A New Method of Character Strings Extraction based on Blanket Binarization

Hiromi YOSHIDA
Kobe University
Graduate School of Maritime Sciences
Japan
070w706w@stu.kobe-u.ac.jp

Naoki TANAKA
Kobe University
Graduate School of Maritime Sciences
Japan
ntanaka@maritime.kobe-u.ac.jp

Takashi NAGAMATSU
Kobe University
Graduate School of Maritime Sciences
Japan
nagamatu@maritime.kobe-u.ac.jp

***Abstract*—In this paper, a binarization method based on fractal dimension for character string extraction and a improving method to reduce the processing time are proposed.**

***Keywords*-binarization; blanket method; Fractal;**

I. Introduction

In this paper, we proposes the new binarization method for character string extraction consist of three steps and a improving method to reduce the processing time are proposed. In first step, plural binary images named CCRI(Candidate Character Region Images) are generated by blanket method. Second step is a noise reduction process which consists of two steps. Third is a confliction solving process of CCRI. Finally, single integrated character string image is generated.

II. Fractal dimension

A. Blanket method

Fractal dimension(FD) has been proposed as a method of texture analysis named ”Blanket method” by SHUMEL[1] in 1984. The range of the dimension is from 2 to 3 and is obtained from the equations 1,2,3 and 4.

$$U_\epsilon = \max\{U_{\epsilon-1}(i,j)+1, \max_{|(m,n)-(i,j)|\le 1} U_{\epsilon-1}(m,n)\} \quad (1)$$

$$b_\epsilon = \min\{b_{\epsilon-1}(i,j)-1, \min_{|(m,n)-(i,j)|\le 1} b_{\epsilon-1}(m,n)\} \quad (2)$$

$$A(\epsilon) = \frac{\sum_{i,j}(U_\epsilon(i,j) - b_\epsilon(i,j))}{2\epsilon} \quad (3)$$

$$A(\epsilon) = F\epsilon^{2-D} \quad (4)$$

B. Fast blanket method for bainary image

Usually, the Blanket method is applied to a gray level image, although our proposed method deals with a binary image. In binary cases, a pixel value of an input image has ether 0(black) or 255(white), equations 1 and 2 can be as following

$U_{\epsilon-1}(i,j) \ge 255 \Rightarrow U_\epsilon(i,j) = U_{\epsilon-1}(i,j)+1$
$U_{\epsilon-1}(i,j) < 255 \wedge (U_{\epsilon-1}(m,n))|(|(m,n)-(i,j)| \le 1) < 255 \Rightarrow U_\epsilon(i,j) = U_{\epsilon-1}(i,j)+1$
$U_{\epsilon-1}(i,j) < 255 \wedge (U_{\epsilon-1}(m,n))|(|(m,n)-(i,j)| \le 1) = 255 \Rightarrow U_\epsilon = 255$

$b_{\epsilon-1}(i,j) \le 0 \Rightarrow b_\epsilon(i,j) = b_{\epsilon-1}(i,j) - 1$
$b_{\epsilon-1}(i,j) > 0 \wedge (b_{\epsilon-1}(m,n))|(|(m,n)-(i,j)| \ge 1) < 0 \Rightarrow b_\epsilon(i,j) = b_{\epsilon-1}(i,j)+1$
$b_{\epsilon-1}(i,j) > 0 \wedge (b_{\epsilon-1}(m,n))|(|(m,n)-(i,j)| \ge 1) = 0 \Rightarrow b_\epsilon = 0$

So, they can be represented as

$$U_\epsilon = \begin{cases} 255, (U_{\epsilon-1}(i,j) < 255 \wedge \underset{|(m,n)-(i,j)|\le 1}{U_{\epsilon-1}(m,n)} = 255) \\ U_{\epsilon-1}+1, (Otherwise) \end{cases} \quad (5)$$

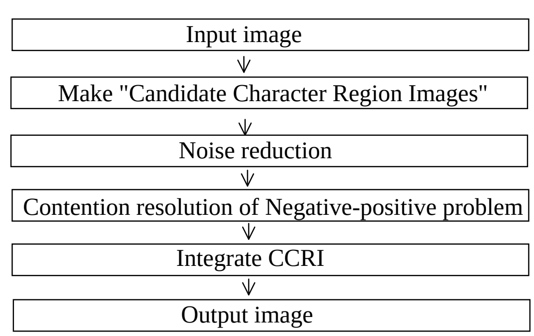
$$b_\epsilon = \begin{cases} 0, (b_{\epsilon-1}(i,j) > 0 \wedge \underset{|(m,n)-(i,j)|\le 1}{b_{\epsilon-1}(m,n)} = 0) \\ b_{\epsilon-1}-1, (Otherwise) \end{cases} \quad (6)$$

This transformation will allow speeding up the blanket method by rejecting the unnecessary calculations.

