

Analog VLSI Systems for Image Acquisition and Fast Early Vision Processing

JOHN L. WYATT, JR., CRAIG KEAST, MARK SEIDEL, DAVID STANDLEY, BERTHOLD HORN, TOM KNIGHT, CHARLES SODINI, HAE-SEUNG LEE, AND TOMASO POGGIO
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA 02139

Received .

Abstract

This article describes a project to design and build prototype *analog* early vision systems that are remarkably low-power, small, and fast. Three chips are described in detail. A continuous-time CMOS imager and processor chip uses a fully parallel 2-D resistive grid to find an object's position and orientation at 5000 frames/second, using only 30 milliwatts of power. A CMOS/CCD imager and processor chip does high-speed image smoothing and segmentation in a clocked, fully parallel 2-D array. And a chip that merges imperfect depth and slope data to produce an accurate depth map is under development in switched-capacitor CMOS technology.

1 Introduction

1.1 Analog VLSI for Vision

In real-time machine vision the sheer volume of image data to be acquired, managed, and processed leads both to very high computational demands and to communication bottlenecks between imagers, memory, and processors. Our group is designing and testing experimental *analog* VLSI systems to overcome these problems. The goal is to determine how the advantages of analog VLSI—high speed, low power, and small area—can be exploited and its disadvantages—limited accuracy, inflexibility, lack of storage capacity, and long design and debugging times—can be minimized. The work is concentrated on *early* vision tasks, that is, tasks early in the signal flow path of animal or machine vision.

The faculty involved in this project since its beginning in September 1988 are Profs. Horn, Lee, Poggio, Sodini, and Wyatt, principal investigator. The completed designs to date include seven different analog chips for image filtering and edge detection [1-3], moment extraction to determine object position and orientation [4-6], image smoothing and segmentation [7-10], depth determination from stereo image pairs

[11], accurate depth determination jointly from imperfect depth and slope data [12], and camera motion determination [13], plus other chips to test novel circuit designs or processing methods [7, 14, 15]. Some of these projects are now complete, while others are in various stages of testing, redesign, or refabrication. The typical subsystem is physically very small and can perform one or more computationally intensive image-processing tasks at hundreds to thousands of frames per second using only tens to hundreds of milliwatts.

There is no single design strategy, but most systems have many of the following features:

- sensors (typically on-chip) tightly coupled to the processing circuitry;
- parallel computation;
- analog circuits for high throughput, low latency, low power, and small area;
- selection of tasks and algorithms requiring low to moderate precision;
- special emphasis on computations that map naturally to physical processes in silicon, for example, to relaxation processes or to resistive grids;
- emphasis on charge-domain processing, for example, CCD and switched-capacitors, for maximal layout density and compatibility with CCD sensors;

- sufficiently fast processing that no long-term storage circuitry is required;
- careful matching of algorithms, architecture, circuitry, and (often custom) fabrication for maximum performance; and
- modular design, with a standardized input and output, for compatibility between subsystems.

The advantages of this analog design approach to early vision hardware are high speed (both in the sense of high throughput and low latency), low power, and small size and weight. High throughput can be important in high-speed inspection processes, for example, for printed materials or PC boards. Low latency is very important for machine-vision-guided closed-loop feedback systems, because they are easily destabilized by image processing delays. Vehicle navigation and robot arm guidance are examples. Low power together with small size and weight are important for airborne and space applications. And finally, small low-power systems tend to be affordable.

1.2 System Design Issues

Analog systems are capable of very fast parallel operation, operating directly on photosensor output signals without analog-to-digital conversion. But they offer limited flexibility and precision. Simulations and experimental chip tests have shown that analog precision equivalent to 6-bit fixed-point arithmetic is adequate for most algorithms that are suitable for analog implementation. Our typical circuit design goal is 6–8 bits precision. These characteristics suggest that analog circuits are of greatest value in preprocessing or early processing steps.

We believe that in some applications a high-speed vision system should consist of a photosensor array followed by an analog processing stage that feeds into a digital processing section. The analog section outputs should be of much lower bandwidth than the image input. For example, the analog output could consist of velocity measurements, image moments, or perhaps a line drawing. This output can be converted to low-bandwidth digital form, easily handled by conventional digital technology.

For each of our designs the first decision is whether it should be implemented as a *single-chip system* or as a *modular component* of a larger multichip system. Processing steps that output an entire grey-scale image

should generally be implemented as modules in a larger system, especially if the image output needs further processing. But algorithms that convert an entire image into a few numbers are best implemented in single-chip systems. In our project these single-chip systems are always *focal-plane processors* with the imager and processor on the same chip.

Chip-to-chip communication between components of a modular vision system is another systems design issue, especially since bandwidths are high and the settling times for analog signals are longer than for binary data. We approach this problem by using wide communication pathways between chips, each pathway consisting of N wires for $N \times M$ images, designed to communicate one image column in parallel. Multichip modules, in which the individual chips are mounted on a substrate inscribed with interconnect lines, would be an ideal technology for this purpose and is under consideration.

A third design issue is the degree of parallelism in processing. In fully parallel designs each pixel of an $N \times M$ array is associated with a distinct processor that communicates directly with neighboring pixels and processors. These systems *with order (NM) parallelism* achieve very fast processing speed, often at a substantial cost in chip area. One possible difficulty is that in some cases only a small portion of chip area is left available for sensors. These low fill-factor sensor arrays have reduced sensitivity due to the small optically sensitive area, and their sparse sampling of the image can cause spatial aliasing problems. Another difficulty is that the processing speed often becomes so fast that system delays are completely dominated by photosensor integration time and communication overhead, that is, much of the processing speed becomes irrelevant to system performance. An alternative is to save area at some cost in speed by having only one processor per image row, that is, *order (N) parallelism*. These systems process one image column at a time. They are particularly convenient to design with CCD sensor arrays that function as shift registers, carrying image data to a column of processors at the edge of the imager.

We have only attempted relatively small arrays, from 29×29 to 64×64 pixels plus test arrays, in this project. These sizes are sufficient for examining the power and limitations of analog design. Fabrication has been done in the MOSIS 2-micron and MIT 1.75-micron processes. Many of the designs would scale directly with fabrication dimensions, yielding 256×256 or larger systems if fabricated in a commercial 0.8-micron

process. Arrays larger than 256×256 would generally require system design changes. The single-chip systems could be expanded by moving the imager to a separate chip, for example. The modular component systems could be expanded by doing processing in parallel on several chips or handling portions of an image sequentially on a single chip.

Interestingly, most routes of expansion yield processing times that grow *more slowly* than the number of pixels. For example a design dominated by processing time that has order N parallelism and a design dominated by communication time that has column-parallel input and output would both have delays proportional only to M as the $N \times M$ pixel array is scaled up. And a design like the one in section 2 with speed dominated by photoreceptor response would have *constant* delay as the array size is increased.

1.3 Analog Design Styles

This project explores two very distinct styles of analog design: continuous-time (i.e., unlocked) processing and discrete-time (i.e., clocked) processing. Both styles are implemented primarily in CMOS (Complementary Metal Oxide Semiconductor) using field effect transistors (FETs). Bipolar transistors play only certain special roles in our designs. The two varieties of clocked designs are based on charge-coupled devices (CCDs) and switched capacitors.

