

CMOS Imager with Embedded Analog Early Image processor

Christophe Basset^[1], Pietro Perona^[1], Guang Yang^[2], and Bedabrata Pain^[3]

^[1] California Institute of Technology, Pasadena, CA 91125, USA. [basset, perona]@vision.caltech.edu
^[2] Dialog Semiconductor, 54 Old Hwy 22, Clinton, NJ 08809, USA
Guang.Yang@diasemi.com
^[3] Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, CA 91109, USA.
bpain@jpl.nasa.gov

Abstract

We present a high quality computational CMOS imager with integrated early image processing general-purpose filter. The early processor identifies in real-time areas of interest within the field of view, and reduces data volume and data-bandwidth to the external processor. Unlike other computational imagers, this approach does not compromise signal integrity, and unlike high-speed imagers, it does not overwhelm the external processor. On focal plane integration of image pre-processing allows an efficient implementation of a variety of computationally intensive applications such as autonomous navigation, object avoidance or intercept, and recognition. A proof of concept imager has been built and high-speed operation of the imager-processor has been confirmed.

Introduction

While high-speed imagers with varying degrees of performance are being developed [e.g.1,2], and high speed digital processors exist, signal I/O from the imager, and image processing at a high update rate required in autonomous navigation or object-avoidance scenarios remain a challenge. Existing systems involving CCD or CMOS imager arrays combined with an external computing chip [3,4] are limited both by the sheer volume of data, as well as by the bottleneck of transferring the data serially from the imager to the processing chip.

On-focal plane units [5] benefit from parallel computing but at the cost of reducing greatly the fill-factor of the pixels. Communication between non-neighboring pixels also becomes an issue. In addition, on-pixel digital or binary systems [6] do not take advantage of the full range of the signal from the imager. Multi-chip and digital systems also suffer from large power consumption. [2,6]

The new single-chip architecture presented in this paper incorporates a layer of analog early-image processing near but separate from the imager array. Thanks to this semi-parallel approach, the data volume is vastly reduced without compromising image quality, enabling efficient implementation of high-quality real-time computational imaging systems.

Architecture overview

A single-chip system combining an Active Pixel Sensor

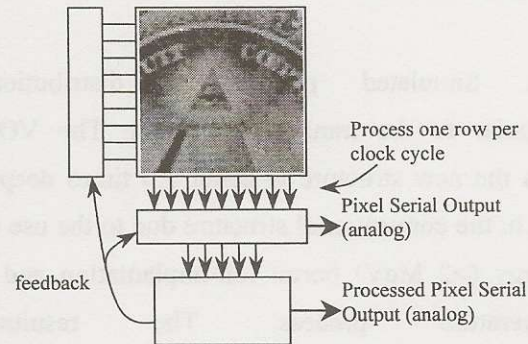


Fig. 1: Early Image Processing concept.

(APS) imager and an early image processor has been developed. Fig. 1 shows the conceptual diagram of the system.

The early-image processor consists of an array of convolution units performing a real-time estimation of the spatial convolution of the image with a template to identify in real-time possible areas of interest. Spatial convolution can be used for a variety of applications in image processing, such as edge detection, motion computation, image gradient measurements, blurring and others, depending on the kernel used. (Laplacian of Gaussians for edge detection, Gaussian for spatial averaging, Sobel derivatives, Gabor patterns for orientation...) Summation of the image pixels is also computed separately to allow the calculation of the normalized correlation when the system is used for other computationally intensive applications such as template matching in real-time tracking systems.

The filtering function applied by the convolution is entirely generic as the kernel is uploaded to the chip as a 7×7 -pixel 8-bit grayscale image. It is stored in a serial-input/parallel-output on-chip $7 \times 7 \times 8 = 392$ -bit digital memory. A 20MHz serial link (as in the test setup built) loads the template in under $20 \mu s$, making on-the-fly update possible. Once uploaded, the kernel does not change, keeping power consumption and transistor switching noise to a minimum.

