iVisual: An Intelligent Visual Sensor SoC With 2790 fps CMOS Image Sensor and 205 GOPS/W Vision Processor

Chih-Chi Cheng, Chia-Hua Lin, Chung-Te Li, and Liang-Gee Chen, *Fellow, IEEE*

***Abstract*—iVisual, an intelligent visual sensor SoC integrating 2790 fps CMOS image sensor and 76.8 GOPS, 374 mW vision processor, is implemented on a 7.5 mm $\times$ 9.4 mm die in a UMC 0.18 μm CMOS Image Sensor process. Light-in, answer-out SoC architecture is adopted to avoid possible privacy problems. A feature processor is designed to eliminate the dataflow mismatch between processor array and scalar processor to increase 36% of average throughput. To increase hardware utilization, an inter-processor synchronization scheme is adopted to increase 23% of average throughput. Memory access is reduced by 94% to save 726 mW of power consumption. A bitplane-based single port memory structure is adopted to reduce SRAM area. The 205 GOPS/W power efficiency and 1.16 GOPS/mm^2 area efficiency are therefore achieved by use of the proposed techniques.**

***Index Terms*—GOPS, intelligent visual sensor, SIMD, single-instruction multiple-data, video analysis, vision processor.**

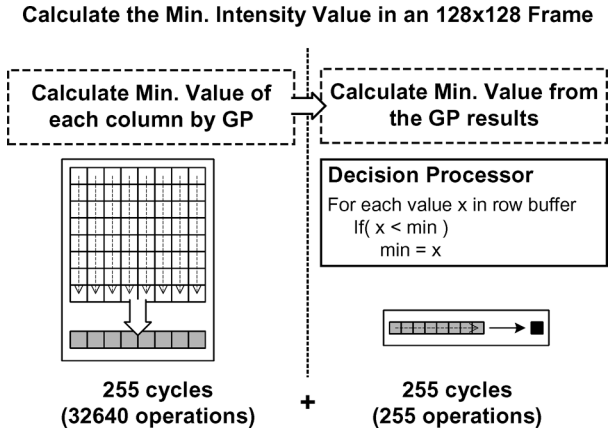

Fig. 1. An example illustrating the throughput bottleneck in vision processors.

I. Introduction

VISUAL sensors combined with video analysis technology can enhance applications in surveillance [1]–[4], healthcare [5], intelligent vehicle control [6], human-machine interface [7] and so on.

Due to the importance and the high computational complexity, hardware solutions exist for video analysis applications [8]–[10]. Analog on-sensor processing solutions [8] feature the integration of an image sensor and a 2-D parallel per-pixel processor array. However, the precision loss issues of analog signal processing prevent those solutions from realizing complex algorithms [11]. These solutions also lack flexibility, for they can handle only frame-in, frame-out operations. Vision processors [9], [10] provide more feasible solutions for handling complex algorithms. In those processors, a SIMD processor array is designed for parallel data in, parallel data out operations, and another separate processor, we call it decision processor, is designed for operations with other dataflows. High GOPS numbers are realized by increasing the parallelism of the processor array. However, the dataflow mismatch between the processor array that produces parallel data and the decision processor that consumes scalars induces a throughput bottleneck. Take XETAL-II [10] for example, the data production rate of the processor array can achieve 430 Gb/s. However, the data consumption rate of the decision processor (named GCP in [10]) is only 2.7 Gb/s. Furthermore, due to the massively-parallel processor array architecture, the memory access bandwidth in the vision processors is significant, and this leads to a high power consumption.

We illustrate the throughput bottleneck in vision processors with a simple example as shown in Fig. 1. The goal of this example is to calculate the minimum intensity value in a 128 $\times$ 128 video frame. This operation is commonly used in algorithms like histogram equalization, contrast enhancement and so on. The computation can be partitioned into two steps. The first step is to calculate the minimum intensity value of each column. This can be done in parallel by using the SIMD processor array scanning the image from top to bottom, and it takes 255 clock cycles to process these 32640 operations. The second step is to calculate the overall minimum intensity value from the minimum values of columns. Because this operation cannot be parallelized, the second step has to be handled by the decision processor sequentially. It still takes 255 clock cycles. However, only 255 operations are processed in this step. The overall system throughput is thus degraded due to this throughput bottleneck.

Privacy invasion is always a critical issue in setting up visual sensors around the living spaces because of the danger of revealing video data during the video analysis processing [12]. In an intelligent visual system with image sensors, frame buffers and vision processors, the video data can be easily revealed by monitoring the inter-chip traffic among those components. The

Manuscript received April 15, 2008; revised August 31, 2008. Current version published December 24, 2008. This work was supported by Himax Technologies Inc. under Grant 96-S-B19.

The authors are with the Graduate Institute of Electronics Engineering, National Taiwan University, Taipei, Taiwan (e-mail: ccc@video.ee.ntu.edu.tw; chlin@video.ee.ntu.edu.tw; zt@video.ee.ntu.edu.tw; lgchen@video.ee.ntu.edu.tw).

Digital Object Identifier 10.1109/JSSC.2008.2007158

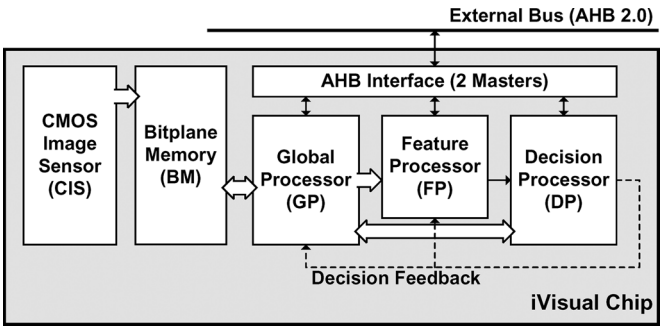

Fig. 2. The iVisual system architecture.

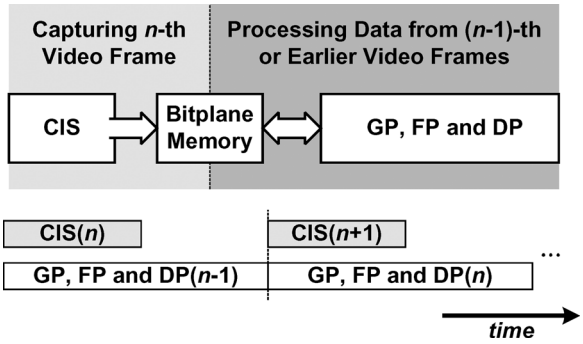

Fig. 3. The frame pipeline scheme in iVisual.

