A SUB-100-MILLIWATT DUAL-CORE HOG ACCELERATOR VLSI FOR REAL-TIME MULTIPLE OBJECT DETECTION

Kenta Takagi, Kosuke Mizuno,
Shintaro Izumi, Hiroshi Kawaguchi, Masahiko Yoshimoto

Graduate School of System Informatics, Kobe University

ABSTRACT

In this paper, a Histogram of Oriented Gradients (HOG) feature extraction accelerator for real-time multiple object detection is presented. The processor employs three techniques: a VLSI-oriented HOG algorithm with early classification in Support Vector Machine (SVM) classification, a dual core architecture for parallel feature extraction, and a detection-window-size scalable architecture with a reconfigurable MAC array for processing objects of different shapes. Early classification reduces the number of computations in SVM classification. The dual core architecture and the detection-window-size scalable architecture enable the processor to operate in several modes: high-speed mode, low-power mode, multiple object detection mode, and multiple shape object detection mode. These techniques expand the processor flexibility required for versatile application. The test chip was fabricated using 65 nm CMOS technology. The proposed architecture is designed to process HDTV resolution video (1920 × 1080 pixels) at 30 frames per second (fps). The performance of this accelerator is demonstrated on a pedestrian detection system.

Index Terms— HOG, object detection, low-power, demonstration system

1. INTRODUCTION

Object detection is a crucial task for many computer vision applications such as surveillance, entertainment, automotive systems, and robotics. HOG [1], a widely accepted feature for object detection, is robust against changes of illumination, attaining high accuracy in the detection of variously textured objects.

Recent progress of high-performance general-purpose processors enables them to achieve real-time object detection. However, those processors require high power consumption, rendering them unsuitable for mobile systems, which have limited battery capacity. Consequently, a low-power and high-performance HOG feature extraction processor is necessary to widen the range of applications.

Figure 1 presents the image resolution versus frame rates reported from several related works of HOG hardware. Zhang et al. [2] proposed object detection with GPGPU. Some FPGA implementations [3]–[8] and an FPGA-GPU architecture [9] have been proposed for real-time applications. A target-reconfigurable object detector for multiple object detection was proposed by Yazawa et al. [6]. However, reloading of parameters for other objects is necessary to detect other objects, so that multiple objects cannot be detected simultaneously. Our previous work [10] on FPGA is superior to other works. However, it particularly targeted pedestrian detection. HOG features are adaptable to widely versatile applications. Therefore, anticipated HOG feature extraction processors must provide higher flexibility. Our goal is to develop design techniques for a real-time HOG feature extraction processor for use in multiple object detection from HDTV-resolution video.

The most common approach used in conventional processors is a window-based approach, for which the number of computations of 447.7 GOPS and memory bandwidth of 55 Gbps are necessary for HDTV resolution because of repetitive computations. Our previous work demonstrated that the computations and memory bandwidth are reduced significantly by the reuse of calculated data and adoption of efficient computation [10]. However, the power consumption remains high for mobile applications. The most dominant and second-most dominant activities are, respectively, cell histogram generation and SVM classification. Achieving low-power object detection demands improvement of the power efficiency of these two dominant processes.

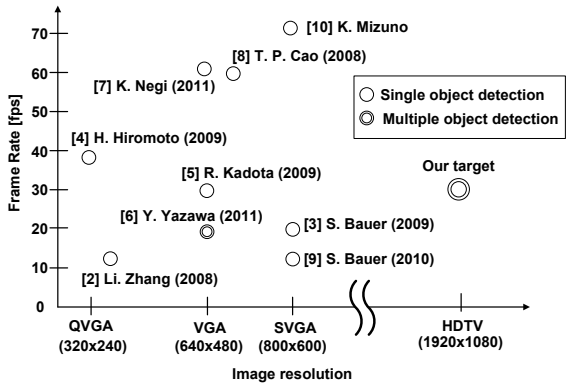

Fig. 1 Previous works of HOG feature extraction processor.

2. ALGORITHM

A simplified HOG algorithm for hardware implementation is presented in Fig. 2. This algorithm is modified from the

original algorithm [1]. The flow in Fig. 2 is based on our previous work [10]. We have confirmed that simplified HOG algorithm reduces the implementation cost maintaining the same detection accuracy as original one on a Detection Error tradeoff simulation [10]. An early classification method is newly introduced to our previous algorithm. A simplified HOG algorithm employs the following seven techniques.

