Integration of Image Capture and Processing – Beyond Single Chip Digital Camera

SukHwan Lim and Abbas El Gamal

Information Systems Laboratory
Department of Electrical Engineering, Stanford University, CA 94305, USA

ABSTRACT

An important trend in the design of digital cameras is the integration of capture and processing onto a single CMOS chip. Although integrating the components of a digital camera system onto a single chip significantly reduces system size and power, it does not fully exploit the potential advantages of integration. We argue that a key advantage of integration is the ability to exploit the high speed imaging capability of CMOS image sensors to enable new applications such as multiple capture for enhancing dynamic range and to improve the performance of existing applications such as optical flow estimation. Conventional digital cameras operate at low frame rates and it would be too costly, if not infeasible, to operate their chips at high frame rates. Integration solves this problem. The idea is to capture images at much higher frame rates than the standard frame rate, process the high frame rate data on chip, and output the video sequence and the application specific data at standard frame rate. This idea is applied to optical flow estimation, where significant performance improvements are demonstrated over methods using standard frame rate sequences. We then investigate the constraints on memory size and processing power that can be integrated with a CMOS image sensor in a 0.18μm process and below. We show that enough memory and processing power can be integrated to be able to not only perform the functions of a conventional camera system but also to perform applications such as real time optical flow estimation.

Keywords: Digital camera, CMOS image sensor, Integration, DSP architecture, High speed imaging

1. INTRODUCTION

Digital still and video cameras are rapidly becoming ubiquitous, due to the increasing demands of multimedia applications. A typical digital camera system today (see Figure 1) employs a CCD image sensor and several other chips for analog signal generation, A/D conversion, digital image processing and compression, control, interface, and storage. Using a CMOS image sensor, the chip count of a digital camera system can be reduced by integrating the analog signal generation, A/D conversion, and some of the control and image processing with the sensor on the same chip.[1–4] Commercially available PC camera chips now routinely integrate A/D conversion, gamma correction, exposure and gain control, color correction and white balance with a CMOS CIF and VGA size image sensor. As CMOS image sensors scale to 0.18μm processes and below, integration of the rest of the camera system becomes feasible resulting in true "camera-on-chip". Although integrating the camera system shown in Figure 1 onto a single chip can significantly reduce system size and power, it does not fully exploit the potential advantages of integration. In this paper we argue that a key advantage of integration is the ability to exploit the high speed imaging capability of CMOS image sensors. Several recent papers have demonstrated the high speed imaging capability of CMOS image sensors.[5–7] Krymski *et al.*[5] describe a 1024×1024 Active Pixel Sensor (APS) with column level ADC achieving frame rate of 500 frames/s. Stevanovic *et al.*[6] describe 256×256 APS with 64 analog outputs achieving frame rate of 1000 frames/s. Kleinfelder *et al.*[7] describe a 352×288 Digital Pixel Sensor(DPS) with per pixel bit parallel ADC achieving 10,000 frames/s or 1 Giga-pixels/s.

The high speed imaging capability of CMOS image sensors can benefit conventional camera systems by enabling more efficient implementations of several applications such as motion estimation,[8] video stabilization, and video compression, and of new applications such as multiple capture for enhancing dynamic

Correspondence: Email: sukhwan@stanford.edu, abbas@isl.stanford.edu; Telephone: 650-725-9696; Fax: 650-724-3648