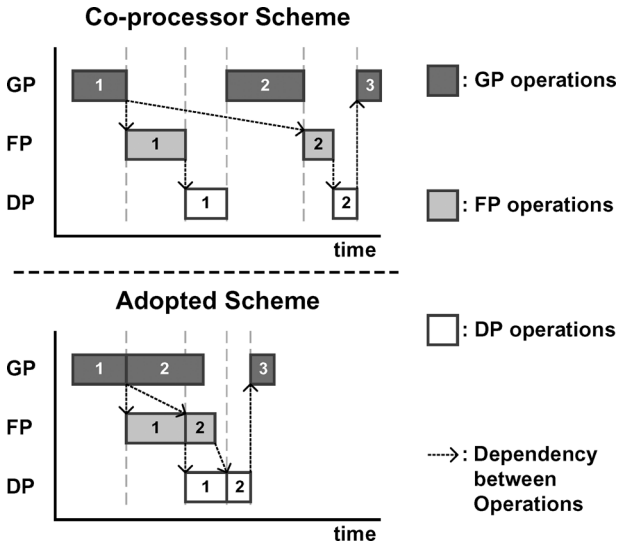

Fig. 4. The inter-processor synchronization scheme.

privacy issues thus exist with the above-mentioned solutions because of the inevitability of inputting or outputting video data.

iVisual [13] is characterized as follows:

1) High level of integration: iVisual is a light-in, answer-out SoC integrating CMOS image sensor, 76.8 GOPS vision processor and 1 Mb storage. No video data need to be revealed outside the chip during the video processing. The possible privacy problems are therefore avoided.
2) New vision processor architecture: feature processor eliminates the throughput bottleneck and increases 36% of average throughput. The inter-processor synchronization scheme ensures minimum communication between processors and further increases 23% of throughput.
3) High power/area efficiency: the 205 GOPS/W power efficiency is achieved by introducing feature processor, processing element (PE) register file and instruction-level gated clock. The 1.16 GOPS/mm^2 area efficiency is achieved by introducing feature processor, bitplane memory structure and reconfigurable storage allocation.

This paper is structured as follows. The top-level system architecture of iVisual is discussed in Section II. Section III presents the architecture designs of important modules. The physical design is described in Section IV, and Section V lists the measured results and chip features. Finally, Section VI concludes this work.

II. System Architecture

Fig. 2 shows the iVisual chip with five major parts: on-chip CMOS image sensor (CIS), bitplane memory (BM), global processor (GP), feature processor (FP) and decision processor (DP). CIS is a high frame-rate, low resolution CMOS image sensor with read-out circuits. The captured video data are buffered in BM. The main processing engine of GP is a SIMD processor array with 128 PEs. GP processes parallel data in, parallel data out operations. The main data memory of GP is BM. BM is thus both the output buffer of CIS and the data memory of GP. DP is a five-stage pipelined processor with MIPS-like instruction set and architecture. It handles scalar in, scalar out operations.

To reduce the throughput bottleneck induced by the dataflow mismatch between GP and DP mentioned in Section I, the FP is designed for iVisual. FP is a processor dedicated for parallel data in, scalar out operations to eliminate the dataflow mismatch. As shown in Fig. 2, the signals in DP can also be sent to GP and FP to control their program execution or change their data. GP, FP, DP, and BM together form the vision processor in iVisual.

A. Frame Pipeline Scheme

The exposure of CIS might take a long period of time, depends on the environmental lighting. If the three processors (GP, FP, and DP) have to wait for the exposure before processing every video frame, the idle time of processors might be long.

To hide this latency, CIS is frame-pipelined with the vision processor to increase hardware utilization as shown in Fig. 3. The bottom part of Fig. 3 shows the corresponding hardware scheduling. When CIS is capturing nth video frame, the three processors are processing $(n-1)$th or earlier video frames.

B. Inter-Processor Synchronization Scheme (IPSS)

To increase the system throughput, the program executions of GP, FP, and DP are nearly mutually independent. However, the correct inter-processor data dependencies are still maintained. Fig. 4 compares the adopted inter-processor synchronization scheme (IPSS) and the co-processor scheme [9]. Each block represents a group of operations, and different colors represent operations of different processors. The arrows represent the dependencies between two groups of operations of different processors.

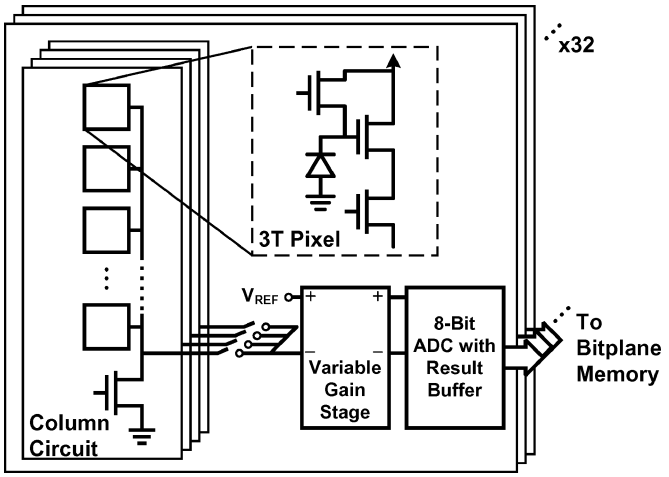

Fig. 5. The pixel circuits and read-out architecture of CIS.

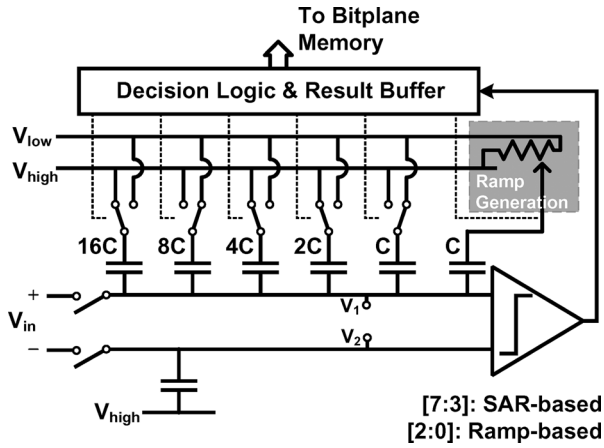

Fig. 6. The ADC circuits of CIS.

