

# A High Speed Digital Vision Chip with Multi-grained Parallel Processing Capability

Takashi Komuro, Shingo Kagami, and Masatoshi Ishikawa

Department of Information Physics and Computing, University of Tokyo  
7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

kom@k2.t.u-tokyo.ac.jp

## Abstract

A vision chip which has massively parallel processing elements integrated with photo detectors is effective for performing high frame rate image processing. However, in designing a general-purpose vision chip, it is a problem that there is a trade-off between flexibility and the number of pixels. This paper shows a new architecture and sample algorithms of a vision chip that has a function to chain processing elements. A prototype chip with  $64 \times 64$  pixels using the  $0.35\mu\text{m}$  CMOS process is also shown.

## 1 Introduction

A vision chip is a CMOS image sensor each pixel of which has a processing element (PE). It can perform not only image capture but also on-chip computation over captured images such as image filtering and feature extraction. Compared with a system in which a sensor and a processor are separated, the vision chip need not to scan out images and transmit them to the processor. Therefore it can handle high frame rate images in real-time not using high-bandwidth communication which consumes much power.

We have been developed vision chips and systems introducing digitally-designed PEs[1, 2, 3, 4, 5], and have applied them to high speed visual feedback for controlling mechanical devices such as cameras, microscopes, and robots[6, 7, 8, 9]. The chips are also expected to be used for inspection, monitoring, human interface and so on.

In designing a vision chip, it is desirable to make it programmable for wide use. However, there is a trade-off between flexibility and the number of pixels. In order to raise the flexibility of image processing, it is necessary to raise the capability of each PE. As a result, the circuit area of the PE grows larger and the number of pixels to be integrated in one chip decreases.

In order to solve the problem, a new architecture of a vision chip has been designed in which several PEs can be chained. In this paper we describe the designed architecture and algorithms, and a prototype chip is shown.

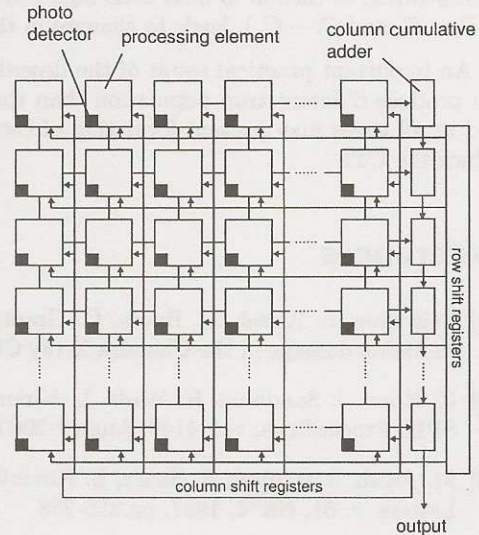


Fig. 1: Structure of the Entire Chip.

## 2 Architecture

The structure of the entire chip is shown in Fig. 1. PEs are arranged in a two-dimensional array and a photo detector (PD) is attached to each PE. Each PE is connected to up, down, left, and right PEs and performs neighbor communication. For every row and column, a common global bus is arranged and data are given to the buses from outside via shift registers. As a processed result, scalar values are output in a bit-serial manner via column cumulative adders connected to right-end PEs.

The structure of the PE is shown in Fig. 2. The PE consists of a 1 bit ALU with a carry register and 24 bit memory, and is controlled by programs. Program control is performed by an external controller and all the PEs performs the same operation at the same time. The inputs from the PD and the row/column common bus, and zero signal are mapped to the memory space and the PE can get them without using special I/O instructions. Communication with a neighbor PE is performed via a latch and it is possible to select

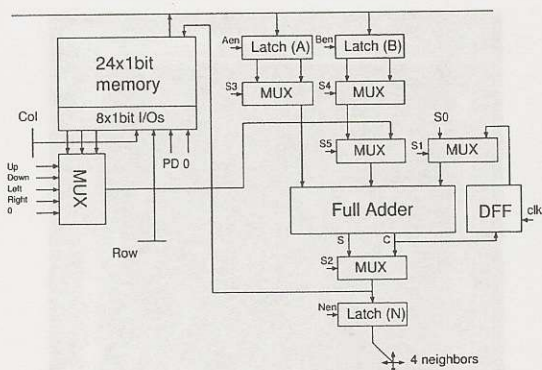
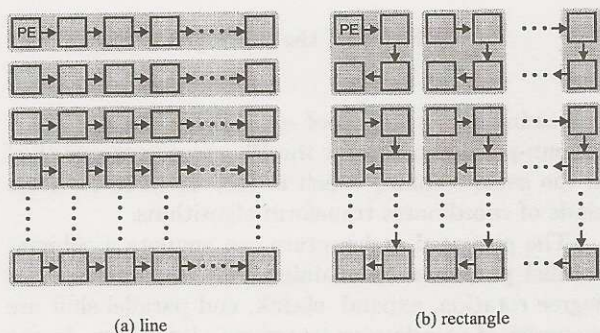


Fig. 2: Structure of the PE.



