Motion-Vector Estimation and Cognitive Classification on an Image Sensor/Processor 3D Stacked System featuring ThruChip Interfaces

Tetsuya Asai*, Masafumi Mori*, Toshiyuki Itou*, Yasuhiro Take†,
Masayuki Ikebe*, Tadahiro Kuroda† and Masato Motomura*
*Graduate School of Information Science and Technology, Hokkaido University
Kita 14, Nishi 9, Kita–ku, Sapporo 060–0814, Japan
Email: see http://lalsie.ist.hokudai.ac.jp/
†Department of Electrical Engineering, Keio University
Hiyoshi 3–14–1, Kohoku-ku, Yokohama, Kanagawa 223–8522, Japan

***Abstract*—1,000 fps motion vector estimation and classification engine for highspeed computational imaging in a 3D stacked imager/processor module is proposed, prototyped, assembled, and also tested. The module features 1) ThruChip interfaces for high fps image transfer, 2) orders of magnitude more area/power efficient motion vector estimation architecture compared to conventional ones, and 3) a cognitive classification scheme employed on motion vector patterns, enabling the classification of moving objects not possible in conventional proposals.**

I. Introduction

Computational imaging is a state-of-the-art digital imaging technology that captures and processes numerous image snapshots to invent perceptually meaningful representation of our visual world (Fig. 1 left). Difficult challenge in computational imaging is to achieve both high-speed imaging, which enables capturing motions not being able to see with human eyes, and low-power image processing. In this paper, we propose a 3-D stacked module for such high-speed computational imaging applications consisting of our low-power CMOS imager [1] and an image-processor die where image snapshots are transferred to the low-power image processor via high-speed ThruChip Interfaces (TCIs) [2] utilizing inductive-coupling between numerous numbers of coils on each die (Fig. 1 right).

II. Proposed Algorithms and Architectures

Our target, in terms of the computational imaging, is hardware motion-vector estimation and its cognitive classification. The key idea is to reduce computational power of motion vector estimation (block matching) in the image processor by utilizing high-speed imaging and high-bandwidth image-data transfer between the imager and processor with TCIs, as illustrated in Fig. 2, based on the fact that movement of real-world subjects on image sensors tends to be limited within 1 pixel under high fps condition. Computational cost of pattern search process ($\sim$ search area) in block matching is obviously reduced as sampling frame-rate increases, since the inter-frame difference is decreased as the frame rate increases.

The minimum computational cost of block matching is obtained under a minimum macro block of 3×3 pixels and search area of 5×5 pixels (corresponding to 1-pixel search range), as shown in Fig. 2 right. Although conventional block-matching hardware employs much larger search ranges (*e.g.*, a few (Fig. 2 left) to a few tens $\sim$ hundreds in [3], [4]), we choose the minimum (1-pixel) search range intentionally because high-speed imaging (1,000 fps and more) enables motion-vector estimation even with the minimum search range. Table I compares simulated PSNR values (precision of motion

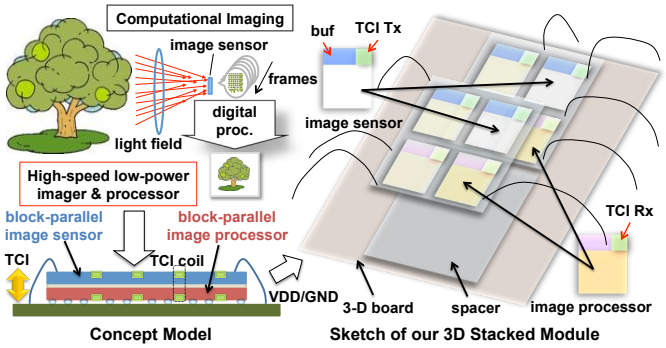


Fig. 1. Overall concept of proposed imager/processor 3D stacked module.

