



**HAL**  
open science

## Distributed mixed-signal architecture for programmable smart image sensors

Juliette Le Hir, Anthony Kolar, Filipe Vinci dos Santos

► **To cite this version:**

Juliette Le Hir, Anthony Kolar, Filipe Vinci dos Santos. Distributed mixed-signal architecture for programmable smart image sensors. *Analog Integrated Circuits and Signal Processing*, 2018, 97 (3), pp.493-501. 10.1007/s10470-018-1342-y . hal-01943432

**HAL Id: hal-01943432**

**<https://centralesupelec.hal.science/hal-01943432v1>**

Submitted on 3 Dec 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Distributed Mixed-Signal Architecture for Programmable Smart Image Sensors

Juliette LE HIR<sup>1</sup>, Anthony KOLAR<sup>1</sup>, Filipe VINCI DOS SANTOS<sup>2</sup>

<sup>1</sup>GeePs | Group of electrical engineering – Paris, CNRS, CentraleSupélec, Univ. Paris-Sud, Univ. Paris-Saclay, Sorbonne Univ.,  
3&11 rue Joliot Curie, Gif-sur-Yvette, France  
juliette.lehir, anthony.kolar@centralesupelec.fr

<sup>2</sup>SANA | Advanced Analog Design Group, CentraleSupélec, 3 rue Joliot Curie, Gif-sur-Yvette, France

**Abstract**— Smart vision systems on a chip are promising for embedded applications. Currently, flexibility in the choice of integrated pre-processing tools is obtained at the expense of total silicon area and fill factor, which are otherwise optimized provided that the sensor performs a specific task. We propose a new architecture based on macropixel-level processing to improve the trade-off by using the same processing elements (PEs) for a whole group of pixels. In this paper, we show through transistor-level simulations the feasibility of using macropixel PEs. Their operative part is analog to avoid the bottleneck of analog to digital converters (ADC) and has digital control which is distributed in and out of the matrix of pixels. PEs are designed to be suitable for coefficient-reconfigurable spatial and temporal filtering. Sharing electronics among several pixels and matching existing algorithms to the target architecture allow for such programmability without degrading too much pixel area nor fill factor.

**Keywords**— Smart image sensor, vision system on a chip (VSoC), focal-plane array, algorithm-architecture matching, mixed analog-digital electronics.

## 1 Introduction

Smart Vision Systems-on-a-Chip (VSoCs) aim at outputting relevant information on the scene by performing low- and middle-level image processing, sometimes at the expense of image quality. Extracting image features such as edges or motion prior to transmitting it for further analysis can be a gain of speed and power consumption provided that the analog and digital processing units are co-designed and spatially distributed [1]. Such integrated imaging systems are becoming attractive for embedded applications such as drone vision thanks to their savings in area, power, weight and communication bandwidth [2]. They are also cost-effective, provided they are fabricated in standard (i.e. planar single-chip) CMOS image sensor (CIS) technology.

This paper is dedicated to the proposal, analysis and design of a new architecture for smart image sensors addressing important issues of smart VSoCs based on standard CIS processes, in particular their poor balance between reconfigurability and pixel optimization. The state of the art of VSoCs presented in section 2 shows an unavoidable trade-off between versatility and pixel pitch as well as fill factor. In section 3, we propose a new design approach to reach an optimized solution regarding this trade-off, thanks to spatial distribution of processing elements. Section 4 details the hardware architecture of a programmable sensor based on this

approach. Section 5 presents results of transistor-level simulations showing the feasibility of such an architecture, before section 6 concludes by the future work needed to implement a hardware prototype.

## 2 VSoCs State of the Art

During the last decade several smart vision sensors have been designed in standard CIS technology [1,3-17]. The increasing resolutions and frame rates result in a large data transfer between the imaging array and the processing unit. In order to avoid this highly energy consuming operation, image processing is moved as close as possible to the focal plane array. The straightforward approach is to implement in-pixel circuitry. Digital implementations offer high programmability: an example of the state of the art [3] performed edge detection, median filtering, histograms and tracking. But focal-plane digital circuits consume a lot of power and silicon area, especially through analog to digital converters (ADCs) [3, 4]. On the other hand, analog computations become more attractive for specialized tasks (implying restricted programmability) since analog operations can run faster at lower power. Tradeoffs between functionalities and surface must then be found through algorithm-architecture matching techniques. Since our work is aimed at tightly-embedded applications, with strong requirements in terms of power consumption and surface, analog implementations seem more appropriate. Therefore, we focus here on analog implementations of common processing tasks such as edge detection using spatial convolution [1, 5], difference of averaged images [6] and neighbours comparison [7]; motion detection using temporal difference [1, 7]; or image enhancement [1, 8].