In the co-processor scheme, there is only one processor active at a time. This can keep the program control simple with the price of long processor idle time as shown in the upper part of Fig. 4. In the adopted IPSS, for each instruction, each processor checks the required hardware resources of the current instruction. A processor will stop its execution only when it requires hardware resources from other processors, and those required resources are not yet available. A simple hand shaking protocol thus exists among processors to communicate the availability of resources. On average, the adopted IPSS can increase 23% of throughput compared with the co-processor scheme due to the increase of processor utilization. The benchmark adopted to estimate the throughput increase is a posture classification algorithm. For each instruction, the clock signal in the hardware resources not required are turned off to save power.

The increase of hardware utilization might result in an increase of peak power. However, in iVisual, most power are consumed by GP and BM, and FP and DP together consume only 11.8% of total power consumption. Therefore, the peak power increase will not induce problems in physical design.

III. Module Architecture Design

A. CMOS Image Sensor and the Read-Out Circuits

Fig. 5 shows the pixel circuits and read-out architecture of CIS. The 3T structure is adopted for pixel cells. The cell area is enlarged to 7 μm $\times$ 7 μm to enable high frame-rate capturing. A parallel read-out architecture is adopted. One set of read-out circuits is shared by four pixel columns. It is theoretically proved that high gain read-out circuits can effectively increase the SNR [14]. We built such a gain stage with four adjustable gains.

For the ADC design, SAR-based architecture [15] and ramp-based architecture [16] are combined. As the bidwidth of converted samples increases, the SAR-based architecture enjoys a linear increase of conversion steps with the price of exponential area increase; on the contrary, the ramp-based architecture enjoys a linear area increase with the price of exponential increase of conversion steps. In the adopted 8-bit ADC, the conversion of the five most significant bits is based on SAR ADC scheme, and the conversion of the least significant three bits is based on the ramp-based scheme. Compared with the conventional SAR architecture, the required cycle count for one sample conversion is increased from 18 cycles to 20 cycles per sample, while ADC area is reduced by 48.1% due to the reduction of capacitor array area. Further increasing the number of bits processed with the ramp-based approach will result in a greater degradation of conversion speed. For example, if four bits are processed with SAR approach and four bits are processed with ramp-based approach, the area can be further reduced by 7.1%. However, the conversion speed will be reduced by 30%. The overall read-out process takes 35 clock cycles per column per row, and a set of read-out circuitry is shared by four columns. With the rolling shuttle scheduling and a 50 MHz clock rate, 2790 fps peak frame rate can be achieved.

Both exposure time and read-out speed influence the frame rate. Enlarged sensor cells and pre-ADC gain stage reduce the exposure time, and the parallel read-out architecture reduces the read-out time. In our read-out scheme, if the exposure time is less than 355 μs, the overall frame rate will be determined by the read-out speed.

B. Global Processor

Fig. 7 shows the architecture of GP. GP handles parallel data in, parallel data out operations. The input data come from BM, and the output data can be written back to BM or sent to FP. DP signals can be inputted to control the program execution of GP. There is also a high bandwidth data link between DP and GP to communicate parallel data.

The main execution unit of GP is a SIMD processor array with 128 PEs. 51 instructions are designed for GP. To reduce pipeline control circuits, the PEs are not pipelined. Each PE has a unique index and its own conditional control circuits to enhance the flexibility. Compound instructions are utilized for logic and arithmetic instructions. A conditional control, input/output bitwidth control, input/output padding and an arithmetic operation can be processed in a clock cycle.

In previous works [9], [10], the PEs read and write the data memory every clock cycle. Because there are 128 PEs in a SIMD array, this parallel memory access induces a high memory access power according to our analysis. To reduce the power consumption, an additional memory hierarchy, named PE register

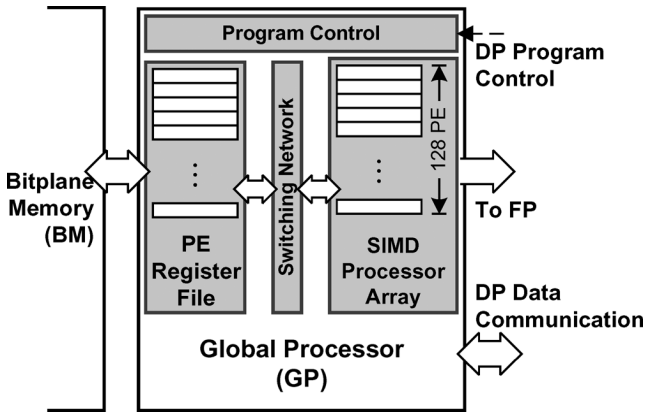

Fig. 7. The global processor (GP).

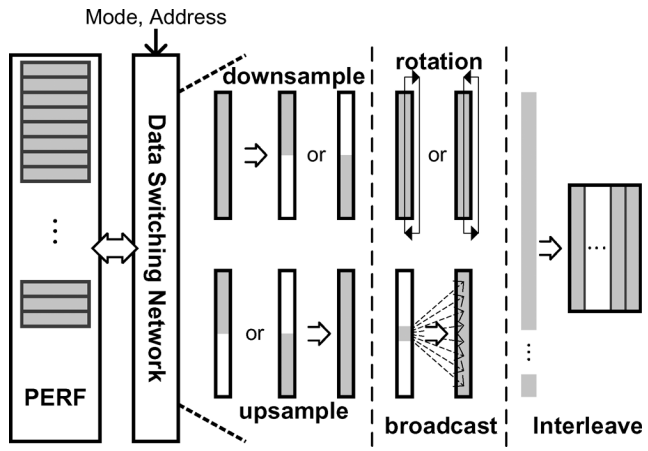

Fig. 8. The data reorganization scheme in GP.

file (PERF), is introduced. The PERF lies between the PE array and BM as shown in Fig. 7. The intermediate results during the video analysis processing can be buffered in PERF rather than BM to reduce access frequency of BM, and 94% of BM access is reduced on average. The benchmark used to estimate the BM access reduction ratio is the posture classification algorithm. This memory access reduction is estimated to be 726 mW. The PERF itself consumes 134 mW.