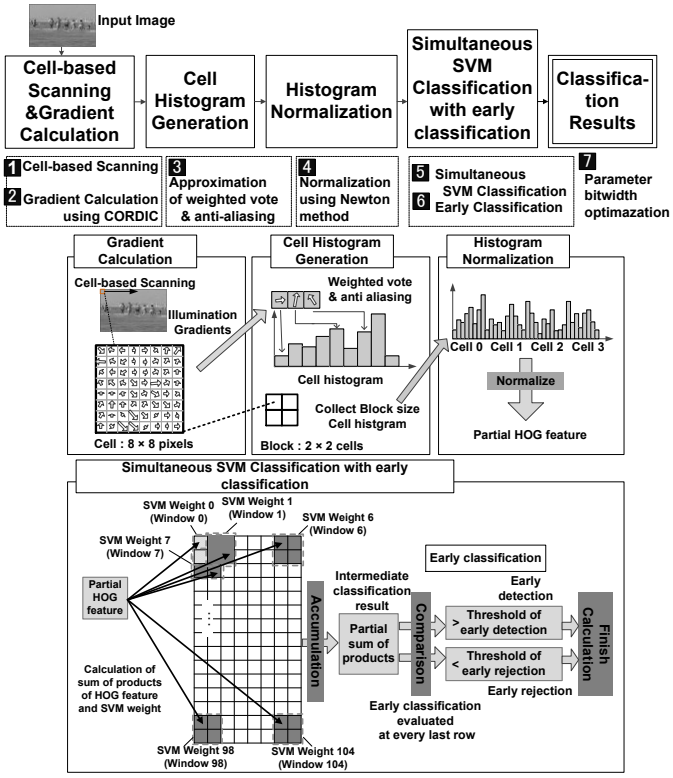

Fig. 2 Simplified HOG algorithm flow.

1. Cell-based scanning

Scanning of an input image is based on local regions called cells (8 × 8 pixels). HOG features are extracted from cell-based calculations. No cell overlaps with other cells. Sharing and reusing of a cell are vital for memory bandwidth reduction.

2. Gradient calculation using CORDIC [11]

3. Approximation of weighted voting for spatial and orientation anti-aliasing

4. Newton method with approximated initial values

5. Simultaneous SVM calculation

Figure 2 presents simultaneous SVM calculations for cell-based processing. Partial HOG features, which belong to 105 windows maximally, are located at different positions in each window. Partial HOG features are multiplied by the SVM coefficients of each window and accumulated. The accumulation result is stored and reused in subsequent SVM calculations. Simultaneous SVM calculation is suitable for parallel computing in hardware.

6. Early classification with the accumulation results

Before finishing all SVM calculations, data can be treated as they are already classified if the intermediate classification result is sufficiently low (or high) as shown in Fig. 3. Thereby, subsequent calculations can be skipped. Classifications on early stages are evaluated 14 times per detection-window.

Pairs of preliminarily learned thresholds are used for the comparison. These represent possible misclassification intervals. The intervals are expressed as [μtar-4σtar, μnon+4σnon]. The means μtar, μnon and the standard deviations σtar, σnon are estimated, presuming that the intermediate classification result of a target class and a non-target class are normally distributed. Data within the two thresholds on an early stage are sent to the later stage and calculations are continued. Early classification reduces the number of computations by 22.3% from 6.34 GOPS to 4.92 GOPS without any degradation of classification accuracy, on a Area Under Curve simulation on INRIA dataset [13].

7. Parameter optimization

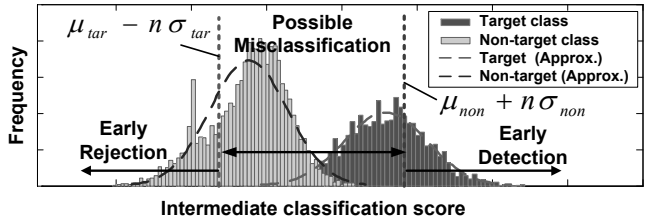

Fig. 3 Distribution of intermediate classification result.

3. ARCHITECTURE

3.1. Dual core architecture with cell-based pipeline

Figure 4 depicts a block diagram of the dual core architecture. The proposed architecture consists of two HOG feature extraction cores, a CPU interface, and a memory interface. The HOG feature extraction core comprises a controller, address generators, cell histogram generation module, histogram normalization module, SVM classification module, and working SRAMs. An external CPU controls the HOG processor. The input grayscale image is loaded from external SRAM to internal SRAM via a memory interface.

