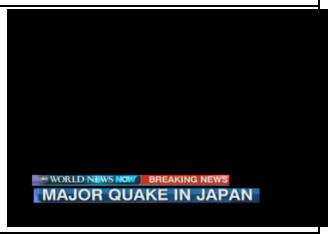
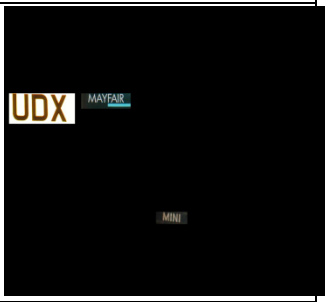
Original images	Detected text regions
	
	
	
	
	
	
	

Table 2. Comparison of methods on ICDAR 2011 data set

Method	Recall	Precision	Hmean
Toan [4]	40.81	64.32	49.94

Hanif [12]	58.43	75.52	65.88
C. Yi [11]	64.91	67.38	66.12
Kumar [10]	75.65	63.85	69.25
The proposed system	**71.10**	**81.23**	**75.83**

The ranking metric used for the Text Localisation task is the Harmonic Mean calculated according to the methodology described in [9].

$$Recall = \frac{N.o.correctly\ detected\ rectangles}{N.o.rectangles\ in\ the\ database}$$

$$Precision = \frac{N.o.correctly\ detected\ rectangles}{Total\ n.o.detected\ rectangles}$$

$$Hmean = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Precision and Recall are calculated cumulatively over the whole test set (all detections over all 102 images pooled together).

4. CONCLUSIONS

In this paper we presented a three stage system for the process of detecting text in images. The system includes an efficient filter to reduce noises in pre-processing stage, a tensor voting based text localization method that generates candidate text regions, and a machine learning refinement in post-processing stage. To overcome the problem of false detected results, a nearest neighbor classifier trained on highly discriminative local binary pattern features is implemented.

The obtained experimental results, which were evaluated using an independent evaluation methodology, show that the implementation of proposed method has much better precision than several previous works.

In future research, we would improve machine learning refinement for better classification and final segmented bounding box of text regions.

ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST)(2012-047759) and by the MSIP(Ministry of Science, ICT&Future Planning), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2013-H0301-13-3005) supervised by the NIPA(National IT Industry Promotion Agency).

REFERENCES

[1] Keechul Jung, Kwang In Kim, Anil K.Jain, “Text information extraction in images and video: a survey”, *Pattern Recognition*, vol. 37, no. 5(May 2004), pp. 977-997.

[2] Victor Wu, Raghavan Manmatha, and Edward M. Riseman, “TextFinder: An Automatic System to Detect and Recognize Text in Images”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, November 1999.

[3] G. Medioni, M. S. Lee, and C. K. Tang, *“A computational framework for segmentation and grouping”*, Elsevier, 2000.

[4] Toan Dinh Nguyen, Jonghyun Park, and Gueesang Lee, "Tensor Voting Based Text Localization in Natural Scene Images*," IEEE Signal Processing Letters*, vol. 17, no. 7, July, 2010, pp. 639-642.

[5] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *International Conference on Computer Vision*, 1998, pp. 839-846.

[6] P. Clark and M. Mirmehdi, "Recognising text in real scenes," *International Journal on Document Analysis and Recognition,* vol. 4, pp. 243-257, 2002.

[7] Z. Guo, L. Zhang, D. Zhang, “A completed modeling of local binary pattern operator for texture classification”, *IEEE Transactions on Image Processing* 19(2010) 1657–1663.

[8] Karatzas, S. Robles Mestre, J. Mas, F. Nourbakhsh, P. Pratim Roy, **“**ICDAR 2011 Robust Reading Competition - Challenge 1: Reading Text in Born-Digital Images (Web and Email)”, *11th International Conference of Document Analysis and Recognition*, 2011, IEEE CPS, pp. 1485-1490.

[9] C. Wolf and J.M. Jolion, "Object Count / Area Graphs for the Evaluation of Object Detection and Segmentation Algorithms", *Int. Journal of Document Analysis*, vol. 8, no. 4, pp. 280-296, 2006.

[10] Deepak Kumar, A. G. Ramakrishnan, “OTCYMIST: Otsu-Canny Minimal Spanning Tree for Born-Digital Images”. *Document Analysis Systems*, 2012, pp.389-393.

[11] C. Yi and Y. Tian, “Text String Detection from Natural Scenes by Structure-based Partition and Grouping”, *IEEE Transactions on Image Processing*, Vol. 20, Issue 9, 2011.

[12] Shehzad Muhammad Hanif, Lionel Prevost, “Text Detection and Localization in Complex Scene Images using Constrained AdaBoost Algorithm”, *ICDAR '09: 10th International Conference on Document Analysis and Recognition*, July 2009.

Minh Hieu Nguyen

He received the B.S degree in Software Engineering from FPT University, Vietnam in 2011. He is currently a MS student in Electronics and Computer Engineering department of Chonnam National University, South Korea. His research interests are image processing, computer vision and text extraction.

GueeSang Lee

He received the B.S. degree in Electrical Engineering and the M.S. degree in Computer Engineering from Seoul National University, Korea in 1980 and 1982, respectively. He received the Ph.D. degree in Computer Science from Pennsylvania State University in 1991. He is currently a professor of the Department of Electronics and Computer Engineering in Chonnam National University, Korea. His research interests are mainly in the field of image processing, computer vision and video technology.