Image Preprocessing for Camera Phone Based Text Recognition

Heejung Kim, Sangwook Oh, Hoonjae Lee, Sanghoon Sull
Korea University, 1, 5-ka Anam-dong, Sungbuk-gu, Seoul, Korea
hjkim@mpeg.korea.ac.kr, osu@korea.ac.kr, hoonjae@mpeg.korea.ac.kr, sull@mpeg.korea.ac.kr

Abstract – ***With the increase of digital camera resolution and the advance of digital image processing technologies, various applications for camera phones are introduced newly. One of them is to recognize texts from a still image taken by camera phones or a preview raw image which is not compressed. These images are usually distorted by a variety of factors such as light and non-flat surface. Since the distortions make the text recognition difficult, an image preprocessing step is needed to reduce the distortion. This paper proposes an image preprocessing method for correcting the skewed texts appearing on the preview image from image sensor of camera phones by using simple user interaction. The proposed method is implemented on a commercially available camera phones. The experimental results demonstrate the feasibility of our method.***

1. Introduction

The information has been expressed as text, image and sound. The text among them is the most popular method to record information. Nowadays the information printed or written pervades our life. We read a newspaper and magazine to know the news, road signs and subway signs to know the direction, a paper and book to study and a menu and recipe to eat.

The information represented as text is printed or written on the paper or film. As we enter the computer age, it becomes very important to transform the analog data that is the information printed or written into the digital data. To satisfy this requirement, it is needed to change the printed or written text on paper into new form automatically. So the scanner was invented and it has used to convert the document to digital data which can be processed by computer.

When the scanner is used as input device, a user places the document from the left-top point on the scanner, so the start point of scanning can be decided easily. Because the surface of document on the glass is flat, the distance between document and image sensor is always equal with preset value. In addition, the scanner uses a backlight to get the reflect image of document, so the brightness of all parts of image are equal and not sensitive to the environment. However the text recognition using a scanner is restricted by time and space. The common scanner can make an image from only document. And it is possible to scan the document only in the room where the computer and scanner are set up

As the camera resolution increases and the popularity of digital camera rapidly grows, the work on the text recognition for a digital camera has started. In [1, 2, 3], the digital camera is used as an image acquirement devices. The image captured by digital camera is preprocessed and then recognized by OCR software. Although the digital camera extends the area of available devices for text recognition, it can't recognize the text by itself. The image captured by digital camera must be processed by computer.

In these days, the mobile phone becomes the basic necessities of life. In 2007, about 1.1 billion of mobiles are sold in the whole world and this is increased more than 14.7% since 2006 [4]. Moreover the camera phones are expected to account for 89% of all mobile phones shipped by 2009 [5]. The rapid growth of camera phones like this creates many new applications. In [6], Jimmy Addison Lee presented a mobile information guide system for efficient recognition of land-marks taken from camera phone. The user transmits a photograph of his surroundings to a server and then the information mapped to the query image by the server is transmitted back to the user. In [7], Tudor Dumitras used an ordinary, Internet-enabled mobile camera phone, Nokia 6620, to snap a photo of an object and automatically send it to a server using GPRS network. And then the server extracts a text from the image and sends the extracted text back to the camera phone. In the case of [6, 7], the server is needed to extract an information from the image.

However there are some difficult problems in text recognition using camera phone. The image contains out-of-focus text, very small text, angled text and so on. These distortions can be divided into 3 types, namely the optical distortion like aberration, flare and smear made by camera lens and image sensor, the geometric distortion resulting from the non-flat object surface, and the perspective distortion like angled text occurred because the camera is positioned not orthogonally or diagonally to the object. These distortions make it hard to extract and recognize text from the image taken by the camera module embedded in mobile phone. By adjusting a lens aperture or using an aplanat, the optical distortion can be minimized. The camera modules equipped in most of camera phones have no aperture. The adoption of special lens like aplanat is difficult due to high price. So it is very hard to remove the optical distortion from camera phone. However the geometric distortion can be controlled easily by image preprocessing relatively.

Because the camera phone doesn't have enough memory and a powerful processor like PC [7], the more efficient memory management and the specialized algorithm for text recognition are needed.

Therefore we propose an effective image preprocessing method for the skewed image to reduce a geometric distortion. In addition to these preprocesses, the interaction with user using the indicator is added to reduce the skew angle of that preview

image. The proposed technique uses the camera preview image and can be efficiently implemented on mobile camera phone.

