*P*roceedings of the 1996 IEEE
International Conference on Robotics and Automation
Minneapolis, Minnesota – April 1996

Anton Philips Prize for the Best Student Paper

A Sorting Image Sensor: An Example of Massively Parallel Intensity–to–Time Processing for Low–Latency Computational Sensors

Vladimir Brajovic and Takeo Kanade

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

The need for low-latency vision systems is growing: high speed visual servoing and vision–based human computer interface. In this paper we present a new intensity–to–time processing paradigm suitable for low–latency massively parallel global computation over fine–grained data such as images. As an example of a low–latency global computation, we have developed a VLSI sorting computational sensor — a sensor which sorts all pixels of an input image by their intensities, as the image is being sensed. The first sorting sensor prototype is a 21 by 26 array of cells. It detects an image focused thereon and computes the image of indices as well as the image's cumulative histogram, before the intensity data are readout. The image of indices never saturates and has uniform histogram. Under user's control, the chip can perform other operations including simple segmentation and labeling.

1 Introduction

As robotics is increasingly interacting with human users, the need for low-latency vision systems is growing. Existing vision systems, however, do not meet this need. For example, a standard video camera takes 1/30 of a second to transfer an image, and in many applications it is too late by the time the system receives the image. As another example, the pipelined dedicated vision hardware can deliver the processing power to update its output 30 times per second, but the latency incurred through the pipeline is typically several frame times. These two examples point to two main sources of latency in vision systems: data transfer bottleneck and the computational load bottleneck. It is clear that an alternative is needed.

The computational sensor paradigm [5] has potential to greatly reduce latency. By integrating sensing and processing on a VLSI chip both transfer and computational bottlenecks can be alleviated. On–chip routing provides high throughput transfer, while an on-chip processor could implement massively parallel computational models.

A great majority of computational sensory solutions so far implement *local operations* on a single light sensitive VLSI chip (for examples see [5] [6] [8]). Local operations use operands within a small spatial/temporal neighborhood of data and thus land themselves to the graceful implementation in VLSI. Typical examples include smoothing and edge detection. Local operations produce preprocessed images; therefore, a large quantity of data still must be read out and further inspected before a decision for an appropriate action is made — usually a time consuming process. Consequently, a great majority of computational sensors built thus far are limited in their ability to quickly respond to changes in the environment.

Global operations, on the other hand, produce fewer quantities for the description of the environment. If computed at the point of sensing, these entities could be routed from a computational sensor through a few output pins without causing the transfer bottleneck. This information will be often sufficient for rapid decision making and the actual image does not need to be readout. However, implementing global operations in hardware is not trivial. The main difficulty comes from the necessity to bring together, or aggregate, all or most of the data in the input data set [2]. This global exchange of data among a large number of processors/sites quickly saturates communication connections and adversely affects computing efficiency in parallel systems — parallel digital computers and computational sensors alike. It is not surprising that there are only a few computational sensors for global operations, all with modest capability and/or low resolution [7].

This work introduces a novel intensity–to–time processing paradigm — an efficient solution for massively parallel *global* computation over large groups of fine–grained data. By using this paradigm we have developed a sorting computational sensor — an analog VLSI chip which sorts pixels of a sensed image by their intensities. The sorting sensor produces *images of indices* that never saturate. It also provides a cumulative histogram — a global image quantity — on one of the pins *before* the image is readout.

0-7803-2988-4/96 $4.00

2 Intensity–to–Time Processing Paradigm

The intensity–to–time processing paradigm implements global operations by aggregating only few of the input data at a time. Inspired by biological vision, it is based on the notion that stronger input signals elicit responses before weaker ones. Assuming that the inputs have different intensities the responses are separated in time and a global processor makes decision only on a few inputs at a time. The more time allowed, the more responses are received; thus, the global processor incrementally builds a global decision first based on several, and eventually based on all the inputs. The key is that some preliminary decision about the environment can be made as soon as the first responses are received. Therefore, this paradigm has an important place in low–latency vision processing.

