REGULAR PAPER

High speed vision processor with reconfigurable processing element array based on full-custom distributed memory

Japanese Journal of Applied Physics **55**, 04EF08 (2016)
http://doi.org/10.7567/JJAP.55.04EF08

REGULAR PAPER

High speed vision processor with reconfigurable processing element array based on full-custom distributed memory

Zhe Chen, Jie Yang, Cong Shi, Qi Qin, Liyuan Liu, and Nanjian Wu*

State Key Laboratory for Superlattices and Microstructures, Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, P. R. China

*E-mail: nanjian@red.semi.ac.cn

Received September 28, 2015; revised December 1, 2015; accepted December 28, 2015; published online March 16, 2016

In this paper, a hybrid vision processor based on a compact full-custom distributed memory for near-sensor high-speed image processing is proposed. The proposed processor consists of a reconfigurable processing element (PE) array, a row processor (RP) array, and a dual-core microprocessor. The PE array includes two-dimensional processing elements with a compact full-custom distributed memory. It supports real-time reconfiguration between the PE array and the self-organized map (SOM) neural network. The vision processor is fabricated using a 0.18 µm CMOS technology. The circuit area of the distributed memory is reduced markedly into 1/3 of that of the conventional memory so that the circuit area of the vision processor is reduced by 44.2%. Experimental results demonstrate that the proposed design achieves correct functions.

1. Introduction

The inspired thinking of integrating an intelligent visual system on a single silicon device opened the research field of the vision chip decades ago.[1,2] Since then, a number of vision chip prototypes have been proposed. Original vision chips integrate a massively parallel processing architecture and an image sensor to achieve high speed early visual functions as the vertebrate retina does.[3–15] Such applications include object position extraction,[3–7] image filtering,[8–12] contour generation,[13] dynamic range extension,[14] and event detection.[15] More advanced vision chip designs focus on more specific visual tasks such as fingerprint recognition,[16] eye tracking[17] and horizon detection.[18] As the required image processing functions become more complex, it becomes such an inevitable problem that the processing circuits occupy a large circuit area, which results in a small fill factor and a limited sensor resolution. Some designs try to solve this problem by replacing the massively parallel processing elements in each pixel with row-parallel processors, which turns out to achieve similar performance for various visual applications, such as the high-speed tracking and image filtering.[7,19–21] Some vision chip designs separate the processing architecture from the image sensor to achieve mid-level or even high-level image processing functions, such as human detection, tracking and posture classification,[22] and face feature extraction.[23] Recently proposed vision chips or application-specific image processors are dedicated for high-level image processing algorithms, especially for image recognition, such as histogram-of-oriented-gradients (HOG) feature extraction,[24] object recognition based on scale invariant feature transform (SIFT),[25] vocabulary forest,[26] convolutional neural networks,[27] and self-organizing map (SOM) neural network.[28] These state-of-the-art vision chips achieve high performance in terms of accuracy, versatility and computing capacity. Since there is an emerging trend to design a three-dimensional (3D) stacked smart image sensor,[29,30] we expect the performance of the vision chips to increase further. This paper focuses on low-cost and high-performance image processing circuit designs for the vision chips.

For high-speed digital vision chips, a two-dimensional processing element (PE) array has been frequently used. It can be programmed in a single-instruction multiple-data (SIMD) manner to accomplish low-level image processing at a 1000-fps high speed.[4] Recently, we have proposed an architecture supporting the reconfiguration of the PE array into the SOM neural network to speed up high-level image processing.[28] Although the proposed architecture achieves rather high performance in multi-level image processing, it remains as a critical issue that the distributed memory in the PE array occupies a large circuit area; this becomes the bottleneck limiting the resolution of the PE array to increase further. It seems obvious that the static random access memory (SRAM) is not area-efficient in implementing the distributed memory with small capacity because the SRAM requires additional sensitive amplifiers. To realize a compact local memory, the three-transistor (3-T) DRAM used as the basic memory cell has been developed;[12] however, the drawback is that the signal state can only be retained for 91.3 ms under 3.3 V operating voltage, and it requires two clock cycles to accomplish a single write or read operation. The latch-based 7-T SRAM has also been reported to implement the local memory;[23] however, the latch does not support write and read operations at the same time, which occurs at a frequency of over 30%.

In this paper, we propose a vision processor based on a 64 bit compact full-custom distributed local memory. The memory architecture is based on both static and dynamic latch structures. The physical layout of the proposed memory is designed in a full-custom way to save the circuit area further. Compared with the conventional method used in our previous design, the proposed memory reduced the circuit area by more than 68.0%. The vision processor also achieves an obvious area reduction against the previous work.[28] Experimental results demonstrate that the proposed vision processor with the dedicated distributed local memory fully achieves high-speed image processing functions.

