TEXT LOCATION IN COLOR SCENE IMAGES FOR INFORMATION ACQUISITION BY MOBILE TERMINALS

Hao Wang   *Jari Kangas*

Nokia Research Center Visual Communications Laboratory
No. 11, He Ping Li, Beijing, P.R.China

ABSTRACT

A camera integrated to a mobile phone for transmitting and receiving information is an objective people expect in the near future. Because texts always contain useful information, it is valuable to extract text or characters from natural pictures. This paper proposes a connected-component-based approach to automatic text location and recognition in color scene images that are taken by a digital camera. A multi-group decomposition scheme is used to deal with the complexity of the color background. Introduction of weak color and grayscale besides hue space improves the performance of the method. Block adjacency graph (BAG) algorithm is employed for extracting connected components in each image layer and alignment analysis is efficient to obtain accurate location. Some new features are applied in block candidate verification. Results of our experiments prove the efficiency for a wide range of real mobile application environments in the terms of character fonts, shooting conditions, and color backgrounds.

Keywords: Text Location, Scene Image, Multi-Group Decomposition, Connected Component, Alignment Analysis, Camera Phone, Information Acquisition.

1. INTRODUCTION

In the future, mobile phones will become multimedia tools: not only the functionality that standard mobile phones include today, such as initiate and receive phone calls, send and receive short messages and email, manage contacts and calendar entries, etc., but also the fact that phones will integrate a whole variety of sensors for multimedia data, for example, a camera will be assembled at least. However, besides the feasibility of constructing such kind of devices, it is important that the user gets a clear benefit from using such a camera phone.

What is the interesting and valuable information in a picture which is taken by a camera? The answer to this question will give us great revelation in order to provide additional values to the users of such devices mentioned above. It is easily understandable that usually texts and human beings are the two important elements in scene images. So this paper will discuss how to extract textual information from complex natural color scene images. Fig. 1 illustrate the conception of using a camera phone to carry out some machine intelligent applications. Of course, text manipulation is included.

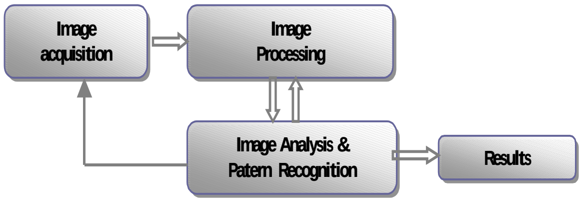


Fig. 1: Block diagram of camera phone

Combined with optical character recognition (OCR) techniques, it is possible to locate and recognize the characters in natural scene images. Most applications of automatic text processing can be divided into three classes: (i) converting texts embedded in color images to electronic data for indexing; (ii) real-time recognizing texts in images which are taken directly by camera for further processing, e.g., recording, translation; and (iii) real-time region-of-interest (ROI) coding for image and video transmission. However, it is obviously that to automatically locate and recognize characters in scene images is much more difficult than that in document images obtained by scanner. The variations of text in terms of character font, size and style, orientation, alignment, texture, diverse language and color, as well as low contrast, complex background, uncontrolled illumination and noise of images make the problem very hard. In addition, a high speed of processing is desired usually.

In this paper, a research effort on a new location method based on connected component analysis is described. Multi-group decomposition using Hue space-Weak color-Grayscale (HWG) scheme is applied and some new features are introduced to verify the text blocks. Fig. 2 shows the flowchart of the proposed method. Texts with uniform color can be detected by considering only those pixels in the layer itself with a specific color. The connected components in each layer are combined according to the rules such as size, eccentricity and alignment. Some assumptions are made in our algorithm: (i) a character has a uniform color;