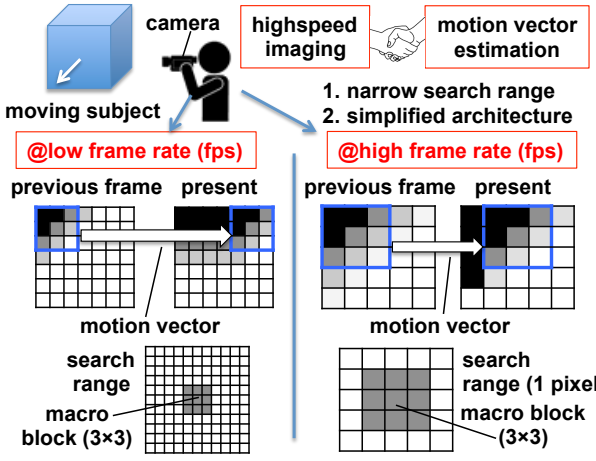


Fig. 2. Concept of 1-pixel search-range block matching under high-speed imaging environment.

TABLE I. COMPARISON OF COMPUTATIONAL COST AND PRECISION.

CPU	Xeon E5-1660 v2 (3.7GHz)
Macro block	8×8 / 3×3 ([5] / proposed)
Search range	7 / 1 ([5] / proposed)
Camera fps	30 / 1,000 ([5] / proposed)

BM algorithm	Avg. PSNR (dB)	Avg. relative runtime
Proposed	22.6	1
FSA [5]	16.6	22.2
SESTSS [5]	15.6	2.5
NTSS [5]	16.3	2.8
DS [5]	15.9	5.9
ARPS [5]	16.0	4.9

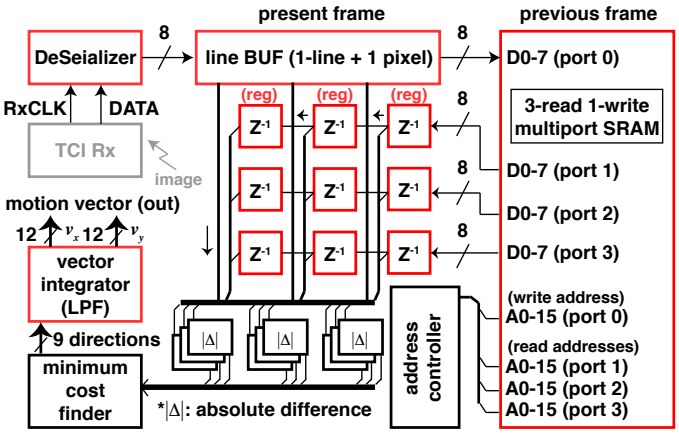


Fig. 3. Architecture of 1-pixel search-range block-matching module.

vector estimation) and averaged relative runtime (computational cost) of conventional block matching methods under 30 fps imaging [5] and the proposed scheme, which indicates that the proposed method exhibits major advantages in both precision and cost of motion vector estimation.

Figure 3 illustrates our proposed architecture of motion-vector estimator implementing the 1-pixel search-range block-matching method. The circuit accepts 14-bit data per pixel (12-bit pixel data plus 1-start and 1-stop bits) sequentially from an imager via TCI. The data are de-serialized to 8-bit depth values, and then are transferred to a line buffer where a part of present frame data is stored in this buffer. The data are further transferred to the multi-port SRAM storing previous frame data, and then forwarded to 3 register chains (Z^{-1}s in Fig. 3 middle). The register chain constructs a macro block buffer of 3×3 for block matching, and absolute differences ($|\Delta|$s) between the pixel values in the line buffer and the block buffer are calculated by nine $|\Delta|$ units in parallel, and then the most plausible vector having the minimum $|\Delta|$ value is selected. Since block matching with minimum (1-pixel) search range gives 8 vectors with magnitude 1 and zero vector only, to obtain multi-level magnitudes of motion vectors, one has to integrate the vectors in time. Instead of employing conventional digital integrator that requires additional frame buffers, we employ a leaky integrator that can be implemented by conventional digital low-pass filters (LPF). With this construction, the motion-vector output is sequentially generated, and is represented by temporal sequences in 12-bit vector form (V_x, V_y).

