A New Method for Fast Skeletonization of Binary Images on Cellular Processor Arrays

Bin Wang, Przemyslaw Mroszczyk and Piotr Dudek
School of Electrical & Electronic Engineering
The University of Manchester
Manchester, M13 9PL
p.dudek@manchester.ac.uk

***Abstract*—The proposed skeletonization algorithm, based on wave propagation principle, modifies the iterative thinning approach by replacing template matching with simple morphological operations, significantly decreasing the computational requirements. Conceptually, this method can be seen as binary wave-propagations in a 2-layer cellular network, where a trigger-wave propagating in one layer computes distances, and another layer is used to detect wave-collisions. Implementations of the algorithm on a fine-grain SIMD cellular processor array (SCAMP-5 vision chip), and on a dedicated VLSI asynchronous cellular processor array are presented. The experimental results conclude the paper.**

***Index Terms*—Skeletonization, vision chips, SIMD processor arrays, asynchronous VLSI, CMOS**

I. Introduction

THE skeleton of an object provides a compact representation of its shape, size, position and orientation. It has been widely used in many computer vision algorithms involving shape recognition, gesture recognition, and human action recognition [1, 2, 3]. Skeletonization algorithms have been intensively researched during the past decades. Some recently proposed skeletonization algorithms require sophisticated hardware, such as GPGPU, to achieve adequate processing speeds (e.g. 24 FPS). This drawback restricts the usage of skeletonization algorithms in situations where the processing platform is not very powerful, e.g. in an embedded system or a smart camera. Our goal, in this work, is to provide a simple, fast and reliable skeletonization algorithm, which is suitable for implementation on a variety of hardware platforms. In this work, we implemented the proposed algorithm on a cellular processor array, i.e. SCAMP-5, and also designed a dedicated fully pixel-parallel, asynchronous chip.

II. The Proposed Algorithm

The proposed skeletonization algorithm is based on a two-layer trigger-wave propagation. Rather than inhibiting the wave propagation before the collision, which is typically done in other algorithms [4], the proposed algorithm allows waves to meet together in one layer of propagation, while detecting skeleton pixels in another layer, coupled with the first one.

Assuming there are two binary pixel arrays **B** and **R**, the input binary image is initially stored in both **B** and **R** where 1s (active pixels) denote the background pixels while 0s (inactive pixels) denote the foreground objects. In each iteration, the propagation in **B** starts first, then the propagation in **R** is processed based on the propagation result in **B**. Propagation in **B** is the ordinary binary trigger-wave propagation. Each inactive pixel (foreground pixel) will be changed to active pixel (background pixel) if one of its direct neighbours (in a 4-connected array) is active. This process can be represented by the following function:

$$B^{n+1}_{i,j} = B^{n}_{i+1,j} \vee B^{n}_{i-1,j} \vee B^{n}_{i,j+1} \vee B^{n}_{i,j-1} \vee B^{n}_{i,j}; \qquad (1)$$

Propagation in **B** generates the wave fronts from the boundaries of the objects according to the propagation rules. The wave collisions will determine the skeleton points. The propagation in **R** provides "delayed" wave fronts, so that the collision points can be marked and saved before collisions. Specifically, each inactive pixel in **R** ($R_{i,j} = 0$) is changed to active ($R_{i,j} = 1$) if the following two conditions are satisfied: Firstly, the pixel (i, j) needs to be inactive in **R** and active in **B** (skeleton pixels are always at the wave front). Secondly, the four direct neighbours of pixel (i, j) are not all active in **B**. The transition rules for **R** can be represented as the following function:

$$R^{n+1}_{i,j} = R^{n}_{i,j} \vee B^{n+1}_{i,j} \wedge \overline{B^{n+1}_{i-1,j} \wedge B^{n+1}_{i+1,j} \wedge B^{n+1}_{i,j-1} \wedge B^{n+1}_{i,j+1}} \qquad (2)$$

When equations (1) and (2) are applied, non-skeleton pixels are removed from **R** in each iteration. Intuitively, the propagation in **B** implements the wavefronts propagating from the boundaries into the foreground objects. Pixels that are activated in **B** can be then activated in **R**, provided there are no wavefront collisions in **B**. A detailed example is shown in Fig.1. During propagation in **B**, pixels P_1, P_2 and P_3 have been activated according to (1). Pixels activated in **B** may become activated in **R**. Because pixel P_1 does not have any inactive neighbour in **B** while pixels P_2 and P_3 have one inactive neighbour in **B** (pixel P_4), pixel P_1 is left inactive. Appling equations (1) and (2) iteratively, a skeleton can be eventually obtained as shown in Fig.2.