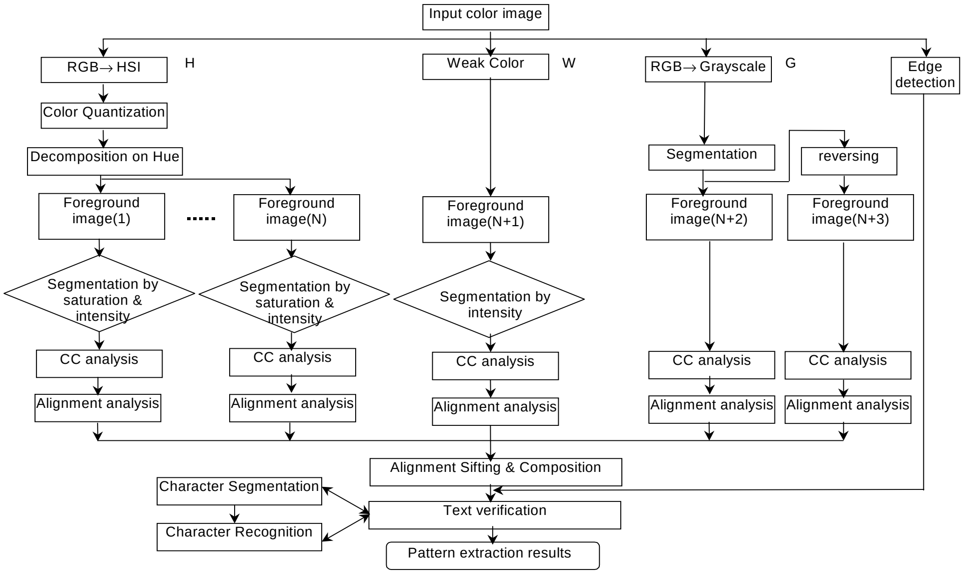


Fig. 2 Flowchart of proposed method

(ii) characters do not touch each other;
(iii) a character is not embedded into graphics with the very color the character has, which is even unable to distinguish by human eyes.

This paper is organized as follows. Section 2 shows related work in text location. HWG decomposition scheme is explained in Section 3. CC extraction combined with alignment analysis is described in Section 4. Section 5 presents block candidates verification and grayscale-based recognition. Experimental results are illustrated in Section 6 and then conclusion is drawn in Section 7.

2. RELATED WORK

Several approaches have been proposed on text location in diverse images including license plates, road signs, book and journal covers, WWW pages and video frames. Those methods can be broadly classified into two types. The first one is texture-based. Ohya *et al*. proposed a method by observing gray-level differences between adjacent regions [1]. Zhong *et al*. used horizontal spatial variance to locate bounding boxes around text components [2]. Sobottka *et al*. proposed a top-down and bottom-up analysis by using the knowledge that regions containing text include at least two colors [3]. Most of the caption location in video uses texture features either in compressed frames or uncompressed frames. Zhong *et al*. used the intensity variation information encoded in the DCT domain directly [4]. The second method is based on connected component. Under the assumption that text is represented with a uniform color (or gray-level), connected components are extracted from each elementary image. Jain *et al*. used color reduction and multivalued image decomposition to extract connected components [5]. Kim used local color quantization to deal with characters on complex background colors and extract candidate text line by merging connected components [6]. Suen *et al*. applied binary edge image to locate character-like components [7]. Furthermore, Zhong *et al*. combined the two kinds of methods in their work [2].

3. MULTI-GROUP (HWG) DECOMPOSITION

A multi-group decomposition is proposed to decompose the color scene image into several binary image layers, in which the connected component analysis will be carried out in the next step. As shown in Fig. 2, there are four groups of layers used in our algorithm. The first group is based on the results of color clustering in hue space. In order to overcome the segmentation difficulty and shortage of HSI spaces when the saturation is very low, the following two groups are generated according to the segmentation results of weak color and grayscale respectively besides using hue component. The last group contains only one layer, namely, an edge image of the original color image, which is helpful for text identification.

3.1. Color reduction

A human observer can perceive millions of natural colors but will not care much about the small variations in the approximately uniform color of an object. Especially, the color variations after digitalization by a digital camera are often not useful and should be eliminated before the decomposition step. Furthermore, digital camera and such kind of devices will bring more or less noise to the images, shadows make the uniform color look different, and over or lack of lightening makes different colors look alike. In our algorithm, Hue-Saturation-Intensity (HSI) format of color image is applied to deal with such problems. Random noise is reduced by quantization in the color format conversion from RGB to HSI. All of the three parameters (H, S, and I) have 36 quantization-levels, which is proved to have good performance in our experiments. Color clustering is implemented in hue space. Weak color and grayscale are also introduced in order to obtain reliable results.