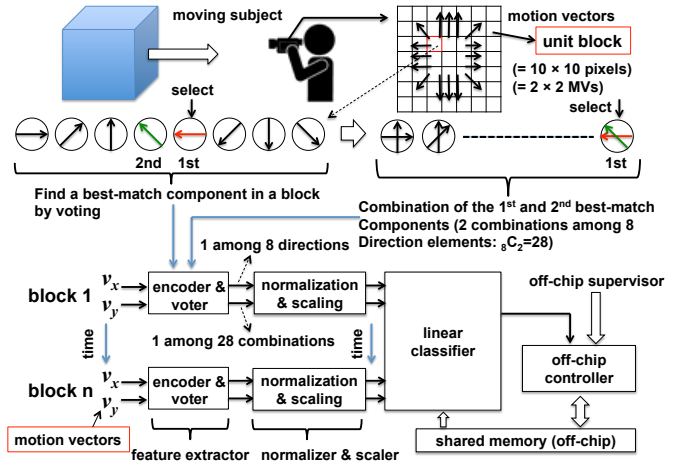


Fig. 4. Feature extraction scheme and its block components for proposed motion classifier.

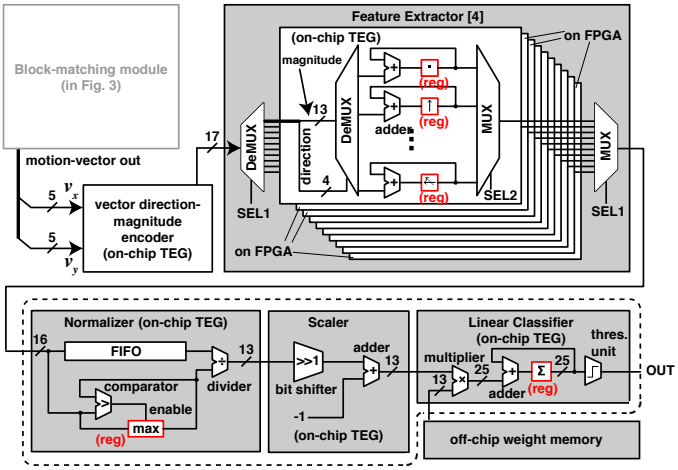


Fig. 5. Architecture of proposed motion-classification subsystem implementing on-/off-chip feature extractors.

The motion vectors are transferred to the motion classification subsystem shown in Figs. 4 and 5. Our classifier recognizes subject's motions, *e.g.*, rotation, zooming in/out, anomaly movement, etc., based on our machine-learning scheme [6] where we employ a part of brain-like structure to recognize motion in the image sequences by using motion vectors. Features in the motion-vector space are extracted by hardware neural circuits based on neurophysiological structures in visual cortex (V1) for generating orientation-selective map and middle temporal cortex (MT) for motion extraction and the subsequent cognitive processing, as shown in Fig. 4.

In the proposed motion classification system and architecture (Figs. 4 and 5), motion vectors (after integration by digital LPF) are forwarded to vector direction-magnitude encoder to adapt motion vectors to neural sparse representation. Then the sparse data is given to a feature extractor, which finds the

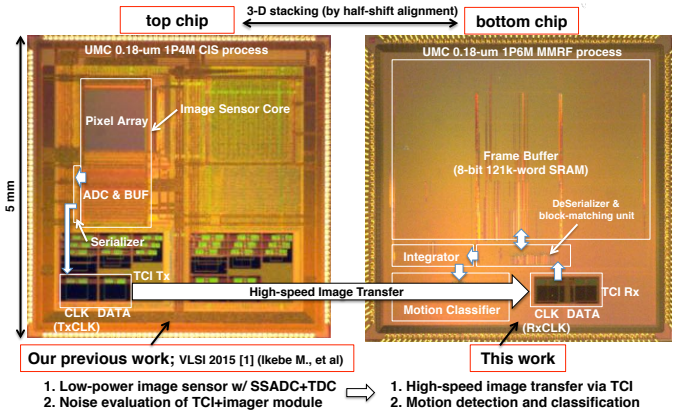
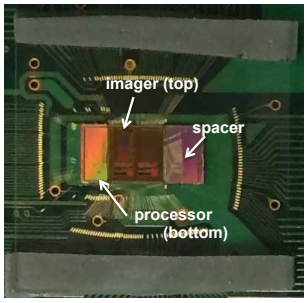


Fig. 6. Chip micrographs of our imager chip having data-transmitter TCI (left) [1] and our motion vector estimator and classifier chip having data-receiver TCI (right, this work).



