2016 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2016, Yugawara, Japan, November 27th-30th, 2016

A Hardware Cellular-Automaton Architecture for Spatial Pattern Generation towards Motion-Vector Estimation of Textureless Objects

Aoi Tanibata†, Miho Ushida†, Alexandre Schmid‡, Masayuki Ikebe†, Tetsuya Asai†, Masato Motomura†

†Graduate School of Information Science and Technology, Hokkaido University
Kita 14, Nishi 9, Kita-ku, Sapporo 060-0814, Japan
‡Microelectronic Systems Laboratory, École Polytechnique Fédérale de Lausanne
CH-1015, Lausanne, Switzerland
Email: tanibata@lalsie.ist.hokudai.ac.jp

Abstract

In [1, 2], we proposed a hardware-oriented cellular-automaton algorithm that generates the spatial patterns of textureless objects and backgrounds to estimate the motion-vectors of textureless moving objects. In this report, we propose the fundamental architecture for the algorithm. The system consists of one-dimensional shift-register arrays and arithmetic operators for diffusion, a lookup table acting as a nonlinear "reaction" function, and several state controllers (counters, multiplexers, etc.), which together act as a one-dimensional reaction-diffusion streaming processor (RDSP). Two-dimensional image processing is performed for texture generation by arranging the one-dimensional RDSP on a two-dimensional mesh in a time-division manner, which reduces the complexity of the circuit system as a whole.

1. Introduction

Motion estimation is used in various applications such as hand gesture user interfaces[3] and automatic anomaly detection in monitoring camera pictures[4]. This technique has been actively researched in recent years. The block-matching method is one of the most common methods for motion estimation. However, when this method is applied to textureless moving objects, it can detect the motion vector of the outline only (Fig.1 (a)). In this case, we cannot perceive it as a textureless object or a frame. To estimate the motion more precisely, we have to detect the motion vectors in the outline of textureless moving objects. Therefore, we propose a hardware-oriented cellular-automaton algorithm that generates the spatial patterns of textureless objects and backgrounds in order to estimate the motion vector of textureless moving objects[1, 2]. The textureless moving objects are regarded as objects with the same patterns as the generated spatial patterns. Accordingly, we can detect the motion vectors in the outline of the textureless moving objects (Fig.1 (b)). In this study, we propose a fundamental module and architecture for this algorithm.

Figure 1: Motion estimation for a textureless object: (a) image of usual motion estimation, (b) image of motion estimation with spatial patterns generated

2. Algorithm

We proposed an algorithm that generates the spatial patterns using the reaction-diffusion (RD) model[1, 2]. The RD model is a well-known method for spatial pattern generation[5]. One-dimensional RD is obtained by iterative updating. This updating consists of three processes: diffusion, subtraction, and amplification. The diffusion process involves iterative blurring. Blur is described as

$$a_i(t+1) = \frac{a_{i-1}(t) + 2a_i(t) + a_{i-1}(t)}{4} \tag{1}$$

where i is the ith pixel in a row of an image, and t is $t$times blurring. The subtraction process involves finding the difference between the before diffusion and after it. The amplification process amplifies the value using the sigmoid function. In addition, we have to reduce the adverse influence of noise. Therefore, we add a filter that updates after every iteration of updating. We perform only diffusion in filter updating because smoother spatial patterns can be generated through this process.

In two-dimensional RD, the two-dimensional input image is first divided into a one-dimensional arrangement, x and y. These arrangements are then repeatedly processed by the one-dimensional RD model. Finally, they are multiplied together [1, 2].