Unlocked analog systems are generally the fastest and require the least power. The unlocked designs in our project use bipolar transistors as photoreceptors and FETs for image processing. The bipolar transistors are called parasitic elements since they are unavoidably present in standard CMOS chips. These designs are somewhat similar to those in Prof. Carver Mead's work [16]. One major difference is that we generally avoid operating FETs in the subthreshold mode, to avoid transistor matching problems and attain greater speed. One could also use photodiodes as sensors or bipolar transistors as processing elements in unlocked designs. Since processing time is generally negligible in these systems (often less than 50 microseconds for an entire image), the limiting factors for system speed are photoreceptor response and image output times.

Many of our clocked designs use CCDs, the most widespread commercial electronic image sensor. They convert incident optical energy into charge, and they can also perform linear and thresholding operations that

are useful for image processing. CCD systems operate in discrete time on continuous-valued packets of charge. Processing operations can be conveniently performed directly on the charge packets delivered by the photosensors. CCDs also provide the best form of analog shift register available. Charge decay times are long enough (on the order of tens of milliseconds for the smallest detectable error in many systems at room temperature) that CCDs can serve as temporary storage devices during processing. CCD systems typically require a substantial number of clock signals (e.g., 10 to 25 in our designs) that are produced off-chip. This is an advantage in that clock signals can be varied to control the data flow, but it is also a significant drawback in that clock generators consume power, clock signal routing consumes chip area, and the clocking operation adds complexity to the design.

Switched-capacitor (SC) systems are a second style of discrete-time or clocked design. Switched capacitors were originally introduced to synthesize on-chip linear resistors from capacitors and transistor switches. Their importance arises from their small area and the controllable conductance that is proportional to switching frequency. More complex multiterminal switched-capacitor elements that are useful in vision processing appear in section 4.

The circuit and system design issues mentioned above are illustrated in three doctoral-level design projects in the next three sections. David Standley's chip in section 2 is a fully parallel, continuous-time, CMOS single-chip system with bipolar phototransistors. Craig Keast's chip in section 3 is a fully parallel, discrete-time, CMOS/CCD system with CCD imagers that can be used as a single-chip system or a modular system component. And Mark Seidel's design in section 4 will be implemented as a fully parallel, switched capacitor, CMOS modular system component chip that relies on an off-chip imager.

2 Fast Imager and Processor Chip for Object Position and Orientation

David Standley, working with Profs. Horn and Wyatt, has designed and tested this analog VLSI chip, which finds the position and orientation of an object's image against a dark background. The algorithm is based on finding the first and second moments of the image brightness [17]. These moments allow the centroid (an indicator of position) and the axis of least inertia (an

indicator of orientation) to be computed; see figure 1. If $I(x, y)$ is the intensity as a function of position, and we (initially) assume that $I(x, y) = 0$ outside the object, then the required quantities are given by

$$\iint I(x, y)h(x, y) dA \quad (1)$$

for all of the following h :

$$h(x, y) = 1, x, y, xy, x^2 - y^2 \quad (2)$$

(x^2 and y^2 are not needed separately). All of the weighting functions h are *harmonic*; that is, the Laplacian vanishes identically:

$$\Delta h(x, y) \equiv 0 \quad (3)$$

This observation is the key to a scheme proposed by Horn [18], in which an analog computer based on a resistive sheet (or resistor grid) can be constructed. In the implementation described here, an $N \times N$ array of discrete intensity data is reduced to a set of $4N$ quantities by a 2-D resistor grid, and is subsequently reduced to a set of just *eight* quantities by 1-D resistor lines, all in a continuous-time, asynchronous fashion—no clocking required. The eight outputs can be digitized, and the centroid and orientation can be found using simple expressions [4, ch. 2]. While earlier systems have used this method to compute position [19, 20], none have used it previously to perform the orientation task, which requires computing second moments.

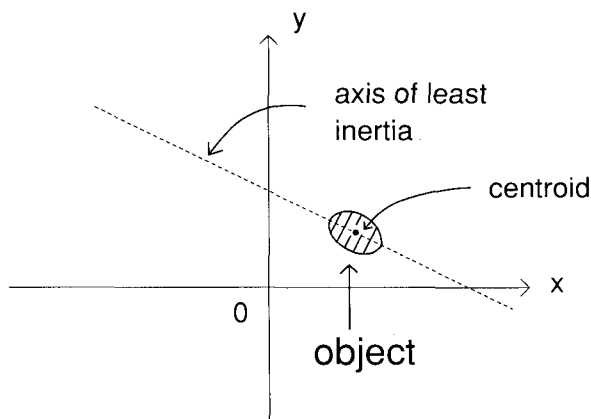
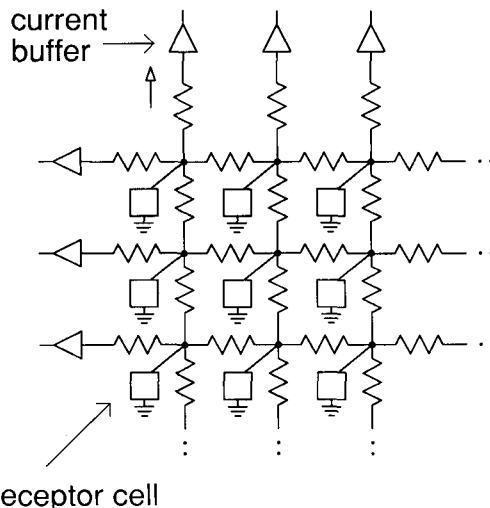


Fig. 1. Example of object centroid and axis of least inertia.

Figure 2 shows the resistor grid and its associated array of *photoreceptor cells*, which are uniformly spaced and occupy most of the chip area. The object image is focused onto the surface of the chip, inside the array. Each photoreceptor cell contains a phototransistor and processing circuitry; it converts the incident light



photoreceptor cell

Fig. 2. Resistor grid and photoreceptor cell array.

intensity into a current that is injected into the grid. Thresholding is available to remove a dim (yet nonzero) scene background, so it does not interfere with the calculation. If the intensity I at a particular cell is below an adjustable threshold value I_{th} , then no current is injected. If $I > I_{th}$, then the output current (which is analogous to the gray-level weighting at that location) is proportional to $I - I_{th}$. The result is a *continuous, piecewise-linear response characteristic*. The array size is 29×29 ; intentional image blurring over a few elements gives substantially increased resolution.

The perimeter of the grid is held at a constant voltage by the *current buffers* in figure 2: this ensures proper operation of the grid as a “data reduction” computer. The buffer outputs are simply copies of the currents flowing into them from the grid. The buffers are needed to isolate the grid from the effects of other circuitry. Figure 3, which shows the complete architecture of the chip, indicates how the (copied) grid currents are fed into resistor lines on the perimeter, how the ends of these lines are connected, and where the eight output currents exit the chip near the corners. These currents are measured by external circuitry which also holds the ends of the lines at ground potential. In this setup, there are two lines on each side: one uniform and one “quadratic” line. These calculate weighted sums of the grid currents, where the weighting is (respectively) a linear or square-law function of the position along the line. The buffer outputs are steered either to the uniform or quadratic lines, so that four outputs are available at a time; that is, multiplexing is required here (but is not necessary in general).