In-pixel processing loosens data throughput requirements in exchange for decreased fill factor. Hence a trade-off has to be made with image quality. Moreover, image processing tasks have been proven to benefit from spatial distribution of processing circuits [9]. Therefore an improvement is to also integrate processing circuits once for the whole matrix [10] or at the bottom of each column. For example, one can take advantage of the column-wise correlated double sampling circuit to perform temporal difference [11].

On top of pixel-wise, column-wise and array-wise processing, one can consider the macropixel approach: blocks of several pixels (e.g. from 3x3 to 32x32 pixels) processed as a whole. Virtual macropixels are used for region-of-interest

detection: pixels are processed together as a virtual cluster by out-of-the-matrix electronics or software. For example, this method is applied for spatial averaging [6, 12], computing of local integration time [13] or memory optimization by pixel interlacing [14]. On the other hand, the concept of macropixels can be implemented in hardware by mutualizing in-matrix circuitry for the block of pixels instead of repeating it in every pixel. Suárez *et al.* [15] proposed such a hardware macropixel: 4 photodetectors share an amplifier and an ADC. A solution for Gaussian filtering is also implemented in [16], but it relies on a full resolution switched-capacitor network which does not really take advantage of the macropixel concept.

In short, smart vision sensors currently perform one or several simple tasks such as: edge and/or motion detection [7], edge detection, high dynamic range and tracking [16], motion detection or low power imaging by programming pairs of pixels [17]. However, none of these systems grants real programmability in the choice neither of algorithms nor of the coefficients.

On the other hand, analog programmable VSoCs have been proposed in [1, 8, 12], but with only in-pixel processing circuits and thus they suffer from very low fill factor (e.g. 5,4% in [12]).

A key observation is that distributing analog processing in the matrix improves the area/programmability trade-off. Though a few programmable sensors do exist, there seems to be a lack of a tightly integrated solution. Therefore, in section 3, we introduce a new design approach, furthering the macropixel concept, for a highly distributed fully configurable smart image sensor.

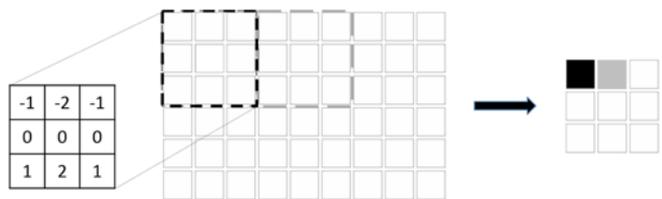
### 3 Algorithm-Architecture Matching for Distributed Electronics

The goal of this work is to develop a smart image sensor, embedding digitally controlled analog processing circuits allowing for fully programmable image pre-processing tasks in the focal plane. This limits data transfers out of the system and thus energy consumption, by extracting relevant information as close as possible to the source. By distributing processing electronics between different levels - pixel, macropixel(s), column and whole matrix -, embedding more electronics for versatility purpose becomes possible without degrading significantly other characteristics such as fill factor or pixel size, so that smart high resolution sensors can be fabricated at low cost on standard CIS technology.

The idea is to map common image processing operations to processing circuits that are distributed all over the matrix. In particular, we consider moving away from pixel by pixel operations towards macropixel-level processing in both spatial and temporal image analysis tasks. Moreover, globalized programmable processing elements allow for electronic resources reuse for different tasks.

#### 1 Spatial Convolution

Spatial convolution is widely used in pre-processing tasks such as edge detection or filtering, so efficiently implemented coefficient-programmable spatial convolution is of great interest. It has been done at pixel-level [1,5] but this implies high sensing surface loss in each pixel. A new solution is



**Fig. 1** Principle of the down-sampled convolution: linear combination of a Sobel mask with pixels of the image is done once instead of nine times for a 3x3 kernel

proposed here using macropixel-level implementation. The idea is to limit the number of processing elements (PEs) and interconnections inside the matrix. Therefore each pixel is linked to only one PE, and one PE manages as many pixels as the size of the mask (i.e. kernel), for ease of use of the control. Each PE is identical and performs the linear combination of the linked pixels weighted by the chosen coefficients of the mask. The result is then a down-sampled convolution since there is no superposition of the kernels (see Fig. 1). Hence drastic data and in-matrix circuitry reduction is obtained (division by the size of the mask) at the cost of quality loss due to downsampling.

