Fast square-area detection algorithm using automata for VLSI implementation

Fast Square-area Detection Algorithm using Automata for VLSI Implementation

（オートマトンを用いた高速矩形領域検出アルゴリズムとその回路構成）

Kazuhiro Maeda†, Junichi Akita (member)†

Abstract

Conventional image sensors, including CCD sensors and smart sensors are considered for just acquiring the image as the matrix of dots, not for recognizing the meaning of image.

In this paper, we propose a novel real-time algorithm to detect the square area in an object using the structure of node automata, and discuss its implementation as a CMOS image sensor, where the pixels and the node automata are integrated in one chip. We also discuss the search algorithm of their position in the pixel plain using area dividing methodology.

Key words: Area detection, Automata, State transition, Tree Structure, Image Sensor

1. Introduction

Conventional image sensors, including CCD sensors and smart sensors are considered for just acquiring the image as the matrix of dots, not for recognizing the meaning of image.

The information of area and position of segments in focal plain is very important, even if they are not accurate, in order to restrict the target area where the further image processing should be executed as a small area. We will discuss the algorithm to detect the rough area and position of segments using pixel-parallel processing by node automata.

In the applications for robot vision, the processing time within about 1ms should be achieved for the fast and accurate servo control[1]. The fast image processing, such as segmentation, can be achieved by special massive parallel processing circuits, such as multi DSP, but most of them need the frame buffer memory where the whole image is stored, and the conventional approach will not make full use of the parallelity of the pixel information, since the basic processing for pixels is perfomed sequentially by each processing circuits.

Some studies on vision chips which perform segmentation processing are reported, but they aim to perform segmentation processing as an image without calculating basic information of the each area, such as its position or area[2], or to perform calculation of centroid of just whole image, not for each object[3].

In this paper, we propose a novel real-time algorithm to detect and determine the central point of each square area in an object in focal plain, which is perfomed in fully pixel parallel processing, using the structure of node automata. We also discuss its implementation as a CMOS image sensor, where the pixels and the node automata are integrated in one chip. We also discuss the search algorithm of their position in the pixel plain using area dividing methodology.

2. Area Detection Algorithm

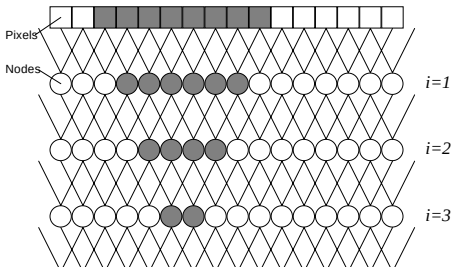
First, we consider the one-dimensional array of pixels and node automata, as shown in Fig.1. Here the squares at the highest level are pixels, whose colors represents whether they belong to the target object (gray, '1') or not (white, '0'). The circles at the lower levels represents the node automata, which have the value of logical AND of the two pixels or nodes at the previous level, as the following procedures. (Here d_j is the value of pixel at the j-th place, and $S_{i,j}$ is the value of node at the i-th level, and j-th place.)

（1） The nodes at the first level, $i = 1$, have the logical AND of value of two neighbor pixels; $S_{1,j} = d_j \cdot d_{j+1}$.

（2） The nodes at the second level, $i = 2$, have the logical AND of values of two neighbor nodes at

Received July 23, 1999; Revised December 9, 1999; Accepted January 5, 2000

† Department of Electric&Computer Engineering, Kanazawa University
(2–40–20 Kodatsuno, Kanazawa, Ishikawa 920-8667, Japan)



Fi .

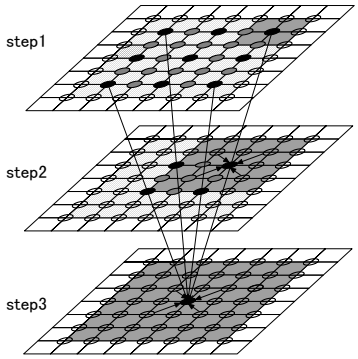


Fig. 2 Detection process in two dimensional case.

the first level; $S_{2,j} = S_{1,j-1} \cdot S_{1,j+1}$.