Each core can share HOG features and intermediate classification results with another. This structure enables the processor to operate in several modes described in Section 3.2 and Section 3.3.

The cell histogram generation module adopts four-way architecture because one cell is shared for four blocks. The cell histogram normalization module adopts two-stage architecture to implement L2-Hys normalization [13]. The architectures including these two modules are the same architectures as reported previously [10].

Our HOG processor architecture adopts a cell-based pipeline flow. Cell-based pipeline processing is conducted as follows:

1. A cell histogram is generated with cell-based scanning.

2. When the process described above reaches the block level (2 × 2 cells) and four cell histograms are collected, a block-level cell histogram is normalized. Then the block-level HOG feature is extracted.
3. Block-level HOG features and its paired SVM coefficients corresponding to each window are multiplied and accumulated.
4. A partial sum of products is compared with early classification thresholds. A window is classified in the early stage if the comparison condition is true.
5. An accumulation result of the entire window level is compared with the SVM threshold. All remaining windows are classified based on this comparison. Then the final detection result is obtained.

The window-based approach requires memory bandwidth of 55 Gbps for HDTV images. The cell-based pipeline architecture reduces it significantly to 0.499 Gbps, preventing reloading of input pixels in different detection windows. However, the cell-based architecture requires circuit area overhead for extra SRAMs to store intermediate cell histograms and classification results.

3.2. Parallel processing and operating modes

In our architecture, the required cycle count for object detection is reduced, sharing the workloads between the cores by dividing an image into two half-images for each core. Thereby, it achieves high-speed processing. In contrast, low-power processing is achieved by minimizing the operation frequency.

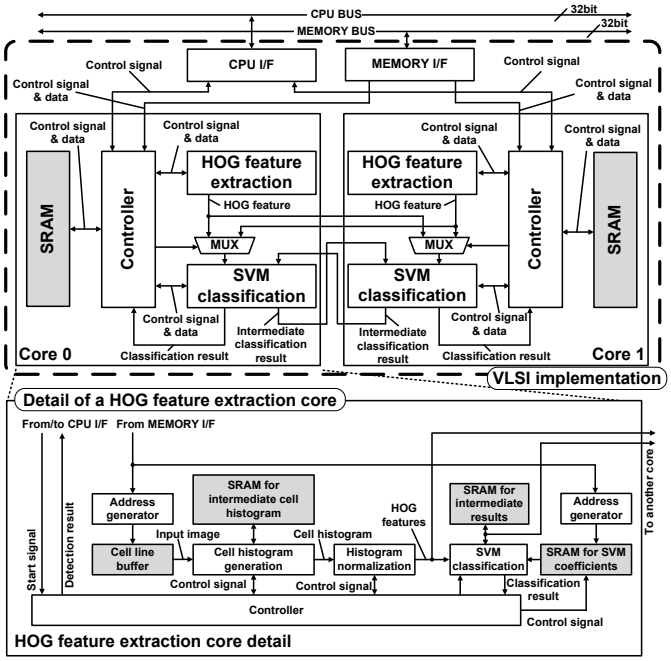

Fig. 4 Overall Architecture of the proposed HOG processor VLSI.

Detection for a single target object is insufficient for recent advanced applications. For example, on-vehicle applications must detect pedestrians, cars, and traffic signs. Conventional architecture requires another processor to detect another target. Furthermore, it wastes power for extraction of the same HOG feature, although feature extraction is the dominant part of the object detection task.

The SVM classification module in the proposed core contains an independent SRAM dedicated for SVM coefficients. Each core stores different SVM coefficients for different objects. Sharing HOG features to another core, feature extraction processes in one core can be turned off completely to reduce power consumption.

3.3. Detection-window-size scalable architecture

Our architecture also provides object detection for different shapes: square objects and vertically/horizontally long rectangular objects. The SVM classification module comprises a reconfigurable MAC array. A MAC array is reconfigured according to the target object shape. Each core processes each rectangular region. The detection of a square object is conducted with the cooperation of two processor cores. One core loads an intermediate result of another core as an initial value. Flexibility for multiple object detection is provided by coordinated processing.

3.4. Performance evaluation of the architecture

The number of cycle counts for each calculation in HOG based object detection was estimated using a Verilog-HDL simulator. Comparison between the proposed architecture, our previous architecture [10] and architecture without parallelization is presented in Fig. 5.