3.1.1. Color clustering in hue space

Since a character has a uniform color comparing with a whole image, the several values represent the character in hue space should be close. Therefore, an unsupervised clustering approach is applied to find clusters of similar colors. All pixels with the colors in a same cluster are labeled with the same color value and then decomposed into the same layer. Fig. 3 shows the fundamental idea of the clustering method: at first, statistical information is evaluated by building a histogram of hue parameter. Secondly, a pointer to the larger neighbor is given to each cell in the histogram. Noting that the hue component makes a round circle, the first cell at the beginning and the last cell in the end of the circle are neighboring. After pointers are set for the whole histogram, several chains of cells pointing to a local maximum are built. The set of cells belonging to such a chain build a cluster. The principle of the method might be described that every local maximum is possible to stand for a color layer which contains some integrated objects. The number of resulting clusters depends on the color quantization-levels, normally 4-9 when the quantization-level is 36. The property of this clustering algorithm is fast and robust for most conditions without knowing the previous information about the input image.

Each cluster is given the hue value of the local maximum in the cluster, which stands for the color of all the pixels belonging to the cluster. This value will be used in character segmentation step.

To deal with the case that texts have the same color with the background but brightness or saturation is different, threshold segmentation technique is optionally used according to the features of histograms calculating in the saturation and intensity spaces.

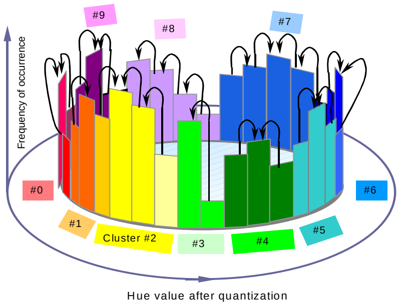


Fig. 3 Example of color clustering

3.1.2. Use of weak color and grayscale

A weakness of HSI format is that the hue component will be very unsteady when the values of R, G, and B become same or very close. Here we use the term *Weak Color* to describe that kind of gray-looking color. Instead of converting from RGB to HSI, each pixel with the color of which the maximum absolute difference of R-G, G-B, and B-G pairs is no more than a given threshold is marked with *Weak Color*. Then all those pixels marked with *Weak Color* are put into a special layer.

Furthermore, there are always some small characters in scene images and those characters can not remain integrated after decomposition using hue values. It can be easily understood that a human observer often pays more attention to the contrast or brightness rather than colors about the details in an image. Grayscale image is applied to deal with the small characters. Gray-level is obtained by mixing the R, B, and B values with unequal weighting, as shown in Eq. (1).

$$Y = \frac{1}{1000}(299 \cdot R + 587 \cdot G + 114 \cdot B) \qquad (1)$$

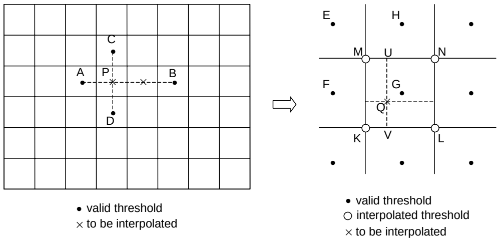
Because there are always different shadows, illumination, and contrast in different zones of a scene image, segmentation with a coordinate-dependent threshold selection is used. Basic idea of the method is to segment the original image into a series of neighboring sub-images, select a threshold of each sub-image and then compute the threshold of each pixel using bilinear interpolation. Fig. 4 illustrates the threshold interpolation operation. In Fig. 4a, A, B, C, D are centers of their sub-images which have valid threshold after checking the histograms. When the histogram of a sub-image is obviously two-peak-shaped, a valid threshold is chosen as the valley position. Otherwise, no threshold is given to that region. Sub-image P is not processed due to its histogram property. To interpolate the threshold of P, Eq. (2)-(4) are employed.