2. Vision processor architecture

2.1 Processor architecture

The vision processor architecture is shown in Fig. 1. It inherits the characteristics of the hybrid parallel processing architecture from our previous vision chip design.[23,28] The architecture consists of a two-dimensional four-connected $N \times N$ PE array, a row-parallel $N \times 1$ row processor (RP)

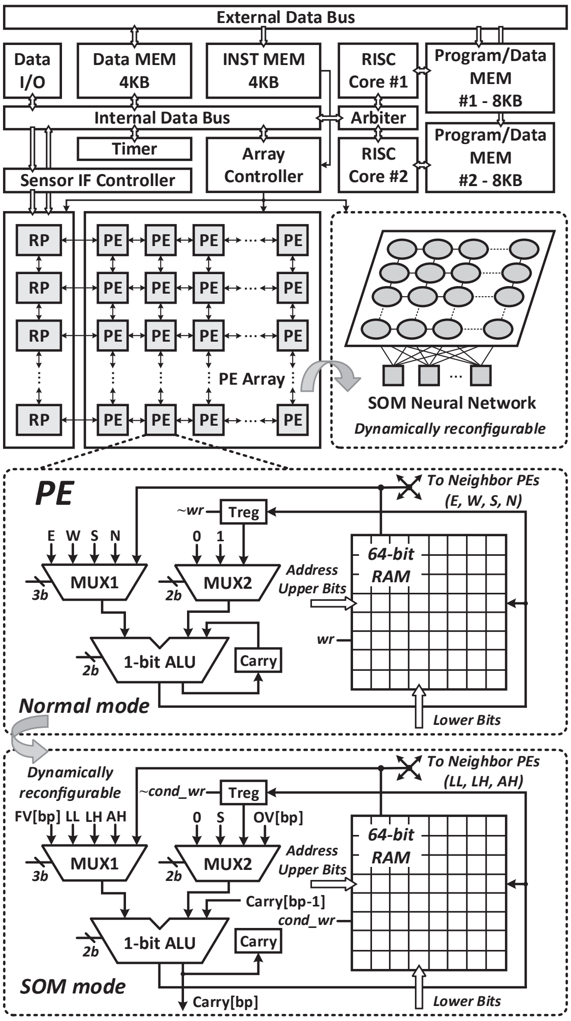


Fig. 1. Architecture of the vision processor with reconfigurable PE.

array, and a dual-core RISC processor. The PE array can be dynamically reconfigured into an SOM neural network. It helps increase the high-level image processing speed.[28] Each PE contains a 1 bit arithmetic logic unit (ALU) and a 64 bit local memory. The ALU can be programmed to perform 1 bit AND, OR, NOT, and ADD operations, and each PE can be programmed as reading (writing) data from (to) the local memory of itself or those of four neighbor PEs. The massively parallel PE array provides a large bandwidth between the local memory and the processing circuits; thus, most low-level image processing algorithms can be realized at high speeds. The RP array also operates in an SIMD fashion and helps speed up some mid-level processing tasks such as histogram statistics, global feature extraction, and transformation between the time domain and the frequency domain. It also supports high-speed low-level image processing tasks such as filtering and edge detection, especially when the processing tasks must be finished within the exposure interval between two rows of pixels. The dual-core RISC is only used to assign tasks among the hybrid parallel processors and manage the data flow along the internal data bus.

The dynamic reconfiguration of the SOM neural network is implemented by switching the neighbors of each PE from eastern (E), western (W), southern (S), and northern (N) PEs to the logical low (LL), logical high (LH), and arithmetical high (AH) PEs. The carry signal generated in each PE is passed onto the LH or AH PE. In this manner, the PE subarray is reconfigured into an SOM neuron. To implement the learning and classification algorithm, the feature vector (FV) and the overflow flag (OV) related to the bit position (bp) must be programmed. Since SOM neurons can be programmed to operate in an SIMD fashion, the parallelism of high-level image processing results in $(N/M) \times (N/M)$.

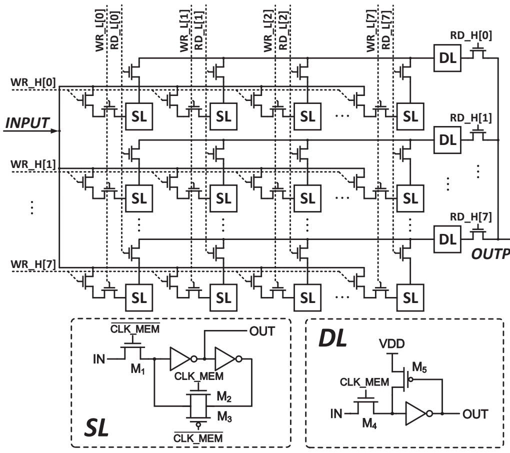


Fig. 2. Architecture of the distributed memory: 64 bit RAM.

