MEMOIRS OF THE FACULTY OF ENGINEERING
FUKUI UNIVERSITY
VOL. 25 No. 1 1977

An Algorithm for Calculating Perimeter Length

Keiji TANIGUCHI*, Hideo OGAWA*, Yutaka NAKANO*

(Received Dec.15, 1976)

The present paper proposes an algorithm for calculating accurately the perimeter length of a picture. The outline of the algorithm is as follows : for each cell or picture element (i,j) on the perimeter, a coefficient which corresponds to the minute perimeter length at the cell is assigned as a function of the situation of neighbor cells (i,j-1), (i-1,j) and (i-1,j-1), and the number of same situation connected along the perimeter.
Then, the total perimeter length can be given as the sum of these coefficients。 According to this methods, approximation accurate to 0。2 percent could be obtained in the blob patterns。

1. Introduction

In picture processing it is often desired to compute the perimeter length of the objects in a picture. When a computer is employed to calcurate the perimeter, the object picture is digitized。 Several such algorithms have been studied.[1) 2) 3)]
However, high accuracy cannot be expected there。 In this paper, considering the same problem, an improved algorithm to obtain the perimeter length with higher accuracy is presented.

2。 Algorithm

As to the calculation of perimeter length of a picture, we assume that the digitization is so fine that any 2×2 cell array includes only one edge (perimeter) and the edge in the array can be regarded as a straight line. Then this total length may be obtained using the following four step algorithm:

Step 1 : Digitize the picture plane into $n\times n$ matrix $\mathbb{P}$ and transform it into the binary matrix $\mathbb{A}$, according to

*Dept. of Electronics。

$a_{ij} = 1$ if $p_{ij} \geqq h$

$$1 \leqq i,\ j \leqq n$$

$a_{ij} = 0$ if $p_{ij} < h$

where p_{ij} is the grey level of the (i,j) element of $\mathbf{P}$ and h is the threshold value.

Step 2 : Construct two matrixes $\mathbb{K}$ and $\mathbb{M}$ from $\mathbb{A}$ by considering the 2×2 pertial matrix α_{ij} defined as $\alpha_{ij} = \begin{pmatrix} a_{i-1\,j-1} & a_{i-1\,j} \\ a_{i\,j-1} & a_{ij} \end{pmatrix}$ (Fig.1).

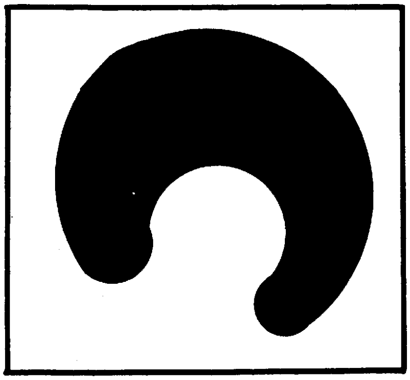
If α_{ij} is one of $\begin{pmatrix} 11 \\ 00 \end{pmatrix}, \begin{pmatrix} 00 \\ 11 \end{pmatrix}, \begin{pmatrix} 10 \\ 10 \end{pmatrix}$ or $\begin{pmatrix} 01 \\ 01 \end{pmatrix}$,

let $k_{ij} = 1$ and $m_{ij} = 0$.

If α_{ij} is one of $\begin{pmatrix} 00 \\ 01 \end{pmatrix}, \begin{pmatrix} 11 \\ 01 \end{pmatrix}, \begin{pmatrix} 00 \\ 10 \end{pmatrix}$ or $\begin{pmatrix} 11 \\ 10 \end{pmatrix}$,

let $k_{ij} = 0$ and $m_{ij} = 1$.

Otherwise, $k_{ij} = m_{ij} = 0$.



