TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.1, January 2014, pp. 592 ~ 598
DOI: http://dx.doi.org/10.11591/telkomnika.v12i1.3359

A Fast Thresholding Technique in Image Binarization for Embedded System

Weijun Zhang, Jinkun Dou*
School of Computer Science, Shenyang Aerospace University, Special Instrument Research Institute of Shenyang Aerospace University, Shenyang, 110000, China
School of Computer Science, Shenyang Aerospace University, Shenyang, 110000, China
*Corresponding author, e-mail: elijahdou@gmail.com

Abstract

In the embedded systems of visual navigation, due to the limitations of images capture working environment conditions, there are sometimes defects of nonuniform illumination and noisy in the captured images. Therefore, global threshold methods are unfeasible. On the contrary, image binarization using local threshold methods are more appropriate. However, local threshold methods which take more time to calculate can't satisfy the requirement of real-time performance in embedded systems. In this paper, taking these two limitations into consideration, we proposed an efficient and fast method to determine the threshold value by using integral image and statistical methods which is similar to the local threshold method and meet the requirement of real-time performance in embedded systems. After experiments, for the proposed method, the result has demonstrated that the processing speed is nearly twice the global threshold method. And the processing quality closes to that of local threshold method.

Keywords: *threshold, local adaptive, binarization, integral image, embedded system*

1. Introduction

With the development of machine vision, guidance using path following is widely applied in the field of automated guided vehicles, for short AGV, based on embedded system. Navigation is the core technology of AGV with an enormous complexity. An AGV system can react to prespecified paths inside environments as factories, hospitals, office buildings and so on. Therefore, the captured images by visual navigation system have their own environmental characteristics. Due to the limitations of images capturing condition and working environment, the biggest defect is nonuniform illumination in the capturing environment. It's difficult to achieve satisfying results of binarization in illumination asymmetry images with global threshold method. Therefore, binarization using local threshold method is more appropriate. However, the program is running in the embedded systems which require real-time computing speed. Unfortunately, local threshold methods which need more computing time can't satisfy the requirement. But, we should on the premise of processing quality give attention to improve the speed of processing as soon as possible.

Image binarization is one of the basic techniques of image processing. Image binarization [1] is the process of separation of pixel values into two groups, white as background and black as foreground. Thresholding plays an very important part in binarization of images. Because, an appropriate threshold value is the basis of the subsequent processing operation, such as feature extraction, segmentation, target recognition and so on. In general, thresholding can be categorized into global thresholding and local thresholding. Global threshold methods, when dealing with the images which have a uniform and obvious contrast between background and foreground, namely the image gray histogram having twin obvious peaks, are relatively simple computation and fast. If the illumination over the images is not uniform, for instance the camera-captured pictures in this embedded system, global threshold methods often fail to take the actual situation in various regions of the image into account, and it is difficult to carry out effective image segmentation. In such complexities, noise, uneven illumination, big gray intensity change of background etc, local threshold techniques have been proposed for binarization which estimate a different threshold for each pixel according to the grayscale

Received May 29, 2013; Revised July 13, 2013; Accepted August 10, 2013

information of the neighbouring pixels. The basic idea of the traditional local threshold segmentation methods [1] is as follows:

First of all, original image is divided into a number of sub-picture, wherein each block of the size of the sub-picture can not be equal. Secondly, we compute and get the corresponding segmentation threshold value for each sub-picture to complete their own binarization. Finally, we merge each sub-picture together to complete the whole image binarization. Although local thresholding methods take the local actual situation into account, the connection between each sub-picture is not strong enough. There are mutations between the threshold values of adjacent sub-pictures, so that the image is prone to have blocky effect, affecting the quality of the image segmentation.

2. Related Work of Thresholding

The threshold method proposed by Otsu [2] is a global adaptive binarization method, which is the best representative of global adaptive thresholding, and try to find a single threshold value for the whole image. Then each pixel is assigned to foreground or background based on its gray value [3]. Because of existing noise and uneven illumination in the captured images by camera, Otsu method is not preferable.

Local adaptive thresholdings, they are preferable to handle the captured images in this system. However, local adaptive thresholdings have higher calculation complexity than that in global adaptive thresholding. Local adaptive threshold techniques need to calculate threshold values for each pixel, based on some local statistics of the neighborhood pixels such as range, variance, or other. Some drawbacks of the local threshold techniques are time consuming and region size dependant, such as background subtraction [4], mean and standard derivation of pixel values [5], and local image contrast [6]. Therefore, using a hybrid approach that applies both global and local thresholding methods [7] also takes too much time to complete binarization.