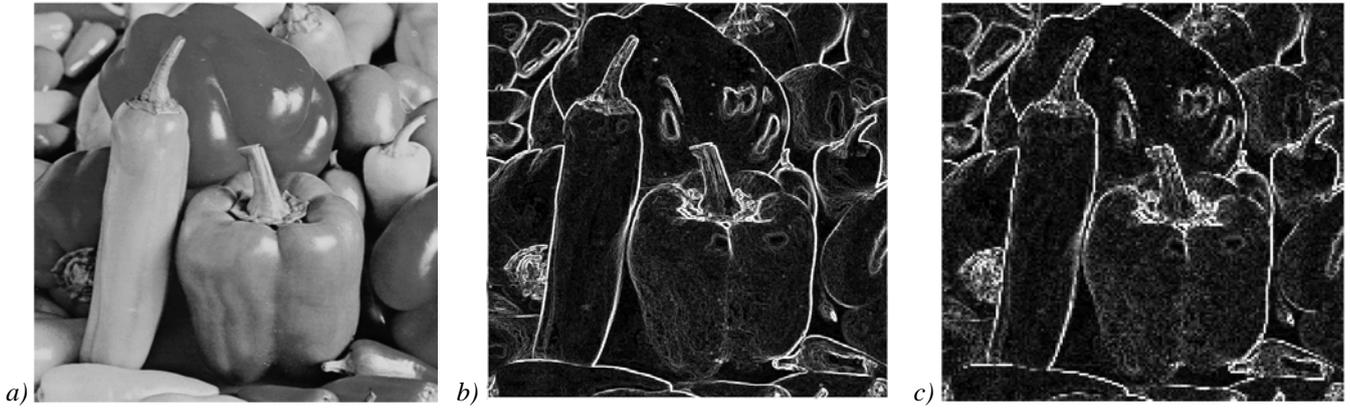
This theoretical adaptation of convolution has been functionally tested through Matlab simulations. An illustrative result is displayed on Fig. 2.

This down-sampled convolution has also been applied to the Histogram of Oriented Gradient (HOG) algorithm, which is widely used for pedestrian detection [18]. The first step of this algorithm is the gradient computation, which can be done by applying a  $\{-1\ 0\ 1\}$  mask or a Sobel mask, or else directly in polar coordinates [19].

Results with SVMs (Support Vector Machines) trained by 600 positive images and 600 negative images from an INRIA dataset, for each algorithm, are listed in Table 1. Tests were conducted on 200 positive and 100 negative images from the rest of the INRIA dataset. Optimizing the training of the SVM is out of the scope of this paper. We simply used the same sets of images to qualitatively compare different low-level algorithms. Table 1 shows that using down-sampled convolution on cartesian or polar gradients gives comparable results to an implementation of classic HOG algorithm for false negative images. The down-sampled convolution shows a much higher rate of false positive detections. This can be explained by the fact that during training of positive images, edges can be lost and

TABLE 1: COMPARISON OF IMPLEMENTATIONS OF HOG ALGORITHM

	Type of HOG				Classic HOG - single large pixel
	Classic HOG	Down-sampling	Polar gradient	Down-sampling of polar gradient	
False positive images (%)	0	14	0	4	13
False negative images (%)	6	8	5	8.5	10



**Fig. 2** (a) Original image and (b,c) results of gradient computation using (b) classic convolution by a Sobel kernel (equivalent to 1 PE per pixel) and (c) down-sampled convolution with same kernel (equivalent to 1 PE per 9 pixels), performed on the classic image of peppers (512x512 pixels result in (c) 170x170 pixels)

thus the SVM is considering non-pedestrian edges as pedestrian edges. For most applications, such as military detection of suspect person or pedestrian detection for automotive avoidance, false positive detections are not a critical issue. So for a negligible loss of quality, the proposed method divides by 9 the amount of electronic processing in the matrix for a 3x3 kernel convolution.

Besides, errors in the calculation of the downsampled convolution were simulated through addition of a normal law of chosen standard deviations. Simulations showed that to keep false positive and negative results comparable to those obtained with the ideal HOG algorithms, the standard deviation of the final error must be kept below 0.25. Having an adapted algorithm still effective with up to 25% error of cumulated computation loosens constraints on an analog implementation.