The intensity–to–time paradigm for parallel processing involves the following steps. The input data are gathered in parallel by focusing an image on the array of sensor–processor cells. An architecture of such an array is shown in Figure 1. Each cell is comprised of radiation sensitive con-

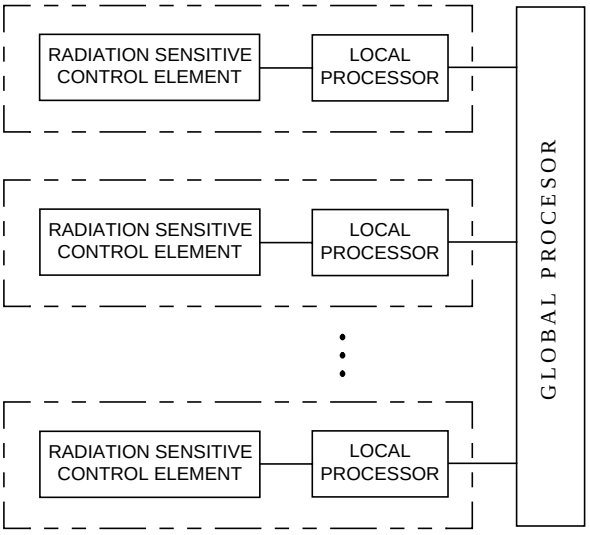

Figure 1: An architecture of computational sensor for global operations.

trol element and a local processor. The local processor in each cell performs one or more predetermined (i.e. pre–wired or pre–programmed) operations. The *instant* when this operation is executed is determined by the radiation sensitive control element and is related to the intensity of the radiation received at the particular cell. A single global processor supervises and services the array of cells. Since local processors trigger at times determined by the magnitude of their input operands, the global processor serves only a few local processors at a time.

The intensity–to–time relationship has been used to improve image segmentation [3] — a local operation, and for image processing in a SIMD architecture [4]. In contrast, our architecture allows global operations and shares some features of traditional MIMD parallel processing. Namely, the local processors perform their operations asynchronously, an essential feature for a quick response and the low latency performance of parallel systems.

3 Sorting Chip

By using the intensity–to–time paradigm we have developed a sorting computational sensor — an analog VLSI sensor which sorts the pixels of an input image by their intensities. The chip detects an image focused thereon and computes *image of indices.* The image of indices has uniform histogram which has several important properties: (1) the contrast is maximally enhanced, (2) the available dynamic range of readout circuitry is equally utilized, i.e. the values read out from the chip use available bits most efficiently, and (3) the image of indices never saturates and always preserves the same range (e.g. from 1 to N). During the computation, the chip computes a cumulative histogram — one global quantity of the detected image — and reports it with low–latency on one of the pins.

3.1 Circuitry and Operation

Shown in Figure 2, the sorting sensor is comprised of a sensor–processor cell array and a global processor. The global processor comprises of the current–to–voltage converter (resistor R), a voltage follower and two wires: W_{in} and W_{out}. The global processor communicates with the

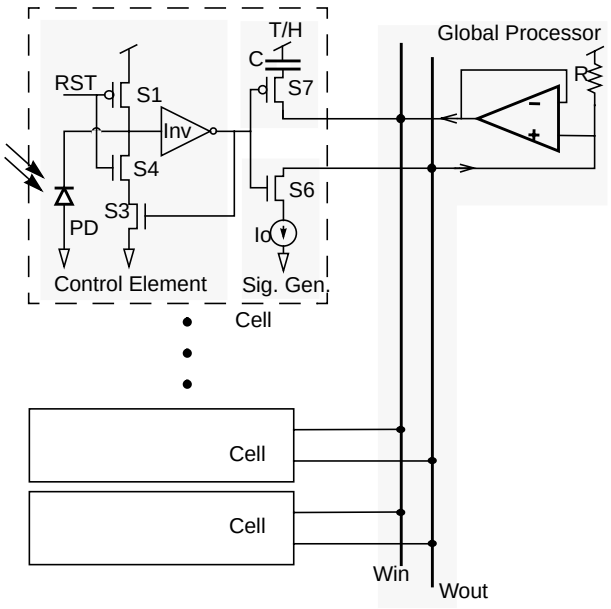

Figure 2: Schematic diagram of the sorting computational sensor

sensor–processor cells over the wires W_{in} and W_{out}. Each cell has a local processor and a photo sensitive control element. The local processor consist of a track–and–hold (T/H) circuit comprised of capacitor C and switch S_7, and a current signal generator comprised of current source Io and switch S_6. Accordingly, the local processors perform two functions: (1) remember data supplied by the global processor via W_{in} in their T/H circuits; and, (2) send their current signals I_o to the global processor via W_{out}. The remaining portion of the cell comprises the photo sensitive control element. It controls the instant when the corresponding local processor executes its functions.