(ŏriginal picture)

(step 1)

$$\begin{matrix}
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&1&0&0&0&0\\
0&0&0&1&1&1&1&1&1&1&0&0&0\\
0&0&1&1&1&1&1&1&1&1&0&0&0\\
0&0&1&1&1&1&1&1&1&1&1&1&0\\
0&0&1&1&1&1&1&1&1&1&1&1&0\\
0&0&1&1&1&0&0&0&1&1&1&1&0\\
0&0&1&1&1&0&0&0&0&1&1&1&0\\
0&0&1&1&1&0&0&0&0&1&1&1&0\\
0&0&0&1&1&0&0&0&0&1&1&0&0\\
0&0&0&0&0&0&0&0&1&1&1&0&0\\
0&0&0&0&0&0&0&0&1&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0
\end{matrix}$$

(matrix $\mathbb{A}$)

(step 2)

$$\begin{matrix}
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&1&0&0&0\\
0&0&0&1&0&0&0&0&0&0&1&0&0\\
0&0&1&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0
\end{matrix}$$

(matrix $\mathbb{M}$)

$$\begin{matrix}
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&1\\
0&0&1&0&0&0&1&1&0&0&0&0&1\\
0&0&1&0&0&1&0&0&0&0&0&0&1\\
0&0&1&0&0&1&0&0&0&1&0&0&1\\
0&0&0&0&0&1&0&0&0&1&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0
\end{matrix}$$

(matrix $\mathbb{K}$)

Fig.1. Construction of the matrixes $\mathbb{K}$ and $\mathbb{M}$.

Step 3 : Select the weight coefficient c_{ij} of m_{ij} using the table 1. This selection depends upon the value n of the chain to which m_{ij} belongs in the matrix $\mathbb{M}$, where the chain is a connected component of m_{ij} and the value of a chain is the number of elements in it. (see Fig. 2)

Table.1. Weight coefficients c of m .

n	1	2	3	equal to or more than 4
c	1.1755	1.2687	1.3105	1.4142

chain
(n = 1)

chain
(n = 2)

chain
(n = 3)

$$
\begin{matrix}
0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0
\end{matrix}
$$

(matrix $\mathbb{M}$)

Fig.2. Value n of a chain

Step 4 : Calculate the perimeter length L from $\mathbb{K}$ and $\mathbb{M}$.

$$L = \ell \sum_{ij} (k_{ij} + c_{ij} m_{ij})$$

where ℓ is the unit length of a cell.

3. Experimental Result

When a original picture is black and white pattern, the binary dicision in step 1 can be approximately made according as the center point of a cell is included in the black region or in the white region. We calculated the perimeter length of the blob patterns, for example as shown in Fig. 3, by using this algorithm. As the result, the approximation accurate to 0.2 percent was obtained in 40×40 digitization. This is highly accurate, and is practical for high speed picture processing.

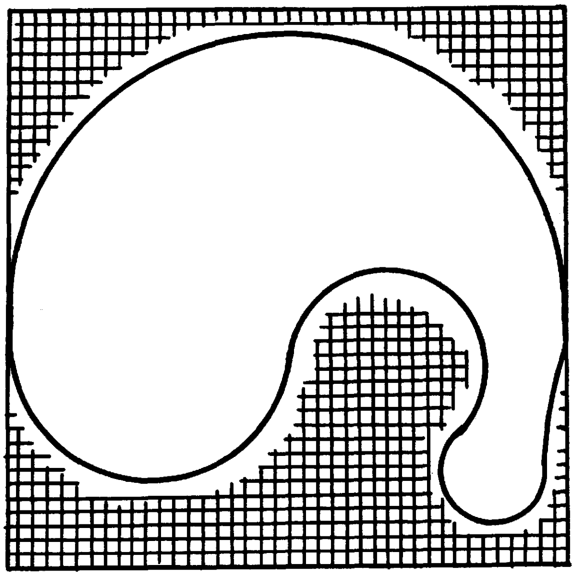


Fig.3. Example of a blob pattern.

References

1) S.B.Gray, : IEEE Trans.Comput., vol.C 20,pp.554-557,May. 1971.

2) J.W.Bacus, E.E.Gose, : IEEE Trans., vol.SMC 2, ,pp.517,Sept. 1972.

3) K.Taniguchi and Y.Nakano, : IECE Trans. vol.58,pp.651-652, Oct. 1975.