The computation units (CU) convolve a 7×7 -pixel neighborhood of the image I_{ij} around the current coordinates with the template T_{ij} :

$$\sum_{i=1}^7 \sum_{j=1}^7 T_{i,j} \cdot I_{i,j} \quad (1)$$

This operation is performed on a portion of the imager by sliding the current neighborhood $I_{i,j}$ over a window of interest (WOI).

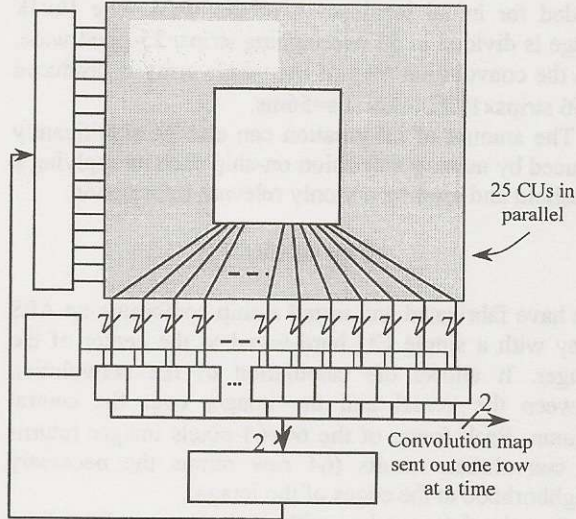


Fig. 2: Complete convolution system diagram.

Each row of pixel is read at once from the imager, opening the possibility of a semi-parallel architecture where the whole row is treated at once by the convolution core. For a 25x25 pixels WOI, a row of 25 CUs are implemented in parallel. After a latency period of 6 row-clock cycles, one convolution result is made available at each new clock cycle.

When appropriate for the application (as in tracking systems), a Winner-Take-All circuit (WTA) is used as a decision-maker to identify the precise location of the relevant feature in the WOI. A feedback loop to the imager address decoder updates the location of the WOI at each frame. See Fig.2 for a diagram of the complete system.

Computation Units

The core of the convolution is an arithmetic unit that multiplies pixel-wise the digital template with the window of interest in the imager. Pixels are sent to the computation units as currents. Details on the current-mode imager implementation can be found in the next section. Each CU contains $7 \times 7 = 49$ of these elementary units. The operation performed by each of them is:

$$T_{i,j} \cdot I_{i,j} \quad (2)$$

with i,j the coordinated of the pixel in the 7×7 window.

Therefore, for the 7×7 kernel/image, the CU receives two inputs: an 8-bit bus from the template $D[0:7]$ and an analog bus of 7 current flows I_{pixel} from the imager.

As shown in Fig. 3, the single-pixel units are split into two halves, independently treating the four LSB and the

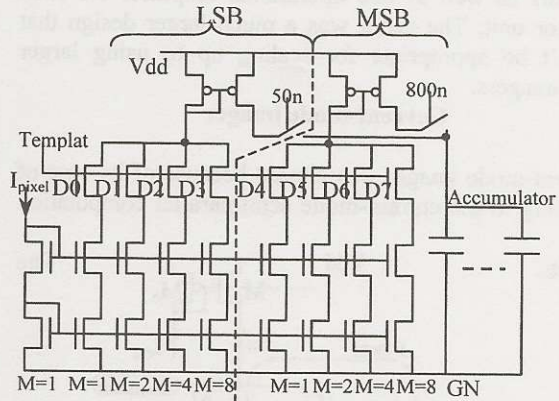
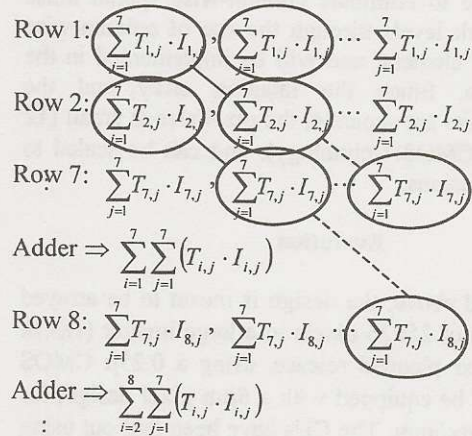