2. Application framework

In this paper, we describe the image preprocessing method used in the application processes. The application framework is shown in Figure 1. The well-focused preview image is captured automatically and then this image is sent into the preprocessing part. In this part, the image goes through 3-steps – image binarization, detection of the highest point and calculation of the skew angle. After determining the skew angle of the word, the indicator used to notify the corrected direction to user is drawn in the display and then the user moves the mobile in accordance with that. When the skew angel satisfies the criterion, the captured preview image is sent to the OCR module. The preview image used in these processes is a raw data not a compressed data like JPEG file. In the following Sections, the preprocessing steps will be presented in more details.

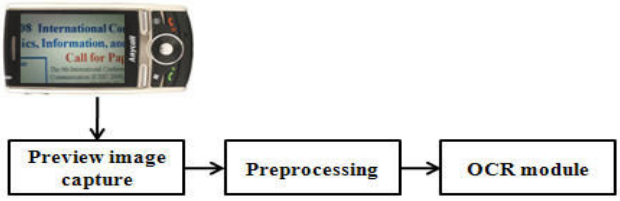

Figure 1: Application framework of recognizing the camera preview image

3. Image binarization

To recognize a text from the required image, the image binarization step is needed to separate a text from the background. During binarizing the image, the thresholds are used to discriminate text pixel and background pixel. Many previous papers presented about a global thresholding method [8, 9] and a local thresholding method [10]. In this paper, we propose an effective binarization scheme for mobile phone. When user wants to translate a word, he locates the word in the prefixed area of the LCD and it is indicated by 4 green marks like Figure 2.

The pixels of one row are used to determine the threshold in Figure 2. From a pixel (w/2, h/2), the values of left and right pixels are checked in shifts. Here the 'w' & 'h' are the width and height of the image respectively.

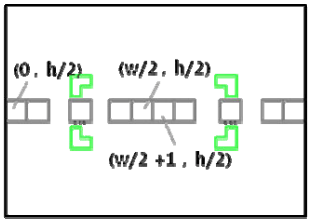

Figure 2: The region of text recognition and the center low used for image binarization

Generally, the text color can be definitely distinguished from the background color but the color distribution of the image has the characteristic of gradual transform like Figure 3 because of the blur effect and noise. In this case, dominant colors or intensities and their range can be found easily by using color histogram thresholding and defined as text value I_{lower} and background value I_{upper} respectively.

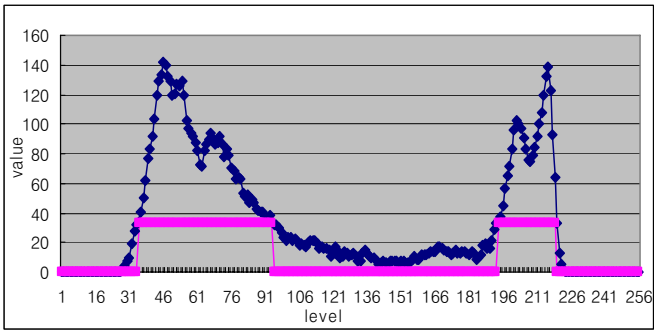

Figure 3: Distribution of the color histogram on the center area image.

After finding the upper and lower dominant intensity value of pixel, the threshold value is calculated like this.

$$Th = (\text{Min } I_{upper} + \text{Max } I_{upper})/2 \qquad (1)$$

Because the color histogram is calculated in only horizontally narrow area, the proposed method is simpler than other thresholding methods and needs less processing time. It is very important to implement a real-time application.

4. Detection of the highest point

During the image binarization, the start point, the end point and the width of target word are determined.

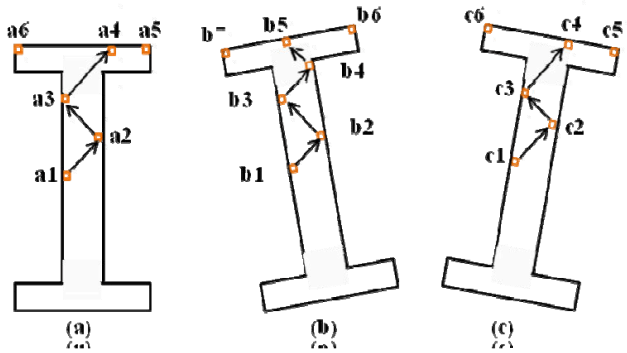

Figure 4: The process of determining the highest point of letter (a): flat case. (b)&(c): skewed cases

