2009 10th International Conference on Document Analysis and Recognition

Scene Text Extraction using Focus of Mobile Camera

Egyul Kim, SeongHun Lee, JinHyung Kim
Artificial Intelligence & Pattern Recognition Lab, KAIST, Korea
{*egkim, leesh, jkim*}*@ai.kaist.ac.kr*

Abstract

Robust extraction of text from scene images is essential for successful scene text recognition. Scene images usually have non-uniform illumination, complex background, and existence of text-like objects. The common assumption of a homogeneous text region on a nearly uniform background cannot be maintained in real applications. We proposed a text extraction method that utilizes user's hint on the location of the text within the image. A resizable square rim in the viewfinder of the mobile camera, referred to here as a 'focus', is the interface used to help the user indicate the target text. With the hint from the focus, the color of the target text is easily estimated by clustering colors only within the focused section. Image binarization with the estimated color is performed to extract connected components. After obtaining the text region within the focused section, the text region is expanded iteratively by searching neighboring regions with the updated text color. Such an iterative method would prevent the problem of one text region being separated into more than one component due to non-uniform illumination and reflection. A text verification process is conducted on the extracted components to determine the true text region. It is demonstrated that the proposed method achieved high accuracy of text extraction for moderately difficult examples from the ICDAR 2003 database.

1. Introduction

Scene text understanding is an attempt to recognize text in an image of a natural scene. Recently, scene text recognition via mobile phones has received a great deal of attention from many researchers. If scene text could be directly recognized from a mobile camera, it would lead to a diversity of new applications and yield enormous benefits for users. For example, as a user simply snaps a photo of a restaurant signboard the internet connected camera can instantly recognize the name and return relevant information about the restaurant.

There are many challenging issues related to separating texts from camera captured images. The images usually have non-uniform illumination due to the lighting conditions and shadows. Hence, the intrinsic properties of scene text, such as homogeneity of text pixel colors and distinctiveness of the text pixels from the background color, are difficult to preserve in real applications. Complex layout and interaction of the content and background are common in outdoor images. When the system scans the whole image for texts, non-text pixels surrounding the text could be confused for text because of similar shape to the texts. As an example, bars of a window could be regarded as a series of 'i's. Such complications make extracting text from scene images remain as an open problem.

Many approaches for the extraction of text from natural scene images have been proposed [2]. Ezaki *et al.* [3] proposed four steps for text extraction based on connected components: Sobel edge detection, Otsu binarization, connected component extraction and rule-based connected component filtering. Gatos *et al.* [4] applied binarization techniques to both gray and inverted gray images and chose the optimum between both binarization results. But image binarization on the gray-scale image cannot distinguish different color components having the same luminance. K.C. Kim *et al.* [5] combined color continuity, gray-level variation and color variance features to extract text regions. Park *et al.* [8] first split color pixels into chromatic and achromatic components and then separated them by histogram-based K means clustering. However those methods mainly work on the images of text with nearly uniform background. All these methods produce a lot of missing and false detections on many natural scene images. From this we may confirm that the text extraction from natural scene images is still a challenging problem, especially for the images of complex background and natural illumination. Interactive vision system could offer a solution, by borrowing the power of user interface. For instance, lazy Snapping cuts [6] and GrabCut [9] adopt interactive graph cuts to segment object from backgrounds.

In this paper, we propose a text extraction method in hand-held camera using a hint on the location of target text.

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.21

Focus, a resizable square rim in the viewfinder of mobile camera, is an interface to help the user point a target text (a red box on the original image in Figure 1). With the hint from the focus on the location of the target text, the color of the target text is easily estimated by clustering colors only within the focused section. Since we know in advance the color of the target text region, text separation would be much easy, even from complex background. In addition, restricting searching area within the focused section can prevent misclassification caused by surrounding non-text regions. After obtaining the target text region within the focused section, the target text region is expanded iteratively by searching neighbors with the updated text color. Such an iterative method would prevent the problem of one text region being separated more than one due to non-uniform illumination and reflection.