Fig.3: 1-pixel multiplying DAC. Each computation unit (CU) is made of seven of these. They share the same accumulating capacitors. One of which is read out at every clock then reset.

four MSB of the template pixel. In each part, the pixel current is mirrored four times (cascode configuration), the transistors being scaled 1-to-1, 1-to-2, 1-to-4 and 1-to-8. The 4 bits of the template control which of the mirrored currents is allowed to flow through.

The two currents produced are added by connecting them together and dumped into accumulating capacitors for different integration times (1 to 16 ratio), increasing the weight of the MSB part by a factor 16 so the reconstructed operation uses binary-scaled amplifiers of gains 1 to 128. Each row is processed in $800\text{ns} + \text{read-out time}$ (total row time is $1\mu\text{s}$ in the current test setup).

The CUs all contain a bank of capacitors storing row-wise convolutions (intermediate results needed to reconstruct the array when all necessary rows have been provided). A rolling selection is made at each cycle so one of the capacitor is readout and immediately reset. Since the pixels come from the imager one row at a time, the computation is parallel in one dimension. This scheme makes the pipeline in the other dimension:



A voltage-mode processing circuit-element was also studied. It allowed the use of better-quality voltage-mode pixels but called for binary-scaled capacitors in lieu of

transistors as well as one operational amplifier for each correlator unit. The result was a much larger design that wouldn't be appropriate for scaling up to using larger format imagers.

Current-mode imager

A current-mode imager was chosen because of its ease of interfacing to the current-mode semi-parallel computation circuit elements.

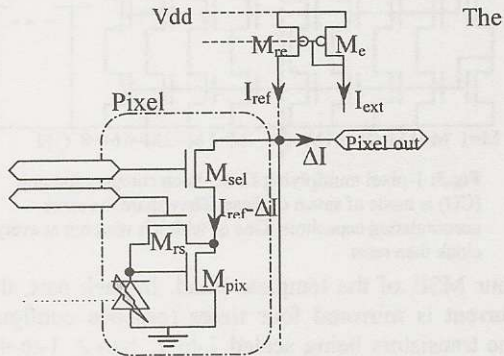


Fig.4: Schematic of a current-mode pixel

accepting currents as the image pixels input, a current-mode pixel was designed. The schematic can be seen in Fig. 4. It is similar in operation to current-mode imagers developed elsewhere [7]

A flow of current I_{ext} is provided by an external source connected to M_{ext} . It is mirrored once per column of pixels in the array (64 identical transistors M_{ref} in the chip built), providing a reference current in the column. When no row is selected, I_{ref} flows down to the read-out circuitry, unmodified.

When exposed to light, the junction capacitance of the photodiode charges, creating a voltage that biases M_{pix} . Selecting a specific row of pixels by opening M_{sel} , allows the current I_{ref} to flow through the corresponding pixel, minus a ΔI , proportional to the bias at the gate of M_{pix} .

It is possible to eliminate column-wise spatial noise and subtract dark levels through the use of column-wise current-memory element, and will be implemented in the next generation. Since the imaging array and the processing circuits are separate, the pixel size is small ($12 \mu m$ in $0.5 \mu m$ CMOS technology), and can be scaled to megapixel dimensions.