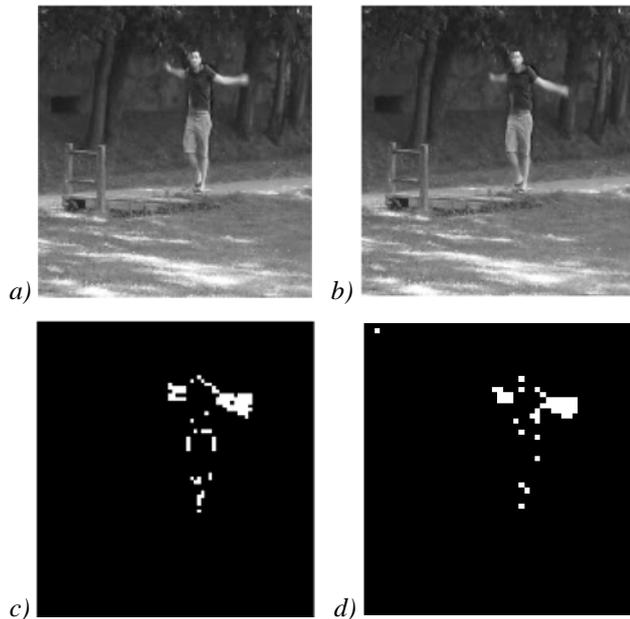
## 2 Temporal Difference

The same design methodology has been applied to temporal difference, which is a common technique for motion and Regions of Interest detection. Downsampling by 3x3 pixels seems to induce too much information loss, but downsampling by 2x2 appears to be a better trade-off between area saving and quality, as shown in Fig. 3.

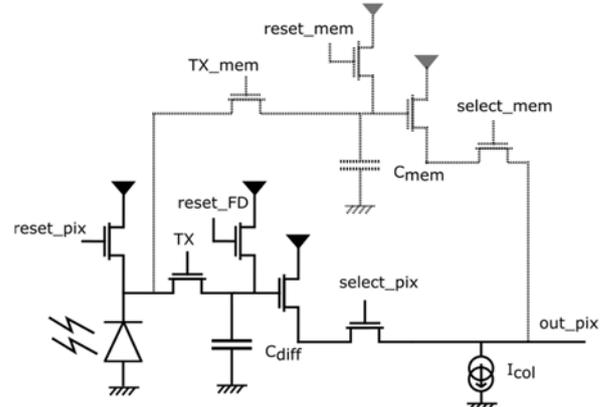
Concerning spatial convolution, for a similar fill factor and final resolution, one could suggest using classic 3x3 convolution, with a single large pixel instead of a group of 9. This scheme is evaluated through HOG algorithm in the last column of Table 1. Results are comparable with other implementations, but a 2x2 temporal difference on fine pixels could not be implemented for instance.

## 3 Resources Reuse

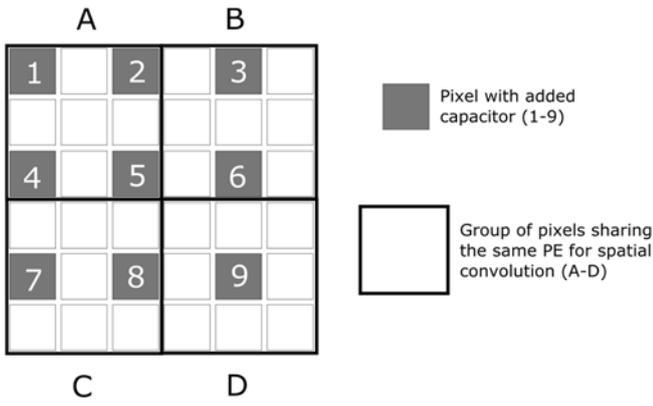
The presence of a coefficient-programmable processing element allows for reusing it for different tasks. For instance, temporal difference can be computed with the same PE as spatial filtering using the appropriate set of coefficients. Note that if one PE is assigned to 3x3 pixels while temporal difference is to be computed on a 2x2 basis, a certain sequence of operations must be carried out. It takes longer than having a PE devoted to each temporal difference in parallel. But this is acceptable given the versatility gained with little added circuitry.



**Fig. 3** Temporal difference: (a) image from a video of a walking and waving man, (b) following image of the video, and thresholded differences of the two previous images with (c) a 1/(2x2) down-sampling or (d) a 1/(3x3) down-sampling. (a) and (b) have a resolution of 240x320 pixels, and so (c) is 120x160 and (d) is 80x107 pixels