Process	UMC 180nm 1P6M CMOS
P. Supply	1.8 V core / 3.3 V I/O
Die area	5.0 mm x 5.0 mm
Gate counts	19.9k gates (nand2) / 121.1 kB (SRAM)
Tx& Rx CLK	~100 MHz (measured) 560 MHz (designed)
System clock	~7.1 MHz (measured) 40 MHz (designed)
Resolution	200×200 pix, 40×40 vector 187 fps@100 MHz (meas) 1k fps @560 MHz (designed)

Fig. 7. 3D stacked module (board) snapshot, and system specification

most similar vector among 8 possible directional candidates, and determines dominant combinations of 2-major motion vectors in each block, through extensive parallel voting process. Registers in the feature extractor, denoted by "(reg)" in Fig. 5, represent the voting box, and after the voting process, the most similar and dominant combinations are selected in each block. The selected values are normalized by conventional normalizer to 0∼1 (13-bit unsigned fixed point values), and then scaled to -1∼1 (13-bit signed fixed point values). Finally, the scaled values are sent to a linear classifier that calculates weighted-sum of the scaled values and off-chip weight vales, accumulates them, and threshold the results.

III. Experimental Results

Figure 6 exhibits chip micrographs of our imager chip having data-transmitter TCI (left), which has previously presented in [1], and our motion vector estimator and classifier chip having data-receiver TCI (right, this work). Figure 7 shows a photograph of the fabricated 3-D stacked board, including the system performance summary table. The gate count was significantly reduced by 97∼99 % as compared to [3] and [4]. Due to our chip I/O and present experimental restrictions, we evaluated the chip and 3-D module at maximum 100 MHz transfer clock (187 fps with 200×200 pixels; 100 MHz / 14-bit = 7.1 MHz system clock) only, although the chip is designed to operate at transfer clock of 560 MHz, which will result in 1,000 fps at 200×200 image resolution.

Figure 8 exhibits raw motion-vector outputs of the motion

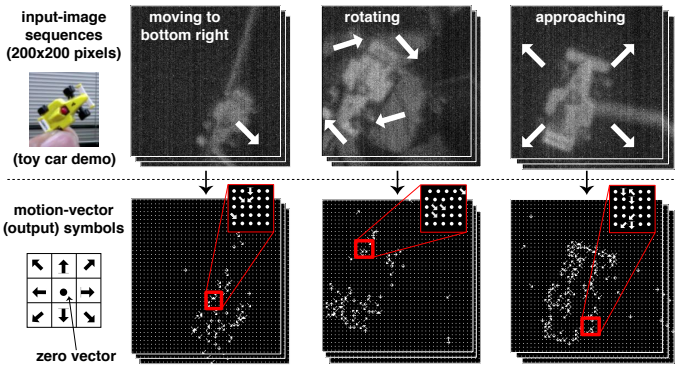


Fig. 8. Motion-vector examples estimated by our 3D stacked module before LPF processing (8 directions and 1 stationary outputs only).