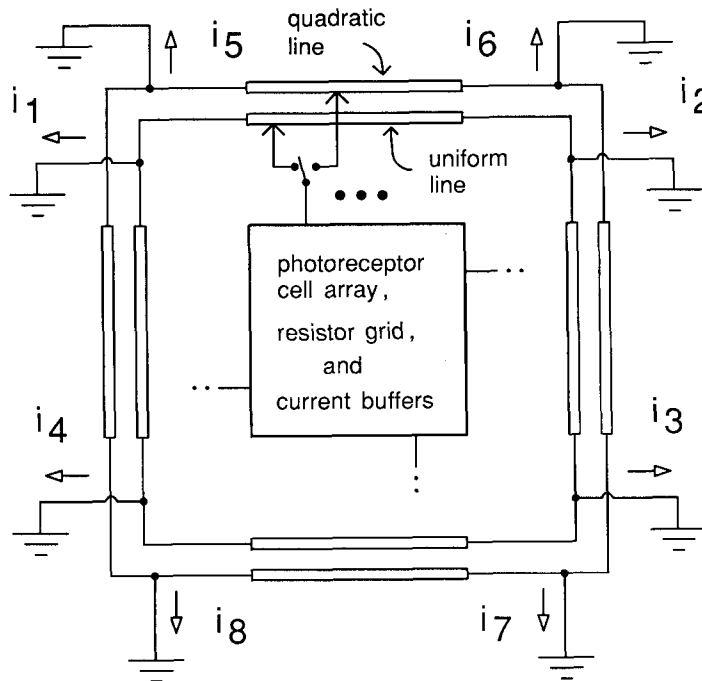


Fig. 3. Main chip architecture.

Working chips have been fabricated using the Mosis service. The die size is 7.9 mm \times 9.2 mm, and the imaging array is a 5.5 mm square. Accuracy is dependent on the object. For moderately sized and sufficiently elongated objects, for example, a diamond of diagonal dimensions 25 by 50 on a (normalized) 100 by 100 image field, orientation is typically determined to within $\pm 2^\circ$. This fully parallel single-chip system operates at a remarkable speed of 5000 frames per second, and power consumption is typically only 30 mW. Further details of the design and testing can be found in [4, 5, 6]. Dr. Standley is now designing a more advanced version of this system with the imager and processor on separate chips, at the Rockwell Corporation.

3 An Integrated Image Acquisition, Smoothing and Segmentation Focal Plane Processor

Craig L. Keast, working with Prof. Sodini, has designed and fabricated a focal plane processor for image acquisition, smoothing, and segmentation. The technology used is CCD/CMOS, since it provides a convenient way to image, process, and read out data. In this architecture, image brightness is transferred into

signal charge using standard CCD imaging techniques. The 2-dimensional Gaussian smoothing operation is approximated by a discrete binomial convolution of the image with a fully controllable support region. The design incorporates segmentation circuits with variable threshold control at each pixel location to preserve edges in the image. Once processed, the image is read out using a standard CCD clocking scheme.

By performing some of the image preprocessing functions in parallel on the image plane, the requirements on subsequent signal processing circuits are reduced. Since the area for collecting pixel charge is a large fraction of the array, the need for a spatial anti-aliasing prefilter is eliminated, in contrast to many fully parallel designs.

3.1 System Architecture

Figure 4 shows a system level block diagram of the processor. The system consists of four functional blocks: the two-dimensional focal plane processing array, a conventional 4-phase CCD shift register, the output circuit, and an electrical input structure used for testing and characterization.

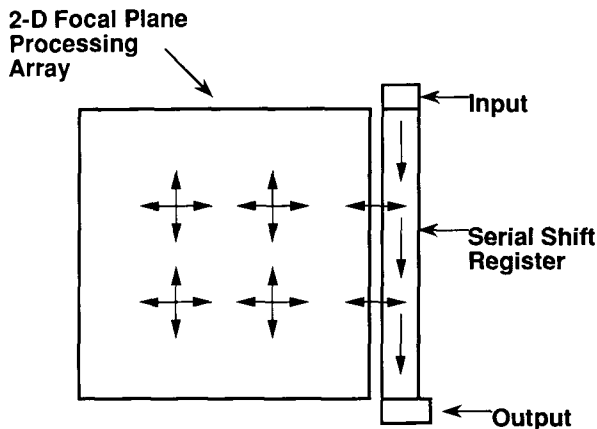


Fig. 4. System level block diagram of the smoothing and segmentation focal plane processor.

2-D Processing Array. Figure 5 shows a plan view of a nine-pixel section of the focal plane array. The backbone of the array consists of an orthogonal mesh of horizontal and vertical CCD transfer channels. This 2-D array is used to perform the spatial binomial convolution of the integrated image charge. It is accomplished by clocking the four gates (phases 2, 5, 7 and

8), surrounding each node gate "high," and then clocking phase 1 "low." After this operation, one quarter of the accumulated charge from each pixel (to within the alignment accuracy of the polysilicon layers and the active region) is stored under each of the four gates. With *straight smoothing*, that is, with the segmentation circuits deactivated, each of the four charge packets is averaged with the quarter charge packets of its nearest neighbors. The first iteration is equivalent to convolving the image with a

$$\frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

mask at each pixel location. The second cycle of the smoothing operation, which begins with the averaging of the four new values at each node location, results in the following convolution mask:

$$\frac{1}{64} \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 8 & 2 & 0 \\ 1 & 8 & 20 & 8 & 1 \\ 0 & 2 & 8 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

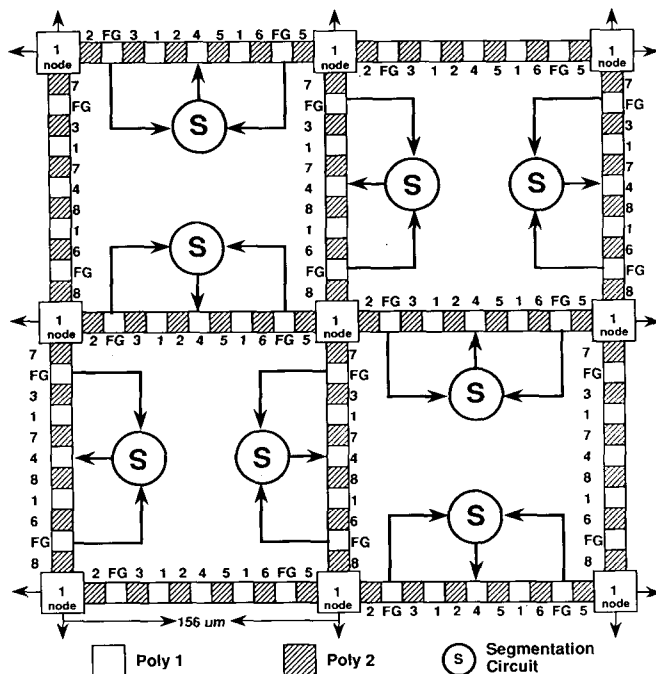


Fig. 5. Plan view of the 2-D CCD focal plane processor architecture. The two levels of polysilicon gate material are denoted Poly 1 and Poly 2. The segmentation circuits prevent charge mixing along lines connecting adjacent pixels if the absolute difference in charge at those pixels exceeds a threshold value.