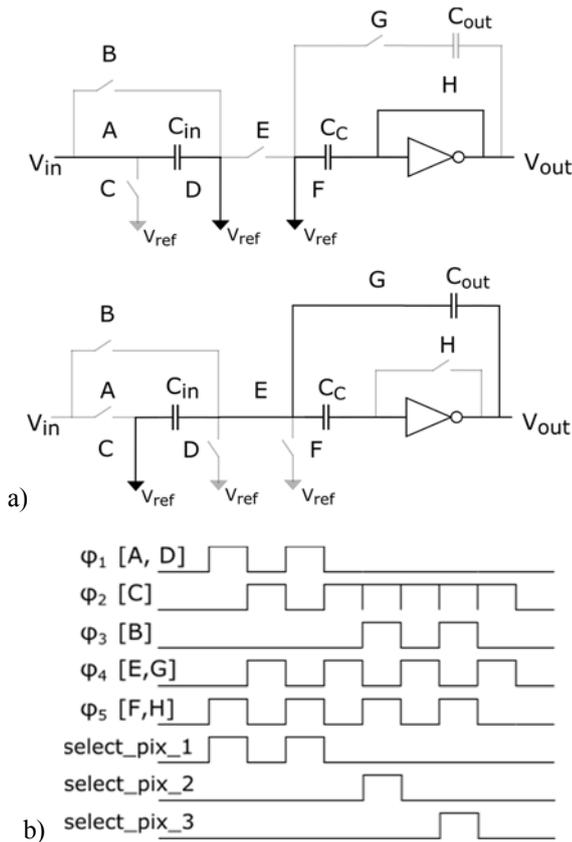


**Fig. 4** Schematic of the 5T-pixel, with additional circuit (in grey) to memorize one frame (only for 1 pixel out of 4)



PE	A	B	C	D
Pixels with memory	1; 2; 4	3; 6	7; 8	5; 9

**Fig. 5** Schematic representation of the 6x6-pixel basic unit of the proposed matrix: 1 PE for 3x3 pixels, and 1 capacitor for 2x2 pixels. This should be replicated as much as necessary to build the whole matrix. The table presents the four PEs of a 6x6-pixel scheme and the pixels with added memory that they respectively manage during a temporal difference. Note that capacitor 5 of the pattern is to be connected to the “D” PE so that each PE manages 3 capacitors at the most



**Fig. 6** (a) Schematic of a switched-capacitor circuit able to perform positive and negative accumulations of a voltage value. Top : forward charging phase. Bottom : transfer phase. (b) Corresponding timing diagram of the control of the analog accumulator for a  $[2 -1 -1, 0 0 0, 0 0 0]$  mask.  $\phi_1$  controls switches A and D,  $\phi_2$  controls C,  $\phi_3$  controls B,  $\phi_4$  controls E and G, and  $\phi_5$  controls F and H

Besides, analog memory of a frame is usually obtained through storing capacitors. A suitable pixel implementation is presented on Fig. 4. It features a 5T-pixel with a photodiode reset to avoid blooming during the pre-processing computations (the pixel value being stored on the floating diffusion  $C_{diff}$ ). A second chain of transfer gate-floating diffusion-source follower-select transistor is added on the pixels used for temporal difference. An available capacitor linked to a pixel permits high dynamic range (HDR) imaging, since this storage capacitor can receive the charge surplus from the photodiode under high illumination. Therefore, this overflow capacitor allows for extended range of sensed illumination [20]. In our proposed implementation, using the memory capacitor as an overflow capacitor would result in a downsampled HDR image.

#### 4 A Distributed Architecture

The proposed architecture consists of macropixels of a fixed size. Most kernels are 3x3 pixels (e.g. Sobel kernel) so we propose to fit this kernel size to the hardware architecture of PEs: macropixels are group of 3x3 pixels in this architecture. Temporal difference is on a 2x2 basis. Therefore there are 4 types of macropixels that differ from each other in terms of position and number of added memories, as shown in Fig. 5, but that are otherwise identical. This results in a 6x6-pixel scheme which is the basic tile for designing a generic resolution image sensor.

An example of operation using these macropixels is as follows. If a spatial convolution is requested, each PE performs it in its own macropixel, i.e. performs the linear combination of 3x3 pixels. The result is a convoluted image with a downsampling of 3x3. If a temporal difference is demanded, first a full resolution image is taken and then a second 2x2 downsampled image is taken and stored in the added memories. Then PEs perform the difference of the two images, on one pixel with memory for each PE, the matrix of macropixels is then read, and the difference is done on the second pixel with memory of each macropixel before another reading of the matrix, etc. Following this scheme, the “D” PE in the 6x6 scheme (Fig. 5) would work only once while the “A” PE would work four times meaning four computations and matrix readouts would be necessary. To limit this loss of speed of the system due to sequential computation, the workload is distributed among the PEs: pixel and memory #5 are managed by the “D” PE when temporal difference is demanded. The final result is a difference of two successive images with a downsampling of 2x2.

Since PEs must be able to perform convolutions, multiply and add operations are required. A parallelized implementation would imply as many multipliers as mask coefficients. The huge area cost takes it out of consideration. Moreover, it would be wasteful since masks containing zero coefficients would leave several multipliers unused. So we chose to have one multiplier and one accumulator for the sake of area and efficiency, at the cost of some velocity due to the sequential flow of operations. Note that those multiplier and accumulator are implemented in the analog domain, so that there is no ADC, which ensures fast computation and area savings.