2. Scene text extraction method using focus

The proposed scene text extraction algorithm consists of three steps: selection of text color candidates, extraction of connected components and text verification (Figure 1). The scene text extraction algorithm is applied within the focus and then also applied outside of the focus to detect the target text region.

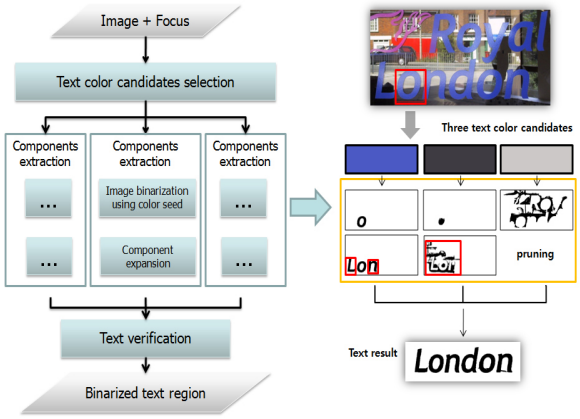

Figure 1. Overview of scene text extraction

First, the system analyzes the small region inside the focus to select color candidates for the target text. Since the focused area contains only a small number of different colors, it is easier to estimate the target text color than searching the whole image. As seen in Figure 1, the system extracts three distinctive colors (blue, black, gray) with a hope that the target text color is included in them.

Second, the system extracts text candidates independently using each text color candidate. Searching is continuously expanded to its neighboring region outside of the focus. For example, using the blue color, 'o' is first extracted within the focus and expanded to find the neighboring character components 'L', 'n' and so on. Images of partial degradations due to uneven illumination or reflection are tough to discern. Conventional method such as global thresholding cannot segment text strings as a single region in such an image. But the iterative region growing method can extract whole text-lines by setting the threshold intelligently.

Finally, the extracted text candidate regions go under a verification process, like Ezaki [3] and Gotos [4], to find the true text components. Simple heuristic rules are used to filter out the false text components. As result, the binarized text region is successfully obtained.

2.1. Selection of text color candidates

Considering that the scene text is designed to be easily visible, it would be effective to use a color model close to human perception of colors. In this regard, we decide to use a perceptually uniform color space, HCL (hue, chroma, and luminance) space [10]. We adopt a color similarity measure called HCL distance (D_{HCL}) to express color difference between text and background in HCL color space. HCL distance between a pixel color (h, c, l) and a seed color (h_s, c_s, l_s) is defined as

$$D_{HCL} = \sqrt{A_L(l-l_s)^2 + A_{CH}\{c^2 + c_s^2 - 2cc_s\cos(h-h_s)\}}, \quad (1)$$

where $A_L = 0.1, A_{CH} = 0.2 + (h - h_s)/2$. A_L is a constant of linearization for luminance, and A_{CH} is a parameter which helps to reduce the distance between colors having a same hue as the hue in the seed color.

HCL distance is more suitable in case of scene text images by emphasizing hue difference. Hue is robust on the illumination changes compared to luminance or RGB color. For example, Figure 2 shows the difference between RGB Euclidean distance and HCL distance in which gray scale represents a distance from the seed color. The red points of two images indicate the seed text colors. In RGB distance-of-color image (Figure 2 (b)) the top and bottom parts have large difference: the bottom parts of the text are rather close to the background of the image. On the other hand, in the HCL distance-of-color image (Figure 2 (c)) every parts of the text region show uniformly darker than the background. We expect that the color variation on the text region would be well handled using the HCL distance measure.

Figure 2. Color distance in RGB and HCL

We need to select a text seed color from the focused area to apply a text extraction method on HCL color space. Text

color is considered as one of the distinctive colors inside the focused section by assuming that text region generally occupies significant portions of the focused area. Color clustering is a common approach to find the major colors from the image. We use the mean-shift clustering method [1] on RGB color space to find the seed colors $(seed_j)$ from sample pixels.

$$seed_j' = \frac{\sum_{r=-w}^{w}\sum_{g=-w}^{w}\sum_{b=-w}^{w} seed_j * n(r_0+r, g_0+g, b_0+b)}{\sum_{r=-w}^{w}\sum_{g=-w}^{w}\sum_{b=-w}^{w} n(r_0+r, g_0+g, b_0+b)}, \quad (2)$$

where w describes the range of the mean-shift. $seed_j$ is the (r_0, g_0, b_0) color value and $n(r, g, b)$ is number of pixels which have (r, g, b) color value.