Different operations may require different types of data organization. A single-cycle data reorganization unit is embedded in the data switching network of GP. Fig. 8 shows this data reorganization scheme in GP. Four modes are supported: data downsample/upsample, data rotation, data interleaving and data broadcasting.

C. Bitplane Memory

In video analysis algorithms, video data with different bitwidths usually appear. For example, a video object mask takes only one bit per pixel. In these cases, the byte-aligned storage structure is wasting. To increase the storage density, bitplane-based physical structure is adopted in iVisual as shown in Fig. 9. The grey regions in Fig. 9 represent an example showing video data with three bits per pixel stored in the bitplane

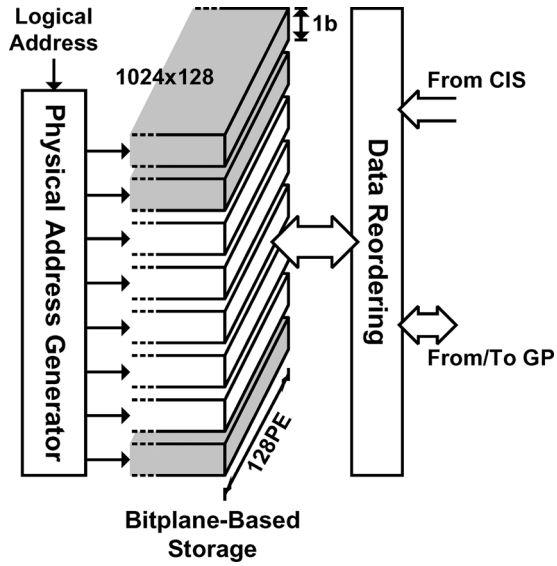

Fig. 9. The physical structure of the bitplane memory.

memory. Each memory bank can store eight video frames with one bit per pixel. The GP/CIS provides logical address and the bitwidth of the video data to be accessed. The corresponding physical address for each memory bank will then be automatically generated by hardware, and the video data will also be reordered to the correct sequence. According to a benchmark of motion object segmentation, the bitplane-based structure can reduce 41% of SRAM requirement. This benchmark includes connected component extraction, different types of FIR filtering operations, dilations and erosions. Because the bitwidths of data vary during the processing, the bitplane-based storage structure can effectively reduce the storage requirement.

BM is both the output buffer of CIS and the data memory of GP as mentioned in Section II. To reduce 64% of SRAM area, the data ports of BM are shared by both CIS and GP. The data port collision conditions are automatically handled by hardware. Thanks to the PERF reducing 94% of BM access from GP, the port collision probability is below 0.1%.

The bitplane-based physical structure reduces the SRAM area from 41 mm^2 to 24 mm^2, and the data port sharing further reduces the SRAM area from 24 mm^2 to 9 mm^2. The above two techniques together save 43% of the total are of the vision processor.

Storage requirements of 20 representative algorithms about video analysis and image enhancement are analyzed, and the 1 Mb storage size combining with the bitplane-based structure is concluded to be sufficient to handle all analyzed algorithms. In case of insufficient storage, off-chip storage can be utilized through use of the embedded AHB 2.0 master interface [17].

D. Feature Processor and Decision Processor

FP is a processor dedicated for parallel data in, scalar out operations. It is mentioned in Section I that there was a throughput bottleneck between the processor array and the decision processor due to the dataflow mismatch between them. FP is therefore designed to eliminate this throughput bottleneck. The input

TABLE I
ILLUSTRATION OF THE FP INSTRUCTION SET

Feature Extraction Operations (17 Instructions)	
Operation Description	Application Examples
Sum of input samples with 8-bit or 16-bit input data (signed or unsigned)	[1] [2] [3] [4] [5] [6] [19]
Bitwise Logic Operations	[1] [2] [3] [4] [5] [6]
Count the number of enabled samples	[1] [2] [3] [4]
Extract the minimum or maximum value among input samples (signed or unsigned)	[1] [2] [3] [4] [5] [6]
Extract the index of the input sample with minimum or maximum value (signed or unsigned)	[1] [4] [19]
Count the number of samples with value in certain range (signed or unsigned)	[3] [4] [19]
...	
Data Manipulation (7 Instructions)	
Operation Description	
Set value of specified input sample	
Shift of all input samples	
Change processing mode between 8-bit and 16-bit input bitwidth	
Padding of input samples	
Clear or enable all enable signals	
...	
Execution Control (9 Instructions)	
Operation Description	
Jump / Jump if results not zero / Jump according to external signal value	
Set break points / break if external signal enabled	
Wait for external signals	
Non-operation / end of file	
...	

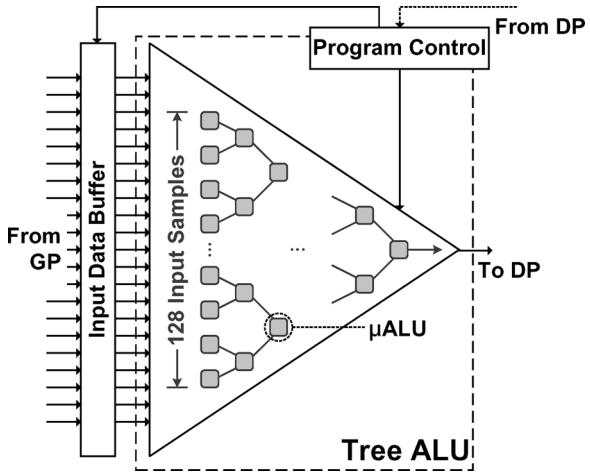

Fig. 10. The FP architecture.

data are the parallel data from GP, and the output is the scalar sent to DP as shown in Fig. 10.

Table I summarizes the instruction set of FP. 17 feature extraction instructions are designed after analyzing the feature extraction operations of video analysis algorithms and the Intel OpenCV library [19]. For example, the index of the minimum input sample can be extracted for calculating object bounding box; the number of input samples with value within a certain range can be extracted for calculating color histograms. Another 16 instructions are designed for data manipulation and program execution control.

To enable the object-based video analysis, the input samples can be enabled or disabled by appending an enable bit on each input sample. When extracting features of a video object, the object mask can be inputted as enable bits, and the FP will extract only the information of the specific object. To further enhance the flexibility, input data can be configured as 8-bit or 16-bit data, signed or unsigned data.