To go further in the sequential computation, one may consider multiplication done by multiple accumulations of the same value. This technique avoids the use of a multiplier but

allows only integers in the convolutive masks. This is not overly restrictive in our application since most of used masks or temporal difference only use integers. Other algorithms studied in the previous section (based on polar gradient) use non-integers coefficients but since their results are of similar quality as the integer-based masks, we focus on the latter.

A potential implementation is to use a switched capacitor circuit since they are well suited for integration (i.e. accumulation). A possible solution is presented in Fig. 6. Such a circuit performs accumulation with two repeated steps: first the input voltage is sampled on  $C_{in}$  capacitor, and then charges are transferred on  $C_{out}$  capacitor where they are accumulated over steps. As an amplifier, an inverter is used because only low gain is needed and this saves area. An offset compensation scheme (capacitor  $C_c$ ) is used to cancel the influence of the offset of the inverter which changes according to process [21, 22]. Switch  $B$  allows for subtraction by reverse charging of the input, while switches  $E$  and  $F$  maintain the offset compensation. Convolution is enabled by connecting the outputs of the 9 pixels successively to the input of such an analog PE.

The PEs are digitally controlled (through the switches in a switched capacitor implementation), and that can be done from the out-of-matrix digital logic. This would mean numerous control buses crossing the whole matrix. Instead, we propose to distribute digital control over the matrix as well as analog operative circuits, since this implies less metal tracks at the cost of only a few logic elements in each macropixel.

This architecture is illustrated in Fig. 7. The exterior digital part controls the pixels (*reset\_pix*, *TX*, *reset\_FD*, *TXmem*, *reset\_Fdmem*, see Fig 4.) and starts the PE with *proc\_enable* (00 for idle, 01 for classic 3x3 mask, 10 for temporal difference). Then the PE sequentially selects the needed pixels or capacitors (*select\_pix* or *select\_mem*), multiplying their output by the corresponding coefficient (*coeff*) coming from the external digital control. Once the accumulation is done, the macropixel acknowledges (*ack*) and waits to be selected and read (*sel\_macro*). If macropixels have digital outputs (e.g. simple 1-

bit thresholding), they can be read all at once, and thus the system would be much faster.

The architecture can be easily modified to also permit classic convolution or 5x5-mask convolution, at the cost of more complex sequences for the control of the analog operative parts. Output rate of the sensor might thus be lowered in those cases, but very few electronics have to be added.

## 5 Transistor level simulations

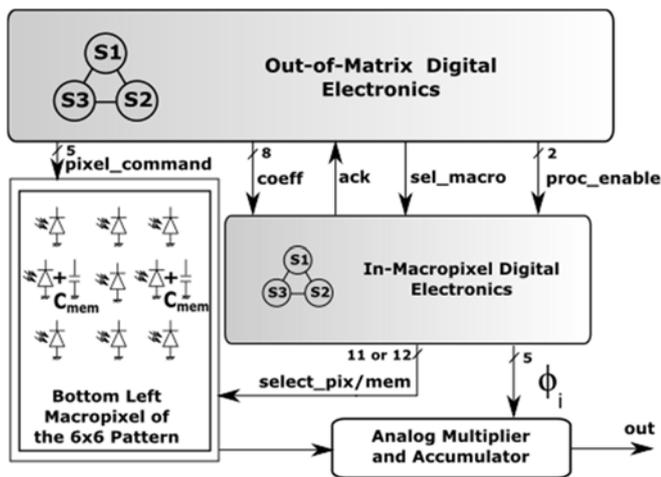
The system was designed and simulated at transistor level in the AMS 0.35 $\mu$ m technology.

### 1 Spatial Convolution

To validate our system, simulations of spatial convolution were conducted. A Sobel mask such as the one presented in Fig. 1 detects edges which can be used for pedestrian detection for instance. When it is applied to a macropixel, the value of the first pixel is charged backward in the accumulator for the -1 coefficient and transferred on the output capacitor. Then the value of the second top pixel is charged backward and accumulated, twice for the -2 coefficient, etc. until the value of the bottom right pixel is charged forward in the accumulator and accumulated. So computation of a Sobel mask corresponds to 8 accumulations only.

The computation of Sobel masks (one for vertical and one for horizontal edges) on a complete image was simulated at transistor level. Results are shown on Fig. 8. The horizontal and vertical edges are clearly visible, as in Fig. 2, which presented results of functional simulations on the same image. So this shows that the analog processing part performs as expected.