2.2 Full custom 64 bit RAM

Figure 2 illustrates the architecture of the proposed local memory, namely, the 64 bit RAM. To reduce the routing overhead, the RAM is designed into a rectangular array to support addressing from both the horizontal and vertical directions.[31] The RAM consists of 8 rows of 8 bit memory stripes. Each stripe behaves in the same manner as the registers based on master and slave stages. The master stage contains eight 7-T static latches (SLs), while the slave stage contains one 4-T dynamic latch (DL). In each SL, the transmission gate made up of M2 and M3 provides a positive feedback loop. When the clock rising edge arrives, the M1 on the input data path is off, and the input signal state is well retained. At the same time, the M4 in the DL is on, and the selected SL passes its stored signal to the DL. Since the M4 is an NMOS transistor, it will cause a threshold voltage drop when passing the high state. The DC simulation shows that the threshold voltage of the NMOS transmission transistor under the standard 1.8 V supply voltage is 0.768 V. The M5 with weak driving force is used to recover the high state level to VDD and thus reduce the static power consumption. According to the result of transient simulation, the signal transmission time of the NMOS transmission transistor turns out to be around 30 ps. It will not become the bottleneck limiting the write and read speed. The DL serves as the read out buffer in the slave stage, and its signal state is automatically refreshed in every clock cycle; thus, the signal stored in the DL will also be well retained. The addressing signals are divided into higher-bit and lower-bit sections. The higher-bit section is used to select the memory stripe, while the lower-bit section is used to select the SL. All of the transistors except the M5 are designed with the minimum feature size, and the physical layout of the memory is designed in full custom. Figure 3 shows the layout of two memory stripes. The circuit area is $30.7 \times 7.7\,\mu m^2$ under the

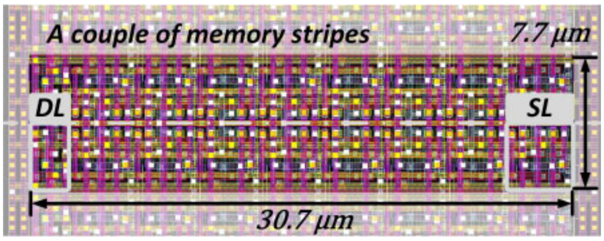


Fig. 3. (Color online) Layout of partial compact distributed memory: a couple of memory stripes.

0.18 μm process. Consider the conventional method used in our previous design, the distributed memory of each PE in a subarray is extracted and gathered into a 2 kB memory block, and the memory block is implemented as an SRAM generated by the memory compiler. In this manner, the average area consumption is as large as 41.1 μm^2/bit.[28] Compared with the developed method, the memory proposed in this work is designed in full custom, and it benefits from the reduced transistor usage, shrunk physical layout and saved routing overhead. As a result, it achieves a markedly reduced circuit area.

2.3 Reconfigurable PE subarray

Figure 4 illustrates the method of dynamic reconfiguration between the PE subarray and an SOM neuron. Each PE subarray contains 4 × 4 PEs, and the reconfigured SOM neuron is formed by connecting the 16 PEs in a snakelike serial style. An additional condition generator (CG) is designed to generate conditional signals for the subarray to implement the required SOM algorithms.[28] The 1 bit ALU and conditional generator are designed for synthesis flow, while the distributed memory and PE connections are implemented in full custom to save the circuit area. To reduce the carry signal transmission distance inside the PE subarray, two types of full-custom PE layouts have been designed. The first-type PE sets the input carry signal on the south and the output carry signal on the north, while the second-type PE reverses the carry signal transmission path. The CG provides conditional control signals for the SOM neuron, including the conditional write, OV, and state control signals. The CG also helps generate a balanced clock tree to synchronize all the processing circuits and the full-custom local memory. The layout of the CG with the clock tree buffer occupies only a 3.5% circuit area of the PE subarray. Global control signals including instructions, addresses and the data are provided by the RISC core in an SIMD fashion, and the selection flag $W_{i,j}$ is used to select and update the local SOM neuron.