Some researchers have proposed many corresponding improved methods for local adaptive thresholdings [8-14], but their common feature is a computational intensive. All these local adaptive thresholdings are not suitable to embedded system. Singh, et al. [15] have proposed an improved method based on Sauvola and Pietaksinen's method [16] and Niblack's method [5]. The three local thresholding methods can get better results especially for stained and badly illuminated images. It is just to meet the requirements of the captured images in this system. Both of them, Sauvola and Pietaksinen's method and Niblack's method, are using local variance technique. In Sauvola's method the local threshold value $T(r,c)$ at pixel position (r,c) is calculated within a window of size w, the threshold value calculation equation of gray-scale images as:

$$T(r,c) = a(r,c)[1 + k(\frac{\partial(r,c)}{128} - 1)] \quad (1)$$

Where $a(r,c)$ and $\partial(r,c)$ are the local mean and standard deviation of the pixels inside the local window and k is a bias, which takes positive values in the range [0.2, 0.5]. The bias k controls the level of adaptation varying the threshold value.

However, in Niblack's method, the calculation equation as:

$$T(r,c) = a(r,c) + k\partial(r,c) \quad (2)$$

According to the contrast in the local neighborhood of the pixels, the local mean $a(r,c)$ and standard deviation $\partial(r,c)$ adapt the value of the threshold. From Eq. (1-2), we can get the message that the computational complexity is $O(mnw^2)$ for an image of size of $m \times n$.

Obviously, the computational complexity is unacceptable for an embedded system, especially when it processes big size and a lot of images. Singh, et al. [15] proposed a new local thresholding based on the integral image, and its running time approaches that of global binarization method.

The concept of integral sum images was popularized in computer vision by Viola and Jones [9]. Suppose an image is w pixels wide and h pixels high. Then the integral of this will be $w+1$ pixels wide and $h+1$ pixels high. The first row and column of the integral image are all zeros. All other pixels have a value equal to the sum of all pixels before it. The value in integral image I at (r,c) of original image G is calculated as:

$$I(r,c)=\sum_{i=0}^{r}\sum_{j=0}^{c}G(i,j) \tag{3}$$

The integral image of any grayscale image can be efficiently computed in a single pass, integral sum of entire pixels at (r,c) as:

$$I(r,c)=I(r-1,c)+I(r,c-1)-I(r-1,c-1)+G(r,c) \tag{4}$$

Integral sum of 1st row at $(0,c)$ and 1st column at $(r,0)$ as:

$$I(0,c)=0 \tag{5}$$

$$I(r,0)=0 \tag{6}$$

Using Eq. (4-6), we can get the integral sum image of any grayscale image in a single pass. Then, the local sum $s(r,c)$ at (r,c) which is the centre of the local window of size $w\times w$ of original image G is the sum of all the pixel intensities within the local window.

Formerly, the sum $s(r,c)$ can be calculated in two passes as:

$$s(r,c)=\sum_{i=r-l}^{r+l}\sum_{j=c-l}^{c+l}G(r,c) \tag{7}$$

where $l=\frac{w-1}{2}$,since w is an odd number.

While we use integral image I, Eq. (7) can change from two passes to single pass without depending on window size w as:

$$\begin{aligned}s(r,c)&=[I(r+d-1,c+d-1)+I(r-d,c-d)]\\&-[I(r-d,c+d-1)+I(r+d-1,c-d)]\end{aligned} \tag{8}$$

where $d=round(\frac{w}{2})$.

In the local window of size $w\times w$ of the original image G, the local arithmetic mean $a(r,c)$ at (r,c) is the average of the pixels. It can be calculated by using Eq. (8) as:

$$a(r,c)=\frac{s(r,c)}{w^2} \tag{9}$$

In this way, the local mean, using integral image, can be calculated efficiently in a single pass without depending on local window size. In the Singh‘s proposed technique, it requires to compute the local mean $a(r,c)$ and mean deviation $\partial(r,c)$ rather than local standard deviation to determine the local threshold as:

$$T(r,c) = a(r,c)[1 + k(\frac{\partial(r,c)}{1-\partial(r,c)} - 1)] \quad (10)$$

Where $\partial(r,c) = G(r,c) - a(r,c)$ is the local mean deviation and k is a bias which can control the level of adaptation varying threshold value. Its range is [0,1] only.

The proposed technique by Singh,et al. calculate the local threshold using Eq. (4-6) (9-11). The Singh's method is faster than Sauvola's method, and its running time approaches that of global binarization method. Morevoer, it is better than other contemporary relevant methods, both in terms of quality and speed.

3. Proposed Technique

The proposed method by Singh, et al. has a drawback that the size w of the window is an odd number. And the method needs the gray intensity value of the centre of the local window to calculate. All that is inflexible. What's more, this local thresholding doesn't take the local actual situation into account, the connection between each sub-picture is not strong enough. There are mutations between the thresholds of adjacent sub-pictures, so that the processed image is prone to have blocky effect, affecting the quality of the image segmentation.