(3) The similar procedures are processed for the lower level, until all of the nodes become '0'; $S_{i+1,j} = S_{i,j-1} \cdot S_{i,j+1}$.

The node value of '1' at $i = 1$ represents the "block" of two pixels, and the node value of '1' at $i = 2$ represents the "block" of two nodes of $i = 1$, or the "block" of four pixels in other words. It is easily derived that the node value of '1' at i-th level represents the block of $2i$ pixels, and all nodes will be '0' at the k-th step if the size of the largest pixel block is $2k$. We can detect the position and size of the block of pixels, by detecting where and when all nodes have become '0.'

It is notable that the nodes determine their values just according to the values of neighbor nodes at the previous step, and they can be easily implemented as the finite state automata connected to only the neighbor nodes.

The above structure and the processes can be extended for the two-dimensional case, by placing matrix pixels with the node automata among the pixels, as shown in Fig.2. In this case, the number of steps needed for detecting the square area whose size is $m \times n (m \leqq n)$ is $n/2$ at maximum, in other words, the number of detection steps is proportional to $O(n)$, which is fast enough for practical image data.

Here, we describe the simulation results of this algorithm for real images. The original image shown in Fig.3(a) are digitized according to color information as shown in Fig.3(b).

It is notable that the digitizing according to each pixel's color information can be performed just refering one pixel's color information, not with the neighbours' colors. Assuming that the original color image is transferred serially, the digitizing procedure by color information can be performed just for the color signal of one pixel in the serial data stream, which does not need the frame memory where the whole image should be stored.

The square area detection discussed above is processed for the digitized image, and the processes are shown in Fig.4(a)-(f).

All the pixels are going to eliminated at the 7th step, just after the Fig.4(f), and the all of detection procedures are finished at this step.

As described above, this algorithm can perform both detecting square area in focal plain and determing central point of this square area by pixel parallel processing, while no other vision chip aim to perform, in much faster time than the conventional frame memory-based system.

3. Position Search Algorithm

In the process discussed above, the 'eliminating' pixel represents the position where the square area exists, where the word of 'eliminating' represents the node automaton making transition from '1' to '0', whose neighbor pixels also make the same transition, as shown in Fig.6(a), while it does not represents the square area if the neighbor pixels does not make transition to '0', shown in Fig.6(b). Thus the flag of 'eliminating' pixel, $f_{i,j}$ is expressed as follows,

$$f_{i,j} = S_{i,j}^{n-1} \cdot \overline{S_{i,j}^{n}} \cdot (\overline{S_{i-1,j-1}^{n}} \cdot \cdots \cdot \overline{S_{i+1,j+1}^{n}}) \quad (1)$$

where $S_{i,j}^{n}$ is the state of node automaton at (i, j) in n-th transition step.

It is unreasonable to scan all pixels to detect the eliminating pixel, since the number of scanning step of all pixel is very large. Fig.7 shows the reasonable process to detect the eliminating pixel.

Here we assume that the logical-OR ('1' for eliminating pixel) of the selected subarea can be directly read out. At the first step(a), the whole area is divided into four subareas, and the logical-OR of upper-left subarea is '1', that implies that there is a eliminating pixel in this subarea. At the following step(b), the upper-left

(a) (b)

Fig. 3 Original sample image(a) and digitized image according to color information(b).