Additional iterations of the smoothing cycle increase the support region over which the convolution is performed. In its limit this operation approximates convolution with a discrete Gaussian operator.

Smoothing with Segmentation. The binomial smoothing with segmentation is a nonlinear operation. It preserves high spatial frequencies associated with charge differences that are greater than the segmentation threshold, while *smoothing* those differences that are smaller than the threshold. In the CCD approach presented here, the threshold of the segmentation circuits is controlled by an externally adjustable voltage via a global bus.

To help in understanding this operation, consider a single horizontal segmentation circuit shown in figure 6, which shows two nodes of the 2-D array and the CCD transfer channel that connects them. This channel is represented by the overlapping CCD gates in the bottom of figure 6. The rest of the circuitry corresponds to the "S" box shown in figure 5. During operation the quarter-pixel charge packets, which are stored under the two gates (phases 2 and 5) adjacent to the two node locations, are transferred under their respective floating gate amplifiers (FGA). The potential of the floating gates, which were reset to a reference voltage VFG on the phase one clock cycle, changes when the charge packet is transferred underneath the gates. The voltages

from the left and right floating gates are buffered using the source followers in figure 6. The buffered potentials are compared using the absolute value of difference circuit (AVDC) shown on the left side of figure 6. The AVDC is a parallel combination of two "fill-and-spill" inputs with their input gates cross-coupled (see figure 7), and delivers a negative charge packet proportional to the absolute value of the difference of the two FGA voltages to the left side of the CMOS sense amplifier. The AVDC on the right side of figure 6 is used to supply a charge packet proportional to the externally controlled segmentation threshold voltage. If the absolute value of the difference of the two FGA potentials is greater than the threshold, the left side of the sense amp will be driven "low" when the V_{sense} is brought "low." After the *sense* operation is completed, the charge packets are transferred to the phase-2 and phase-5 gates on either side of the mixing gate (shown in the bottom center of figure 6). The "low" on the left side of the sense amp is inverted to a high, which disables the p-channel phase-4 pass transistor, and enables the n-channel phase-1 pass transistor. During the smoothing cycle, phase 4 goes "high" and phase 1 is held "low." Therefore, when the difference value of the two FGA potentials is greater than the segmentation threshold, the two charge packets are prevented from mixing. When the difference value is lower than the segmentation threshold, the left side of the sense amp goes

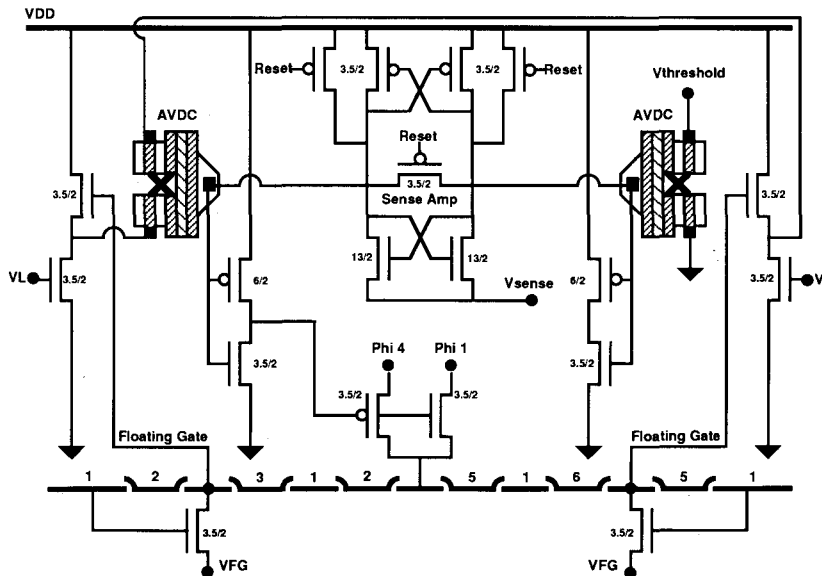


Fig. 6. Schematic representation of two nodes of the focal plane processor and the horizontal CCD transfer channel and segmentation circuitry that connect them.

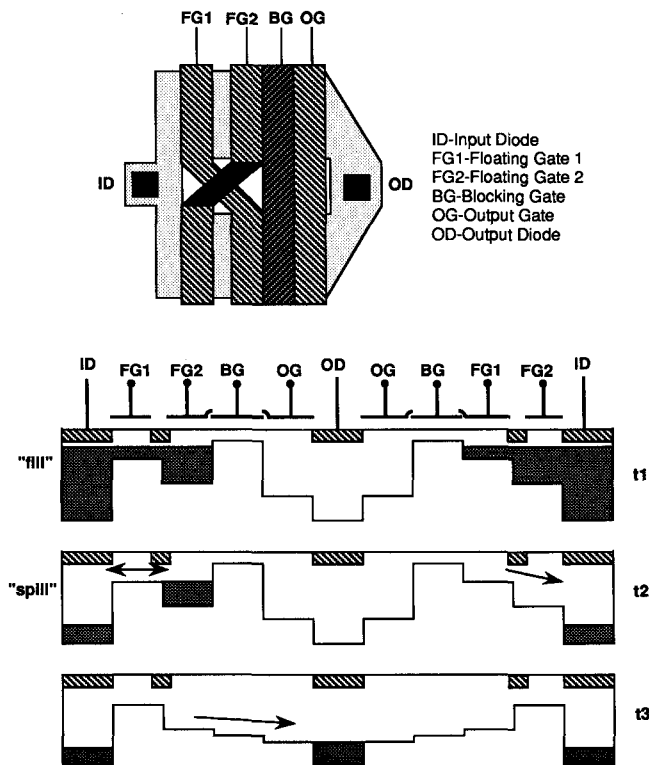


Fig. 7. Schematic plan view of the absolute value of difference circuit (AVDC). Here the signal on floating gate 2 exceeds that on floating gate 1. A quantity of charge proportional to that difference is trapped under gate 1 on the left and then dumped into the output diode.

“high” and the phase-4 pass transistor is enabled, allowing the two charge packets to be averaged. During normal operation, this cycle occurs in parallel along all vertical and horizontal transfer channels resulting in true 2-dimensional processing.

3.2 Results

Craig Keast has also developed a 4-phase buried channel CCD capability to enhance MIT's 1.75-micron baseline CMOS process [9]. He has fabricated both a 4×4 test array and a 40×40 pixel system in this process. The 4×4 array has been fully characterized and the 40×40 system is currently under evaluation. Figure 8 shows experimental data comparing the input image and a series of processed images obtained from a 4×4 processor test array after 1, 5, 10, 20, and 40 cycles. With a high segmentation threshold, none of the edges are preserved and the processed images are identical to the straight smoothing case; however, with a low segmentation threshold, the image edges are preserved.

Figure 9 shows the simulated results of the smoothing and segmentation algorithm implemented on

a 256×256 8-bit image. This figure shows the original input image, the image after 500 *straight smoothing* cycles, and the image after 500 smoothing and segmentation cycles with a threshold level of 12 out of a possible 256 levels. With a pixel clock of 10 MHz and a segmentation clock of 1 MHz, images of this size could be processed at > 100 frames per second with 500 smoothing and segmentation cycles per frame. Further details on this system can be found in [9, 10].