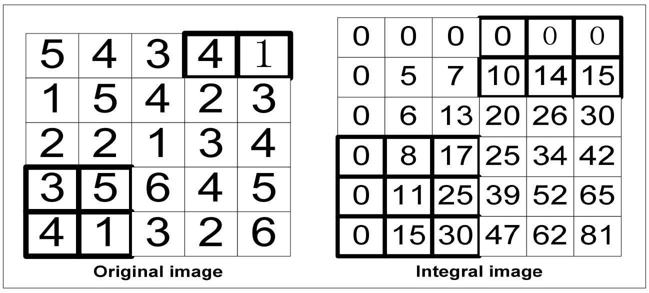


Figure 1. Schematic diagram of integral image

See the integral image in the Figure 1, every pixel is the summation of the pixels before it (above and to the left). Now, to calculate the summation of the pixels in the black box, we take the corresponding box in the integral. We sum as: (Bottom right + top left – top right – bottom left)

$$s_{x-window} = s_{br} + s_{tl} - s_{tr} - s_{bl} \quad (11)$$

Where $s_{x-window}$ is the gray intensity summation of pixels in the box, s_{br} is the bottom right point of the box, s_{tl} is the top left point of the box, s_{tr} is the top right point of the box, s_{bl} is the bottom left point of the box. So for the 3,5,4,1 box, the calculations would go like this: (30+0-17-0 = 13). For the 4,1 box, it would be (0+15-10-0 = 5). This way, we can calculate summations in rectangular regions rapidly.

Based on the individual captured image characteristics, the method we proposed is as follows:

Step 1: Use Eq. (4-6) calculate to get the integral image I from the original image G.

Step 2: Select a window size w (determine the optimum value of w by experiments and can be an odd or even number, according to the characteristics and size of captured images) for each local rectangle window and each window size w maybe not equal, so that we can divided the image G into a number of different size sub-pictures as we need by using Eq.(11).

Step 3: Then we calculate the local threshold $a(r,c)$ of each sub-picture using Eq.(10)(12) , and get the average $mean$ of all the local thresholds and the standard deviation ∂ between $mean$ and the set of local threshold values $a(r,c)$.

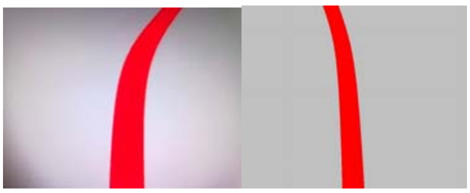
Step 4: Based on Eq.(10), we can get a new formule using $mean$ and ∂ as:

$$T(r,c) = mean[1 + k(\frac{\partial}{1-\partial} - 1)] \quad (12)$$

Using Eq. (9) (11-12), we can overcome the defects mentioned above.The key of proposed method is to determine the appropriate window size w and bias k , both of them determined by experiment and taking a major role in determining threshold value. The lower value of w may lead to be no statistical significance and the higher value can not rule out the impact of noise and uneven illumination. The lower value of bias k makes the threshold value higher and higher value of bias k lowers the threshold value.

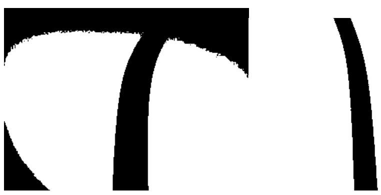
4. Experimental Result

To verify the validity of proposed method, we compare the performance of the proposed method with Otsu method and Singh's method. The experiment for the proposed method was carried out using Linux 3.01 OS on S3C6410 development board based on core of ARM11 with the following configration: 256M DDR memory, 2G NAND Flash and 533MHz. There are three comparison groups for this experiment, and two different illumination images are peocessed with the above three method. Both qualitative and quantitative analysis are carried out in comparison of the proposed technique with the oters. Figure 2(a-b) respectively are uneven illumination and uniform illumination of image with the same size of 320×240 . Figure 3(a-b) are the processed images by Otsu method. Figure 4(a-b) are the processed images by Singh's method. Figure 5(a-b) are the processed images by proposed method. Under the set experimental conditions above, the calculation program run for 5 consecutive times, then we figure out the average time in user space of the 5 times (do not compare the real time and system space time). Experiment data is shown in Table 1.