$$Th1 = [Th(B) - Th(A)] \cdot (\overline{AP} / \overline{AB}) + Th(A) \qquad (2)$$

$$Th2 = [Th(D) - Th(C)] \cdot (\overline{CP} / \overline{CD}) + Th(C) \qquad (3)$$

$$Th(P) = (Th2 - Th1) \cdot \min(\overline{AP}, \overline{PB}) / [\min(\overline{AP}, \overline{PB}) + \min(\overline{CP}, \overline{PD})] + Th1 \qquad (4)$$

After getting the thresholds of all sub-images, threshold of any pixel, e.g., pixel Q in Fig. 4b, is computed. The threshold of point M is taken by averaging the thresholds of the nearest sub-images (E, F, G and H), and so do K, L, and N.



(a) (b)

Fig. 4 Bilinear interpolation of threshold

$$Th(U) = [Th(N) - Th(M)] \cdot (\overline{MU} / \overline{MN}) + Th(M) \qquad (5)$$

$$Th(V) = [Th(L) - Th(K)] \cdot (\overline{KV} / \overline{KL}) + Th(K) \qquad (6)$$

$$Th(Q) = [Th(V) - Th(U)] \cdot (\overline{UQ} / \overline{UV}) + Th(U) \qquad (7)$$

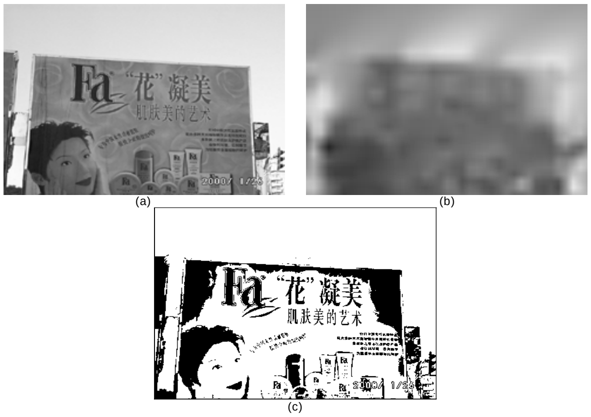


Fig. 5 Example of coordinate-dependent segmentation
(a) original image, (b) threshold surface, (c) segmentation result

All the sub-images touch the edge of the original image are processed using some specific ways in order to get uniform results. Fig. 5 shows an example of the coordinate-dependent segmentation in our experiments. The segmented binary image can be used to extract connected components (especially for small ones) directly.

3.2. Decomposition

The purpose of decomposition is to generate a series of binary images. Those binary images are used to implement CC analysis, which will be explained later. The name HWG derives from the groups divided during the color reduction.

Hue space group: N is defined as the total number of clustering colors in hue space. Thus the quantized image can be decomposed into N binary layers. In the *i*th layer, pixels in the quantized image with color value H_i are assigned to 1, and pixels with other color values are assigned to 0. Furthermore, each layer in this group is optionally segmented depending on the statistical properties of intensity and saturation. If the histogram of intensity/saturation is obviously two-peak-shaped and each peak contains enough black pixels (foreground), the layer will be segmented. Therefore, there are altogether N to 2N layers in the hue space group.

Weak color group: In this group, all pixels marked with *Weak Color* are assigned 1 in a layer, and others are assigned 0. Like layers in hue space group, the layer of weak color is optionally segmented by brightness (intensity). Thus there are at most 2 layers in the group.

Grayscale group: This group contains 2 layers, one is reversed from the other. Coordinate-dependent threshold segmentation is applied to obtain the layers, as described before. These two layers contain some redundant information besides the hue space group and weak color group. However, they have great benefit for extracting small characters in scene images.

Binary edge image: Edge image works as an assistant to the HWG scheme. It can be directly used to extract character-like connected components as described in [7]. However, to create a binary edge image accurately is not an easy task especially when the background is complex with much noise and the contrast is very low. If the edge of a character is not closed, it is not reliable to extract the character in the edge image. So edge image is used in the text verification step rather than in character location directly. The edge detection technique used is based on gradient operations.