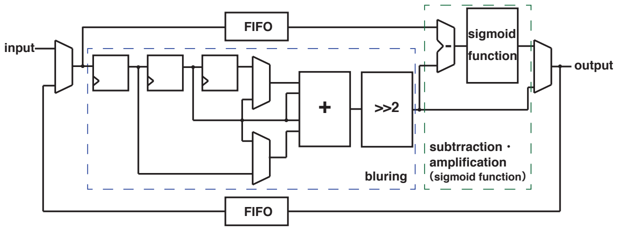

Figure 2: Module for One-dimensional Reaction-Diffusion

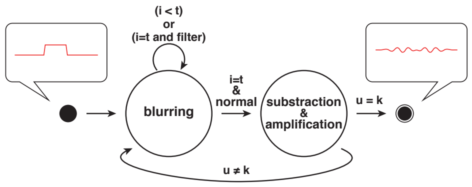

Figure 3: State Transition Diagram

3. Circuits

3.1. Module for One-dimensional Reaction-Diffusion

The proposed module for one-dimensional RD is shown in Fig.2. In addition, the state transition diagram of the module is shown in Fig.3. The one-dimensional RD algorithm is considered as switching and repetition of two operations: blurring, and subtraction and amplification. Therefore, we prepare the module as a state machine in this study as shown in Fig.3.

In Fig.2, i is the number of blurrings and u is the number of updates. In this case, diffusion is obtained by blurring t times. A one-dimensional RD is obtained by updating k times. *Normal* refers to the normal updating consisting of three processes including diffusion, subtraction, and amplification. *Filter* refers to filter updating.

We also devised a state controller consisting of counters. These counters count the number of blurrings, updates, and pixels in one row of an image. The multiplexers and control signals for the first-in first-out memories (FIFO) consisting of shift registers are controlled by the signals from the controller, and the module can be switched between states.

The module goes into the blurring state during diffusion and filtering. It transmits a pixel value as the output from the imager to the shift-register. It obtains the necessary pixel values from the shift-register. The module loads from the last burring result instead of the imager from the second iteration onward. It saves the pixel values loaded as the values obtained before diffusion in the FIFO if blurring occurs immediately after updating.

The module becomes goes into the subtraction and amplification state during subtraction and amplification. It subtracts the values after diffusion from the values before diffusion and amplifies the result using the sigmoid function. The sigmoid function is made up of the LUT (Lookup table).

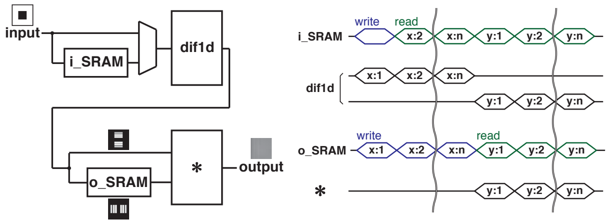

Figure 4: Architecture of Two-dimensional RD and its timing chart

3.2. Architecture for Two-dimensional reaction-diffusion

The proposed architecture for two-dimensional RD is shown in Fig.4. Dif1d is the module used for one-dimensional RD.

We also prepared a state controller consisting of some counters similar to the one used in one-dimensional RD. The controller controls the signals for static random access memories (SRAM) and the direction of readout pixel values, in addition to the signal for the one-dimensional RD module. We have to maintain the initial value in two-dimensional RD. The initial values are divided into a one-dimensional arrangement, x and y in rotation. Therefore, the architecture begins by saving the pixel values loaded from the imager in the SRAM for input. At the same time, it processes x though the module for one-dimensional RD and saves the result in SRAM for output. After this, the architecture obtains the initial values in y from the SRAM for input and processes in y. Finally, it multiplies the results of one-dimensional RD of x and y and provides it as the results of two-dimensional RD.

Two-dimensional image processing for texture generation is performed by arranging the module for one-dimensional RD on a two-dimensional mesh in a time-division manner, which reduces the complexity of the circuit system as a whole.

4. Results

4.1. One-dimensional reaction-diffusion

The bit width of the pixel values affects the precision of the result of one-dimensional RD in terms of module one-dimensional RD. We process data with widths of 12 bit and 8 bit through one-dimensional RD and determine the Fast fourier transform (FFT) of the result.

In this case, diffusion is achieved by blurring 25 times. A one-dimensional RD is obtained by updating 10 times. We add a filter updating after every 4 iterations of updating. The gain of the sigmoid function is 5.