2.4 Balanced clock tree

To synchronize the proposed distributed memory with massively parallel processing circuits, we designed a two-layer balanced clock tree. The first layer of the clock tree is formed by a clock inverter driving the SOM neural network so that all of the neurons are synchronized. The second layer of the clock tree is implemented in the CG in each SOM neuron, and it helps synchronize the PE processing circuits with the full-custom local memory. The proposed memory requires a pair of mutually inverted clocks enabled by both the write (wr) and control signals provided by the CG. The noninverting clock CLK_MEM should be synchronized with the processing circuit clock CLK_ALU. In accordance with the adopted 0.18 μm CMOS technology, the capacitive loads of the inverting, noninverting, and CLK_ALU clocks are estimated to be 4.1, 2.3, and 0.1 pF respectively. Considering the capacitive load, the clock driving circuit in the CG can be established by selecting logic gates with dedicated driving forces, as shown in Fig. 5. In this manner, the balanced clock tree achieves both the symmetry and the compact size.

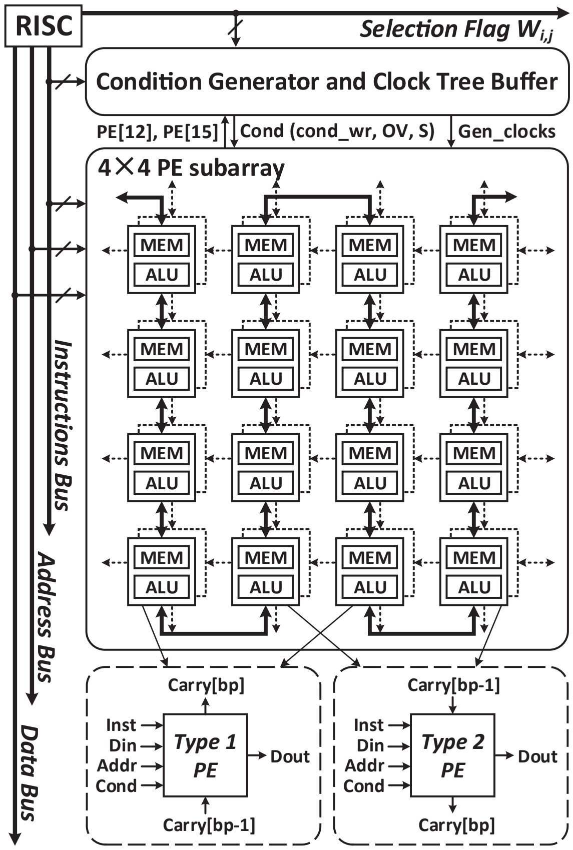


Fig. 4. Reconfiguration of the 4 × 4 PE subarray by switching the connections among PEs.

3. Experimental results

The proposed vision processor has been fabricated in a 0.18 μm 1P5M CMOS technology. Figure 6 shows the chip microphotograph. The processor contains a 64 × 64 PE/SOM array, a 64 × 1 RP, and a dual-core RISC. The measurement platform is shown in Fig. 7(a). It is equipped with a chip under test, a 60 fps QVGA image sensor, and a Cyclone III FPGA. The FPGA can be programmed to accomplish the raw image subsampling and the data exchange between the vision processor and the PC terminal. Another high-speed image sensing and processing measurement platform is implemented by combining the processor with a 1000 fps 800 × 600 CMOS image sensor based on the high-speed 4-T pixel,[32] as shown in Fig. 7(b). For each generated frame, the FPGA stores an arbitrary 256 × 256 image and the RISC saves a 64 × 64 subimage into the data input/output (I/O) memory. An external 2 GB DDR memory is used to record the high-speed image, and the 1000 fps image recording based on the event detection and image recognition tasks can be realized.