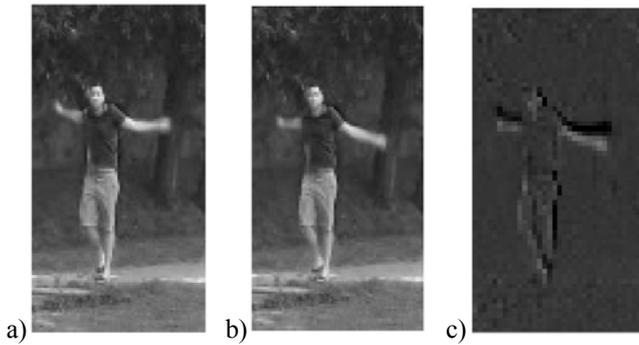
The histogram of oriented gradient (HOG) algorithm was applied to the electrically simulated images, corresponding to the second column of Table 1 (downsampling of classic HOG). The same SVM algorithm showed rates of 11% of false positive and 8% of false negative images. These results are as good as theoretical results (see Table 1) using the same machine learning and HOG algorithms, whose optimization is out of the scope of this work. This indicates this transistor-level implementation of such a smart sensor is appropriate for pedestrian detection for example.



**Fig. 7** Schematic representation of the proposed sensor: out-of-matrix digital electronics control a macropixel PE, composed of digital control interfaced with the general out-of-matrix control, and an analog operative part: 1 multiplier and 1 accumulator per macropixel (3x3 pixels)



**Fig. 8** Gradient computed from the results of Sobel transistor-level simulations of our system on the image of the peppers (see Fig. 2)



**Fig. 9** Results of transistor-level simulations: a) first image (132x66 pixels); b) second image of the video; c) result of temporal difference with our system (66x33 pixels)

## 2 Temporal Difference

The implementation was also simulated for temporal difference. Results from the same video as in Fig. 3 of a man walking and waving his arms are presented on Fig. 9. The simulation shows that moving limbs are perfectly identifiable, as expected from the previous behavioral simulation results.

Considering results on pedestrian detection and temporal difference, both the functional approach of spatial distribution of the PEs on a macropixel-level and its implementation through the described switched-capacitor circuit are shown to be appropriate for our applications.

The computation time of the switched capacitor circuit is  $0.2\mu\text{s}$ /accumulation. The different macropixels perform the operations in parallel (like a Single Instruction Multiple Data processor), so that the size of the focal plane array has no effect on the computation time. Since a Sobel kernel for example requires only 8 accumulations, and a temporal difference  $3\times 2$  accumulations, the computation time is very short compared to the light integration time for this kind of technology (standard AMS  $0.35\mu\text{m}$ ), or the readout of the array of pixels. The sole bottleneck of the system could be that 3 readouts of the array are required for each pair of successive images in the temporal difference algorithm.

## 6 Conclusion

This paper has described a new approach to designing smart image sensors. To increase versatility while keeping reasonable fill factor and pixel area, the concept of hardware macropixel is used. Thanks to algorithm-architecture matching, spatial filtering and temporal difference are adapted to be computed using digitally controlled analog processing elements distributed in each macropixel. We show through simulations that the loss of quality is inconsequential for a subsequent high-level image processing such as pedestrian detection, whose circuit implementation is out of the scope of this work. A general architecture for such a sensor has been presented, using an analog accumulator along with part of the digital control in the macropixel, and general digital control out of the matrix. An implementation of the accumulator with a switched-capacitor circuit has been presented. Transistor-level simulations with a  $0.35\mu\text{m}$  technology demonstrated that it is sufficient for both temporal difference and Sobel convolution for pedestrian detection. The layout of a chip implementing these concepts is

currently underway. We aim at achieving 30% fill factor with one switched capacitor circuit (operative part) per macropixel. In the target AMS  $0.35\mu\text{m}$  technology, the same kind of pixel without a PE nor memories reaches 37% fill factor which means the loss of fill factor is quite limited. Our architecture is aimed at focal plane arrays, but would find even greater benefit when used with 3D stacking technology, as could be discussed in furtherwork.