Figure 3 shows the simulation of the circuit operation for the sorting sensor with four cells. A photodiode *PD* operating in the photon flux integrating mode detects the light. In this mode of operation the capacitance of the diode is charged to a high potential and left to float. Since the diode capacitance is discharged by the photocurrent, the voltage decreases approximately linearly at a rate proportional to the amount of light impinging on the diode (Figure 3, top graph).

The diode voltage is monitored by a CMOS inverter *Inv.* Once the diode voltage falls to the threshold of the inverter, the inverter's output changes state from low to high (Figure 3, second graph). A switch S_3 is included to force rapid latching action.

The output of the inverter represents a control signal produced by the photo sensitive control circuit. It controls the *instant* when the capacitor C in the T/H memorizes the voltage supplied on the wire W_{in}. It also controls the instants when the current I_o is supplied to the wire W_{out}. This is achieved by two complementary switches S_6 and S_7; one turns on the internal current source I_o, and the other disconnects the storage capacitor C from the global input wire W_{in}.

Currents from all cells are summed up on a output wire W_{out}; therefore, this wire functions as a global summer. The voltage on the resistor R (Figure 3, third graph) represents the index of a cell that is changing state and is supplied to the global input wire. The capacitor within each cell follows this voltage until it is disconnected, at which point a capacitor C retains the index of the cell (Figure 3, bottom graph). The bottom graph shows that the cell with the highest intensity input has received the highest "index", the next cell one "index" lower, and so on.

The sorting sensor computes several important properties about the image focused thereon. First, the time when a cell triggers is approximately *inversely* proportional to the input radiation received. Second, by summing up the currents I_0 from all the local processors the global processor knows at each given time how many cells have been trig-

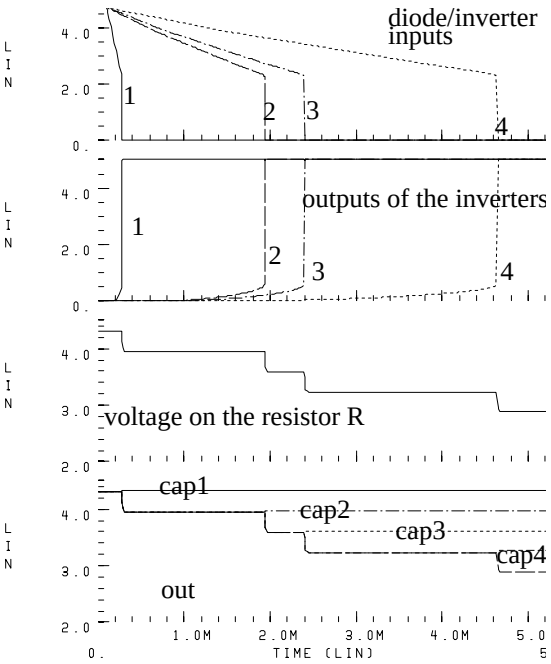

Figure 3: Sorting computational sensor: a four cell simulated operation

gered. This time waveform is closely related to a cumulative histogram of the input image. The time derivative of this signal is related to a histogram of the input image [1]. This is one global property of the input image that is reported by the chip with very low latency.

4 VLSI Realization and Evaluation

A 21 x 26 cell sorting sensor has been built in 2μ CMOS technology. The size of each cell is 76μ by 90μ, with 13% fill factor. The micrograph of the chip is shown in Figure 4. A next generation sorting computational sensor is currently being fabricated in 1.2μ CMOS technology with 38μ by 38μ pixel, and 32% fill factor.

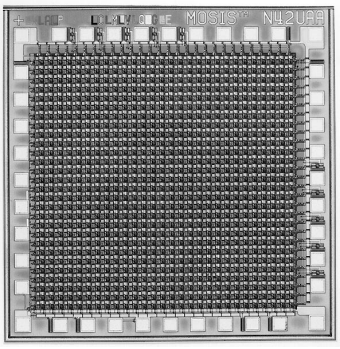

Figure 4: Micrograph of the sorting chip.

An image was focused directly onto the silicon. The cumulative histogram waveform, as well as the indices

from the sorting sensor were digitized with 12 bit resolution. In order to facilitate a hard copy reproduction the 26x21 images obtained by the sorting chip are interpolated and magnified by the factor of 2.