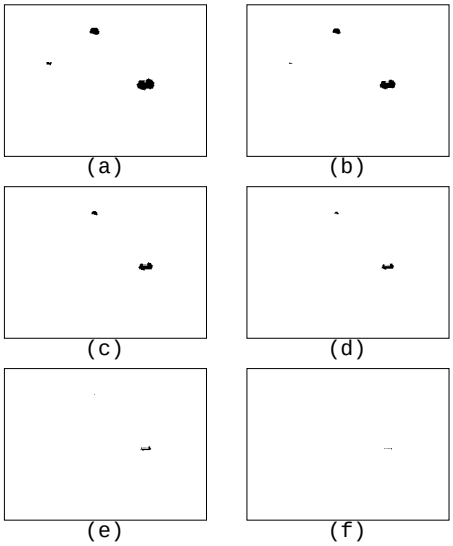


Fig. 4 Detection steps.

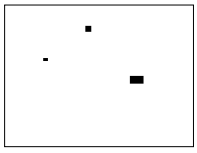
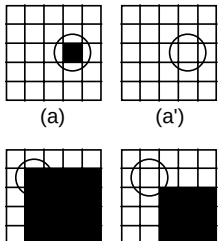


Fig. 5 Detected results of each square area.



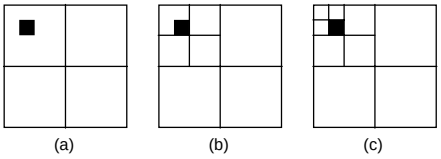
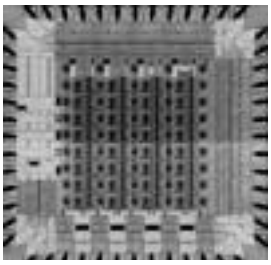


Fig. 7 Area division process to detect black ('eliminating') pixel.



Fig

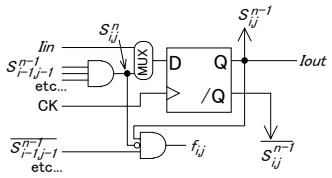


Fig. 9 Circuit of the node automaton.

subarea is divided into four smaller subareas, and then the upper-left smaller subarea should be divided into four subareas at the step(c).

The above scan procedure of image plain generates a kind of encoded image, and it is reported that it is more effective especially in case of less number of eliminating pixels than to scan all pixels[4].

The all of the detection and search procedures are summerized as follows.

(1) Read in the digitized image.

(2) Make transition of all node automata.

(3) If there are any eliminating node automata, scan the position of them by executing area division steps.

(4) Repeat the 2. and 3. steps until all node automata transits to '0'.

4. Implementation of Fast Area Detection Algorithm

4.1 Design of state transition evaluation circuit

We have designed the evaluation circuit of processing just the detection procedures. The original image is fed into the circuit serially, and the state transitions of node automata are occurred until all the nodes goes to '0'. The state of each node is read out as the logical-OR of nodes in both the vertical and the horizontal line.

Fig.8 shows the designed circuit using CMOS 1.2μm, 2 layers metal process*. The chip size is 2.3mm×2.3mm, and it contains 5×6 pixels and 4×5 node automata. The measured minimum operation clock cycle is 7.8ns.

* The VLSI chip in this study has been fabricated in the chip fabrication program of VLSI Design and Education Center(VDEC), the University of Tokyo with the collaboration by Nippon Motorola LTD., Dai Nippon Printing Corporation, and KYOCERA Corporation.

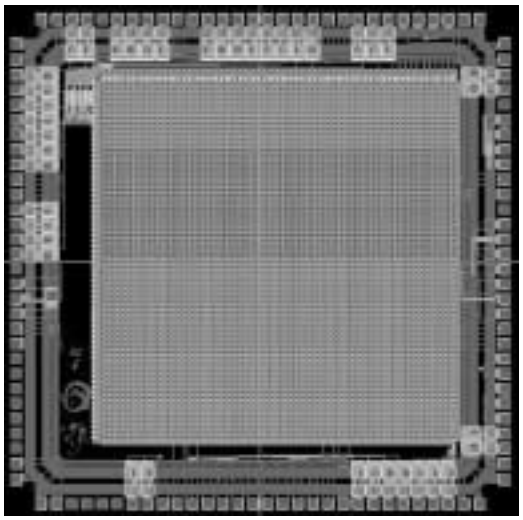


Fig. 10 Chip photograph of the designed second evaluation circuit.