(a) line

(b) rectangle

Fig. 3: Examples of the PE chains.

each of up, down, left, right, and zero as an input of neighbor communication according to the status of the condition registers in the memory space.

In addition, two neighboring PEs can be connected together by setting one of the ALU inputs to *neighbor* and enabling the latch N. By connecting PEs in series, several PEs can be chained. The procedure is that 0 is selected as the inter-PE input in the PE at the starting point and then the PEs are connected one after another to the terminal point. Examples of the PE Chains are shown in Fig. 3.

The chained PEs process cumulative operations. By connecting the output of the ALU to the ALU input of the neighboring PE, the ALUs are multi-staged. Examples of cumulative operations are shown in Fig. 4. By selecting logical OR, addition, or carry calculation as an operation, cumulative logical OR, cumulative addition, or multi-bit addition is performed respectively.

In the chained PEs, the ALUs are multi-staged via a latch, not a flip-flop. This makes them perform cumulative operations efficiently. Compared with the chain via a latch and a flip-flop, the total of each PE

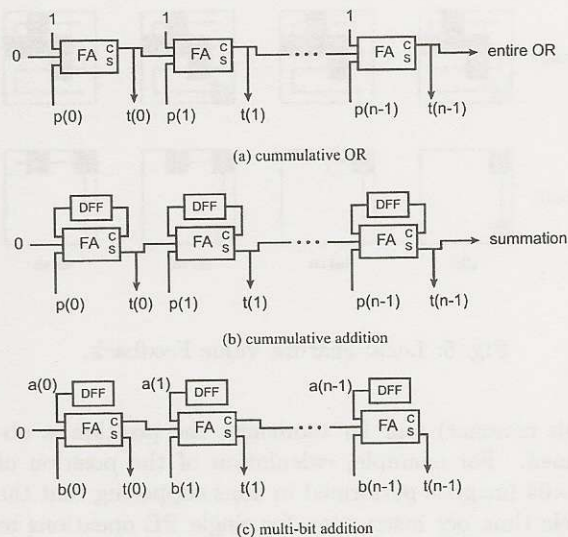


Fig. 4: Examples of Cumulative Operations.

delay is almost same but in the former the variation of each PE delay is canceled. Also, one flip-flop for one PE requires very high control frequency but generally there is an upper limit in control frequency.

Such chained PEs can be regarded as a large PE. If  $n$  PEs are chained, The ALUs are multi-staged and behave as an  $n$  bit ALU. Memory capacity is also multiplied by  $n$ . Therefore, it is possible to raise PE performance by chaining many PEs. Also, in chained PEs, data broadcast using cumulative logical OR can be performed freely.

### 3 Algorithms

The mesh-connected network hardware the proposed architecture introduces is effective in the image preprocessing called early visual processing. For example, supposing the cycle time per instruction (read data from memory, write result to memory, write result to the latch, etc.) to be 100ns, edge detection over a binary/6bit image is performed in 3.1/19 $\mu$ s. Smoothing over a binary/6bit image is performed in 4.7/15 $\mu$ s. Using the previous frame information, target tracking over a binary image is performed in 3.4 $\mu$ s.

The position of a target is indispensable information to visual feedback. Chaining PEs in rows and performing cumulative addition operations, row summations can be calculated efficiently. The summation of an entire image is calculated by column adder from the row summations. Calculating summations of images masked by signals from common global bus, moments can also be calculated. From the summation

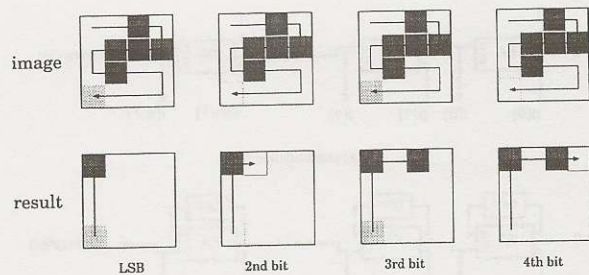


Fig. 5: Local Feature Value Feedback.