Scene *1* of an office environment was imaged by the sorting chip under common office illumination coming from the ceiling. Figure 5 shows the cumulative histogram of *the scene* and the image of indices both computed by the chip. We evaluated the histogram of the indices. It is shown as the bottom graph in Figure 5. Most pixels appeared to have different input intensities and, therefore, received different indices. Occasionally as many as 3 pixels were assigned the same index. Overall the histogram of indices is uniform, indicating that the sorting chip has performed correctly.

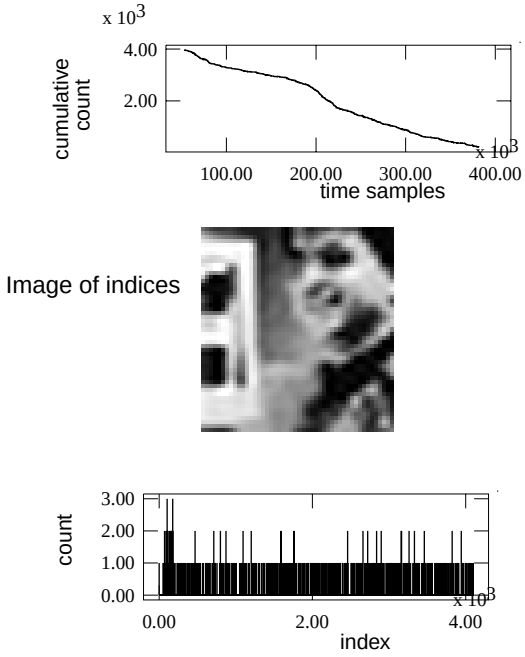

Figure 5: Scene 1 imaged by the sorting sensor.

There is a total of 546 pixels in this prototype, and most of them received different indices. This means that without special considerations as to the illumination conditions, low–noise circuit design and temperature and dark current control, our lab prototype readily provided images with more then 9 bits of resolution. Furthermore, the range of indices remains unchanged (from 0 to 545) and the indices maintain uniform histogram regardless of the range of input light intensity or its histogram.

Scene *2* from the same office was also imaged. Figure 6 shows the scenes's cumulative histogram and image of indices, as well as histogram of those indices. Scene 1 (Figure 5) contains more dark regions than Scene 2 (Figure 6) because the moderately bright wall in the background is replaced by the dark regions of the person in the

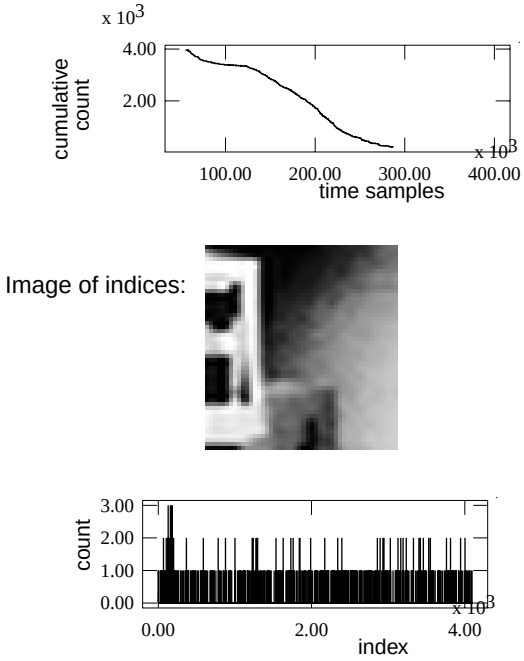

Figure 6: Scene 2 imaged by the sorting sensor.

partial shadow. Therefore, the chip takes longer to compute Scene 1 than Scene 2, but the dynamic range of the output indices is maintained. The total time shown on the time sample axis of the cumulative histograms is about *200ms.*

By producing the cumulative histogram waveform and the image of indices, the sorting computational sensor provides all the necessary information for the inverse mapping — the mapping from the indices to the input intensities. Figure 7a shows the image of indices for Scene

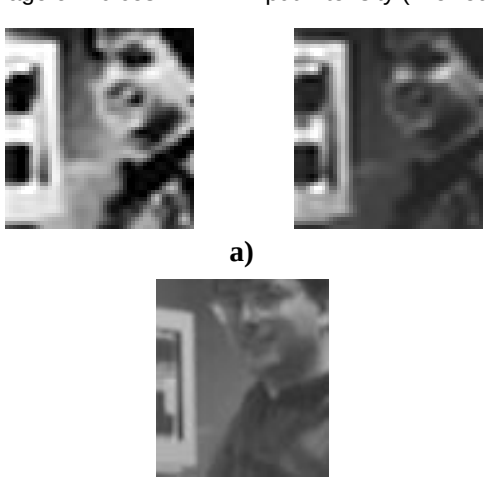

Figure 7: a) Indices from the sorting sensor and inferred input intensity, b) CCD camera image.