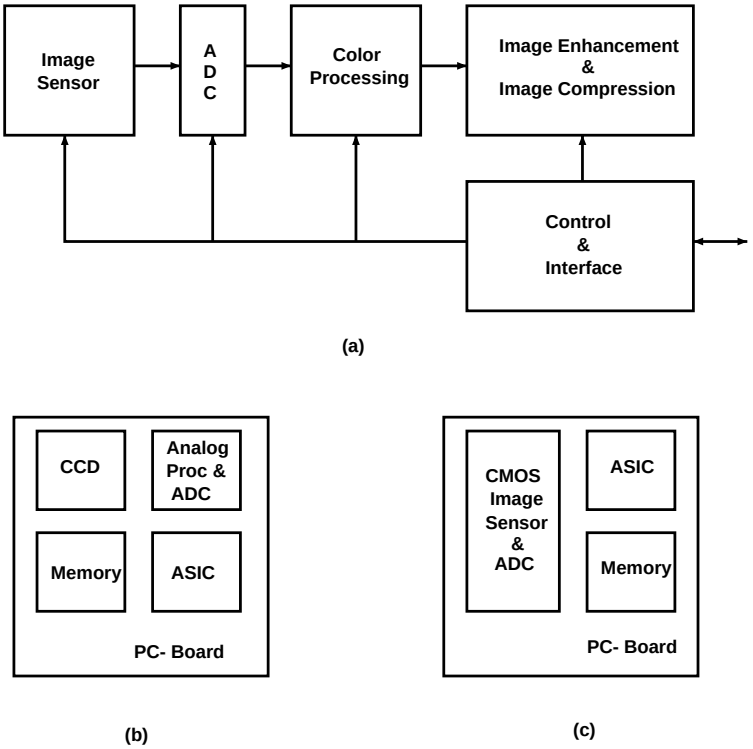


Figure 1. Digital Camera System: (a) functional block diagram, (b) implementation using CCD, and (c) implementation using CMOS image sensor.

range[9,10] and motion blur-free capture.[11] Digital still and video cameras, however, operate at low frame rates and it would be too costly, if not infeasible, to operate them at high frame rates due to the high output data rate requirements of the sensor, the memory, and the processing chips. Integrating the memory and processing with the sensor on the same chip solves the high output data rate problem and provides an economical way to exploit the high speed capability of a CMOS image sensor. The basic idea, which will be explored in this paper (see Figure 2 and Handoko *et al.*[8]), is to (i) operate the sensor at a much higher frame rate than the standard frame rate, (ii) exploit the high on-chip bandwidth between the sensor, the memory and the processors to process the high frame rate data, and (iii) only output the images with any application specific data at the standard frame rate. The on-chip memory and processing power, although increasing according to Moore's law, are quite constrained. These constraints in turn impose constraints on the types of algorithms that can be implemented.

In the following section we apply our basic idea to optical flow estimation. We demonstrate significant performance improvements over conventional optical flow estimation that use standard frame rate image sequences. In section 3 we investigate the constraints on memory size and processing power that can be integrated with a CMOS image sensor in 0.18μm process and below. We show that the integration of an entire video camera system on a chip is not only feasible at 0.18μm, but in fact underutilizes the possible on-chip processing power. Further, we show that the on-chip processing power and memory are sufficient to perform applications such as optical flow estimation by operating the sensor at high frame rate. As technology scales, applications that require even more processing power and memory such as tracking, pattern recognition, and 3D structure estimation can be implemented.

2. APPLICATION TO OPTICAL FLOW ESTIMATION

In this section we demonstrate the idea of using high speed imaging to improve the performance and efficiency of an existing application. The example we present is optical flow estimation, which provides the basis for

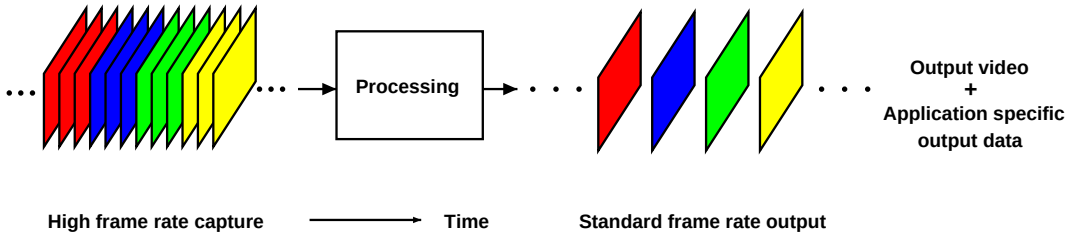


Figure 2. High frame rate capture – standard frame rate output.

many video applications, such as compression, 3D motion and structure estimation, and video stabilization. Thus, being able to perform optical flow estimation in real time would make it possible to perform many of these applications in a conventional digital video camera.

Optical flow estimation based on standard frame rate video sequences has been extensively researched.[12,13] The developed methods can be classified into several categories, including gradient-based, region-based matching, energy-based, Bayesian, and phase-based methods. These methods require storing many frames and performing large numbers of operations per pixel to achieve acceptable estimation accuracy. Moreover, many applications that use optical flow require greater accuracy than can be estimated with current methods.

The gradient based method by Lucas-Kanade was shown[12] to be among the most accurate and computationally efficient methods. This method is particularly attractive when applied to a high frame rate sequence for the following reasons.