Fig. 10 shows the FP architecture. The ALU is designed as μALUs connected in a tree structure rather than a sequential structure to ensure a short timing path. According to our implementation results, a 128-to-1 feature processor with 16-bit data bitwidth enjoys a timing path shorter than an 8-bit by 8-bit multiplier.

We compare the throughput between vision processor with FP and without FP by using the minimum intensity value example mentioned in Section I. The goal of this example is to calculate the minimum intensity value in a 128 $\times$ 128 video frame. We can use GP to calculate the minimum intensity value of each column in parallel, and it takes 255 clock cycles. The second step is to calculate the overall minimum intensity value from the minimum values of columns. Without FP, this has to be handled by DP processing sequentially, and it takes 255 clock cycles. With FP, however, the second step is a single cycle instruction.

DP is a 32-bit five-pipelined processor with MIPS-like instruction set and out-of-order control on instructions involving inter-processor communication. The register file of DP is enlarged to access the parallel data in GP through a high bandwidth link, and 256 Byte data can be communicated between GP and DP in a single cycle. The DP can also control the program execution of GP and FP.

IV. PHYSICAL DESIGN

The design flow of iVisual comprises a full-custom design flow, a cell-based design flow and a flow for co-simulation and layout-merging. FPGA-workstation co-simulation is also included for the simulation of huge test pattern.

A software system model including external traffic is firstly built for performance estimation. The specifications of CIS and vision processor are defined in this step.

CIS is designed with the full-custom flow, and the vision processor is designed with the cell-based design flow. In addition to being designed separately, the CIS and vision processor are also simulated together during the design process. The pre-layout

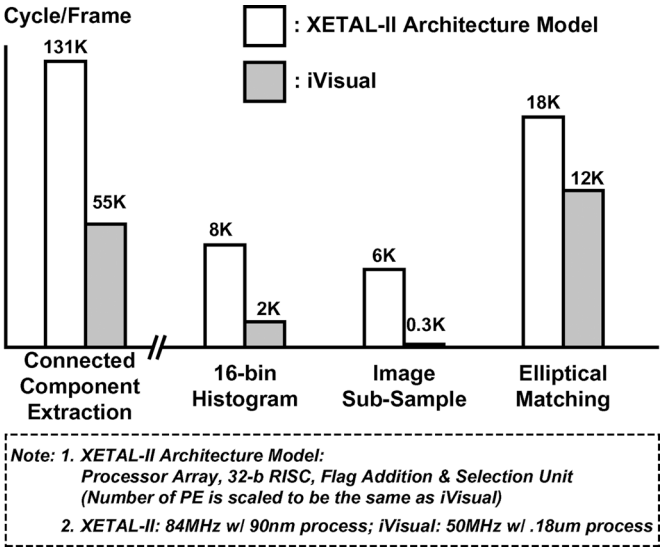

Fig. 11. An architectural comparison between iVisual and the state of the art.

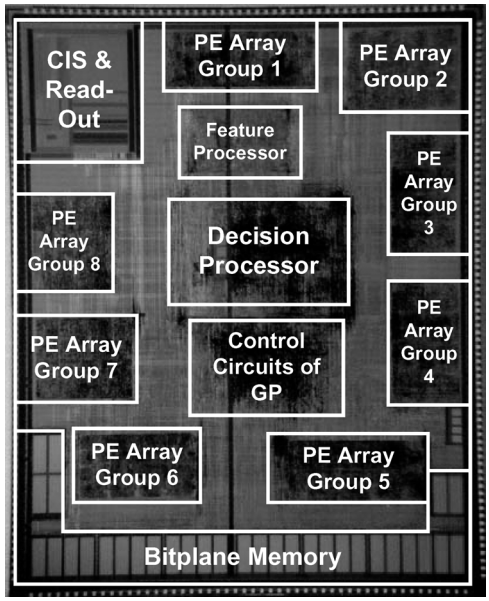

Fig. 12. The chip micrograph of iVisual.

and post-layout transistor-level netlists are simulated with the RTL and gate-level Verilog netlists by mixed-signal co-simulation tools to ensure the functionality and performance.

Due to the huge bandwidth of the parallel data in the processor array, signal routing area is an important issue in the place and route (P&R) phase. In the design of iVisual, a bottom-up P&R methodology is adopted. The GP is partitioned into eight PE groups. The whole design is firstly coarsely routed given the coarse floorplan of modules. The the PE group boundary information is then extracted such that each PE group can be placed and routed independently. The routed modules are then merged with necessary routing modification. Finally, the CIS layout, P&R information and the cell library are merged together.

23 design tools are utilized in the iVisual design process for simulation, verification, layout and so on.

V. Experimental Results

A. Architectural Comparisons

To make an architectural comparison with the state of the arts, we built an architecture model according to our knowledge of XETAL-II [10]. The model contains a processor array, a 32-bit RISC and a flag addition/selection unit. The number of PE is scaled to be the same as iVisual.

Fig. 11 shows such an architectural comparison of throughput in terms of required clock cycles per video frame. Connected component extraction is used in nearly all kinds of video object segmentation algorithms; intensity histogram is widely used in image enhancement and feature extraction; image subsampling is essential for multi-resolution object detection; elliptical matching is widely used for human detection and posture classification. Due to the introduction of FP, the throughput of connected component extraction, intensity histogram and elliptical matching can be increased as shown in the figure. The throughput increase of image subsampling comes from the introduction of the data reorganization unit in GP.

TABLE II
Chip Summary

Technology	UMC 0.18μm 2P4M CMOS Image Sensor Process
Core Area	7.5mm×9.4mm
Package	BGA 256pin
Working Frequency	50MHz
Supply Voltage	2.1V
Temperature	25°C
Power Consumption	CIS: 81mW Vision Processor: 374mW
Peak Throughput	76.8GOPS
Internal Buffer	1Mb
External Interface	Two AHB 2.0 Masters
CIS Resolution	128×128
Pixel Structure	3T Pixel Structure

B. Chip Implementation

iVisual is implemented on a 7.5 mm $\times$ 9.4 mm die in a UMC 0.18 μm two-poly four-metal (2P4M) CMOS image sensor process. Fig. 12 shows the chip micrograph. Table II shows the chip summary. iVisual embedded 1 Mb bitplane-structured on-chip storage and can achieve 76.8 GOPS of peak throughput. The average power consumption is 374 mW for vision processor and 81 mW for CIS. The adopted test pattern for power measurement contains image capturing, histogram equalization, image upsample/downsample and on-chip/off-chip video data access. The shmoo plot generated by the Agilent 93000 series mixed-signal SoC test platform is shown in Fig. 13.