The proposed architecture is superior to others on HDTV resolution. Results show that processing with two cores reduces the number of cycle counts significantly compared with our previous architecture. The reduction rates of histogram generation, histogram normalization, and SVM classification are, respectively, 48.5%, 42.5%, and 50%. In the proposed dual core architecture, the overall process requires 1.43×10^6 cycles per frame. Therefore, the proposed architecture can process HDTV resolution video at 30 fps with 42.9 MHz.

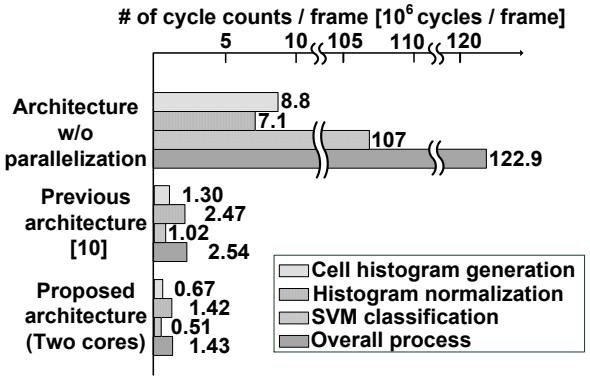

Fig. 5 Reduction of cycle count.

4. VLSI IMPLEMENTATION

A test chip has been designed as presented in Fig. 6. The design includes the VLSI-oriented algorithm and a dual core architecture. This chip, which was fabricated in 65 nm CMOS technology, occupies 4.2 × 2.1 mm^2 containing 502 Kgates and 1.22 Mbit on-chip SRAMs. The static timing analysis after back annotation shows that 110 MHz operation is attained at nominal supply voltage of 1.1 V. Chip specifications are presented in Table 1.

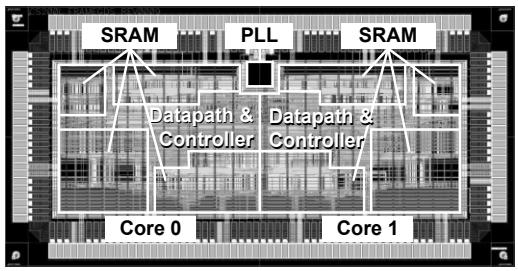

Fig. 6 Chip layout.

Table 1 Chip specifications

Technology	65 nm CMOS
Chip size	4.2 × 2.1 mm^2
Core size	3.3 × 1.2 mm^2
Power supply	1.1 V
Max frequency	110 MHz
Gate count	502 Kgates
Memory size	1.22 Mbit (610 Kbit for one core)
Image resolution	HDTV (1920 × 1080 pixels) @ 30 fps
Power	99.52 mW @ 42.9 MHz 1.1 V 40.30 mW @ 42.9 MHz 0.7 V (min)

5. APPLICATION FOR DETECTION SYSTEM

The proposed HOG processor consumes less than 100 mW. Therefore, it is suitable for mobile applications with limited power sources. As an example of the applications of proposed HOG processor, we have developed a pedestrian detection system. A block diagram of the system is presented in Fig. 7. The HOG processor receives image data and control signals from chip controller module on FPGA board at 25 MHz. After the processor finishes feature extraction and classification, a detection result is sent back to the FPGA. Then the result is displayed as a rectangle. On this system, HDTV resolution image is available for input. The demonstration system in our previous work [10] processes limited 800 × 600 resolution image and detects only a single target object. Because the proposed processor employs a dual-core architecture and a window-size scalable architecture, proposed demonstration system can detect not only pedestrians, but also other objects such as vehicles at once. Figure 8 is a sample image of the detection result of the demonstration system.

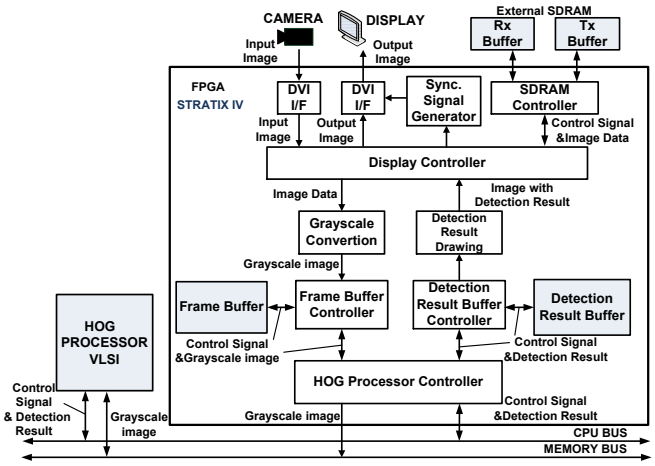

Fig. 7 Demonstration system.