(0th moment) and 1st moments, the position is obtained. For example, calculation of the position of  $64 \times 64$  image is performed in  $40 \mu s$  supposing that the cycle time per instruction for single PE operations to be  $100 ns$  and for  $n$ -stage chained PE operations  $0.5 ns$  (figures under  $100 ns$  is rounded up)

In chained PEs, local feature value feedback is realized by calculating a feature value using summation and then distributing the result to the PEs using broadcast with each bit stored to one PE. All processes are performed in a bit-serial manner and do not consume temporary memory. The procedure is shown in Fig. 5. A PE in a block is used as a pixel in an image at one time and is used as a bit in a word at another time. This provides more effective memory use.

Parallel block matching is an example of an algorithm using local feature value feedback. It is an algorithm to search, from two images, the direction where each block in one image shifts in another image. By calculating matching of two successive frames, an optical flow is obtained that can be applied to motion measurement and motion stereo. Every SAD (summation of the absolute of difference) is calculated as one image is moved spirally, and the minimum SAD and the index are updated when the SAD is larger than the old minimum SAD.

Normally, it takes considerable time to calculate a summation for obtaining an SAD and it is difficult to repeat the calculation many times. However, since the summation in a block can be calculated efficiently, it can be performed in a reasonable time in this chip. For example, in the case the block size is  $8 \times 8$  and search area is  $5 \times 5$  and in the same condition for position calculation, the processing time is  $1.3 ms$ . Also, the SAD, the minimum, and the index are distributed to several PEs dispersedly without using much memory.

Among conventional SIMD processors for image processing, there are column-parallel processors, in which each column has a PE. These types of processors are slower than fully-parallel processors as repeating operations are needed. However, they seem to raise the

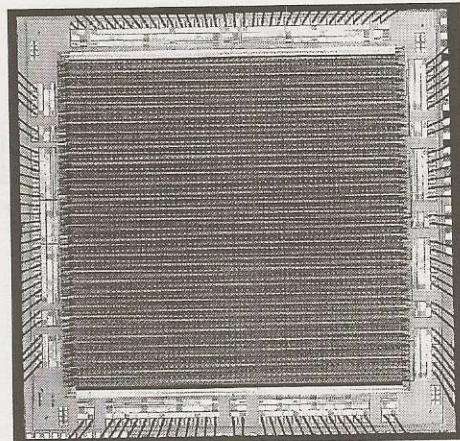


Fig. 6: Photo of the prototype chip.

processing performance of each PE. Moreover, in a column-parallel processor the PE can access any pixel in the same column, which is very effective for some kinds of coordinates transform algorithms.

The proposed architecture can emulate a column-parallel processor by chaining PEs in a column. 90 degree rotation, expand, shrink, and parallel shift are examples of coordinates transform algorithms. In the 90 degree rotation algorithm, a column is transposed to a row for every column via the diagonal line. Data transfer in the same column or row is performed using broadcast. Expand, shrink and parallel shift can be achieved by copying the pixel values in the row direction for every column, and then doing the same in the column direction for every row. In the case the number of pixels is  $64 \times 64$  and in the same condition for position calculation, the processing time of 90 degree rotation is  $330 \mu s$  and that of expand, shrink or parallel shift is  $400 \mu s$ .

## 4 Prototype Chip

Based on the proposed architecture, we have developed a prototype chip. The chip has  $64 \times 64$  pixels in a  $5.4 mm \times 5.4 mm$  area using the  $0.35 \mu m$  TLM CMOS process. The area of the PE is as small as  $67.4 \mu m \times 67.4 \mu m$ . This means that  $256 \times 256$  pixels can be integrated in about a  $1.8 cm \times 1.8 cm$  chip and the standard pixel number as an image processing device is achievable. Fig. 6 shows a photograph of the chip.

Using this chip, some image processing is performed. Though it is proper usage that the vision chip output only the scalar feature values, the images are output here for showing the visual results.