The mean shift algorithm is a kind of non-parametric clustering technique which does not require prior knowledge on the number of clusters, and does not constrain the shape of the clusters. By the mean shift clustering algorithm, a few (say 2 to 5) most distinctive colors are selected as seed colors. Since we do not know which seed color is obtained from the text region, we repeat the connected component extraction process with each color seed.

When clustering all pixels inside the focused area, the color of text boundary pixels can be chosen as a representative color dropping the true text color. It would bring up an unexpected result such that the text and background are combined or the text is segmented into small pieces. To prevent this adverse effect of boundary pixels, we sample non-boundary pixels from the homogeneous areas which have minimum edge values within 3*3 windows. The edge value is obtained as maximum magnitude $M(x, y)$ of Sobel edge among R,G,B color channels.

$$M(x,y) = max(M^R(x,y), M^G(x,y), M^B(x,y)), \quad (3)$$
$$M^i(x,y) = \sqrt{M_x^i(x,y)^2 + M_y^i(x,y)^2} \ \ (i \in R, G, B).$$

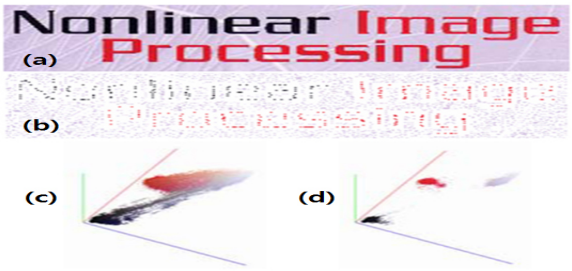

Figure 3. Color distributions of pixel samples

Since most pixels having the minimum edge values belong to text region or background region, we can avoid the undesirable effect of boundary pixels. Figure 3 illustrates the color distribution of the sampled pixels of the original image. The sampled non-boundary pixels are shown in Figure 3 (b). The color distribution of the original image shows that the colors of the text and the background are mixed without distinction (Figure 3 (c)). On the other hand, the color distribution of the sampled pixels shows that they are well separated (Figure 3 (d)).

2.2. Extraction of connected components

In order to achieve robust extraction of text components, we apply binarization on a small region and expand searching to its neighboring areas (Figure 4). An image binarization technique with a seed color is conducted in the HCL color space to classify the area into two regions, i.e., one in similar colors to the seed color and the other in the different colors. The binarization method can effectively separate scene text from complex background in the case that the text pixels have similar HCL color values distinguishable from the background. Furthermore, it has a tendency to extract the text region as a single component even the text color varies smoothly due to the reflection or uneven illumination.

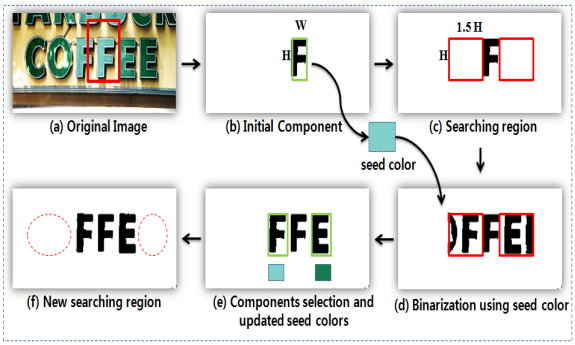

Figure 4. Components expansion

Binarization method needs to set a threshold of a border between two regions. In contrast to the global binarization method which uses a fixed global threshold, the adaptive binarization method finds thresholds adaptively for each pixel. As a result, the local binarization method can handle complex images of low contrast on HCL distance between text and background. For example, Figure 5 shows the difference between the two binarization methods on the HCL distance-of-color image. Adaptive binarization (Figure 5 (d)) shows better result than its counterpart global binarization (Figure 5 (c)): text region is well separated from the background.