Evolution

As discussed above, the design is meant to be arrayed to cover a WOI of 25×25 pixels in a large imager ($1k \times 1k$ pixels). The next planned release, using a $0.25 \mu m$ CMOS technology will be equipped with a $6 \mu m$ pixel design, for an array of $6mm \times 6mm$. The CUs have been laid out using $0.5 \mu m$ technology and are $600 \mu m \times 1.7mm$. Since scalability is a concern, they are stackable in width and a pair fits in only $950 \mu m$, for a total of under $12mm$. We expect the transition to $0.25 \mu m$ technology to reduce the size to about

$7mm$, making it fit under the imager/address decoder bloc. Placing several blocs of 25 CUs vertically is also planned, as it would allow several kernels to be applied to the image at once.

Although it is meant to be used on a WOI at a current speed of $1 \mu s$ per row= $25 \mu s$, a scan of the entire imager is needed for initial localization of the WOI. The $1k \times 1k$ image is divided in 56 overlapping strips, 25-pixel wide, and the convolution map of the whole array is produced in $56 \text{ strips} \times 1000 \text{ rows} \times 1 \mu s = 56ms$.

The amount of information can also be significantly reduced by making a decision on-chip such as applying a threshold and sending out only relevant information.

Test results

We have fabricated and tested a chip combining an APS array with a single CU hard-wired to the center of the imager. It allows the calculation of the convolution between the kernel and the imager over the central column. Each frame of the 64×64 pixels imager returns 58 convolution results (64 row minus the necessary neighborhood at the edges of the image).

Sample pictures taken with the imager can be seen in



Fig.5: Sample images from the 64×64 pixels current-mode imager

Fig.5. FPN correction through dark frame subtraction has been done off-chip. The images fed to the convolution core are, however, unprocessed.

The more important feature tested on this chip, the convolution, showed very good performance. Fig. 6 shows the convolution linearity obtained by using a fixed template and varying image intensity. Fig. 7 indicates the convolution linearity obtained by varying the image intensity is equally good, demonstrating accurate on-chip image convolution. Deviations from the ideal linear response are small (less than 1%). Fig.8 demonstrates the convolution on an actual example: the image of a square

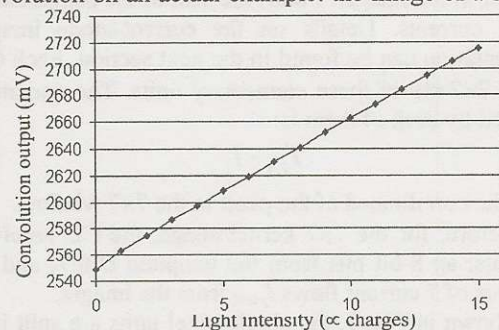


Fig.6: Convolution linearity. Flat template and image of varying intensity.

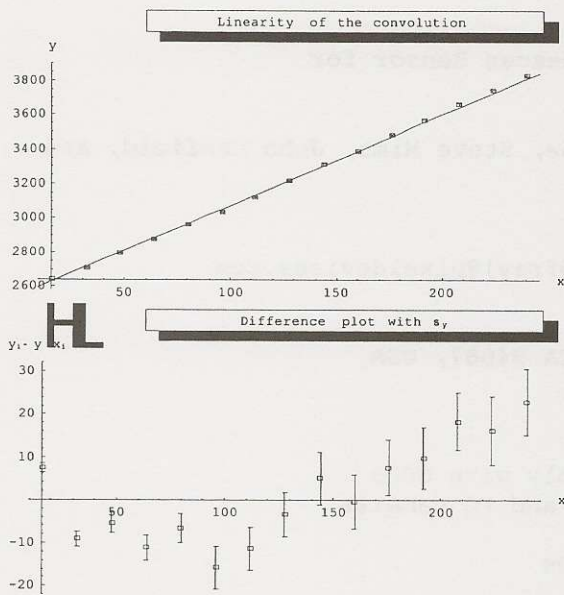


Fig.7: Convolution linearity. Fixed image, template of varying intensity. The line in (a) shows the ideal response, and (b) shows the difference plot between the ideal and the measured responses.

slides from left to right and the convolution with a template representing a square beam. The output is a triangle, as expected. The peak is slightly rounded due to the high input intensity that give high enough results to show the non-linearity of the capacitors. (Also observed in Fig.7b.) This artifact is easy to work with by properly defining the range of operation of the imager (by setting the reference current I_{ref}) and of the template.