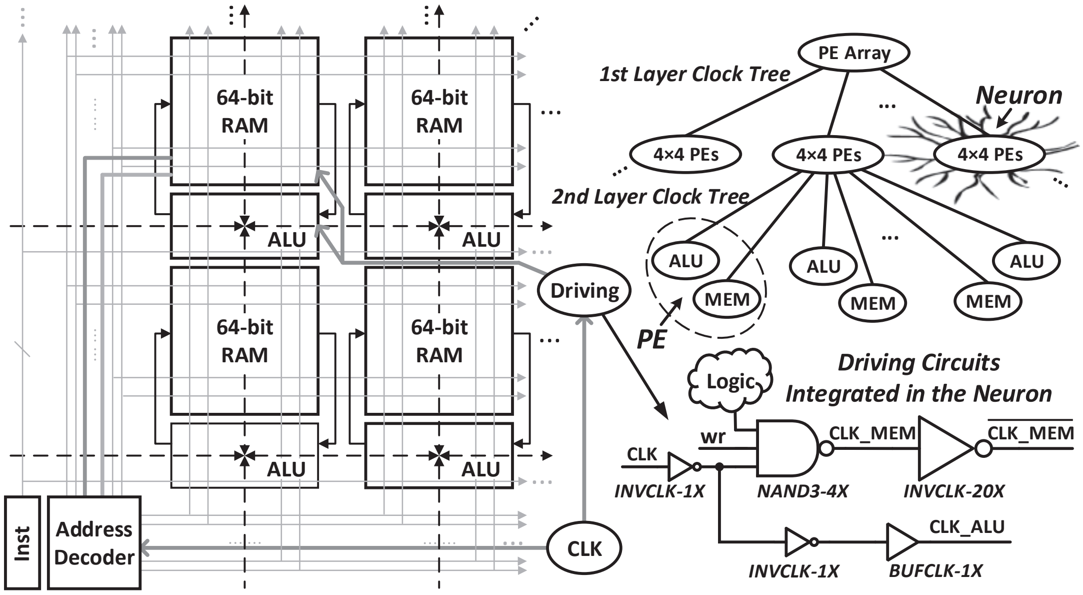


Fig. 5. Implementation of the balanced clock tree.

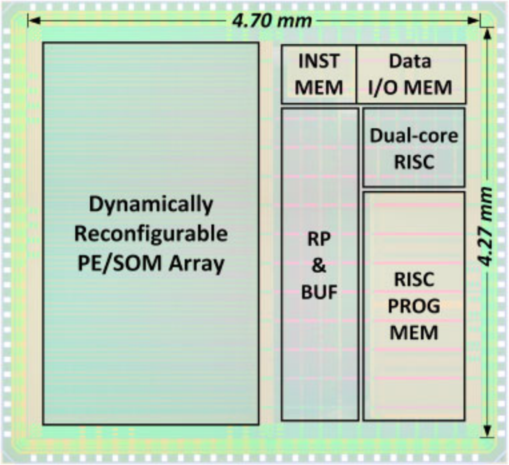
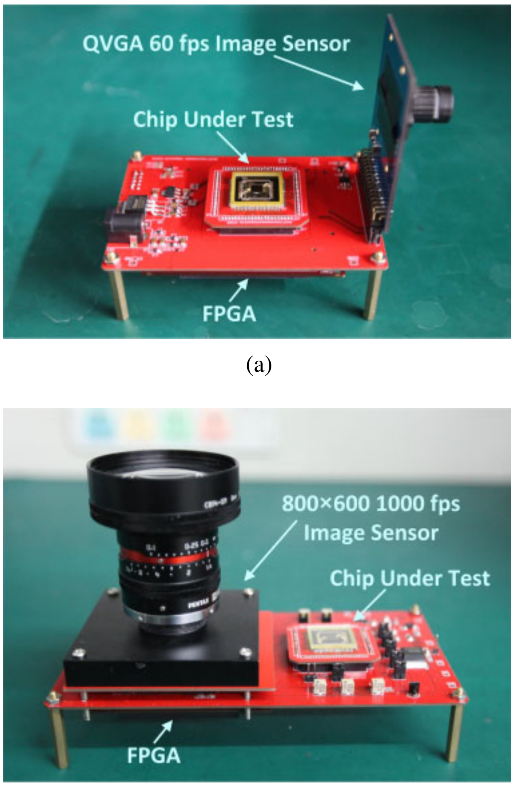


Fig. 6. (Color online) Microphotograph of the fabricated chip.

Test results show that low-level image processing algorithms such as image inversion and morphology-based edge detection can be fulfilled at high speeds. Figure 8(a) shows the measured image processing results. The time consumed on data transmission from the I/O memory to the PE array is not determined. Even considering the data transmission time, most low-level image processing algorithms can still be finished within 1 ms under a 50 MHz clock frequency. Figure 8(b) shows the number of consumed instructions considered for various low-level and mid-level image processing algorithms. Figure 8(c) shows the circuit area of the architecture presented in this paper compared with that in our previous work. According to the measurement, the circuit area of the distributed memory has been reduced by almost 70%, while the circuit area of the rest of the circuits in the architecture is reduced by almost 30%. The vision processor achieves area saving of over 40% while it well preserves the image processing functions of the previous architecture.