4 An SC Coupled Depth/Slope Network Implementation

Mark N. Seidel, working with Profs. Knight and Wyatt, has designed a switched-capacitor implementation of Harris' coupled depth/slope network [21]; it is currently being fabricated. This network effects surface reconstruction by computing the dense depth and slope maps given sparse depth and slope measurements; these sparse measurements are outputs of other early vision modules.

Because of the sparsity of the input data, there can be infinitely many surfaces that might correspond to

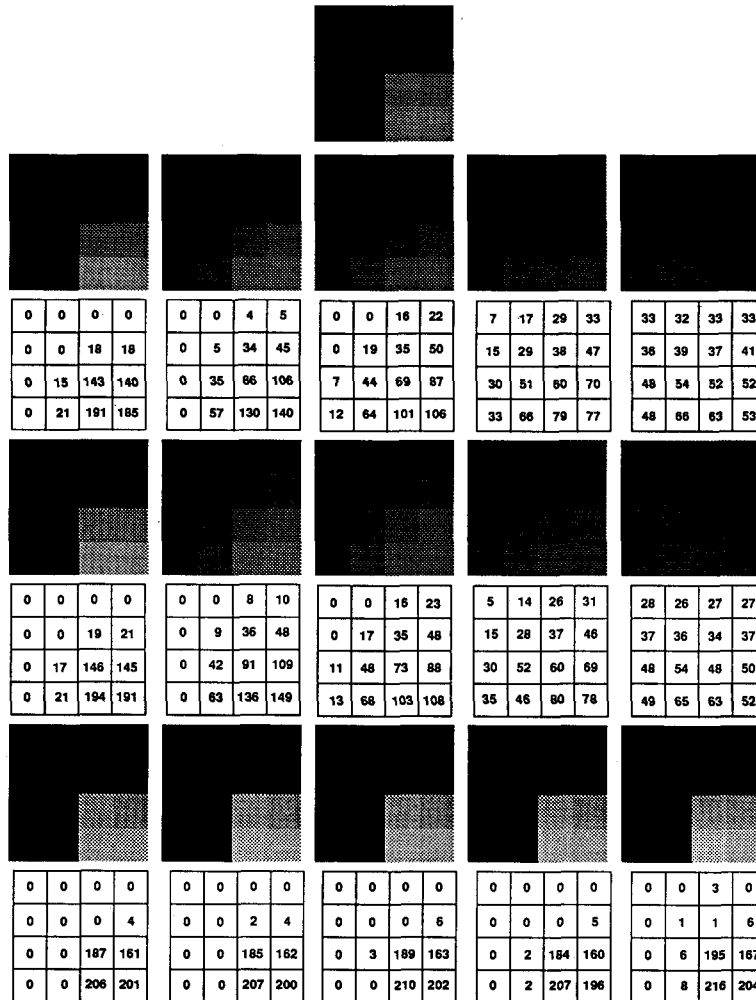


Fig. 8 Images obtained from a 4x4 processor test array. The input image (top) and a series of processed images obtained after 1, 5, 10, 20, and 40 cycles: first row—straight smoothing, second row—smoothing with segmentation (high threshold), and third row—smoothing with segmentation (low threshold). Below each row are the corresponding 8-bit intensity values.

the measurements. Alternatively, due to conflicting measurements, there might be no such solution. Our approach is to construct and minimize an energy functional consisting of terms penalizing deviation from the input data as well as terms penalizing deviation from a measure of smoothness.

4.1 Implementing Cost Function Minimization

A beautiful application of analog resistor networks is in the implementation of cost function minimization. As a direct consequence of Maxwell's minimum heat theorem, reciprocal resistive networks have solutions characterized by an extremum principle, and therefore

appropriately designed networks automatically relax to the minimum cost solution. Specifically, if the energy functional is constructed to represent the total resistive cocontent of the network, with each term representing the cocontent [22] of a reciprocal resistive element, then the set of voltages and currents in the network (after all transients die out) is that unique set that corresponds to the minimum total resistive cocontent. Since this minimum cocontent also corresponds to the minimum of the cost function, the network solution is the solution to the minimization problem.

Reciprocal resistive elements play an important role in this theory, and constraint boxes [23, 24] allow one to design and implement such elements in VLSI. Constraint boxes are multiterminal resistive elements that

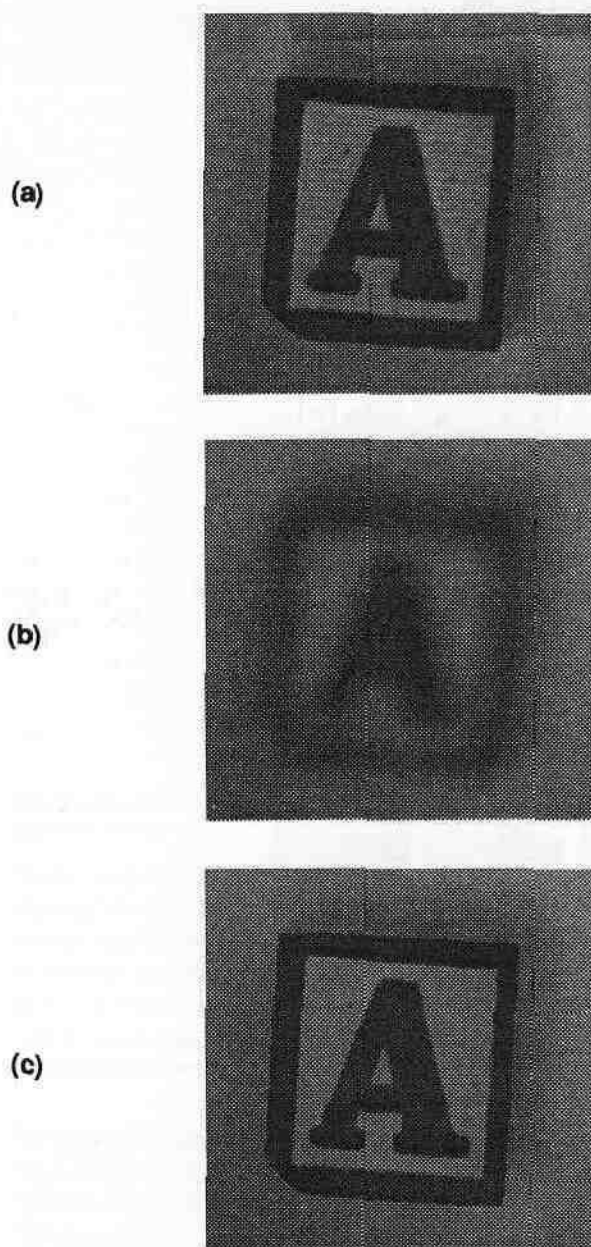


Fig. 9. Simulation of the smoothing and segmentation algorithm on a 256×256 8-bit gray level image. (a) Original image, (b) output after 500 smoothing cycles, and (c) output after 500 smoothing and segmentation cycles with a threshold of $12/256$.

