A 33 μW 64 $\times$ 64 Pixel Vision Sensor Embedding Robust Dynamic Background Subtraction for Event Detection and Scene Interpretation

Nicola Cottini, Massimo Gottardi, *Member, IEEE*, Nicola Massari, *Member, IEEE*, Roberto Passerone, *Member, IEEE*, and Zeev Smilansky

***Abstract*—A 64 $\times$ 64-pixel ultra-low power vision sensor is presented, performing pixel-level dynamic background subtraction as the low-level processing layer of an algorithm for scene interpretation. The pixel embeds two digitally-programmable Switched-Capacitors Low-Pass Filters (SC-LPF) and two clocked comparators, aimed at detecting any anomalous behavior of the current photo-generated signal with respect to its past history. The 45 T, 26 μm square pixel has a fill-factor of 12%. The vision sensor has been fabricated in a 0.35 μm 2P3M CMOS process, powered with 3.3 V, and consumes 33 μW at 13 fps, which corresponds to 620 pW/frame.pixel.**

***Index Terms*—CMOS vision sensors, energy autonomous low-power sensors, low-power, visual processing, wireless sensor networks.**

I. Introduction

VISUAL information is the richest source of information describing our surrounding environment. At present, imagers are largely used in battery-powered consumer electronics, such as mobile phones, camcorders, tablets and toys, embedding one or two high resolution cameras, which are used to take pictures, record movies and communicate through video and audio. These imagers, which typically have megapixel resolution, continuously deliver data at frame rate requiring large communication bandwidth and power. Even a low-cost VGA imager consumes about 70 mW, which represents a significant value for a mobile device. In fact, a battery-powered system cannot afford to keep a conventional camera working for long periods, without cutting down dramatically the battery lifetime.

There are however applications which do not require accurate and highly resolved images to be able to function and take a decision. These kind of applications deal, for instance, with monitoring of people or objects in domestic rooms or moving around a dangerous or an off-limit zone. Infrastructures represent a big obstacle in the diffusion of such systems, which need to be placed all around the area to be monitored. This makes battery-operated systems an attractive solution, reducing both installation and maintenance costs. Present day, conventional wireless video systems still consume too much power to enable long-term battery powered operation. For these systems, however, the imager can be customized for an efficient use of the available energy resulting in a significant improvement of the system lifetime. Ultra-low power devices must therefore continuously monitor the scene, extracting useful information and delivering data only when significant events have been detected, delegating high-level processing to an external computing platform.

The interest on ultra-low power imaging started years ago [1]–[11] and is constantly growing, aiming toward μW and sub-μW cameras targeting also applications such a implantable retinas [12] and distributed wireless sensor networks [3], [10], where power consumption is of main concern. In this paper, we describe the implementation and experimental results of a novel ultra-low power vision sensor, embedding a robust, VLSI-oriented algorithm for temporal contrast detection [13], forming the low-level part of an image processing algorithm for scene interpretation [14]. The 4 k pixel sensor consumes 33 μW at 13 fps, powered at 3.3 V. The 45-transistor pixel detects and binarizes anomalous intensity changes with programmable response time, according with the specific application. The sensor is programmed and controlled in a closed loop with the external off-chip logic.

Section II covers the basic techniques for image background subtraction. Section III discusses the VLSI-oriented algorithm for Dynamic Background Subtraction. Section IV addresses the pixel implementation issues, while details on the sensor architecture are covered in Section V. Experimental results are reported in Section VI. Section VII finally compares our work and results with other sensors which have been recently implemented.

II. Background Subtraction

Detecting an event occurring in the scene requires the following problem to be addressed:

"Given a frame sequence from a fixed camera, detect the foreground objects."

Manuscript received October 05, 2012; revised November 22, 2012; accepted December 01, 2012. Date of current version February 20, 2013.This paper was approved by Associate Editor Hideto Hidaka. This work was supported in part by the Project "A Battery Operated Vision System for Wireless Applications" (BOViS) within the Italy–Israel R & D Cooperation Agreement, 2009–2010.

N. Cottini, M. Gottardi, and N. Massari are with Fondazione Bruno Kessler, I-38123 Povo (TN), Italy (e-mail: gottardi@fbk.eu).