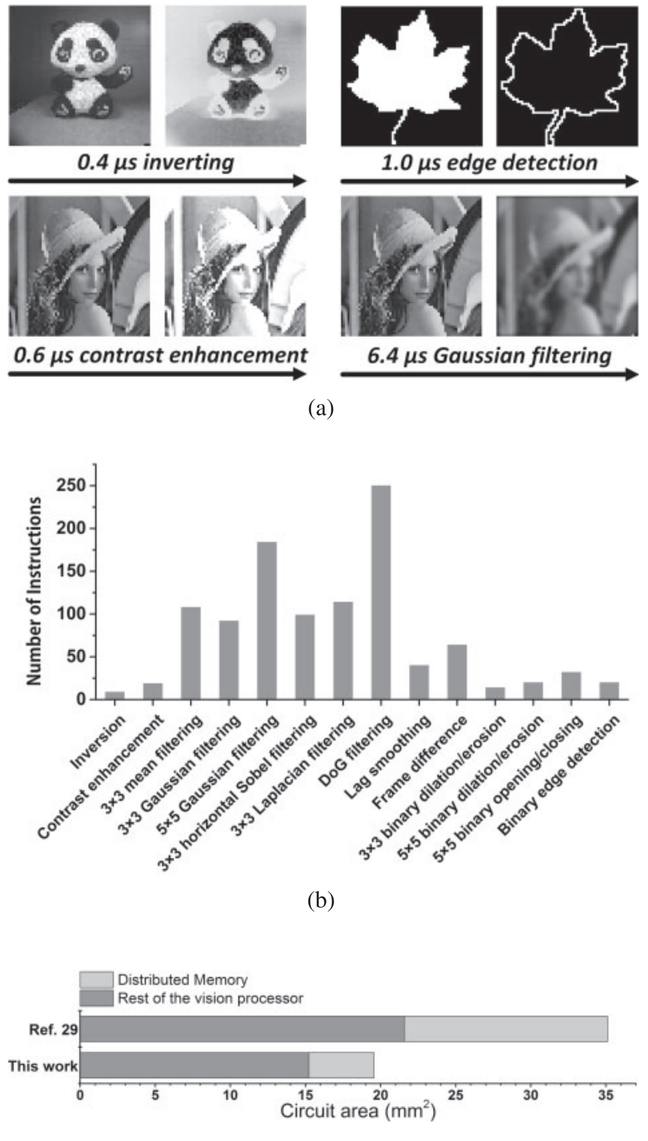
The averaged principal-edge distribution (APED) feature has been demonstrated as a robust descriptor for high-speed object recognition and tracking.[33] Figures 9(a) and 9(b) present APED feature extraction results. The vision processor presented in this paper can be programmed to extract the APED feature within 1 ms, and image recognition can be realized by reconfiguring the PE array into the SOM neural network. In our test, we chose 40 categories in the ORL face database[34] for both training and testing. Figure 9(c) shows the recognition rates achieved by using the minimum distance classifier supported by the SOM neural network under



(a)

(b)

Fig. 7. (Color online) Test platform for the vision processor. (a) Test board equipped with a QVGA 60 fps image sensor. (b) Test board equipped with an 800 × 600 1000 fps high-speed image sensor.

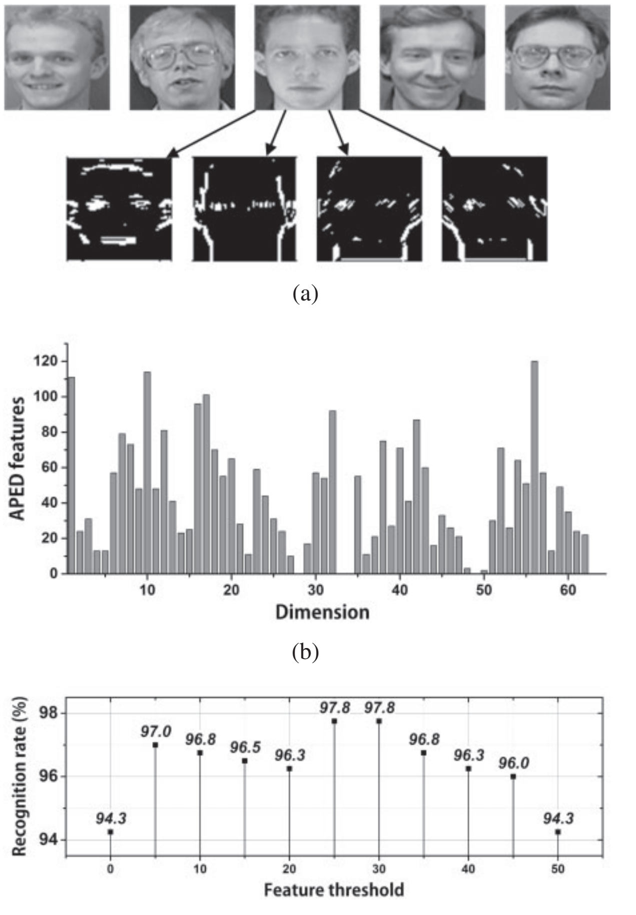


(a)

(b)

(c)

Fig. 8. (a) Test image processing results with consumed time under 50 MHz clock frequency. (b) Numbers of instructions considered for various image processing algorithms. (c) Breakdown of the circuit area of the proposed distributed memory and the processor compared with that in the previous work.