Fig. 9 shows a photograph of the die of the fabricated chip, and Table 1 summarizes its most relevant characteristics.

Conclusion

A new architecture for an early processing of images is

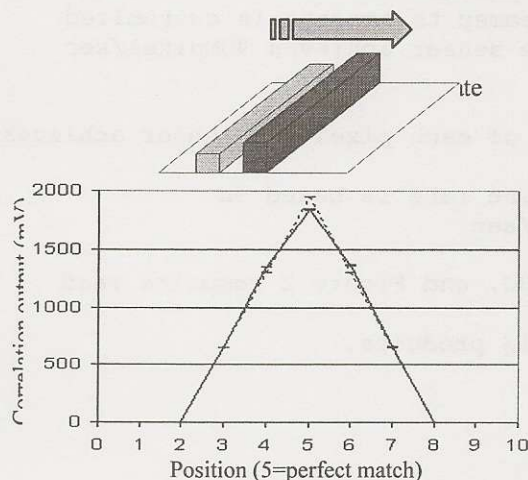


Fig. 8: Correlation of two squares. Expected response (dashed line) and measured (solid line)

presented. A current-mode APS imager interfaces on-chip with a convolution computational unit. A digital template encodes the kernel of the desired filtering function. The filter was shown to have a linear response, necessary for accurate early-image processing. The 19mW power dissipation at a frame rate of 60 images per seconds makes this approach very attractive for various filtering functions including template-matching of elementary targets – a computationally intensive task requiring real-time processing and necessary for applications such as face recognition.

Further work is planned for developing the next generation, including an improved imager and an addressable window of interest by using the filter output through a feedback loop.

Acknowledgement

The authors would like to acknowledge the help they received from Chao Sun, Bruce Hancock, Chris Wrigley and Pavani Peddada at various stages of the project.

References

- [1] Krymski et al., *IEEE Trans. Electron Devices*, vol. ED 50, pp. 136-143, 2003.
- [2] Kleinfelder et al., *IEEE J. Solid State Circuits*, vol. SC-36, pp. 2049-2059, 2001.
- [3] Biyabani et al., *Proc. IEEE Int. Solid State Circuits Conf. 1999*, pp 82-85
- [4] Gottardi et al., *Proc. IEEE Int. Solid State Circuits Conf. 1993*, pp. 194-195, 289, 1993.
- [5] Pesavento et al., *Proc. 33rd Asilomar Conference on Signals, Systems and Computers, Monterey, USA, 1999*.
- [6] Komuro et al., *IEEE Trans. On Electron Devices*, vol 50, No.1, January 2003
- [7] McIlrath et al., *IEEE Trans. on Electron Devices*, vol. ED-44, pp. 1706-1715, 1997.

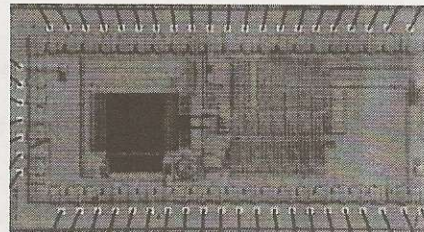


Fig. 9: Photograph of the die.

Characteristics	Values	Comments
Imager Format	64x64 pixels	
Pixel Size	12 μ m x 12 μ m	Photo-diode
Fill-factor	26%	8.5 μ m x 4.6 μ m PD
CMOS tech.	0.5 μ m 1P3M	
Chip area	6.2x2.7mm	
Computation Unit	7x7 window	Hard-wired to center columns of imager
FPN (with compensation)	~0.53%	Of saturation level
Power dissipation	<5.9 mW	Imager only (90Hz)
	<19.1 mW	With convolution

Table 1. Performance Characteristics