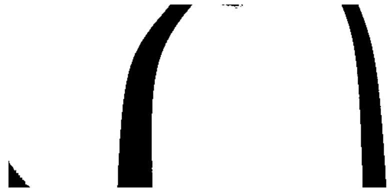
a) Uneven illumination b) Uniform illumination

Figure 2. Original images



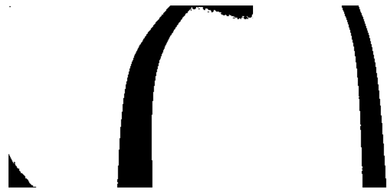
a) Uneven illumination b) Uniform illumination

Figure 3. Processed by Otsu method



a) Uneven illumination b) Uniform illumination

Figure 4. Processed by Singh method(bias k =0.10,windows size w =11)



a) Uneven illumination b) Uniform illumination

Figure 5. Processed by proposed method (optimal k = 0.17,window size w =10)

The figures 3-5 b) show that all the three methods can obtain satisfactory results when uniform illumination image is processed. In the figures 3-5 a) show the test results of the three methods for comparison when uneven illumination image is processed. Otsu method can not distinguish the foreground from background and the result is far from the ideal one, while the Singh's method and the proposed, the two methods have approximative performance. According to Table 1, running time of Singh's method approaches that of Otsu method. However, the running time of the proposed method is half of them.

Table 1. Running time in user space

Figure	OTSU	Singh's Method	Proposed Method
Figure 1 a)	0.041s	0.046s	0.020s
Figure 1 b)	0.040S	0.044s	0.019s

5. Conclusion

In this paper, we present a fast and efficient threshold method for the uneven illumination captured images which is suitable for embedded systems. Based on the individual characteristics of captured images, integral image is used to compute the grayscale average of pixels in the local window so that we can reduce the computation time consuming, because the running time does not depend on the local window size to compute mean in local windows. The running time is half of Otsu method and Singh's method. So the proposed method has a good real-time performance in embedded systems. For the uneven illumination images, the quality of the processed images is better than that of Otsu method, and is approximative with that of Singh's method. What's more, the proposed method can be applied to the fast filtering. The proposed method is suitable for embedded systems to process images binarization extremely.

References

[1] Han Siqi, Wang Lei. A Survey of Thresholding Methods for Image Segmentation. *Systems Engineering and Electronics*. 2002; 24(6): 91-95.

[2] N Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. Systems, Man, and Cybernetics*. 1979; 9(1): 62-66.

[3] Weijun Zhang, Dongli Li. Research on Barcode Image Binarization in Barcode Positioning System. *IJCSI International Journal of Computer Science Issues*. 2012; 9(5-3): 108-112.

[4] B Su, S Lu, CL Tan. *Document Image Binarization Using Background Estimation and Stroke Edges*. Proc. Intl. Journal on Document Analysis & Recognition. 2010; 13(4): 303 -314.
[5] J Sauvola, M Pietikainen. Adaptive document image binarization. *Pattern Recognition*. 2000; 33(2): 225-236.
[6] B Su, S Lu, CL Tan. *Binarization of Historical Document Images Using the Local Maximum and Minimum*. Proc. Intl. Workshop on Document Analysis Systems. 2010: 159-165.
[7] B Gatos, I Pratikakis, SJ Perantonis. Adaptive Degraded Document Image Binarization. *Pattern Recognition*. 2006; 39(3): 317-327.
[8] Yuniarti Anny, Nugroho Anindhita Sigit, Amaliah Bilqis, Arifin Agus Zainal. Classification and numbering of dental radiographs for an automated human identification system. *TELKOMNIKA Telecommunication Computing Electronics and Control*. 2012; 10(1): 137-146.
[9] Qingfu Li, Ye Yuan, Junxia Chen. Segmentation for fabric weave pattern using empirical mode decomposition based histogram. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 1(4): 1996-2001.
[10] Huang He, Li Qingwu, Fan Xijian. Adaptive Local Threshold Image Segmentation Algorithm. *Opto Electronic Technology*. 2011; 31(1): 10-13.
[11] Wang Liangliang, Wang Li, Gao Xiaorong, Wang Zeyong. Two Improved Local Threshold Algorithms. *Image Analysis*. 2009; 14: 78-80.
[12] He Lin, Li Jianwei. Selection of dynamic threshold based on zero-crossing detector. *Opto-Electronic Engineering*. 2003; 30(4): 65-69.
[13] Wu Bing, Qin Zhiyuan. New Approaches for the Automatic Selection of the Optimal Threshold in Image Binarization. *Journal of Institute of Surveying and Mapping*. 2001; 1(4): 283-287.
[14] Zhang Aihua, Yu Shengsheng, Zhou Jingli. A Local Threshold Segment Algorithm Based on Edge-detection. *Mini-Micro Systems*. 2003; 24(4): 661-664
[15] T Romen Singh, Sudipta Roy, O Imocha Singh, Tejmani Sinam, Kh Manglem Singh. A New Local Adaptive Thresholding Technique in Binarization. *IJCSI International Journal of Computer Science Issues*. 2011; 8(6-2): 271-276.
[16] Niblack W. An introduction to digital image processing. *Prentice-Hall, Englewood Cliffs, NJ*. 1986: 115-116.