1 and the image of inferred input intensities. Figure 7b

includes an image taken by a commercial CCD camera for showing natural light conditions in the office environment from which Scene 1 was taken. The inferred input intensities closely resemble the natural condition in the environment.

5 Sorting Sensor Image Processing

The data that are stored in the local processors are provided by the global processor. These global data — a function of time — define a mapping from the input intensities to output data. For the sorting operation this global function is the cumulative histogram computed by the chip itself. In general, when appropriately defined this global function enables the sorting sensor to perform numerous other operations/mappings on input images. Examples of such operations include histogram computation and equalization, arbitrary point-to-point mapping, region segmentation and adaptive dynamic range imaging. In fact, in its native mode of operation — sorting — the chip provides all the information necessary to perform any mapping during the readout.

Histogram Equalization. When the voltage of the cumulative histogram (computed by the chip itself) is supplied to the local processors, the generated image is a histogram equalized version of the input image [1]. This is the basic mode of operation for the sorting chip and has been illustrated in the previous section.

Linear Imaging. When the waveform supplied to the input wire is inversely proportional to time, the values stored in the capacitors are proportional to the input intensity, implementing a linear camera. The results of such mapping have been illustrated in Figure 7. As expected, the result is similar to the image obtained by the linear CCD imager.

Scene Change Detection. Analyzing the change in the histogram pattern is a basic technique to classify images or detect a scene change. The sorting computational sensor computes the cumulative histogram at real–time and can be used for low-latency scene discrimination/surveillance without requiring the image to be read out.

Image Segmentation. Thresholding is a rudimentary technique to segment an image into regions. The cumulative histogram can be used to determine this threshold. Pixels from a single region often have pixels of similar intensity that appear as clusters in the image histogram [1]. The values which ought to be stored in the cells can be generated to correspond to the "label" of each such region. The global processor can perform this labeling by updating the supplied value when the transition between the clusters in the (cumulative) histogram is detected. An example of segmentation is shown in Figure 9b and Figure 9c in which the illuminated and shadowed regions respectively are "colored" as a black region.

Adaptive Dynamic Range Imaging. For faithful imaging of scenes with strong shadows, a huge dynamic range linear camera is needed. For example, the illumination of the scene which is directly exposed to the sunlight is several orders of magnitude greater than the illumination for the surfaces in the shadow. Due to the inherently large dynamic range of the sorting sensor, both illuminated and shadowed pixels can be mapped to the same output range during a single frame.

Figure 8: A scene with back lit objects as captured by a conventional CCD camera.

We demonstrate this concept with back illuminated objects. Figure 8 shows a global view of this scene as captured by conventional CCD camera. Due to the limited dynamic range of the CCD camera, the foreground is poorly imaged and is mostly black. (The white box roughly marks the field–of–view for the sorting sensor.)

When the scene is imaged with the sorting sensor (Figure 9a), the detail in the dark foreground is resolved, as well as the detail in the bright background. Since all 546 indices are competing to be displayed within 256 levels allowed for the postscript images in this paper, one enhancement for purpose of human viewing is to segment the image and amplify only dark pixels. The result is shown in Figure 9b. Conversely, as shown in Figure 9c, the bright pixels can be spanned to the full (8 bit) output range. Finally, if these two mappings are performed simultaneously the shadows are removed (Figure 9d.)

The same method can be applied to the image obtained from a standard CCD camera. If the CCD image of Figure 8 is cropped to the white box, and such an image is histogram equalized, we arrive at the result shown in