Because the pixel ‘a1’ which is the start point of target word is a text pixel, the upper-right pixel is tested if it is a text pixel or a background pixel. Until no more upper-right text pixel exists, this process is repeated. There is no neighbor upper-right text pixel at ‘a2’, so the direction for searching the text pixel is changed to the upper-left. At ‘a3’ and ‘a4’ are found by these searching processes, and there are no upper text pixels at ‘a4’. It means that the ‘a4’ is a top point of the letter. After finding a top point, the leftmost and rightmost point of the extreme row are tried to find and then the ‘a5’ and ‘a6’ are found. Through these processes, the 3-top points (a4, a5, a6) are found and the positions of these points are compared to select the one highest point of this letter. In case of the flat letter like Figure 4a, the heights of three top points are same. However in case of the skewed letters like Figure 4b and Figure 4c, only one point is determined as the highest point. Because the height of ‘b7’ is smaller than that of ‘b6’, the ‘b6’ is the highest point of Figure 4b and the ‘c6’ is the highest point of Figure 4c. Through these processes, only one point is determined as the highest point per letter.

In these processes, the neighbor upper pixels of current text pixel are tested. If all neighbor upper pixels are background pixels, no more upper pixels are tested. Thereupon the small character components such as the top part of ‘i’, ‘j’ and ‘ä’ make no impact on the above processes.

After determining the highest point of one letter, the searching to find the next character is processed. In Figure 4, after determining the highest point among ‘a4’~’a6’, the right pixels of ‘a1’ are tested to find the next letter. Some text pixels and background pixels are found successively and then new text pixel is found. This pixel is a new start point of center row and the process to determine the highest point of new letter is run again. These steps are repeated until there is no more text pixel in center row.

Tall	A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z b, d, f, h, k, l, t
Small	a, c, e, g, i, j, m, n, o, p, q, r, s, u, v, w, x, y, z

Table 1: Character classification based on the highest point

5. Calculation of the skew angle

The letter of the alphabet can be classified by the position of the highest point. The height of all capital letters have the same value, however the small letters have different heights. So we divide all characters into 2 types. Table 1 shows the classification based on the highest point. All uppercase letters are assorted as ‘Tall’, however the lowercase letters are divided into 2 types.

Figure 5 shows the gradients of lines between the highest points of any two letters in same word. In case the two letters are belongs into the different groups, the slopes made by those have various values. Figure 5a shows that the lines between the capital letter ‘I’ and small letter ‘n’, ‘o’ and 'r' have different gradients. In the other way, the lines between two letters of same classification have same angle. The small letter ‘n’, ‘o’, ‘r’ and ‘m’ are members of Small group and the lines between ‘n’ ~ ’o’, ‘n’ ~ ‘r’, ‘n’ ~ ‘m’, ‘o’ ~ ’r’ and ‘r’ ~ ‘m’ have same gradient in Figure 5b. This gradient is same with that of the line between ‘I’ and ‘f’. The ‘I’ and ‘f’ are different character types, but they are belongs into the Tall group together.

(a) The lines between a capital letter and small letters

(b) The lines between two letters in same classification

Figure 5: The lines between the highest points of two letters

The gradients of line made by two letters in the same group are equal with that of base line. In other word, the line which connects the highest points of two characters in same group is parallel with the base line. By the same token, two gradients have same value in Figure 6 and this shows the described principle is right even the skewed word.

Figure 6: The gradients of base line and Tall group letters

In the case of ‘n’, ‘r’ and ‘m’, there are more than one top point – the local maximum point and the global maximum point. In Section 4, one point between the local and global maximum point is determined as the highest point. Because the local and global maximum points are located on the almost same line, the difference of angles made by any local maximum points and any maximum points is few. Moreover the proposed method uses the preview image, so the preview image which has the skew angle that satisfies the criterion is selected by interaction between mobile and user. In the proposed method, the

difference between local and global maximum point is very small, so any maximum points can be used as the highest point.

Figure 7 shows the proposed processes used to determine the angle of skewed word with rapidity. In the first place, we select the first letter of the word used as a basis. In the second place, the skew angle of the line which links the highest points of the first and second letter is calculated. Next the first and third letters are utilized to calculate the skew angle of line. If an angle is found over *n*-times, that angle is determined as the skew angle of the word. If there is no angle found over *n*-times, the basis is changed from the first letter to the second letter. And then the foregoing processes are carried out afresh.