(a)

(b)

(c)

Fig. 9. (a) APED feature map generation. (b) APED feature extraction within 1 ms. (c) Image recognition rates given by minimum distance classifier related to the feature extraction threshold.

different feature extraction thresholds. Reference vectors are generated in the off-line training process in order to speed up the recognition process. The on-line learning of the SOM neural network is based on the learning vector quantization (LVQ) method.[35] It can also be accomplished on the proposed vision processor and the recognition rate could be expected to increase further.

4. Discussion

The proposed vision processor can be programmed in a general purpose manner for a wide scope of high-speed visual tasks, ranging from low-level image processing to high-level image recognition. By designing the reconfigurable PE array with the full-custom local memory, the circuit area of the vision processor is markedly reduced. Table I shows the performance comparison among the state-of-the-art digital vision chips. The processor proposed in this paper achieves the same processing performance as the previous hybrid reconfigurable processing architecture.[28] An external high-speed image sensor with high resolution can be integrated to further improve the performance of the image sensing and processing system. As the process scales down, the proposed vision processor and local memory design method can be used to save the circuit area and improve the processing efficiency further. In our test, we observed that some PEs failed to operate appropriately under the standard 1.8 V supply voltage. It is because we had not checked the timing between the full-custom PE array and the automatically designed RISC processor. We infer that the difference in transmission delay on the clock and the control signal transmission paths for the PE array may lead to the setup time violation. When we increase the supply voltage or decrease the clock frequency, the violation will be avoided, and the processing function will be corrected. As a result, the power consumption turns out to be relatively high as the supply voltage is set to be 2.3 V, as shown in Table I. For future designs, the problem must be solved by introducing strict timing check between the full-custom PE array and the rest of the processing circuits.

5. Conclusion

A high-speed reconfigurable vision processor based on a full-custom distributed memory has been proposed in this paper. The memory adopts a two-stage latch-based architecture to

Table I. Performance comparison among the state-of-the-art digital vision chips.

	This work	Ref. 28	Ref. 23	Ref. 12	Ref. 29[a]
Technology	0.18 µm 1P5M	0.18 µm 1P5M	0.18 µm 1P6M	0.35 µm 2P4M	0.15 µm FD-SOI
Image sensor	w/o	w/	w/	w/	w/o
Sensor resolution	800 × 600	256 × 256	128 × 128	19 × 22	320 × 240
Chip area (mm^2)	24.8	82.3	13.5	9	37.2
Package	QFP 88 pin	BGA 312 pin	N/A	N/A	N/A
Clock frequency (MHz)	50	50	100	75	100
Supply voltage (V)	2.3	1.8/3.3	1.8/3.3	2.5	1.5
Power consumption (mW)	499	630	450	26.4	450
PE resolution	64 × 64	64 × 64	32 × 128	64 × 64	8 × 8
Parallelism	Pix./Row/Vector	Pix./Row/Vector	Pix./Row	Pix.	Pix./Vector
MPU	32 bit dual-core	32 bit dual-core	8 bit single-core	Not integrated	Not integrated
Performance (GOPS)	12	12	44	1	6.4
P_A (GOPS/mm^2)	0.634	0.342	3.4	0.205	0.172
P_E (MOPS/mW)	24	19	97.8	38	14.22

a) Only tier 1 of the 3D integrated vision chip is taken for comparison.

achieve the compact size. The dynamic reconfiguration between the PE subarray and an SOM neuron has been realized by adding dedicated connections in the subarray. Finally, a balanced clock tree has been realized in CG to accomplish synchronization within the vision processor. The vision processor benefits from these proposed design methods and achieves an area reduction of over 44%, while the image processing functions supported by the processing architecture are well preserved.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant Nos. 61234003, 61434004, and 61504141). The authors would like to thank Runjiang Dou for setting up the measurement platform.