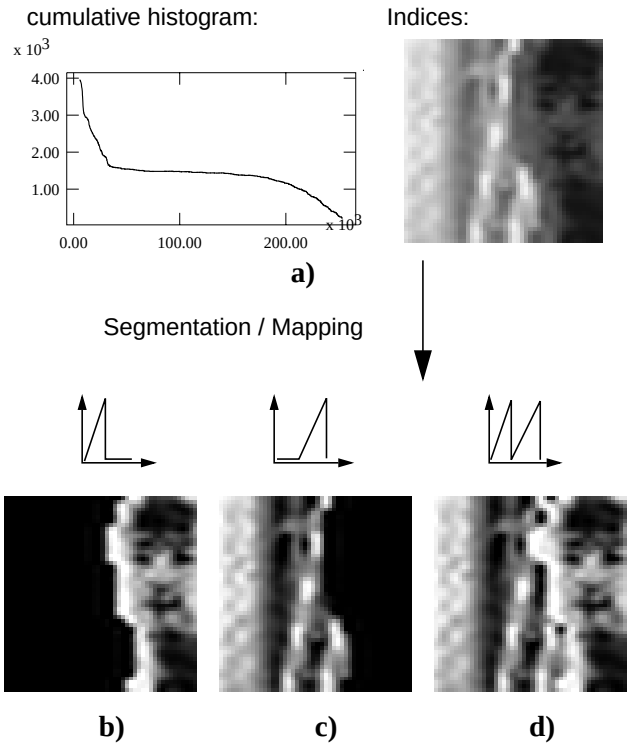

Figure 9: Sorting sensor processing: a) data from the sensors; b) segmentation (viewing the shadowed region); c) segmentation (viewing illuminated region); d) segmentation and shadow removal.

Figure 10a. This image is analogous to the image of indices obtained by the sorting sensor (Figure 9a.) Due to the limited dynamic range, noise and quantization, the CCD image only resolve the face with 2–3 bits. The histogram equalized image from the CCD is used for further mapping using the same steps as for Figure 9d. Due to the obvious reasons, the result is poor. In contrast, the sorting computational sensor allocates as many output levels (i.e. indices) as there are pixels within the dark region, or the entire image for that matter. By comparing Figure 9d and Figure 10b, the superior utilization of the sensory signal with the sorting chip is obvious.

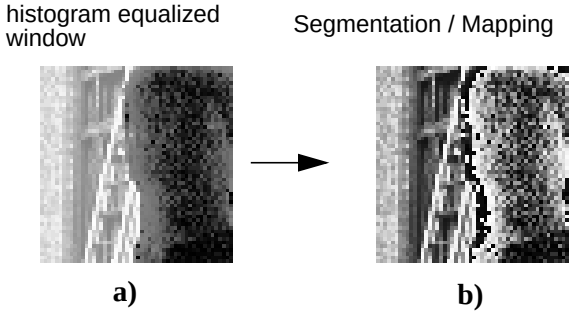

Figure 10: Conventional CCD camera processing: a) histogram equalization of the window; b) segmentation and shadow removal.

6 Conclusion

The intensity–to–time processing paradigm enables computational sensor to be a massively–parallel computational engine which makes some global computation or overall decisions about the sensed scene and reports such decisions on a few output pins of the chip with low latency. The power of this paradigm is demonstrated with an analog VLSI implementation of sorting — an operation still challenging in computer science when performed on large groups of data. This work shows that if an appropriate relationship is maintained between the circuitry, algorithm and application, a surprisingly powerful performance can be achieved in a fairly simple but high resolution VLSI vision computational sensors.

References

[1] Ballard, D.H. and C.M. Brown, *Computer Vision, Prentice-Hall*, 1982.

[2] Brajovic, V. and T. Kanade, "Computational Sensors for Global Operations", *IUS Proceedings*, pp. 621-630, 1994.

[3] Burgi, P.Y. and T. Pun, "Asynchrony in Image Analysis: using the luminance–to–response–latency relationship to improve segmentation," J. Opt. Soc. Am. A, Vol. 11, No. 6, June 1994, pp. 1720–1726

[4] Forchheimer R. and A. Astrom, "Near–Sensor Image Processing: A New Paradigm," IEEE Trans. on Image Proc., Vol. 3, No. 6, pp. 736–746, November 1994.

[5] Kanade, T. and R. Bajcsy, "Computational Sensors: A Report from DARPA workshop", *IUS Proceedings*, 1993.

[6] B. Mathur and C. Koch, *Visual Information Processing: From Neurons to Chips*, eds., *Proc. SPIE*, Vol. 1473, 1991.

[7] D. Standley, "An Object Position and Orientation IC with Embedded Imager," *IEEE Journal of Solid-State Circuits*, Vol. 26, No. 12, pp. 1853-1860, 1991.

[8] B. Zavidovique and T. Bernard, "Generic Functions for On–Chip Vision," ICPR, Conference D, pp. 1–10, The Hague, The Netherlands, 1992.