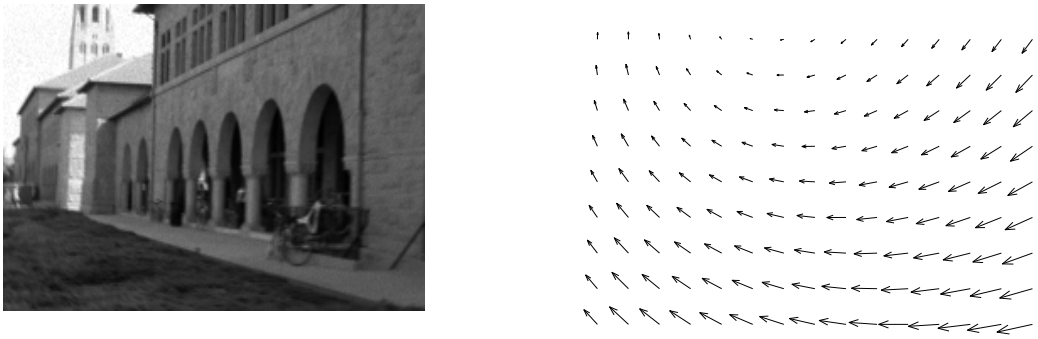
- The assumption of brightness constancy, which states that the rate of change in intensity along the motion trajectory is zero, becomes more valid as frame rate increases.
- Motion (temporal) aliasing, which adversely affects optical flow estimation, also becomes less significant as frame rate increases.
- Temporal derivatives can be more accurately estimated.
- Because of the smaller displacements between consecutive frames, smaller kernel sizes for smoothing and computing gradients can be used, which lowers the memory and computational requirements.

The authors[14] describe an optical flow estimation method that obtains high accuracy optical flow estimation between consecutive high speed frames based on the Lucas-Kanade method and then accumulates and refines these optical flow estimates to find an estimate of the optical flow betwen two consecutive standard frame rate images. Using 5×5 kernels, the algorithm requires approximately 190 operations/pixel·frame and 12 bytes of frame memory per pixel.

To demonstrate the high accuracy obtained by our algorithm, we apply it to a synthetically generated 120 frames/s video sequence of a natural outdoor image (see Figure 3(a)). Perspective warping was applied to the image and noise and motion blur were added. Three differently warped sequences were generated. The horizontal and vertical components of the maximum velocity were limited to no more than 1 pixels/frame. The true optical flow is shown in Figure 3(b). Table 1 lists the average angular error between the optical flow computed using our algorithm and the true optical flow for the three sequences. For comparison, the table also lists the average angular error when the Lucas-Kanade method is applied to 30 frames/s versions of the three sequences. The density of optical flow is equal for both methods and is around 50%.

3. MEMORY AND PROCESSING INTEGRATION LIMITS

In this section we explore the limits of integrating memory and processing with a CMOS image sensor in 0.18μm process and below. Our purpose is to demonstrate that only integrating the camera system in Figure 1 underutilizes the possible on chip processing power and to show that applications such as optical flow estimation can be performed on a single chip imaging system.



(a) **(b)**

Figure 3. (a) One frame of the test sequence and (b) true optical flow.

Scene	Lucas-Kanade method(A)		Our method(B)	
	Angular error	Density	Angular error	Density
1	4.43°	55.0%	3.43°	55.7%
2	3.94°	53.0%	2.91°	53.4%
3	4.56°	53.5%	2.67°	53.4%

Table 1. Average angular error using Lucas-Kanade at 30 frames/s and using our method at frames at 120 frames/s

The single chip imaging system architecture we consider is shown in Figure 4. It comprises an APS with column level ADC or a DPS (with pixel level ADC), frame memory with wide write bus from the sensor, a SIMD processor array, and a controller. This is a natural choice since most image processing algorithms are spatially local and shift invariant. Several researchers have investigated implementations of this generic architecture.[15–18] Forchheimer *et al.*[15] describe a 1.2μm CMOS chip with an APS with 8bit ADC, 8bit bi-directional shift register, 128bits of memory, and a bit-serial processor per column. Hong *et al.*[16] describe an array of SIMD processors performing video compression using vector quantization. Each processor performs 87 operations/pixel·frame at 30 frames/s and handles 16 columns. Hsieh *et al.*[18] discuss a video compression architecture for single-chip digital CMOS camera. Each processor handles 16 columns and is designed for MPEG2 encoder and DV encoder, which require 1.8 billion operations per second.

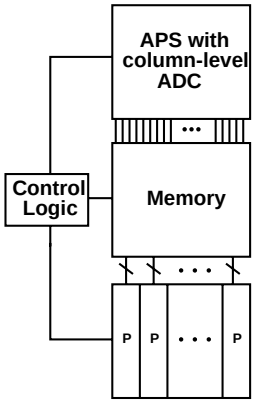


Figure 4. Single chip imaging system architecture.