4. 2 Design of detection and search evaluation circuit

We have designed the evaluation circuit of processing these procedures. The circuit consists of two parts; the node automaton including detection circuit of 'eliminating' pixel, the search controller of the positions of 'eliminating' pixels.

Fig.9 shows the circuit of the node automaton. The current state of each node, $S_{i,j}^{n-1}$ and the inverse of it, $\overline{S_{i,j}^{n-1}}$ are distributed to the neighbor pixels. The state of the next step is generated as the logical-AND of the current states of neighbor pixels, and the flag indicating 'eliminating' pixel, $f_{i,j}$ is also generated as Eq.(1).

This node is considered as the pixel at first step, and it makes its transition as the node automaton at the later step. These two mode is selected by the multiplexor placed at the input of flip-flop.

Fig.10 shows the designed layout of this circuit using CMOS 0.6μm, 3 layers metal process*. The chip size is 4.5mm$\times$4.5mm, and the number of the nodes is 64×57.

The image is fed into the nodes serially in this evaluation circuit, that emulates the original image generation, and it will be integrated with photo detectors as smart sensor.

The operation clock frequency estimated by the `spice` simulation is about 5MHz, and the operation time for detection and search of 10 objects whose size is 10$\times$10 pixels, is about 40μs.

This chip is now under fabrication, and it will be evaluated afterward.

5. Conclusions

We proposed the fast square-area detection algorithm using node automata placed among pixels, which can performed in $O(n)$ for the object size of n. We also discussed its implementation on VLSI sensor integrated with photo detectors, and have designed the evaluation circuit including node automata for state transition and search controller for detecting positions of objects.

Acknowledgements

This study is aided by Foundation For C&C Promotion.

〔References〕

1) I.Ishii et al.: "Target tracking algorithm for 1ms visual feedback system using massive parallel processing", *Proc. IEEE Int.Conf.Robotics and Automation*, pp.2309-2314 (1996)
2) C.L.Keast and C.G.Sodini: "A CCD/CMOS-based imager with integrated focul plane signal processing", *IEEE Journal of Solid-State Circuits*, Vol.28, No.4, pp.431-437 (1993)
3) S.P.DeWeerth: "Analog VLSI circuits for stimulus localization and centroid", *International Journal of Computer Vision*, Vol.8, pp.191-202 (1992)
4) J.Akita and K.Asada: "An Image Scanning Method with Selective Activation of Tree Structure", *IEICE Trans. on Electronics*, Vol.E80-C, No.7 (1997)

Kazuhiro Maeda He was born in Aichi, Japan, on Nov. 21, 1973. He received B.S. and M.S. in Department of Electrical and Computer Engineering Faculty of Engineering, Kanazawa University, Ishikawa, Japan in 1997 and 1999. He has been working as a designer in Digital Media Systems R&D Division, Hitachi, Ltd. His interest is in human seeing system and object recongition for smart image sensor.

Junichi Akita He was born in Aichi, Japan, on Aug. 22, 1970. He received B.S. and M.S., and Ph.D. in in electronics engineeringfrom the University of Tokyo, Tokyo, Japan in 1993 and 1995, and 1998, respectively. He joind the Faculty of Engineering, Kanazawa University as a research associate in 1998. His interest is in the implemenation of LSI systems for real-time processing of real-world information, and implementation and modeling of low-power and intelligent integrated circuits.
He is a member of Institute of Electronics, Information and Communication Engineers of Japan (IEICEJ), Information Processing Society of Japan (IPJ), Japanese Society for Artificial Inteligence (JSAI), and Robotics Society of Japan (RSJ).

* The VLSI chip in this study has been fabricated in the chip fabrication program of VLSI Design and Education Center(VDEC), the University of Tokyo with the collaboration by Nippon Motorola LTD., Dai Nippon Printing Corporation, and KYOCERA Corporation.