After obtaining the target text region within the focus, the target text region is expanded iteratively by searching

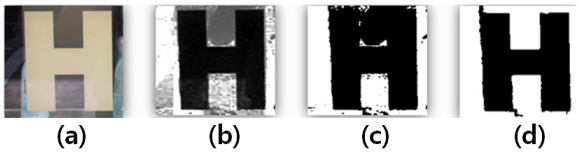

Figure 5. Binarization result: (a)original image (b)HCL distance on seed color (c)global binarization result (d)adaptive binarization result

neighboring area. A text candidate, which is a fully connected component in the binarization result, is first extracted as the initial component (Figure 4 (b)). We search neighbor components of the same color bounded by certain distance (Figure 4 (c)). In the vast majority of cases, a text string is aligned as horizontal in the image and, therefore, neighbor text regions are usually found within a certain distance.

In the searching region we conduct the same component extraction process; an adaptive binarization but with an updated seed color. The seed color is updated by setting as the average color of the newly found neighbor component. By exploiting the characteristic that the nearer in distance causes the less variation for the same colored text regions, neighboring text regions are well extracted with the iterative region growing methods. Searching neighbor components ends when there is no more text component or new component does not satisfy the characteristic of the character.

Five heuristic conditions to stop the component expansion are listed as below. Heuristic rules take into account certain limits for the height, width and location of the newly found connected component (C_2) along with the appearance of the existed component (C_1). The features of the character such as aspect ratio and compactness are also considered. When there is no component satisfying the conditions, we ignore the newly found component and stop searching.

(1) $Exist(C_2) == false$

(2) $Compactness(C_2) = \frac{Area(C_2)}{ContourLength(C_2)^2} \leq t_1$

(3) $t_2 \leq \frac{Width(C_2)}{Height(C_2)} t_3$

(4) $t_4 \leq \frac{Height(C_1)}{Height(C_2)} \leq t_5$

(5) $Overlap_x(C_1, C_2) \geq t_6$

2.3. Text verification

When the component expansion process on each color seed has finished, we decide which component is a text string. Four conditions are used to determine text string from component candidates by checking the global consistency of the text string. In most images, text characters do not appear alone, but together with other characters. All components of text strings are also assumed to be roughly horizontal, nontheless could vertical string be added as well if needed. Characters are subjected to certain geometric restrictions, i.e., their height, width and compactness usually fall into specific ranges of values. The system compares all text candidates which are obtained from each seed color, and then selects the final text region which has minimum variations on the following rules. For example, the components of the top in Figure 6 are selected as the final text region and the text candidate in the bottom is rejected due to the conditions.

(1) Number of components ≥ 3

(2) Variation of distance between components

(3) Variation of heights of components

(4) Variation of compactness of components

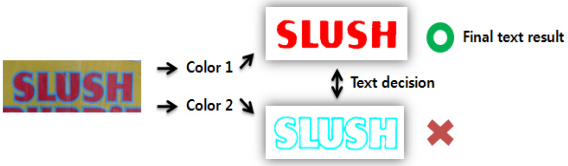

Figure 6. Text verification

3. Experimental result

For evaluating the performance of the proposed method, we used the dataset of the ICDAR 2003 Robust Reading Competition [7]. From the dataset we selected a total of 70 realistic images to include non-uniform illumination, complex background and high variation on shape, size and color of text. Various sizes and locations of the focus are used on the same images to learn the effect of the focus. In sum, a total of 170 cases are used for the evaluation.

We used a similar evaluation method as that of the ICDAR 2003 competition. It is based on the notions of precision and recall which are calculated in terms of number of pixels [3]. Precision p is defined as the number of correct estimates (C) divided by the total number of estimates proposed by our algorithm (E). Recall r is defined as the number of correct estimates (C) divided by the total number of target which is manually labeled text area (T). We then computed the average precision and recall over all the images in the dataset.

$$p = C/|E|, \qquad r = C/|T|.$$

Table 1 showed that the proposed method achieved high precision rate in test images. However, overall recall rate is low (0.51). Since the component searching with the focused section only conducted in the horizontal direction, text strings in the rest parts (above and below the target textline) are ignored. When we measure the recall on the user's

Table 1. Text detection result on test images

	Precision	Recall
Proposed method	0.90	0.51 (0.89)

interested area which is text string regions indicated by focus, high recall rate is achieved (0.89).