To explore the memory size and processing that can be integrated with an image sensor at 0.18μm process

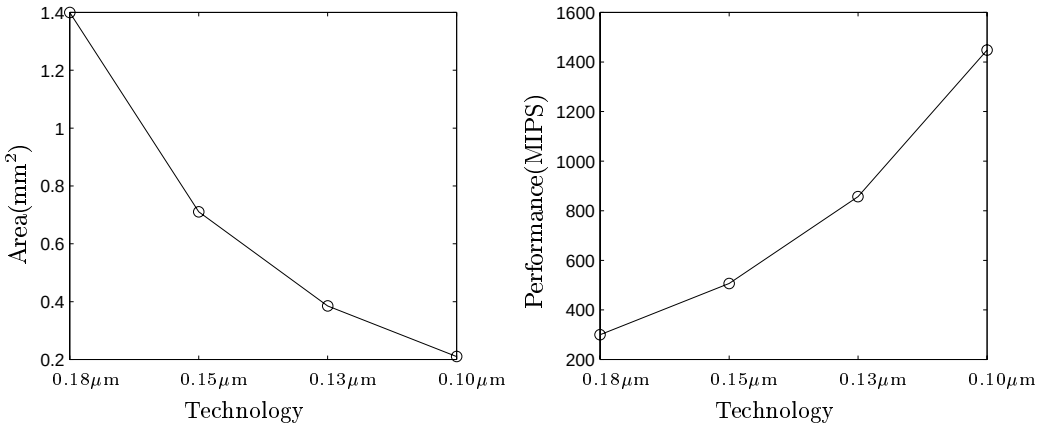


Figure 5. Area and performance of embedded processor as a function of process generation.

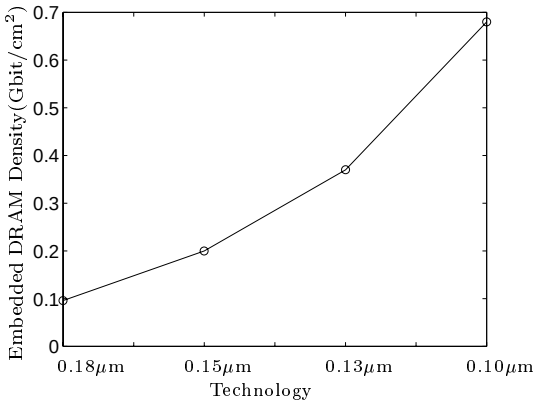


Figure 6. Embedded DRAM density as a function of process generation.

and below, we make the following assumptions:

- We assume that the sensor captures video sequence at constant frame rate of f_s frames/s, and outputs video sequence at a standard frame rate of 30 frames/s. We denote the oversampling factor by $OV = \frac{f_s}{30}$.
- We assume a fixed die core area of 1cm×1cm.
- We assume a 640 × 480 image sensor array size with 5μm×5μm pixel size. Thus it occupies an area of 3.2×2.4 mm^2. We assume that the row readout speed of the sensor is fast enough and is not the limiting factor in determining the throughput.
- We assume SIMD processor array with each processor comprising a 32-bit RISC core with a dedicated hardware MAC and occupying 1.4mm^2.[19] The area and performance of the processor as a function of process generation is given in Figure 5 assuming no change in processor architecture.[20,21]
- We assume that the memory uses embedded DRAM, which typically lags commodity DRAM by one process generation. The embedded DRAM density (Gbit/cm^2) including overhead for 0.18μm technology and below is provided[20] in Figure 6.
- We assume that 68mm^2 of chip core area is available for frame memory and SIMD processor array after subtracting off the image sensor area and an estimate of 24.32mm^2 of ADC, control logic, and routing overhead.

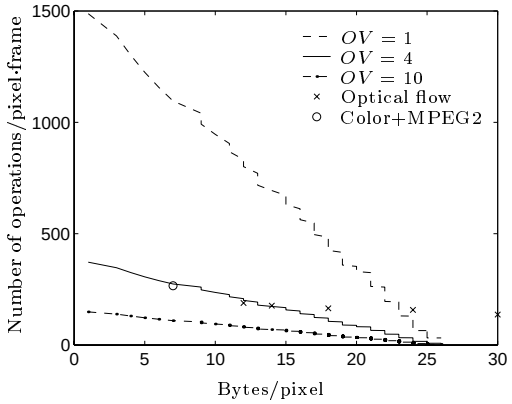


Figure 7. Maximum number of operations/pixel·frame vs. maximum number of bytes/pixel in 0.18μm CMOS process.