C. Measured Throughput Data

Fig. 14 summarizes the measured CMOS image sensor responses with different read-out gain settings. The solid lines are measured in an indoor environment with weak lighting (estimated to be 50 lux), and the dashed lines are measured in a dark environment. If the dark environment is perfectly dark, the dashed lines represent noise signals. As shown in the figure, the exposure time can be as short as 24.5 μs while maintaining enough SNR through use of the enlarged sensor cells and the

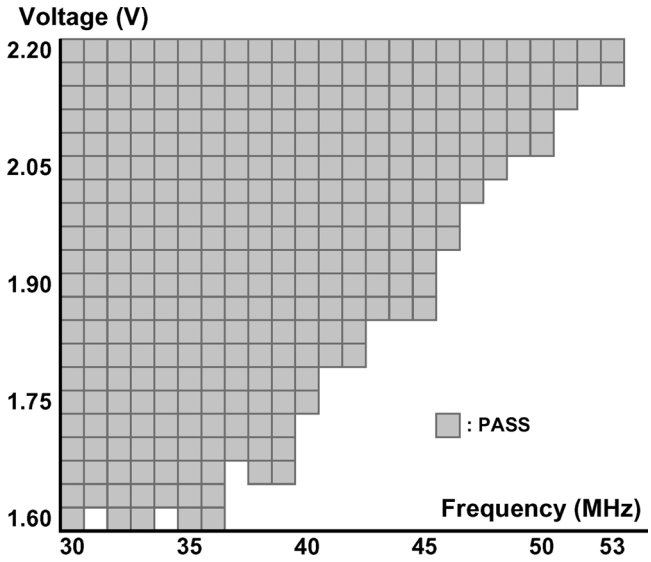

Fig. 13. The shmoo generated by the Agilent 93000 mixed-signal SoC test system.

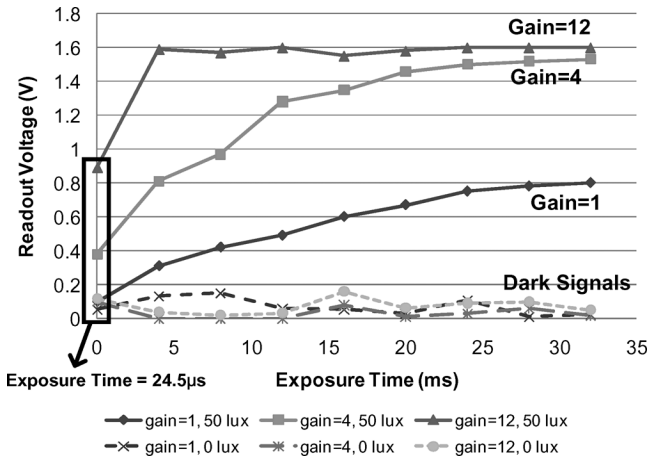

Fig. 14. The measured responses of CMOS image sensor and gain stage.

pre-ADC gain stage to enhance the SNR. High frame rate imaging is thus achieved.

Table III summarizes the throughput of commonly used operations for video analysis algorithms. High throughput is achieved by FP that eliminates the throughput bottleneck and IPSS that maximizes the hardware utilization. To show the capability of iVisual while handling complex algorithms, a posture analysis algorithm is also illustrated in Fig. 15. This benchmark algorithm comprises a complete video analysis processing flow including video capturing, image enhancement, motion segmentation, human detection, human tracking and posture analysis. A throughput of 360 fps is achieved by iVisual while processing seven people simultaneously in the video scene.

Fig. 16 illustrates the throughput increase by use of the proposed techniques. The benchmark is the posture classification algorithm. The average throughput is increased by 36% by introducing the FP to eliminate the throughput bottleneck. The average throughput is further increased by 23% by introducing the IPSS to increase the hardware utilization.

TABLE III
iVISUAL THROUGHPUT OF COMMONLY-USED OPERATIONS

Operation Description	Throughput
Frame-Level Processing	
3×3 FIR Filtering	0.25cycle/pixel
Sobel Image Gradient	0.05cycle/pixel
Harris Corner Detector	2.2cycle/pixel
Image Erosion/Dilation	0.05cycle/pixel
3×3 Median Filtering	0.48cycle/pixel
Row-Based Pipeline Among GP, FP and DP	
6-Dimensional (Affine Model) Motion Estimation	7.95cycle/pixel
Object Bounding Box Extraction	0.06cycle/pixel
2-D Projective Histogram	0.03cycle/pixel
Histogram Equalization	0.07cycle/pixel
16-Bin Intensity Histogram	0.13cycle/pixel
Frame-Based Pipeline Among GP, FP and DP	
Elliptical Matching	0.74cycle/pixel
Connected Component Extraction	3.38cycle/pixel
Horn Optical Flow Calculation	5.23cycle/pixel
Integral Image on 3 Resolutions	0.88cycle/pixel
Object Area Extraction	0.02cycle/pixel

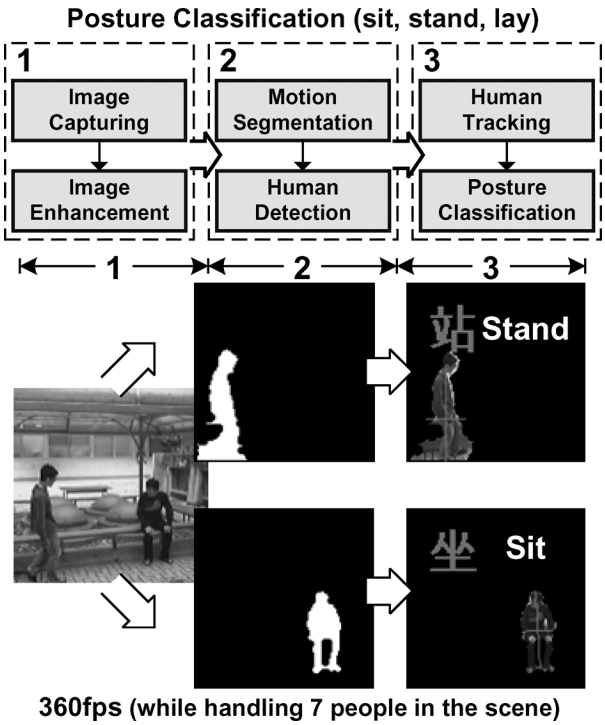

Fig. 15. The iVisual throughput while handling posture classification algorithm.