## References

- [1] N. Massari, M. Gottardi, L. Gonzo, D. Stoppa and A. Simoni (2005). "A CMOS image sensor with programmable pixel-level analog processing", *IEEE Trans. Neural Netw.*, Vol. 16, pp. 1673-1684.
- [2] E. Fossum (1997). "CMOS image sensors: electronic camera-on-a-chip", *IEEE Trans. Electron Devices*, Vol. 44, pp.1689-1698.
- [3] J. Schmitz, M. Gharzai, S. Balkir, M. Hoffman, D. White and N. Schemm (2017). "A 1000 frames/s vision chip using scalable pixel-neighborhood-level parallel processing", *IEEE J. Solid-State Circuits*, Vol. 52, pp. 556-568.
- [4] A. Zarándy, *Focal-plane sensor-processor chips*, Springer, 2011.
- [5] J. Dubois, D. Ginhac, M. Paindavoine and B. Heyrman (2008). "A 10 000 fps CMOS sensor with massively parallel image processing", *IEEE J. Solid-State Circuits*, Vol. 43, pp. 706-717.
- [6] N. Katic, A. Schmid and Y. Leblebici (August 2014). "A retina-inspired robust on-focal-plane multi-band edge-detection scheme for CMOS image sensors", *Midwest Symp. on Circuits and Systems*.
- [7] D. Kim and E. Culurciello (2013). "Tri-mode smart vision sensor with 11-transistors/pixel for wireless sensor networks", *IEEE Sensors J.*, Vol. 13, pp. 2102-2108.
- [8] J. Fernández-Berni, R. Carmona-Galán and L. Carranza-González (2011). "FLIP-Q: a QCIF resolution focal-plane array for low-power image processing", *IEEE J. Solid-State Circuits*, Vol. 46, pp. 669-680.
- [9] H. Li, Z. Zhang, J. Yang, L. Liu and N. Wu (November 2015). "A novel vision chip architecture for image recognition based on convolutional neural network", *IEEE 11th International Conf. on ASIC*.
- [10] B. Zhao, X. Zhang, S. Chen, K.-S. Low and H. Zhuang (2012). "A  $64\times 64$  CMOS image sensor with on-chip moving object detection and localization", *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 22, pp. 581-588.
- [11] N. Massari and M. Gottardi (2007). "A 100dB dynamic-range CMOS vision sensor with programmable image processing and global feature extraction", *IEEE J. Solid-State Circuits*, Vol. 42, pp. 647-657.
- [12] R. Carmona-Galán, J. Fernández-Berni and A. Rodríguez-Vázquez (2015). "Automatic DR and spatial sampling rate adaptation for secure and privacy-aware ROI tracking based on focal-plane image processing", *Image Sensors Workshop*. <http://imagesensors.org/2015-paper/> Accessed 10 January 2018.
- [13] A. Peizerat *et al.* (2015). "A 120dB DR and  $5\mu\text{m}$  pixel pitch imager based on local integration time adaptation", *Image Sensors Workshop*. <http://imagesensors.org/2015-paper/> Accessed 10 January 2018.
- [14] J. Martel, M. Chau, M. Cook and P. Dudek (August 2015). "Pixel interlacing to trade off the resolution of a cellular processor array against more registers", *European Conf. on Circuit Theory and Design (ECCTD)*.
- [15] M. Suárez *et al.* (September 2013). "A  $176 \times 120$  pixel CMOS vision chip for gaussian filtering with massively parallel CDS and A/D conversion", *European Conf. on Circuit Theory and Design (ECCTD)*.
- [16] C. Yin, C.-F. Chiu and C.-C. Hsieh (2016). "A 0.5V, 14.28kframes/s, 96.7dB smart image sensor with array-level image signal processing for IoT applications", *IEEE Trans. Electron Devices*, Vol.63, pp1134-1140.
- [17] J. Choi, S. Park, J. Cho and E.Yoon (2014). "A  $3.4\text{-}\mu\text{W}$  object-adaptive CMOS image sensor with embedded feature extraction algorithm for motion-triggered object-of-interest imaging", *IEEE J. Solid-State Circuits*, Vol. 49, pp.289-300.
- [18] N. Dalal and B. Triggs (June 2005). "Histograms of oriented gradients for human detection", *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*.

- [19] C. Bourrasset, L. Maggiani, C. Salvadori, J. Sérot, P. Pagano and F. Berry (2013). "FPGA implementations of Histograms of Oriented Gradients in FPGA", *Workshop on Architecture of Smart Cameras (WASC) 2013*. <http://www.eunevis.org/wasc2013/index.php/en/programwasc.html> Accessed 10 January 2018.
- [20] S. Sugawa, N. Akahane, S. Adachi, K. Mori, T. Ishiuchi, K. Mizobuchi (February 2005). "A 100dB dynamic range CMOS image sensor using a lateral overflow integration capacitor", *IEEE International Solid-State Circuits Conf.*
- [21] Y. Chae and G. Han (2009). "Low voltage, low power, inverter-based switched-capacitor delta-sigma modulator", *IEEE J. Solid-State Circuits*, Vol. 44, pp.458-472.
- [22] P. Bisiaux, C. Lelandais-Perrault, A. Kolar, P. Benabes, F. Vinci dos Santos (2017). "A 14-b two step inverter-based sigma-delta ADC for CMOS image sensor", *IEEE International New Circuits and Systems Conference (NEWCAS)*.