penalize any deviation from the constraint by dissipating power proportional to the square of the deviation. This power dissipation arises as the terminal currents attempt to restore the constraint. As an example, consider a two-terminal linear resistor with value R . The resistor constraint is to keep the terminal voltages (v_1 and v_2)

equal, with a cocontent penalty of $(1/2R)(v_2 - v_1)^2$ when they are not. More generally, for a constraint that can be written as $F(v_1, \dots, v_n) = 0$ for a grounded $(n + 1)$ -terminal element, then the cocontent of this element is given by

$$\hat{G}(\mathbf{v}) = \frac{1}{2} F^2(\mathbf{v}) \quad (4)$$

and the terminal currents by

$$\mathbf{i}(\mathbf{v}) = \nabla \hat{G}(\mathbf{v}) = \begin{bmatrix} F \frac{\partial F}{\partial v_1} \\ \vdots \\ F \frac{\partial F}{\partial v_n} \end{bmatrix} \quad (5)$$

As an example of the above ideas, consider the coupled depth/slope network proposed by Harris [21, 24]. This network incorporates sparse depth and slope (surface orientation) data, and naturally generalizes to arbitrary levels of smoothness. To understand the operation of the network, consider the underlying cost functional (in 1-D)

$$E(u, p | v, r) = \frac{1}{2} \int \left[g_1(u - v)^2 + g_2 \left(\frac{du}{dx} \right)^2 + g_3 \left(p - \frac{du}{dx} \right)^2 + g_4 \left(\frac{dp}{dx} \right)^2 + g_5(r - p)^2 \right] dx \quad (6)$$

where v and r are the depth and slope inputs, respectively, u and p are the computed dense depth and slope maps, respectively. The cost functional penalizes deviations from the data (the first and last terms), as well as large gradients in depth (second term) and slope (fourth term). The third term penalizes differences between du/dx , the slope based on the dense depth map, and p , the computed dense slope map.

The discretization of the above energy functional yields

$$E(u, p | v, r) = \frac{1}{2} \sum_k \left[g_1(u_k - v_k)^2 + g_2(u_{k+1} - u_k)^2 + g_3(p_k - u_{k+1} + u_k)^2 + g_4(p_{k+1} - p_k)^2 + g_5(r_k - p_k)^2 \right] \quad (7)$$

This functional represents the total cocontent of the network shown in figure 10. Four of the terms correspond to linear two-terminal resistors, and one (the g_3 term) corresponds to the subtractor constraint box. These boxes, indicated by circles in figure 10, attempt to enforce the constraint $p_k = u_{k+1} - u_k$ (or $p = du/dx$ in the continuous case). The intuitive idea is that the depth data (v) are smoothed by the depth input resistive net (resulting in u), while the slope data (r) are smoothed by the slope input resistive net (resulting in p). The depth and slope are coupled via the (bidirectional) constraint boxes, and the network settles to its minimum energy state.

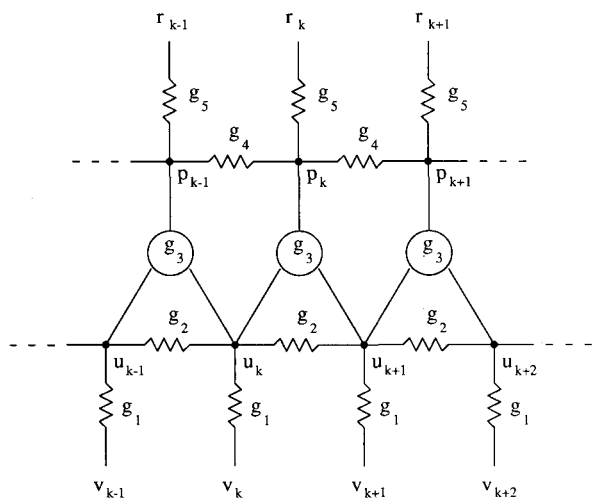


Fig. 10. One-dimensional coupled depth/slope network.

4.2 The SC Coupled Depth/Slope Architecture

A straightforward implementation of figure 10 suffers from a few drawbacks. First, low-power dissipation demands large resistor values that are difficult to obtain compactly in standard VLSI processes. Second, implementing the subtractor constraint box is nontrivial [23, 24]. Third, controlling the relative resistor values is difficult in standard VLSI; this controllability allows for tailoring the network to the image in a dynamic way. For all of these reasons, we believe a switched-capacitor (SC) implementation is a viable alternative.

The basic parallel SC resistor equivalent is shown in figure 11. If the two terminals are connected to voltage sources, then the charge transferred from terminal 1 to terminal 2 over one clock cycle is $\Delta q = C(v_1 - v_2)$. This charge transfers in time $T = 1/f$

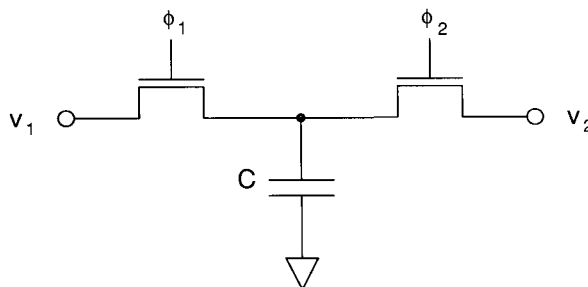


Fig. 11. Parallel SC realization of a continuous resistor.

(where f is the switching frequency and T is the switching period), and the current (charge transferred per unit time) is

$$i = \frac{\Delta q}{T} = fC(v_1 - v_2) \quad (8)$$

The equivalent resistance, therefore, is given by $R_{eq} = 1/fC$. Typical values are $C = 1$ pf and $f = 1$ MHz, yielding an equivalent resistance of $R_{eq} = 1$ M Ω .

SC networks can also be thought of as discretized versions of continuous time elements (or collections of elements). For example, as observed by Tom Knight and Sandy Wells [unpublished], the SC network shown in figure 12(a) has the q - v characteristic

$$\begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = C \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (9)$$

When one sets $\bar{i}_k = fq_k$ (where the overbar indicates a time-averaged value) then the network is (in some sense) equivalent to that shown in figure 12(b) for which

$$\begin{pmatrix} i_1 \\ i_2 \end{pmatrix} = \frac{1}{R_1 + R_2} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (10)$$

Correspondence occurs when $R_1 + R_2 = 1/fC$. Note that this subcircuit implements the subtractor constraint box, with the associated cocontent $\tilde{G}(v_1, v_2) = (fC/2)(v_2 - v_1)^2$.

4.3 The SC Implementation

A schematic of a portion of the network is shown in figure 13. The depth and slope input resistors (g_1 and g_5) are implemented using the parallel SC resistor equivalents described earlier (C_1 with ϕ_3 and ϕ_3 bar, and C_5 with ϕ_8 and ϕ_8 bar). The depth and slope smoothing resistors (g_2 and g_4) are implemented using a double capacitor charge-sharing line with alternate clocking (C_2 with ϕ_1 and ϕ_2 , and C_4 with ϕ_6 and ϕ_7).