To improve the quality of obtained skeletons, more complex

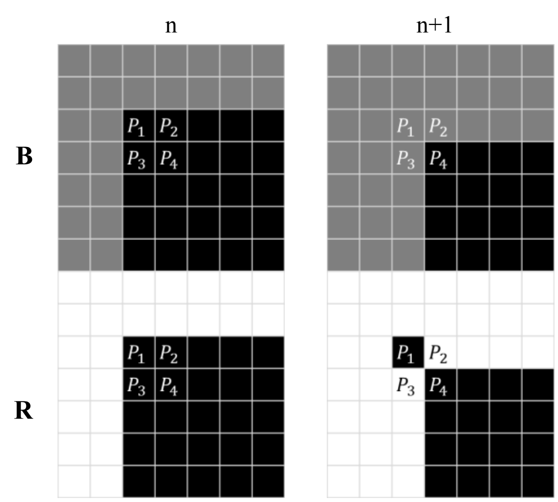

Fig. 1 Skeletonization process of using equation (2) and (3). The propagation in **B** activates pixel P_1, P_2 and P_3. In **R**, pixel P_1 is classified as skeleton pixels because it does not have any inactive direct neighbour in **B,** while P_2 and P_3 are been activated.

978-1-4799-6007-1/14/$31.00 ©2014 IEEE

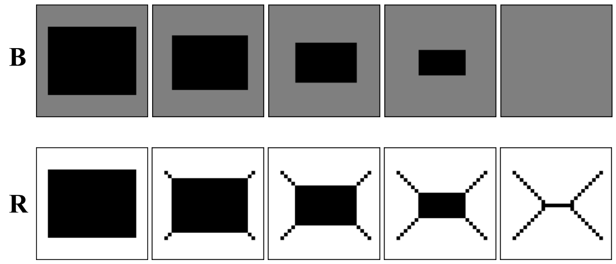

Fig.2 Propagation procedure using equation (1) and (2) (A skeleton is obtained at the end of the procedure in **R**) (black pixels in each image indicate inactive pixels)

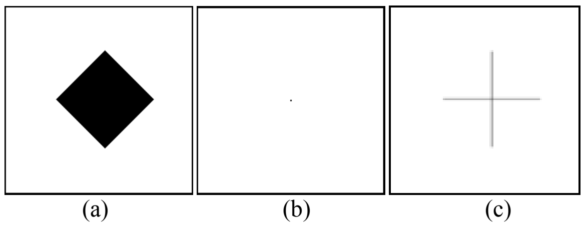

Fig.3 Skeletonization results generated by the proposed algorithm. (a) is the original image; (b) is the skeletonization result generated using equation (2), it reduces the skeleton to a single pixel; (c) is a good skeleton obtained using equation (3).

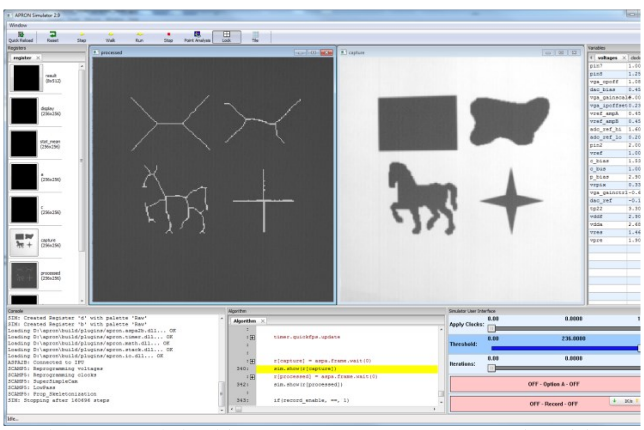

Fig.4 Proposed algorithm running on SCAMP-5 (screenshot of the graphical user interface). The image on the right side is the input image, while the image on the lift side contains generated skeletons.