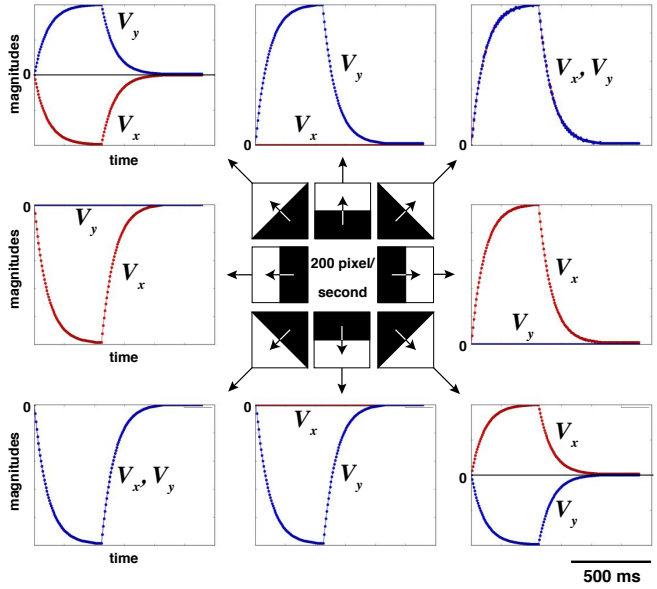


Fig. 9. Results of motion-vector integration by digital LPF with 200 pixel/s movements.

vector estimator on our 3-D stacked module. Motion sequences were captured by our imager (top chip), and then transferred to the processor (bottom chip) via TCI, and the bottom chip produced the vector outputs (the bottom pictures represent vectors before integration by digital LPF). Since the vectors shown in Fig. 8 bottom are raw vectors before vector integration by LPF, some error vectors, which can be attenuated by the LPF, were observed. Time courses of motion vectors $[V_x(t), V_y(t)]$ after on-chip LPF processing were shown in Fig. 9 for 8 different movements (200 pixel/s) of dark edges shown in the center. Smoothed vectors were successfully obtained, which resulted in smoothed motion vectors $[V_x(t), V_y(t)]$, as shown in Fig. 10.

Figure 11 exhibits 6 examples of motion classification that our system was able to recognize. The training was conducted offline, *i.e.*, we prepared datasets of various motions, and trained the classifier model on PC. The trained weights were transferred to the off-chip shared flash memory. Due to chip size limitation, we could implement one feature extractor on the chip, and the rest extractors were implemented on FPGA

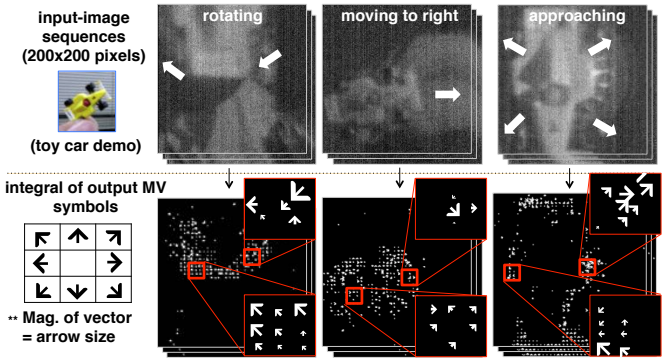


Fig. 10. Motion-vector examples estimated by our 3D stacked module after LPF processing.

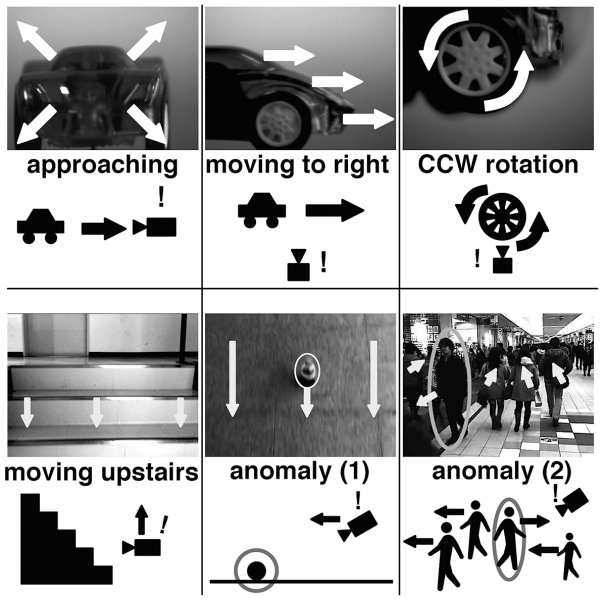


Fig. 11. Examples of tested motion-classification environment.

as expanding units, as shown in Fig. 5.