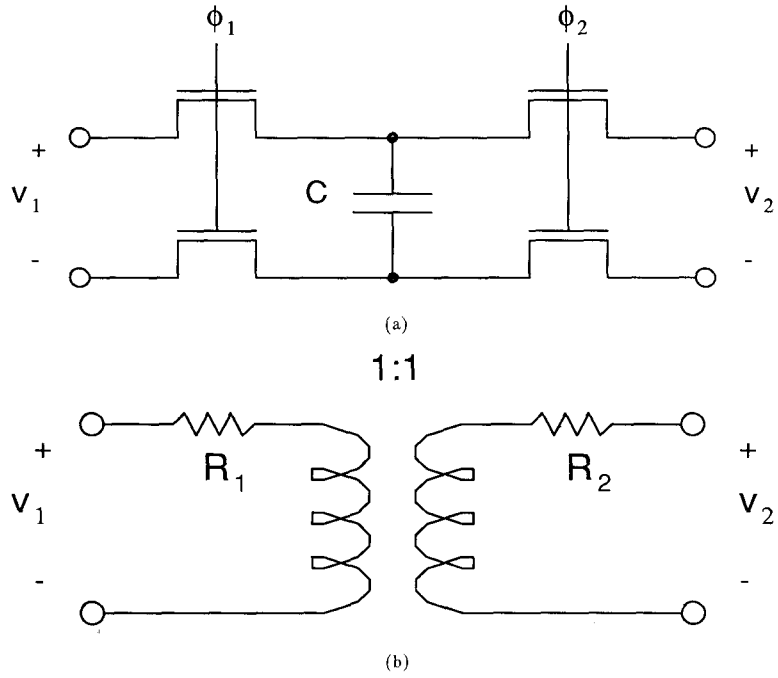


Fig. 12. (a) SC equivalent to a transformer with loss, and (b) its continuous-time equivalent.

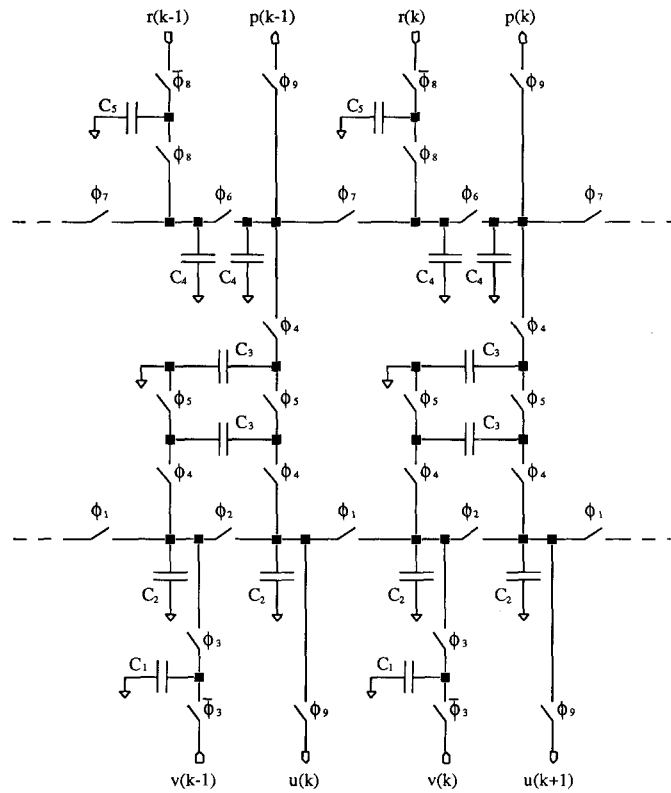


Fig. 13. SC 1-D coupled depth/slope switch-level schematic.

The double capacitors representing each depth and slope node allow for simultaneous derivative sampling, as well as for symmetry of parasitic capacitances. The subtractor constraint box is also implemented using a double capacitor (C_3 with ϕ_4 and ϕ_5). This doubling allows for an unbiased depth derivative and slope sampling. The ϕ_9 switches allow for read-out of the dense depth and slope maps. Finally, by altering the relative clocking frequencies of the various subcircuits, the network response can be tailored dynamically.

Mark Seidel has designed a 1-D version of this system [12]. Test circuits are being fabricated in the MOSIS 2 micron CMOS process.

5 Additional Designs and Related Projects

A more complete description of the MIT project can be found in [25].

The first CCD system in this project was a 64×64 imager and Laplacian of a Gaussian filter conceived and designed by Woodward Yang in collaboration with Dr. Alice Chiang of Lincoln Laboratories and Prof. Poggio [1-3]. This system component chip runs at 1000 frames per second with order N parallel processing (one processor per image row) and produces a filtered image suitable for edge detection. He is currently redesigning this system for fabrication in a more modern 2-micron process.

Mikko Hakkarainen, working with Prof. Lee, has designed and tested a 40×40 CMOS/CCD stereo chip [11] that has been fabricated by MOSIS. It is based on the computationally attractive Marr-Poggio-Drumheller algorithm [26, 27].

Paul Yu and Prof. Lee have designed and tested a new compact CMOS resistive fuse circuit for continuous-time image smoothing and segmentation [7]. A 32×32 array with phototransistor imagers has also been designed and tested [8].

Ignacio McQuirk, working with Profs. Horn, Lee, and Wyatt, has designed and simulated a 64×64 CMOS/CCD motion chip that finds the focus-of-expansion of a scene [13]. The algorithm is based on direct methods for recovering motion without feature extraction [28]. The circuit components of this chip have been fabricated by MOSIS and tested, and the entire chip is now under test.

Lisa Dron, working with Prof. Horn, has created an image segmentation and compression algorithm designed for fully parallel implementation using CCDs

[29]. She has designed and is testing a chip to implement this algorithm.

Chris Umminger, working with Prof. Sodini, has designed and tested a switched-capacitor implementation of resistive lines for use in image smoothing and segmentation algorithms [15].

Steve Decker and Prof. Wyatt have designed an extremely compact resistive fuse circuit using nonconventional depletion-mode CMOS. Mr. Decker, working with Prof. Sodini, has hand-fabricated and tested this system in MIT's Integrated Circuits Laboratory [7].

Andrew Lumsdaine, working with Prof. Jacob White and Prof. Wyatt, has written a very fast, fully parallel Connection-Machine-based simulator for analog vision circuits and used it to study resistive fuse segmentation and smoothing circuits [30].

Mark Seidel, working with Prof. Wyatt, has discovered a set of closed-form bounds for the settling time of large-switched capacitor networks [31].

6 Conclusions

Analog VLSI systems offer some exciting performance advantages for early vision, particularly in the areas of speed, power, and size. We have found a speed advantage of roughly $1\frac{1}{2}$ orders of magnitude over comparable special-purpose digital systems, though this figure varies from example to example. The power and size advantages should make analog systems cheaper in cases where their greater design, test, and debugging costs can be amortized over large production volumes.

Since individual analog systems are not readily reprogrammable, each early vision task requires a separate design. A system consisting of several separate modules can perform a number of distinct tasks, depending on the signal routing from one module to another. The performance of initial designs, some of which were reported here, leads us to expect that analog early vision systems will find application in many special niches where speed, power, size, or cost constraints are crucial.

Acknowledgments

This work has been supported by NSF and DARPA under Contract MIP-8814612 and by the DuPont Corporation, and NSF under contract MIP-9117724.