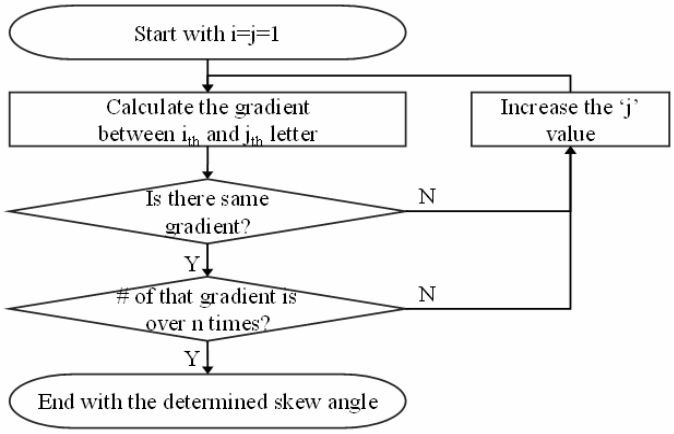

Figure 7: The processes of determining the skew angle of word

6. Experimental results

The proposed preprocessing method is implemented and tested in a commercial mobile. This mobile has a 2-mega pixels CMOS camera that supports the auto-focusing and digital zoom. Because there is no built-in flash, the text recognition in dark environment is very difficult. The preview image processing including the preprocessing proposed needs only 100KB memory and the text recognition engine uses 500KB memory for running.

Because the processing time is important for the real-time application, we check the run time at every step. The first step to select an appropriate preview image for text recognition consumes 0.5 second. The auto-focusing for detecting the text more sharply takes 1.4 second. These steps are needed only one time during one image recognition. To capture the preview image, it needs 0.3 second. Image processing including the proposed method makes use of 0.5 second. It's almost same with the time taken before adopting the proposed method. So the image processing time for text recognition is 0.8 seconds totally. It is available time for the real-time system.

7. Conclusion

This paper presents the image preprocessing method that calculates the gradient in the skewed image for text recognition in camera phone. The proposed preprocessing method uses the simple and specialized algorithms to overcome the limitations of mobile and then the text recognition can be done with only camera phone. The characters are classified into 2 groups based on the highest point and the gradient of line between letters is used to calculate the skew angle of word. Experiment results show our method is very useful for the real-time text recognition in camera phone.

References

1. Shijian Lu and Chew Lim Tan, "Camera Text Recognition based on Perspective Invariants," *International Conference on Pattern Recognition*, Hongkong, Ausust. 2006, Vol. 2, pp. 1042-1045.
2. Wojciech Bieniecki, Szymon Grabowski and Wojciech Rozenberg, "Image Preprocessing for Improving OCR Accuracy", *International Conference on Perspective Technologies and Methods in MEMS Design*, Lviv-Polyana , May, 2007, pp. 75-80.
3. Jun Sun, Yoshinobu Hotta, Yutaka Katsuyama and Satoshi Naoi, "Camera based Degraded Text Recognition Using Grayscale Feature" *International Conference on Document Analysis and Recognition*, Seoul, August, 2005, Vol. 1, pp. 182-186.
4. CIBC World Markets, "2008 Handset Market Guidebook" Jan. 2008
5. InfoTrends "InfoTrends/CAP Ventures Releases World wide Mobile Imaging Study Results" (http://www.infotrends.com/public/Content/Press/2005/01.10.2005.c.html)
6. Jimmy Addison Lee and Kin Choong Yow, "Image Recognition for Mobile Applications", *International Conference on Image Processing*, San Antonio, Texas, September, 2007, Vol. 6, pp. 177-180.
7. Tudor Dumitras, Matthew Lee, Pablo Quinones, Asim Smailagic, Dan Siewiorek and Priya Narasimhan, "Eye of the Beholder: Phone-Based Text-Recognition for the Visually-Imapired", *International Symposium on Wearable Computers*, Montreux, October, 2006, pp. 145-146.
8. R. Lienhart and A. Wernicke, "Localizing and segmenting text in image and videos", *IEEE Transactions on circuits and systems for video technology*, July, 2002, Volume 12, pp. 256-268.
9. N. Otsu, "A threshold selection method from gray-level histogram", *IEEE Transactions on Systems, Man and Cybernetics*, 1979, Vol. 9, pp. 62-66.
10. J. Sauvola, T.Seppnen, S. Haapakoski and M. Pietikinen, "Adaptive Document Binarization", *International Conference on Document Analysis and Recognition*, 1997, pp. 147-152.