1) C. Mead, Proc. IEEE **78**, 1629 (1990).
2) M. Ishikawa, A. Morita, and N. Takayanagi, Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 1992, p. 373.
3) J. Akita, A. Watanabe, O. Tooyama, M. Miyama, and M. Yoshimoto, IEEE Trans. Electron Devices **50**, 184 (2003).
4) T. Komuro, I. Ishii, M. Ishikawa, and A. Yoshida, IEEE Trans. Electron Devices **50**, 191 (2003).
5) T. Komuro, S. Kagami, and M. Ishikawa, IEEE J. Solid-State Circuits **39**, 265 (2004).
6) I. Ishii, K. Yamamoto, and M. Kubozono, IEEE Trans. Electron Devices **53**, 1797 (2006).
7) W. Miao, Q. Lin, W. Zhang, and N. Wu, IEEE J. Solid-State Circuits **43**, 1470 (2008).
8) C. Yin and C.-C. Hsieh, IEEE Asian Solid-State Circuit, 2013, p. 97.
9) W. Jendernalik, G. Blakiewicz, J. Jakusz, S. Szczepanski, and R. Piotrowski, IEEE Trans. Circuits Syst. I **60**, 279 (2013).
10) G. L. Cembrano, A. Rodriguez-Vazquez, R. C. Galan, F. Jimenez-Garrido, S. Espejo, and R. Dominguez-Castro, IEEE J. Solid-State Circuits **39**, 1044 (2004).
11) J. Dubois, D. Ginhac, M. Paindavoine, and B. Heyrman, IEEE J. Solid-State Circuits **43**, 706 (2008).
12) A. Lopich and P. Dudek, IEEE Trans. Circuits Syst. I **58**, 2420 (2011).
13) T. Morie and K. Youngjae, Proc. ISSCC Dig. Tech. Pap., 2009, p. 478.
14) N. Massari and M. Gottardi, IEEE J. Solid-State Circuits **42**, 647 (2007).
15) N. Cottini, M. Gottardi, N. Massari, R. Passerone, and Z. Smilansky, IEEE J. Solid-State Circuits **48**, 850 (2013).
16) S.-J. Kim, K.-H. Lee, S.-W. Han, and E. Yoon, IEEE J. Solid-State Circuits **43**, 2558 (2008).
17) D. Kim and G. Han, IEEE J. Solid-State Circuits **44**, 2581 (2009).
18) T. K. Horiuchi, IEEE Trans. Circuits Syst. I **56**, 1566 (2009).
19) A. Graupner, S. Schreiter, S. Getzlaff, and R. Schuffny, IEEE J. Solid-State Circuits **38**, 948 (2003).
20) T. Komuro, A. Iwashita, and M. Ishikawa, IEEE Micro **29** [6], 58 (2009).
21) H. Yamashita and C. G. Sodini, IEEE Trans. Electron Devices **56**, 2534 (2009).
22) C.-C. Cheng, C.-H. Lin, C.-T. Li, and L.-G. Chen, IEEE J. Solid-State Circuits **44**, 127 (2009).
23) W. Zhang, Q. Fu, and N. Wu, IEEE J. Solid-State Circuits **46**, 2132 (2011).
24) J. Choi, S. Park, J. Cho, and E. Yoon, IEEE J. Solid-State Circuits **49**, 289 (2014).
25) J.-Y. Kim, M. Kim, S. Lee, J. Oh, K. Kim, and H.-J. Yoo, IEEE J. Solid-State Circuits **45**, 32 (2010).
26) G. Kim, K. Lee, Y. Kim, S. Park, I. Hong, K. Bong, and H. J. Yoo, IEEE J. Solid-State Circuits **50**, 113 (2015).
27) L. Camunas-Mesa, C. Zamarreno-Ramos, A. Linares-Barranco, A. J. Acosta-Jimenez, T. Serrano-Gotarredona, and B. Linares-Barranco, IEEE J. Solid-State Circuits **47**, 504 (2012).
28) C. Shi, J. Yang, Y. Han, Z. Cao, Q. Qin, L. Liu, N. Wu, and Z. Wang, IEEE J. Solid-State Circuits **49**, 2067 (2014).
29) R. Carmona-Galán, Á. Zarándy, C. Rekeczky, P. Földesy, A. Rodríguez-Pérez, C. Domínguez-Matas, J. Fernández-Berni, G. Liñán-Cembrano, B. Pérez-Verdú, Z. Kárász, M. Suárez-Cambre, V. Brea-Sánchez, T. Roska, and A. Rodríguez-Vázquez, J. Syst. Archit. **59**, 908 (2013).
30) I. Hong, K. Bong, D. Shin, S. Park, K. Lee, Y. Kim, and H. J. Yoo, Proc. ISSCC Dig. Tech. Pap., 2015, 18.1.
31) Z. Chen, J. Yang, C. Shi, and N. Wu, IEEE Int. Conf. ASIC, 2013, P1-14.
32) Z. Cao, Y. Zhou, Q. Li, Q. Qin, L. Liu, and N. Wu, Proc. Int. Image Sensor Workshop, 2013, p. 229.
33) P. Zhao, H. Zhu, H. Li, and T. Shibata, IEEE Trans. Circuits Syst. Video Technol. **23**, 503 (2013).
34) Web [http://www.cl.cam.ac.uk].
35) T. Kohonen, Proc. IEEE **78**, 1464 (1990).