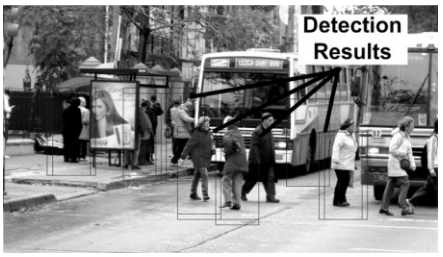

Fig. 8 Example of pedestrian detection with HOG processor.

6. CONCLUSION

A novel architecture and its VLSI implementation of real-time HOG feature extraction are proposed. The architecture employs a simplified HOG algorithm with early classification, a dual core architecture with a cell-based pipeline, and a detection-window-size scalable architecture. The early classification method achieves 22.3% reduction of the amount of computations in classification without accuracy degradation. The dual core architecture and detection-window-size scalable architecture reduce the required cycle count and power consumption. This architecture provides high flexibility for multiple object detection. A pedestrian detection system using the proposed processor is also presented. The proposed architecture is highly adaptable to multiple object detection. It satisfies the demands of recent advanced applications such as on-vehicle application and intelligent robots.

ACKNOWLEDGMENTS

The VLSI chip in this study has been fabricated in the chip fabrication program of VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with STARC, e-Shuttle, Inc., and Fujitsu Ltd. This research has been supported by the Semiconductor Technology Academic Research Center (STARC). This development was performed by the author for STARC as part of the Japanese Ministry of Economy, Trade and Industry sponsored “Silicon Implementation Support Program for Next Generation Semiconductor Circuit Architectures”.

7. REFERENCES

[1] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in Proceedings of the 2005 International Conference on Computer Vision and Pattern Recognition, vol. 2. Washington, DC, USA: IEEE Computer Society, pp. 886–893, 2005.

[2] Li Zhang and Ramakant Nevatia, “Efficient Scan-Window Based Object Detection using GPGPU,” IEEE, CVPRW, 2008.

[3] Sebastian Bauer, Ulrich Brusmann, Stefan, and Stefan Schlotterbeck-Macht, “FPGA Implementation of a HOG-based Pedestrian Recognition System,” MPC-Workshop, July, 2009.

[4] Masayuki Hiromoto, Ryusuke Miyamoto, “Hardware Architecture for High-Accuracy Real-Time Pedestrian Detection with CoHOG Features,” IEEE ICCVW 2009.

[5] R. Kadota, H. Sugano, M. Hiromoto, H. Ochi, R. Miyamoto, and Y. Nakamura, “Hardware Architecture for HOG Feature Extraction,” in Proceedings of the 2009 International Conference on Intelligent Information Hiding and Multimedia Signal Processing. Washington, DC, USA: IEEE Computer Society, pp. 1330–1333, 2009.

[6] Y. Yazawa, T. Yoshimi, T. Tsuzuki, T. Dohi and H. Fujiyoshi, “FPGA Hardware with Target-Reconfigurable Object Detector by Joint-HOG,” in Proceeding of SSII. Yokohama, Japan, 2011.

[7] K. Negi, K. Dohi, Y. Shibata, K. Oguri, “Deep pipelined one-chip FPGA implementation of a real-time image-based human detection algorithm,” IEEE FPT 2011.

[8] T. P. Cao and G. Deng, “Real-Time Vision-Based Stop Sign Detection System on FPGA,” in Proceeding of Digital Image Computing: Techniques and Applications. Los Alamitos, CA, USA: IEEE Computer Society, pp. 465–471, 2008.

[9] Sebastian Bauer, Sebastian Kohler, Konrad Doll, and Ulrich Brunsmann, “FPGA-GPU Architecture for Kernel SVM Pedestrian Detection,” IEEE CVPRW 2010.

[10] K. Mizuno, Y. Terachi, K. Takagi, S. Izumi, H. Kawaguchi and M. Yoshimoto, "Architectural Study of HOG Feature Extraction Processor for Real-time Object Detection", IEEE SiPS, Oct. 2012.

[11] J. E. Volder, “The CORDIC Trigonometric Computing Technique,” IRE Trans. Electron. Computers. EC-8:330-334, 1959.

[12] INRIA Person Dataset. http://pascal.inrialpes.fr/data/human/

[13] D. G. Lowe, “Distinctive image features from scale invariant keypoints,” International Journal of Computer Vision, Vol.60, No.2, pp.91-110, 2004.