4. CC ANALYSIS

To get rid of the Isolated noise points distributed around the characters, average filtering is used in each decomposed binary image. Run length smearing [10] operation is also implemented to deal with the separated parts of a character, which is very common for Chinese characters. Then connected components are extracted in every layer by using BAG (block adjacency graph) algorithm [8]. Fig. 6 shows an example of a smeared character and its BAG representation. Each connected component is bounded by a rectangular block. After the CC analysis, we get a series of connected components in each layer. Some components are deleted first if one of the following conditions occurs:

(i) components are very small or very large;
(ii) components contain very few black pixels;
(iii) components touch the edge of the image.

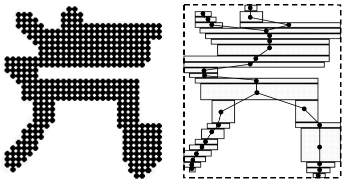


Fig. 6 A smeared character and its BAG

5. CHARACTER VERIFICATION AND RECOGNITION

Alignment is an important property of texts in a natural scene image because characters are always arranged in a certain order. In our algorithm, heuristic alignment analysis is used to combine the separated parts of a character, split the coherent characters, compose the blocks in multiple layers and verify the block candidates of characters. Alignment is measured according to the block sizes, shapes, and arranging positions, and align values are assigned to all of the components.

After confirming all connected components in each layer, different results from different layers are mixed together.

Overlapped blocks are checked by their sizes and align values. Normally, blocks with high align values are retained and weak-alignment blocks are deleted.

Statistical features such as block-eccentricity, foreground-saturation, black-pixel-distribution, and edge-density variation together with recognition confidence are also adopted to verify the block candidates. Character recognition is implemented directly based on grayscale image block. The recognition engine is from the OCR lab of Tsinghua University.

6. EXPERIMENTAL RESULTS

More than 300 images of different character fonts, shooting conditions, and color background in our database are tested and the detection-rate is acceptable to real applications. Fig. 7(a) gives an example of sunshine and shadows. Because shadows have less effect on the hue component than that on brightness, location results are quite good. Fig. 7(b) was taken inside with a flash, so there are sudden illumination variance and much noise. Some of the characters are damaged and hard to recognize even by human eyes. Since color quantization and clustering can restrain noise and run length smearing combined with alignment analysis can link separated parts of characters, the locating result is acceptable. Fig. 7(c) shows an example of high contrast that is good enough to locate both the Chinese characters and English caption. Fig. 8(d) gives a running interface of our demonstration system including character recognition and translation functions. Generally, the HWG decomposition and CC analysis contribute promising results to automatic text location in natural scene images. Future work will consider more complex conditions such as weak-alignment, high noise, extremely diverse fonts, and low illumination.

7. CONCLUSION

An automatic text location scheme based on connected component analysis is discussed in this paper. The description above focuses on the decomposition of color images using our multi-group method. The hue space, weak color and grayscale contribute different weights in the framework and are balanced by the CC analysis step. It is proved that the algorithm is effective especially for the text or characters with uniform color and obvious alignment. Compared with texture-based methods, our scheme has less sensitivity of natural noise, fonts and background effects. Because of introduction of weak color group and grayscale group decomposed layers, the decomposition step is more robust for CC analysis than the existing work introduced in Section 2. So it is more suitable for real applications by using a digital camera.

A good example of mobile usage with the text location techniques is the mobile text input and translation system [9]. It frees the user from actually writing the text himself, which might be difficult or even impossible due to user ignorance or mobile terminal deficiencies (e.g., keyboard limitation). It uses automatic recognition and translation of texts, which can provide much benefit such as signboard recognition, note taking, and Internet data retrieval, etc. It is believed that automatic text location and recognition will play an important role in the mobile info-acquisition system.

8. REFERENCES

[1] J. Ohya, A. Shio, and S.Akamatsu, "Recognizing Characters in Scene Images," *IEEE Trans. on PAMI.*, v16, n2, pp. 214-220, 1994.

[2] Y. Zhong, K. Karu, and A.K. Jain, "Locationg Text in Complex Color Images," *Pattern Recognition*, v28, n10, pp. 1523-1535, 1995.