D. Measured Power Efficiency and Area Efficiency

Fig. 17 shows the impact made by the proposed techniques on the power efficiency and area efficiency. FP can increase 50% of peak throughput while consuming only 5% of total power consumption because of its dedicated structure. The power efficiency and the area efficiency thus is increased by 50% by introducing FP. The PERF reduces the power consumption of vision processor by 62%, and the power efficiency is thus further increased. The bitplane-based physical structure and the sharing of data ports reduces 43% of vision processor area, and the area efficiency is thus increased.

Fig. 18 shows the comparisons between iVisual and the state of the arts on power efficiency and area efficiency with process

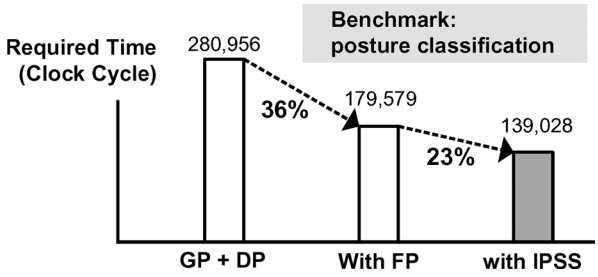

Fig. 16. The throughput increase induced by the proposed techniques.

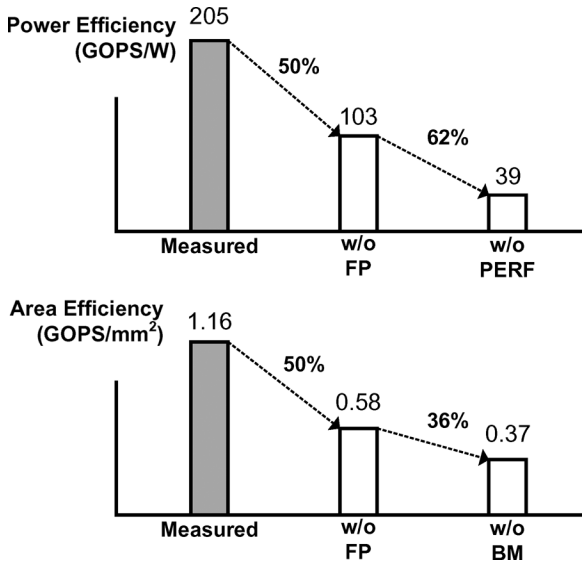

Fig. 17. The impact made by the proposed techniques on power efficiency and area efficiency.

scaling on [10]. The power scaling is done according to the dynamic power equation since leakage power in [10] occupies only 0.3% of its total power. The power efficiency and area efficiency are greatly increased because of the proposed techniques and a more dedicated processor dataflow. In the power efficiency comparison, the clock frequency doesn't matter since both throughput and power are proportional to the clock frequency.

VI. Conclusion

iVisual is an intelligent visual sensor SoC with a light-in, answer-out architecture to avoid possible privacy problems. On average, 51% of throughput is increased by use of the proposed vision processor architecture. Power consumption is reduced by reducing 94% of memory access; SRAM area is also reduced by introducing the bitplane-based memory structure and sharing the data ports. High power efficiency and high area efficiency are therefore achieved.

Acknowledgment

The authors thank Peter Chang, Samuel Chang, and the UMC University Program for support on chip manufacturing.

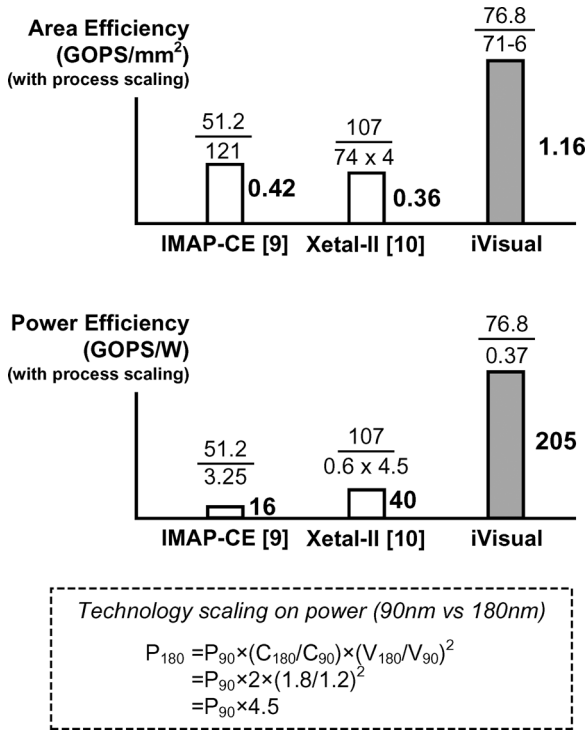

Fig. 18. The comparisons between iVisual and state of the arts on power efficiency and area efficiency.

References

[1] T. Zhao and R. Nevatia, "Tracking multiple humans in complex situations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 9, pp. 1208–1221, Sep. 2004.

[2] L. Wang, "From blob metrics to posture classification to activity profiling," in *18th IEEE Int. Conf. Pattern Recognition*, 2006, vol. 4, pp. 736–739.

[3] S. J. McKenna, S. Jabri, Z. Duric, H. Wechsler, and A. Rosenfeld, "Tracking groups of people," *Computer Vision and Image Understanding: CVIU*, vol. 80, no. 1, pp. 42–56, 2000.

[4] P. Guha, A. Mukerjee, K. S. Venkatesh, and P. Mitra, "Activity discovery from surveillance videos," in *Proc. 18th IEEE Int. Conf. Pattern Recognition*, 2006, vol. 1, pp. 433–436.

[5] S.-G. Miaou, P.-H. Sung, and C.-Y. Huang, "A customized human fall detection system using omni-camera images and personal information," in *Proc. 1st Transdisciplinary Conf. Distributed Diagnosis and Home Healthcare*, 2006, pp. 39–42.

[6] J. Wu and M. M. Trivedi, "Visual modules for head gesture analysis in intelligent vehicle systems," in *Proc. 2006 IEEE Intelligent Vehicles Symp.*, 2006, pp. 13–18.