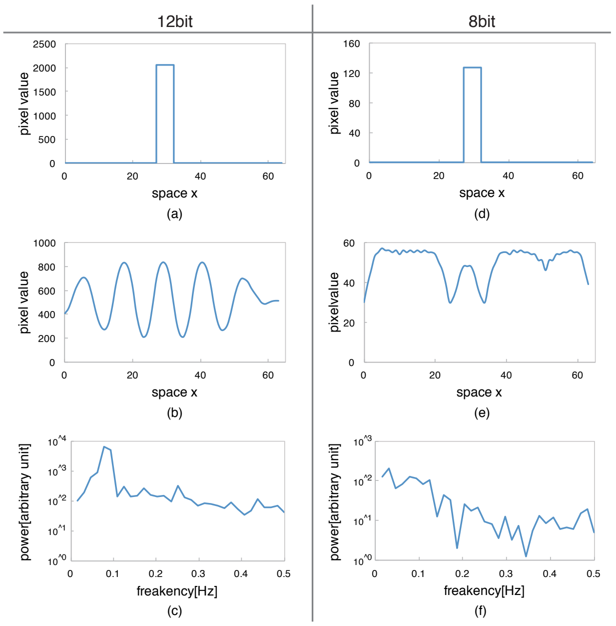

Figure 5: Comparison results of 12-bit and 8-bit in one-dimensional reaction-diffusion 1: (a)(d) initial values, (b)(e) result of one-dimensional reaction-diffusion, (c)(f) result of FFT

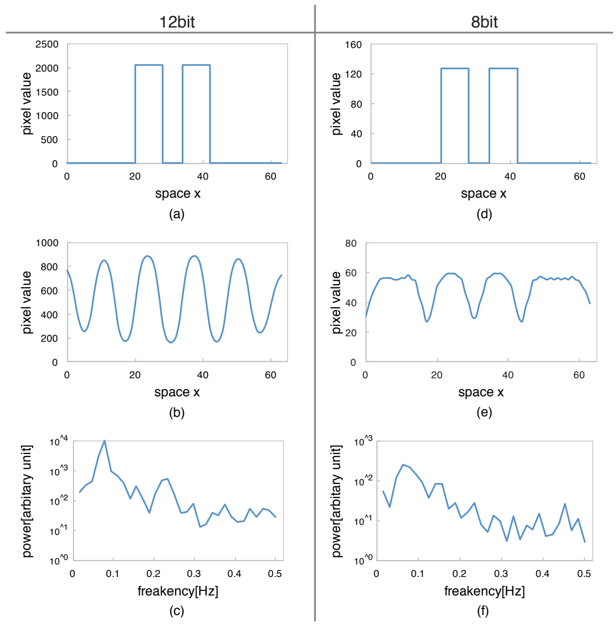

Figure 6: Comparison results of 12-bit and 8-bit in one-dimensional reaction-diffusion 2: (a)(d) initial values, (b)(e) result of one-dimensional reaction-diffusion, (c)(f) result of FFT

The simulation results are shown in Fig.5. The waveform of the 12-bit data is smoother than that of the 8-bit data. The waveform of 8-bit data is sufficiently smooth at a glance. However, we have to consider the result of the FFT. In the result of the 12-bit data, the necessary frequency component has more digits than the noise component. On the other hand, the necessary frequency component in the result of the 8-bit data has approximately the same number of digits as that in the noise component. In this case, we can reduce the noise easily in 12-bit data, but we cannot do so in the 8-bit data. In Fig. 6, we repeat the simulation by changing the initial values and obtain the same result. Therefore, we need a width of at least 12-bit for the pixel values.

4.2. Two-dimensional reaction-diffusion

The generated spatial pattern of the two-dimensional input image is shown in Fig.7. In this case, we used a picture of 120*120 pixels. We assume that we detect the motion vectors of textureless moving objects and prepare a picture of textureless objects against a real background.

The spatial patterns are generated for textureless objects and backgrounds. The same spatial patterns are generated between two frames for textureless objects, regardless of the changing position of textureless objects. Therefore, it appears that the detected motion vectors are in the outline of the textureless moving objects. In Fig. 8, we repeat the simulation by changing the backgrounds and obtain the same result.