Figure 7 shows several examples of the text extraction results. In almost every case, the text areas are detected well as shown in the final binary images. Non-text areas are also eliminated effectively. The proposed method worked successfully even in the case with non-uniform text color. Some separation errors occur on a few scene images. We found that the adaptive binarization with the HCL distance measure is sensitive to cause errors due to a little difference between text and background on the Hue axis. Excessive color change in the same component also causes error. In addition, small strokes often lost during the binarization process. However, total results showed that the target text regions are extracted well from even complex background in the most of cases.

Figure 7. Example of text detection results

4. Conclusion

In this paper, we proposed a text extraction algorithm by utilizing the focus information on scene images. First step, pixel sampling and a mean-shift algorithm are used to choose the text color candidates within the focused section. Second, all pixels in the image are compared to the target seed color in HCL distance measure. And then the adaptive binarization method classifies them into the two regions to form connected components. An iterative region search method then finds the neighboring components. In the last step, text verification based on the heuristic rules is used to determine the true text components. Our method was tested with variations of focuses on the dataset from ICDAR 2003 competitions.

By indicating the location of the target text with the focus interface, the proposed method resolves the difficulties of text extraction on natural scene images caused by non-uniform illumination, complex backgrounds and the existence of text-like objects. We confirmed the feasibility of our method for hand-held camera applications. Restricting the search area within the focused section prevents misclassifications caused by the surrounding non-text regions. While the current method can only extract a single text line from the image, the jump to multi-line texts is also feasible.

Acknowledgements

This work was supported by the Korea Science and Engineering Foundation(KOSEF) grant funded by the Korea government(MOST) 2009-0078943, and by Defense Acquisition Program Administration and Agency for Defense Development of Korea under contract UD080042AD.

References

[1] D. Comaniciu and P. Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 603–619, 2002.

[2] D. Doermann, J. Liang, and H. Li. Progress in camera-based document image analysis. In *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, pages 606–616, 2003.

[3] N. Ezaki, M. Bulacu, and L. Schomaker. Text Detection from Natural Scene Images: Towards a System for Visually Impaired Persons. In *International Conference on Pattern Recognition*, pages 683–686, 2004.

[4] B. Gatos, I. Pratikakis, K. Kepene, and S. Perantonis. Text detection in indoor/outdoor scene images. In *Proc. First Workshop of Camera-based Document Analysis and Recognition*, pages 127–132, 2005.

[5] K. Kim, H. Byun, Y. Song, Y. Choi, S. Chi, K. Kim, and Y. Chung. Scene Text Extraction in Natural Scene Images Using Hierarchical Feature Combining and Verification. In *Proceedings of the 17th International Conference on Pattern Recognition*, volume 2, pages 679–682, 2004.

[6] Y. Li, J. Sun, C. Tang, and H. Shum. Lazy snapping. In *International Conference on Computer Graphics and Interactive Techniques*, pages 303–308. ACM Press New York, NY, USA, 2004.

[7] S. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. ICDAR 2003 Robust Reading Competitions. In *Proc. of the ICDAR*, pages 682–687, 2003.

[8] J. Park, H. Yoon, and G. Lee. Automatic Segmentation of Natural Scene Images Based on Chromatic and Achromatic Components. *LECTURE NOTES IN COMPUTER SCIENCE*, 4418:482, 2007.

[9] C. Rother, V. Kolmogorov, and A. Blake. " GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (TOG)*, 23(3):309–314, 2004.

[10] M. Sarifuddin and R. Missaoui. A New Perceptually Uniform Color Space with Associated Color Similarity Measure for Content-Based Image and Video Retrieval. In *Proc. of ACM SIGIR 2005 Workshop on Multimedia Information Retrieval (MMIR 2005)*, pages 1–8, 2005.