Table II shows a comparison table between state-of-the-art VLSIs [7], [8] and the proposed classifier, indicating 2 major advantages against them. First, their classification targets are static image, whereas the proposed system is able to handle dynamic image (motion). Second, although the number of input neurons of the proposed system was larger than that of [7], [8] by two orders of magnitude, power dissipation of the proposed system was lower than that of [7]. In our system, 90 % of the feature extractor was implemented on FPGA. Therefore, when all the feature extractors are implemented on a chip, the power dissipation will further be decreased. Since our classifier TEG (37-input feature extractor, normalizer, scaler, and linear classifier) consumed 7.2 mW@100 MHz transfer clock (7.1 MHz system clock), and the feature extractor consumed around 15 % of the total gate counts, we could estimate power dissipation of the single on-chip feature extractor as 1.1 mW. The FPGA feature extractor handles 3,700 inputs, therefore, if all the extractors are implemented on the chip, the estimated total power dissipation becomes 110 mW.

TABLE II. COMPARISON BETWEEN PROPOSED AND LATEST NEURAL-NET-BASED HARDWARE CLASSIFIERS.

	ISSCC 2014 [7]	ISSCC 2015 [8]	This work
Process	65nm 1P8M CMOS	180nm 1P6M CMOS	180nm 1P6M CMOS+FPGA
Target	Static Image (HMD Apps)	Static Image (Seizure)	Image Seq. (Motion)
Power	<778 mW	N/A	<7.2 mW / <497 mW
Gate count	8.32M	N/A	32k (nand2) / 29k (ALUT)
# of input	16	16	37 / 3,700
Classifier	Multi-layer Perceptron	Linear SVM	Linear SVM

IV. CONCLUSION

This paper has demonstrated that high fps image snapshots are the enabler for area/power efficient motion-vector estimation and classification systems. We previously showed, on the other hand, it is feasible to employ TCIs in high speed imagers since their noise interference is negligible when coils are placed in a right manner [2]. We hence conclude 3D stacking of imager/processor using TCIs, requiring just metal coils instead of costly TSVs, can become an attractive solution for high-speed computational imaging applications.

ACKNOWLEDGEMENT

This research was partly funded by the Semiconductor Technology Academic Research Center (STARC), Japan.

REFERENCES

[1] M. Ikebe, *et al.*, "Image sensor/digital logic 3D stacked module featuring inductive coupling channels for high speed/low-noise image transfer," Symposium on VLSI Circuits, pp. C82–83, 2015.

[2] D. Ditzel, *et al.*, "ThruChip wireless connections", Hot Chips 26, 2014.

[3] J. Zhou, *et al.*, "A 1.59Gpixel/s motion estimation processor with -211-to-211 search range for UHDTV video encoder," Symposium on VLSI Circuits, pp. C286–287, 2013.

[4] S.-Y. Jou, *et al.*, "Fast motion estimation algorithm and design for real time QFHD high efficiency video coding," IEEE Trans. Circuit Syst. Video Tech., vol. 25, no. 9, pp. 1533–1544, 2015.

[5] S. Sudhakar *et al.*, "Evaluation and comparison of different fast block matching algorithms using motion estimation", in Proc. Int. Conf. Sci. Eng. & Man. Res., pp. 1–13, 2014.

[6] T. Itou, *et al.*, "A new architecture for feature extraction to perform machine learning by using motion vectors and its implementation in an FPGA," Proc. RISP Int. Workshop on Nonlinear Circuits, Communications and Signal Processing, pp. 294–297, 2015.

[7] G. Kim, *et al.*, "A 1.22TOPS and 1.52mW/MHz augmented reality multi-core processor with neural network NoC for HMD applications," ISSCC Dig. Tech. Papers, pp. 182–184, 2014.

[8] M. Altaf *et al.*, "A 16-ch patient-specific seizure onset and termination detection SoC with machine-learning and voltage-mode transcranial stimulation," ISSCC Dig. Tech. Papers, pp. 394–396, 2015.