[3] Sobottka, H. Bunke, and H. Kronenberg, "Identification of Text on Colored Book and Journal Covers," *Proc. of the Fifth ICDAR.*, pp. 57-62, 1999.

[4] Yu Zhong, Hongjiang Zhang, and Anil K. Jain, "Automatic Caption Localization in compressed Video," *IEEE Trans. On PAMI.*, v22, n4, pp.385-392, April 2000.

[5] A. K. Jain and B. Yu, "Automatic Text Location in Images and Video Frames," *Pattern Recognition*, v31, n12, pp. 2055-2076, 1998.

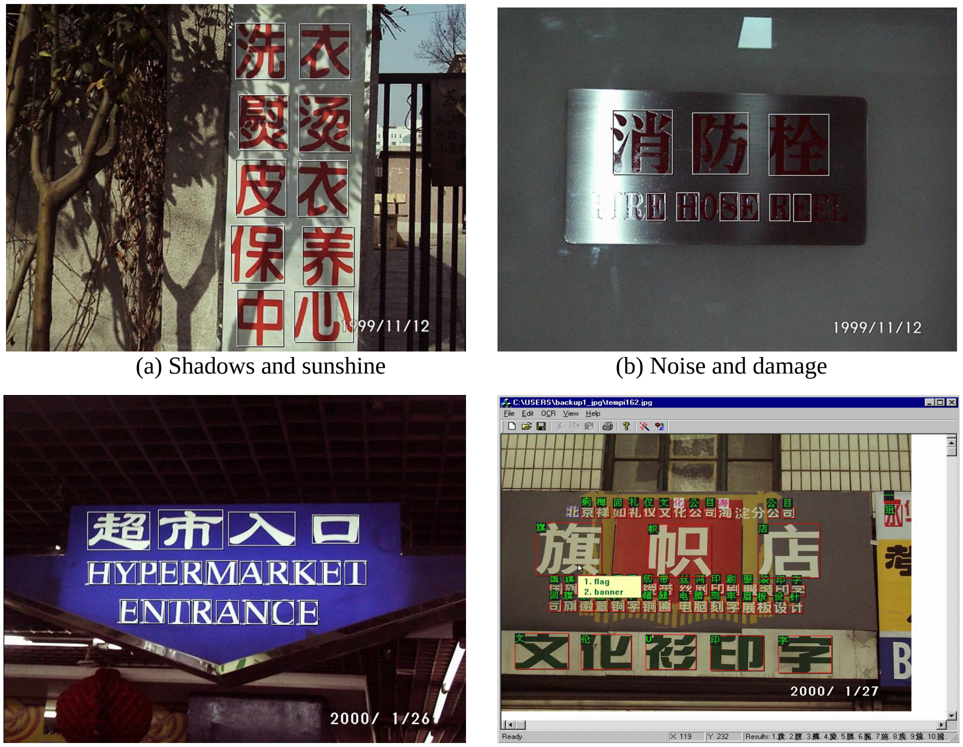
[6] Pyeoung-Kee Kim, "Automatic Text Location in Complex Color Images Using Local Color Quantization," *TENCON 99. Proc. of the IEEE Region 10 Conference*, v1, pp.629–632, 1999

[7] H.-M. Suen, J.-F. Wang, "Segmentation of Uniform-Coloured Text from Colour Graphics Background," *IEE Proc.-Vis. Image Signal Process.*, v144, n6, pp.317-322, 1997.

[8] B. Yu and A. K. Jain, "A Generic System for Form Dropout," *IEEE Trans. on PAMI.*, v18, n11, pp. 1127-1134, 1996.

[9] H. Fujisawa, H. Sako, Y. Okada, and Seong-Whan Lee, "Information capturing camera and developmental issues," *Proc. of the Fifth ICDAR.*, pp. 205-208, 1999

[10] W.Y.Chen, S.Y. Chen, "Adaptive page segmentation for color technical journal's cover images", *Image and Vision Computing*, 16: pp. 855-877, 1998



(a) Shadows and sunshine

(b) Noise and damage

(c) High contrast

(d) Demo

Fig. 7 Experimental results