[7] P. Robertson, R. Laddaga, and M. Van Kleek, "Virtual mouse vision based interface," in *Proc. 9th Int. Conf. Intelligent User Interfaces*, 2004, pp. 177–183.

[8] R. M. Philipp *et al.*, "A 128 x 128 33 mw 30 frames/s single-chip stereo imager," in *IEEE ISSCC Dig. Tech. Papers*, 2006, pp. 506–507.

[9] S. Kyo, T. Koga, S. Okazaki, R. Uchida, S. Yoshimoto, and I. Kuroda, "A 51.2 GOPS scalable video recognition processor for intelligent cruise control based on a linear array of 128 4-way VLIW processing elements," in *IEEE ISSCC Dig. Tech. Papers*, 2003, vol. 1, pp. 48–49.

[10] A. Abbo, R. Kleihorst, V. Choudhary, L. Sevat, P. Wielage, S. Mouy, and M. Heijligers, "XETAL-II: A 107 GOPS, 600 mW massively-parallel processor for video scene analysis," in *IEEE ISSCC Dig. Tech. Papers*, 2007, vol. 1, pp. 270–271.

[11] P. Dudek and P. J. Hicks, "A general-purpose processor-per-pixel analog SIMD vision chip," *IEEE Trans. Circuits Syst. I*, vol. 52, no. 1, pp. 13–20, Jan. 2005.

[12] A. J. Lipton *et al.*, "The intelligent vision sensor: Turning video into information," in *Proc. 2007 IEEE Int. Conf. Video and Signal Based Surveillance*, 2007, pp. 63–68.

[13] C.-C. Cheng, C.-H. Lin, C.-T. Li, S. Chang, C.-J. Hsu, and L.-G. Chen, "iVisual: An intelligent visual sensor SoC with 2790 fps CMOS image sensor and 205 GOPS/W vision processor," in *IEEE ISSCC Dig. Tech. Papers*, 2008, pp. 306–307.
[14] N. Kawai and S. Kawahito, "Noise analysis of high-gain, low-noise column readout circuits for CMOS image sensors," *IEEE Trans. Electron Devices*, vol. 51, no. 2, pp. 185–194, Feb. 2004.
[15] I. Takayanagi *et al.*, "A 1.25-inch 60-frames/s 8.3-m-pixel digital-output CMOS image sensor," *IEEE J. Solid-State Circuits*, vol. 40, no. 11, pp. 2305–2314, Nov. 2005.
[16] S. Yoshihara *et al.*, "A 1/1.8-inch 6.4 Mpixel 60 frames/s CMOS image sensor with seamless mode change," in *IEEE ISSCC Dig. Tech. Papers*, 2006, pp. 492–493.
[17] AMBA Specification. [Online]. Available: http://www.amba.com
[18] W.-C. Kao, S.-H. Wang, L.-Y. Chen, and S.-Y. Lin, "Design considerations of color image processing pipeline for digital cameras," *IEEE Trans. Consumer Electron.*, vol. 52, no. 4, pp. 1144–1152, Nov. 2006.
[19] Open Source Computer Vision Library. [Online]. Available: http://www.intel.com/technology/computing/opencv/

Chih-Chi Cheng was born in Taipei, Taiwan, R.O.C., in 1982. He received the B.S. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, in 2004. He is currently pursuing the Ph.D. degree at the Graduate Institute of Electronics Engineering, National Taiwan University. His research interests include the algorithms and VLSI architectures of intelligent video signal processing and image/video coding.

Chia-Hua Lin was born in Hsinchu, Taiwan, in 1983. He received the B.S. degree in Electrical Engineering from National Taiwan University in 2006 and the M.S. degrees in Electronics Engineering from National Taiwan University in 2008. His research interests include the algorithms and VLSI architectures of intelligent video signal processing and image/video coding.

Chung-Te Li was born in Taipei, Taiwan, R.O.C., in 1984. He received the B.S. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, in 2006. He is currently pursuing the Ph.D. degree at the Graduate Institute of Electronics Engineering, National Taiwan University. His research interests include the algorithms and VLSI architectures of 3D-related image processing.

Liang-Gee Chen was born in Yun-Lin, Taiwan, in 1956. He received the BS, MS, and Ph.D degrees in Electrical Engineering from National Cheng Kung University, in 1979, 1981, and 1986, respectively.

He was an Instructor (1981–1986), and an Associate Professor (1986–1988) in the Department of Electrical Engineering, National Cheng Kung University. In the military service during 1987 and 1988, he was an Associate Professor in the Institute of Resource Management, Defense Management College. From 1988, he joined the Department of Electrical Engineering, National Taiwan University. During 1993 to 1994 he was Visiting Consultant of DSP Research Department, AT&T Bell Lab, Murray Hill. At 1997, he was the visiting scholar of the Department of Electrical Engineering, University, of Washington, Seattle. Currently, he is Professor of National Taiwan University. From 2004, he is also the Executive Vice President and the General Director of Electronics Research and Service Organization (ERSO) in the Industrial Technology Research Institute (ITRI). His current research interests are DSP architecture design, video processor design, and video coding system.

Dr. Chen is a Fellow of IEEE. He is also a member of the honor society Phi Tan Phi. He was the general chairman of the 7th VLSI Design CAD Symposium. He is also the general chairman of the 1999 IEEE Workshop on Signal Processing Systems: Design and Implementation. He serves as Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from June 1996 until now and the Associate Editor of IEEE TRANSACTIONS ON VLSI SYSTEMS from January 1999 until now. He was the Associate Editor of the Journal of Circuits, Systems, and Signal Processing from 1999 until now. He served as the Guest Editor of The Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology, November 2001. He is also the Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: ANALOG AND DIGITAL SIGNAL PROCESSING. From 2002, he is also the Associate Editor of Proc. IEEE.

Dr. Chen received the Best Paper Award from ROC Computer Society in 1990 and 1994. From 1991 to 1999, he received Long-Term (Acer) Paper Awards annually. In 1992, he received the Best Paper Award of the 1992 Asia-Pacific Conference on Circuits and Systems in VLSI design track. In 1993, he received the Annual Paper Award of Chinese Engineer Society. In 1996, he received the Outstanding Research Award from NSC, and the Dragon Excellence Award for Acer. He is elected as the IEEE Circuits and Systems Distinguished Lecturer from 2001–2002.