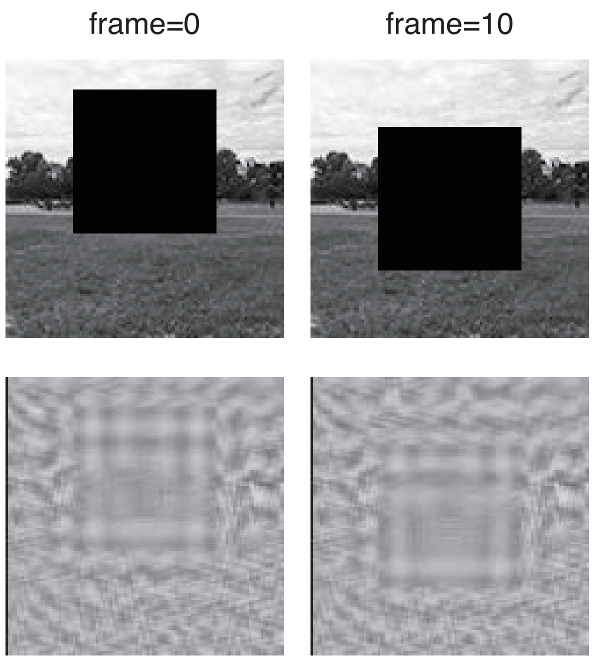

Figure 7: Result of two-dimensional reaction-diffusion with a real background 1

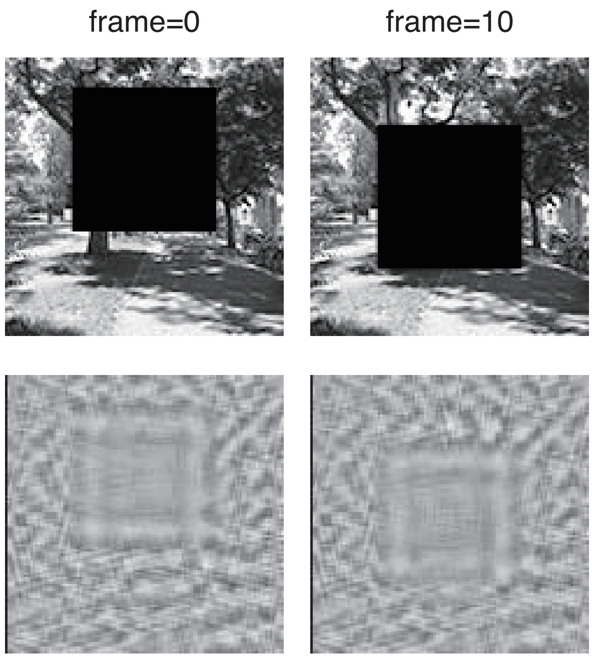

Figure 8: Result of two-dimensional reaction-diffusion with a real background 2

5. Summary

In this study, a module is used as the state machine in one-dimensional RD and arranged on a two-dimensional mesh in a time-division manner in two-dimensional RD. Therefore, we can reduce the complexity of the circuit system as a whole. In addition, we determine the appropriate bit width of pixel values for one-dimensional RD. In future, we seek to develop a hardware implementation that enables real-time processing.

Acknowledgments

This study was supported by the JSPS Grants-in-Aid for JSPS Fellows, and a Grant-in-Aid for Scientific Research on Innovative Areas [2511001503] from the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan.

References

[1] Ushida M., Schmid A., Asai T., Ishimura K., and Motomura M., "Motion vector estimation of textureless objects exploiting reaction-diffusion cellular automata," International Journal of Unconventional Computing, vol.12, no. 2-3, pp.169-187 (2016).

[2] Ushida M., Ishimura K., Schmid A., Asai T., and Motomura M., "Motion vector estimation of textureless objects exploiting reaction-diffusion cellular automata," 2015 International Symposium on Nonlinear Theory and its Applications, pp. 85–88, Hong Kong, City University of Hong Kong, China (Dec. 1–4, 2015).

[3] M. Tanaka, Japan patent Kokai JP2014-52934A, 2014.

[4] M. Kuwahara, K. Kawashima, Japan patent Kokai JP2014-7498A, 2014.

[5] Adamatzky A., de Lacy Costello B., and Asai T. (2009) Reaction diffusion computers, Springer Verlag, London.