Figure 7 shows the integration limits for 0.18μm technology using the above assumptions. The lines represent the maximum number of operations/pixel·frame versus the maximum number of bytes/pixel possible for OV values of 1, 4 and 10.

Application	Operations/pixel·frame	Number of bytes/pixel
Color processing	32	3
JPEG	68	3
MPEG2	220	7

Table 2. Processing and memory required to implement digital video camera system.

Table 2 lists the memory and processing requirements for performing the applications in a conventional digital video camera system (see Figure 1) operating at the standard 30 frames/s rate. The estimated numbers provided in the table assume the following:

- Color processing includes color interpolation, white balancing, color correction and gamma correction.
- Color interpolation is performed using bilinear interpolation with kernel size of 3×3.
- White balancing is performed using the simple Gray world algorithm.
- Color correction is performed via 3×3 matrix multiplication.
- MPEG2 is assumed to have 50% of B-frames.

The memory and processing requirements for implementing color processing and MPEG2 encoding functions with the image sensor on a single chip are 252 operations/pixel·frame and 7 bytes of memory per pixel, respectively. This is plotted in Figure 7 for 0.18μm technology. Note that the requirements are not only easily satisfied, but that the available on-chip processing power is not fully utilized.

The processing and memory requirements for the optical flow estimation algorithm described in section 2 are also plotted in Figure 7. Different points represent the trade-offs between processing and memory requirements. It is clear from the plot that we can perform optical flow estimation at 120 frame/s with $OV = 4$ in 0.18μm technology.

Figure 8 shows the integration limits for 0.15μm down to 0.1μm technologies. The lines represent the maximum number of operations/pixel·frame versus the maximum number of bytes/pixel possible for the different technology generations at $OV = 10$. The figure clearly demonstrates the main point in this paper –

that merely integrating the functions of a conventional digital camera does not fully exploit the potential advantages of integration. As technology scales more compute intensive applications that can take advantage of high speed imaging such as tracking, motion segmentation, and 3D structure estimation can be implemented on a single chip imaging system.

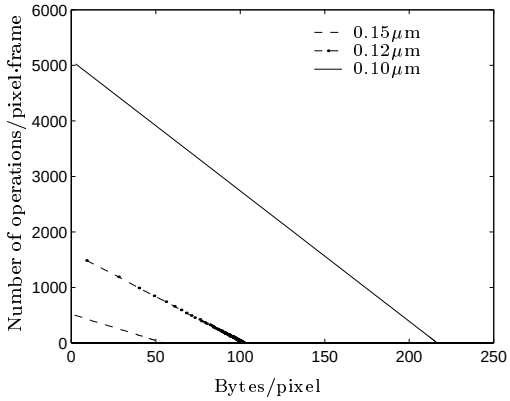


Figure 8. Maximum number of operations/pixel·frame vs. maximum number of bytes/pixel in 0.15μm technology and below at $OV = 10$.

4. CONCLUSION

An important trend in the design of digital cameras is the integration of capture and processing onto a single CMOS chip. Although integrating the components of a digital camera system onto a single chip significantly reduces system size and power, it does not fully exploit the potential advantages of integration. The paper argues that a key advantage of integration is the ability to exploit the high speed imaging capability of CMOS image sensors to enable new applications and to improve the performance of existing applications such as optical flow estimation. The idea we explored is to capture images at much higher frame rates than the standard frame rate, process the high frame rate data on chip, and output the video sequence and the application specific data at standard frame rate. We applied this idea to optical flow estimation and showed significant performance improvements over methods using standard frame rate sequences. We investigated the constraints on memory size and processing power that can be integrated with a CMOS image sensor in a 0.18μm process and below and showed that enough memory and processing power can be integrated to perform real time optical flow estimation. As technology scales, applications that can benefit from high speed imaging and require even more processing power and memory than optical flow estimation, such as tracking, pattern recognition, and 3D structure estimation, can be performed on a single chip digital imaging system.

ACKNOWLEDGMENTS

The work in the paper is supported under Programmable Digital camera Program by Agilent, Canon, HP, Interval Research and Kodak. The authors wish to thank Prof. Wandell, Khaled Salama, Ting Chen, Xinqiao Liu and Jeff DiCarlo for helpful discussions.