condition can be implemented for propagation in **R**. For example, instead of (2), using the following propagation rule can guarantee to generate correct skeletons in some extreme cases that fail when using (2), as shown in Fig.3.

$$R_{i,j}^{n+1} = R_{i,j}^{n} \vee B_{i,j}^{n+1} \wedge \overline{B_{i-1,j}^{n+1} \wedge B_{i+1,j}^{n+1} \wedge B_{i,j-1}^{n+1} \wedge B_{i,j+1}^{n+1} \wedge B_{i-1,j-1}^{n+1} \wedge B_{i+1,j-1}^{n+1} \wedge B_{i-1,j+1}^{n+1} \wedge B_{i-1,j-1}^{n+1}} \quad (3)$$

III. Hardware Implementations

A. SCAMP-5

SCAMP-5 [5], with a resolution of 256x256, is a general purpose vision chip which combines image sensor and array processor on a single die. Because of the parallel nature of the proposed algorithm, implementing it on SCAMP-5 is straightforward. The SCAMP-5 interface and some skeletonization results are shown in Fig.4. The proposed algorithm can achieve 13889 fps (frame per second) on SCAMP-5 when processing the image shown in Fig.4, while a traditional thinning based algorithm (8 templates) can only deliver 2105 fps on the same hardware.

Fig.5. Skeletons obtained using asynchronous processor array.

B. Asynchronous Cellular Logic Array

The propagation-based skeletonization method discussed in Section II was implemented on a dedicated asynchronous processor array executing functions (1) and (2) [6]. The propagation mechanism from (1) is using an array of five-input logic OR gates, with four inputs corresponding to the logic states of pixels in **B** and one used to trigger the wave propagation, assuming that initial state of **B** is zero. The collision detection circuit, operating according to (2), in each processor, consists of a four-input logic AND gate, monitoring the state of the neighboring cells and signaling the collision when all of them are in a high logic state. This, however, is only a necessary condition, since all the AND gates in the array will eventually turn to a high state seemingly suggesting collisions. To assure the correct operation of the circuit, the output of the AND gate has to be saved in a latch, enabled only for a short time, corresponding to the propagation time of one cell, and generated using a simple tunable delay gate. The obtained experimental results from the test chip are shown in Fig.5.

In order to reduce power consumption and area occupation of the circuit, the proposed asynchronous processing cell was designed in a dynamic logic fashion using only 24 MOS transistors. The circuit is capable of performing up to 2 million skeletonization cycles per second while consuming less than 1.7 mW of power. The total size of the array is 840 μm × 1200 μm including the I/O scan registers for image data transfers.

IV. Conclusion

A new skeletonization algorithm based on a two-layer trigger-wave propagation has been introduced in this paper. The proposed algorithm is fast and easy to implement on various processing platforms. Implementations of the proposed algorithm on a general purpose fine-grain cellular processor array and a dedicated VLSI asynchronous cellular processor array have been presented.

References

[1] J X. Bai, "Skeleton-Based Shape Classification Using Path Similarity", International Journal of Pattern Recognition and Artificial Intelligence, Volume 22, Issue 4, 2008;

[2] S. Celebi, A. S. Aydin, T. T. Temiz and T. Arici, "Gesture Recognition Using Skeleton Data with Weighted Dynamic Time Warping", 8th International Conference on Computer Vision Theory and Applications, 2013;

[3] S. M. Yoon and A. Kuijper, "Human Action Recognition Based on Skeleton Splitting", Expert System with Applications, Volume 40, Issue 17, December 2013, Pages 6848-6855;

[4] A.Lopich and P.Dudek, Hardware Implementation of Skeletonization Algorithm for Parallel Asynchronous Image Processing, Journal of Signal Processing Systems, Springer, Volume 56, Number 1, pp. 91-103, July 2009;

[5] S.J.Carey, A.Lopich, D.R.W.Barr, B.Wang and P.Dudek, "A 100,000 fps Vision Sensor with Embedded 535 GOPS/W 256x256 SIMD Processor Array", VLSI Circuits Symposium 2013, Kyoto, June 2013;

[6] P.Mroszczyk and P.Dudek, "Trigger-Wave Collision Detection Asynchronous Cellular Logic Array for Fast Image Skeletonization", IEEE International Symposium on Circuits and Systems, ISCAS 2012, pp. 2563-2566, Seoul, May 2012;