III. Proposed algorithm

A. Summary of algorithm

Summary of proposed method is shown in the following chart.



B. Candidate Character Region Images

The algorithm for getting "Candidate Character Region Images" is obtained by following steps.

- $\boldsymbol{Step1}$: Binarize the input image $\boldsymbol{I(x)}$ with the threshold values from $\mathbf{0}$ to $\mathbf{255}$, and obtained the 256 binarized images $\boldsymbol{I_{b_i}(x)}$ $\boldsymbol{(i = 0, 1, ..., 255)}$ respectively.
- $\boldsymbol{Step2}$: The $\boldsymbol{FD(i)}$ values can be calculated on $\boldsymbol{I_{b_i}(x)}$ images by the Blanket method.
- $\boldsymbol{Step3}$: We treat the $\boldsymbol{FD(i)}$ to be a function respect to $\boldsymbol{i}$, and find the stable interval of $\boldsymbol{FD(i)}$ by taking the first derivation. we get binarized images and their inversed images from the stable interval(They are CCRI).

C. Noise reduction process

We introduce a noise reduction process to remove the non-character regions. They are explained following.

1) Noise reduction 1 "Bounding-Box": "Bounding-Box" (BB) is a rectangle that encloses the connected-component(CC)s on a CCRI. The noise reduction process on the height, width, aspect ratio and the size of BB are widely used to refine the image in many related researches. The parameters are obtained by equation 7.

$$\begin{aligned} \frac{width(BB)}{width(CCRI)} &\leq \frac{1}{5} \\ \frac{height(BB)}{height(CCRI)} &\leq \frac{1}{5} \\ area(BB) = width(BB) \times height(BB) &\geq \\ \frac{2 \times width(CCRI) \times height(CCRI)}{100 \times 100} & \\ \frac{1}{8} \leq \frac{width(BB)}{height(BB)} &\leq 8 \\ \frac{area(CC)}{area(BB)} &\geq 0.15 \end{aligned} \tag{7}$$

where *width()*, *height()* and *area()* mean width, height and whole of number of pixels respectively. CCs with BB which is less than threshold range are rejected as a noise.

2) Noise reduction 2 "Edge Contrast Feature": A character string in a scene image are designed intentionaly to be easy to read , so the included character region has high contrast against to the background. This "Edge Contrast Feature" is presented at equation 8.

$$\begin{aligned} &Feature\_Edge = \tfrac{Border(CC) \cap Edge(Picture)}{Border(CC)} \\ &Edge(Picture) = \\ &Canny(Picture) \cup Sobel(Picture) \end{aligned} \tag{8}$$

where *Canny(Picture)* and *Sobel(Picture)* are the normalized Canny and Sobel edge of an input image respectively. And *Border(CC)* is the border pixels of the CC on CCRI. By threshold processing of $Feature\_Edge$, the CCs with small edge closure are rejected as noises.

D. resolve of confliction

After the noise reduction, the "Negative-positive problem" in CCRI are solved and integrated into one binary image. The process is obtained following.

$$\begin{aligned} ResultCC = MAX(cCC1, cCC2, ..cCCn) \\ cCCl, cCCm : CC'l \cap CC'm \neq 0 \\ CCn' = (CCn \oplus r_3Bsquare) \end{aligned} \tag{9}$$

$r_3Bsquare$ means square shape structure element with a width of 3 pixels. $\oplus$ represents a dilation operation of the mathematical morphology. Through this process, Negative-positive problems are solved.

IV. Experiment result and conclusion

Binarized images by proposed algorithm are shown at Fig 1.

(a)Input image (b)Output image

Figure 1. The examples of output image

The thresholding method we introduce here generates several binary images correspond to each character string, remaining region conflictions and a noise problem. However, by using the noise reduction process and confliction resolving process, these problems are solved successfully.

References

[1] S. Peleg, J. Naor, R. Hartley, D. Avnir, "Multiple resolution texture analysis and classification", IEEE Trans. Pattern Analysis and Machine Inteligence, vol6 no.4, pp.518-523, July 1984