REFERENCES

1. E. R. Fossum, "Digital camera system on a chip," **18**, pp. 8–15, May-June 1998.
2. M. J. Loinaz, K. J. Singh, A. J. Blanksby, D. A. Inglis, K. Azadet, and B. D. Ackland, "A 200mW 3.3V CMOS color camera IC producing 352 × 288 24-b video at 30 frames/s," *IEEE Journal of Solid-State Circuits* **33**, pp. 2092–2103, December 1998.

3. S. Smith, J. Hurwitz, M. J. Torrie, D. J. Baxter, A. A. Murray, P. Likoudis, A. J. Holmes, M. J. Panaghiston, R. K. Henderson, S. Anderson, P. B. Denyer, and D. Renshaw, "Single-chip 306 × 244 pixel CMOS NTSC video camera and a descenedant coprocessor device," *ISSCC Digest* **33**, pp. 2104–2111, December 1998.
4. B. Ackland and A. Dickinson, "Camera on a chip," *ISSCC Digest* **39**, pp. 22–25, February 1996.
5. A. Krymski, D. V. Blerkom, A. Andersson, N. Block, B. Mansoorian, and E. Fossum, "A High Speed, 500 Frames/s, 1024 × 1024 CMOS Active Pixel Sensor," *Symposium on VLSI Circuits* , pp. 137–138, 1999.
6. N. Stevanovic, M. Hillegrand, B. Hostica, and A. Teuner, "A CMOS Image Sensor for High Speed Imaging," *ISSCC Digest of Technical Papers* , February 2000.
7. S. Kleinfelder, S. H. Lim, X. Liu, and A. E. Gamal, "A 10,000 Frame/s 0.18μm CMOS Digital Pixel Sensor with Pixel-Level Memory," *To be published in ISSCC Digest* , February 2001.
8. D. Handoko, S. Kawahito, Y. Takokoro, M. Kumahara, and A. Matsuzawa, "A CMOS Image Sensor for Focal-plane Low-power Motion Vector Estimation," *Symposium of VLSI Circuits* , pp. 28–29, June 2000.
9. D. Yang, A. E. Gamal, B. Fowler, and H. Tian, "A 640 × 512 CMOS Image Sensor with Ultra Wide Dynamic Range Floating-Point Pixel-Level ADC," *ISSCC Digest* , pp. 308–309, February 1999.
10. O. Yadid-Pecht and E. Fossum, "CMOS APS with autoscaling and customized wide dynamic range," *IEEE Workshop on Charge-Coupled Devices and Advanced Image Sensors* , pp. 48–51, June 1999.
11. X. Liu and A. E. Gamal, "Photocurrent estimation from multiple nondestructive samples in CMOS image sensors," *SPIE 2001* **4306**, January 2001.
12. J. Barron, D. Fleet, and S. Beauchemin, "Performance of Optical Flow Techniques," in *International Journal of Computer Vision*, vol. 12, pp. 43–77, February 1994.
13. C. Stiller and J. Konrad, "Estimating Motion in Image Sequences," in *IEEE Signal Processing Magazine*, pp. 70–91, July 1999.
14. S. H. Lim and A. E. Gamal, "Optical flow estimation using high speed sequence," *Submitted to ICIP 2001* .
15. R. Forchheimer, K. Chen, C. Svensson, and A. Odmark, "Single-Chip Image Sensors With a Digital Processor Array," in *Journal of VLSI Signal Processing*, vol. 5, pp. 121–131, 1993.
16. S. H. Hong and W. Yang, "An Embeddable Low Power SIMD Processor Bank," *ISSCC Digest* , pp. 192–193, February 2000.
17. J. Gealow and C. Sodini, "A Pixel-Parallel Image Processor Using Logic Pitch-Matched to Dynamic Memory," in *IEEE Journal of Solid-State Circuits*, vol. 34, pp. 831–839, June 1999.
18. J. Hsieh and T. Meng, "Low-Power Parallel Video Compression Architecture for a Single-Chip Digital CMOS Camera," in *Journal of VLSI Signal Processing*, vol. 21, pp. 195–207, 1999.
19. http://www.mips.com, "Mips technologies, inc.,"
20. http://public.itrs.net, "International Technology Roadmap for Semiconductors 1999 Edition,"
21. M. J. Flynn, P. Hung, and K. W. Rudd, "Deep-submicron Microprocessor Design Issues," in *IEEE Micro*, vol. 19, pp. 11–22, July-August 1999.