References

1. W. Yang and A.M. Chiang, A full fill-factor CCD imager with integrated signal processors, *Proc. ISSCC*, San Francisco, CA, pp. 218-219, February 1990.
2. W. Yang, The architecture and design of CCD processors for computer vision, Ph.D. thesis, Dept. Electrical Engineering and Computer Science, MIT, Cambridge, MA, August 1990.
3. W. Yang, Analog CCD processors for image filtering, *SPIE Intern. Symp. on Opt. Eng. Photon. Aerospace Sensing*, Orlando, FL, pp. 114-127, April 1991.
4. D.L. Standley, Analog VLSI implementation of smart vision sensors: Stability theory and an experimental design, Ph.D. thesis, Dept. Electrical Engineering and Computer Science, MIT, Cambridge, MA, P2, January 1991.
5. D.L. Standley and B.K.P. Horn, An object position and orientation IC with embedded imager, *Proc. IEEE Intern. Solid-State Circuits Conf.*, San Francisco, CA, pp. 38-39, February 1991.
6. D.L. Standley, An object position and orientation IC with embedded imager, *IEEE J. Solid-State Circ.* 26(12):1853-1859.
7. P.C. Yu, S.J. Decker, H.-S. Lee, C.G. Sodini, and J.L. Wyatt, Jr., CMOS resistive fuses for image smoothing and segmentation, *IEEE J. Solid-State Circ.* 27(4):545-553, April 1992.
8. P.C. Yu and H.-S. Lee, A CMOS resistive fuse processor for 2-D image acquisition, smoothing and segmentation, submitted to 1992 European Solid-State Circuits Conference, Copenhagen, Denmark.
9. C.L. Keast and C.G. Sodini, An integrated image acquisition, smoothing and segmentation focal plane processor, to appear in *VLSI Circuit Symposium*, Seattle, WA, June 1992.
10. C.L. Keast, An integrated image acquisition, smoothing and segmentation focal plane processor, Ph.D. Thesis, Department of Electrical Engineering and Computer Science, MIT, February 1992.
11. M. Hakkarainen, J. Little, H.-S. Lee, and J.L. Wyatt, Jr., Interaction of algorithm and implementation for analog VLSI stereo vision, *SPIE Symp. Opt. Eng. Photon. Aerospace Sensing*, Orlando, FL, pp. 173-184, April 1991.
12. J.L. Wyatt, Jr., C. Keast, M. Seidel, D. Standley, B. Horn, T. Knight, C. Sodini, H.-S. Lee, and T. Poggio, Analog VLSI systems for early vision processing, to appear in *Proc. 1992 IEEE Intern. Symp. on Circuits and Systems*, May 1992, San Diego, CA.
13. I.S. McQuirk, Direct methods for estimating the focus of expansion in analog VLSI, S.M. thesis, Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA, September 1991.
14. C.L. Keast, and C.G. Sodini, A CCD/CMOS process for integrated image acquisition and early vision signal processing, *Proc. SPIE Charge-Coupled Devices and Solid State Sensors*, Santa Clara, CA, pp. 152-161, February 1990.
15. C.B. Umminger and C.G. Sodini, Switched capacitor networks for monolithic image processing systems, to appear in *IEEE Trans. Circ. Syst.*
16. C. Mead, *Analog VLSI and Neural Systems*. Addison-Wesley: Reading, MA, 1989.
17. B.K.P. Horn, *Robot Vision*, MIT Press: Cambridge, MA, and McGraw-Hill, New York, pp. 48-57, 1986.
18. B.K.P. Horn, Parallel networks for machine vision. In *Research Directions in Computer Science: An MIT Perspective*, A. Meyer, G.V. Gutttag, R.L. Rivest, and P. Szolovits, eds., MIT Press: Cambridge, MA, pp. 531-572, 1991.
19. J.T. Wallmark, A new photocell using lateral photoeffect, *Proc. Inst. Radio Engin.* 45:474-483, April 1957.
20. S.P. DeWeerth and C.A. Mead, A two-dimensional visual tracking array, *Advanced Research in VLSI*, J. Allen and F.T. Leighton, eds., MIT Press: Cambridge, MA, pp. 259-275, 1988.
21. J.G. Harris, The coupled depth/slope approach to surface reconstruction, Technical Report TR-908, MIT AI Laboratory, Cambridge, MA, June 1986.
22. J. Harris, C. Koch, J. Luo, and J. Wyatt, Resistive fuses: Analog hardware for detecting discontinuities in early vision. In *Analog VLSI Implementation of Neural Systems*, C. Mead and M. Ismail, eds., Kluwer Academic, Norwell, MA, pp. 27-55, 1989.
23. J.G. Harris, Analog models for early vision, Ph.D. thesis, Cal Tech, Pasadena, CA, 1991.
24. J.G. Harris, Solving early vision problems with VLSI constraint networks. In *Neural Architectures for Computer Vision*, AAAI-88 Workshop, Minneapolis, MN, August 1988.
25. B. Horn, H.-S. Lee, C. Sodini, T. Poggio, and J. Wyatt, The first three years of the MIT vision chip project—Analog VLSI systems for integrated image acquisition and early vision processing, VLSI Memo No. 91-645, Microsystems Technology Laboratory, MIT, October 1991.
26. D. Marr and T. Poggio, Cooperative computation of stereo disparity, *Science* 194:283-287, October 1976.
27. M. Drumheller, and T. Poggio, On parallel stereo, *Proc. IEEE Intern. Conf. Rob. Autom.*, San Francisco, 1986.
28. B.K.P. Horn and E.J. Weldon, Jr., Direct methods for recovering motion, *Intern. J. Comput. Vis.* 2(1):51-76, 1988.
29. L. Dron, The multi-scale veto model: A two-stage analog network for edge detection and image reconstruction, submitted to *Intern. J. Comput. Vis.*
30. A. Lumsdaine, J.L. Wyatt, Jr. and I.M. Elfadel, Nonlinear analog networks for image smoothing and segmentation, *J. VLSI Sig. Proc.* 3:53-68, 1991.
31. M.N. Seidel and J.L. Wyatt, Jr., Settling-time bounds for switched-capacitor networks, *Proc. IMACS World cong. Comput. Appl. Math.* Dublin, Ireland, pp. 1669-1670, July 1991.

INTERNATIONAL JOURNAL OF **COMPUTER VISION**

C O N T E N T S

Special Issue: VLSI for Computer Vision
Guest Editor: James J. Clark

Guest Editor Introduction, James J. Clark.....	175
<hr/>	
A VLSI Pyramid Chip for Multiresolution Image Analysis <i>Gooitzen S. van der Wal and Peter J. Burt</i>	177
<hr/>	
Analog VLSI Circuits for Stimulus Localization and Centroid Computation <i>Stephen P. DeWeerth.....</i>	191
<hr/>	
Computing Motion Using Analog VLSI Vision Chips: An Experimental Comparison Among Different Approaches <i>Timothy Horiuchi, Wyeth Bair, Brooks Bishofberger, John Lazzaro, Andrew Moore, and Christof Koch</i>	203
<hr/>	
Analog VLSI Systems for Image Acquisition and Fast Early Vision Processing <i>John L. Wyatt, Jr., Craig Keast, Mark Seidel, David Standley, Berthold Horn, Tom Knight, Charles Sodini, Hae-Seung Lee, and Tomaso Poggio</i>	217
<hr/>	
Dynamic Wires: An Analog VLSI Model for Object-Based Processing <i>Shih-Chii Liu and John Harris</i>	231