Fig. 7 shows the results of early visual processing

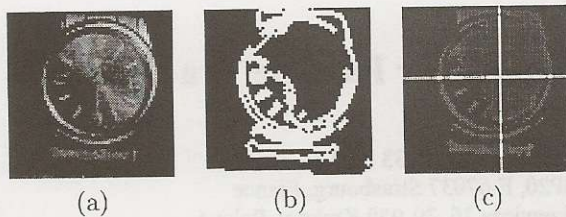


Fig. 7: Sample Output of Early Visual Processing and Global Feature Extraction

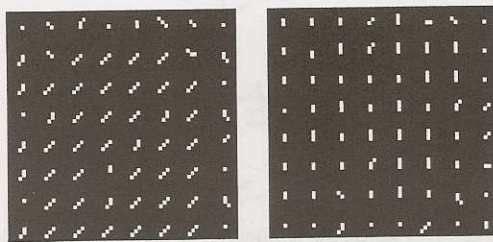


Fig. 8: Sample Output of Parallel Block Matching

and global feature calculation. A lighted watch is used as an input image. In (a) the image is captured in gray-scale. In (b), the captured image is binarized and edge-detected. In (c) the centroid of the captured image is calculated and displayed.

Fig. 8 shows the results of calculating optical flows of transmitting light masked by a sheet on which random dots are printed using parallel block matching. As the sheet is shifted to up, down, right and left in parallel, it is observed that all of the motion vectors turn in the same direction except around the edge of the screen.

(a) of Fig. 9 is the result of 90 degree rotation to transmitting light masked by a sheet on which letters are printed. (b) is the result of enlargement. The magnification is 1.5 and the image is expanded centering on the middle of the image.

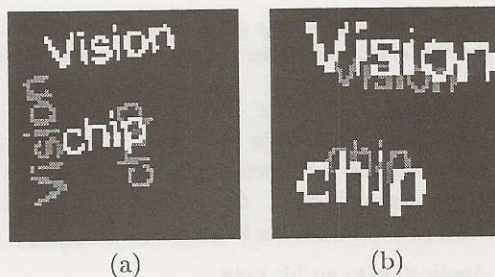


Fig. 9: Sample Output of Coordinates Transform Algorithms

## 5 Conclusion

We describe the new architecture of the vision chip that can perform flexible and high level image processing by chaining several PEs. In our proposed architecture, multi-grained image processing is realized. The  $64 \times 64$  pixels prototype chip has successfully been developed. We expect that this vision chip will widen the application range of high speed visual feedback.

## References

- [1] M. Ishikawa, A. Morita, and N. Takayanagi, "High Speed Vision System Using Massively Parallel Processing," *Proc. 1992 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS'92)*, pp.373-377, 1992.
- [2] T. Komuro, I. Ishii, and M. Ishikawa, "Vision Chip Architecture Using General-Purpose Processing Elements for 1ms Vision System," *Proc. 4th IEEE Int. Workshop on Computer Architecture for Machine Perception (CAMP'97)*, pp.276-279, 1997.
- [3] M. Ishikawa and T. Komuro, "Digital Vision Chips and High-Speed Vision Systems (Invited)," *Dig. Tech. Papers of 2001 Symposium on VLSI Circuits*, pp.1-4, 2001.
- [4] T. Komuro, I. Ishii, M. Ishikawa and A. Yoshida: A Digital Vision Chip Specialized for High-speed Target Tracking, *IEEE transaction on Electron Devices*, Vol.50, No.1 (2003) (to appear)
- [5] Y. Nakabo, M. Ishikawa, H. Toyoda, and S. Mizuno, "1ms Column Parallel Vision System and Its Application of High Speed Target Tracking," *Proc. IEEE Int. Conf. Robotics and Automation*, pp.650-655, 2000.
- [6] Y. Nakabo and M. Ishikawa, "Visual Impedance Using 1ms Visual Feedback System," *Proc. IEEE Int. Conf. Robotics and Automation*, pp.2333-2338, 1998.
- [7] A. Namiki and M. Ishikawa, "Optimal Grasping Using Visual and Tactile Feedback," *Proc. IEEE Int. Conf. Multisensor Fusion and Integration for Intelligent Systems*, pp.589-596, 1996.
- [8] A. Namiki and M. Ishikawa, "Vision-Based Online Trajectory Generation and Its Application to Catching," *Proc. 2nd Joint CSS/RAS Int. Workshop on Control Problems in Robotics and Automation*, to appear, 2002.
- [9] H. Oku, I. Ishii, and M. Ishikawa, "Tracking a Protozoon Using High-Speed Visual Feedback," *Proc. 1st Annual Int. IEEE-EMBS Special Topic Conf. on Microtechnologies in Medicine & Biology*, pp.156-159, 2000.