R. Passerone is with DISI—University of Trento, Italy (e-mail: roberto.passerone@unitn.it).

Z. Smilansky is with Emza Visual Sense Ltd., Kfar Sava, Israel (e-mail: zeev@emza-vs.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSSC.2012.2235031

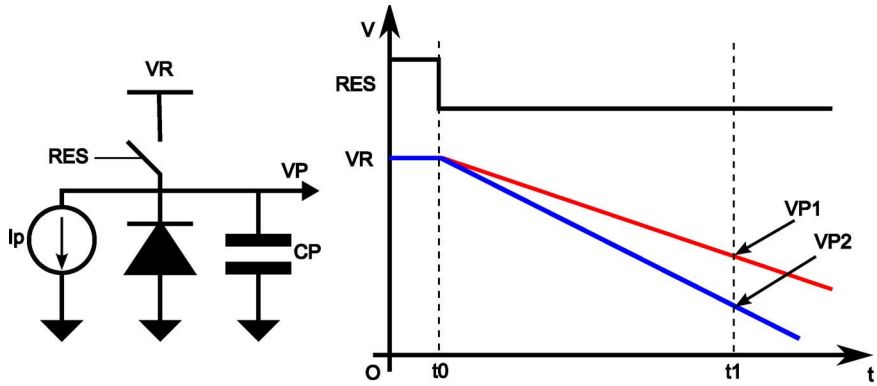


Fig. 1. Operating principle of a photodiode working in storage mode. CP is the junction capacitance while Ip represents the photo-generated current due to the light impinging on the photodiode itself. The two voltage ramps VP1, VP2 show the response of the pixel in two illuminating conditions, where VP2 is related to the higher light intensity.

Although this statement looks relatively simple to be satisfied from a *computer vision* perspective, some considerations need to be made in order to transfer this concept into silicon. The simplest way of approaching this issue is represented by the basic background subtraction:

$$|F_i - B| > \mathrm{TH}; \tag{1}$$

where F_i is the current frame, taken by the camera, B is the background corresponding to a reference frame and TH is a user-defined threshold. As long as (1) is satisfied, foreground objects are present in the scene. Despite its simplicity, this approach is not reliable enough. In fact, in a real scenario, the background constantly changes. It can change due to varying illuminating conditions, suddenly such as moving clouds or slowly, like the moving sun. The background can also change due to camera oscillations, a tree in a windy day, the sea waves and so on. According to these considerations, a more reliable approach is to adopt the well known frame difference, where the background is assumed to be equal to the previous frame:

$$|F_i - F_{i-1}| > \mathrm{TH}. \tag{2}$$

This method is simple and straightforward to be embedded into a sensor and can be executed very quickly. Many implementations of vision sensors with frame difference operations have been proposed in the past [4], [12], [15]–[19], using different design techniques. Unfortunately, this technique is very sensitive to the threshold TH and works properly only under certain values of frame-rate and object speed. Alternatively, the background can be modeled in a more accurate way, by averaging n past frames:

$$B_i = \frac{1}{n}\sum_{j=0}^{n-1} F_{i-j}. \tag{3}$$

This model takes into account the background variations, according with the value of n. It is very memory consuming, requiring n more frames to be stored in an off-chip memory. It also requires extra computation and memory access operations that are in conflict with the low-power target of this work. One of the best trade-offs that can be achieved, meeting the functional robustness of the system with the CMOS technology constraints, is represented by the running average:

$$B_i = \alpha F_i + (1-\alpha)B_{i-1}; \tag{4}$$

where $0 < \alpha < 1$ is called *learning rate*, defining the time response of the running average with respect to background variations. Equation (4) does not require extra memory and can be tuned to the specific scenario by changing the value of α. Using (4), the dynanic background subtraction can be written as:

$$F_i - B_{i-1} = F_i - [\alpha F_{i-1} + (1-\alpha)B_{i-2}]; \tag{5}$$

The present implementation takes advantage of (5) for the implementation of a robust dynamic background subtraction with VLSI-oriented characteristics. This operation is performed at the pixel-level turning the sensor into a massively parallel, digitally-programmable, analog processor.

III. Adaptive Dynamic Background Subtraction

This section describes the algorithm for adaptive background subtraction, which has been embedded in the sensor. The algorithm is inspired by a vision processing algorithm for scene interpretation [14] which pre-filters the photo-generated signals of the pixels, extracting binary information on temporal changes. This information is then processed by a higher level algorithm that discriminates anomalous situations occurring in the scene. The robustness of the algorithm requires that each pixel be able to detect anomalous temporal changes with high reliability. This drastically reduces system false positives, thus decreasing the operating duty-cycle and the power consumption of the entire vision system.

Before describing the details of the algorithm, we recall the basic operating principle of a photodiode. Fig. 1 shows a photodiode working in storage-mode. After a reset phase (RES), where the photodiode is pre-charged to the reverse voltage VR, VP starts discharging by the photo-generated current (I_P). For small voltage variations, the discharge rate is approximately linear with the light intensity impinging on the junction. After the exposure time ($t_1 - t_0$), the final value of the photodiode is sampled. The larger the light intensity, the lower the voltage VP2. Hence, low voltages refer to high illuminating levels, while large voltages refer to low illuminating conditions.

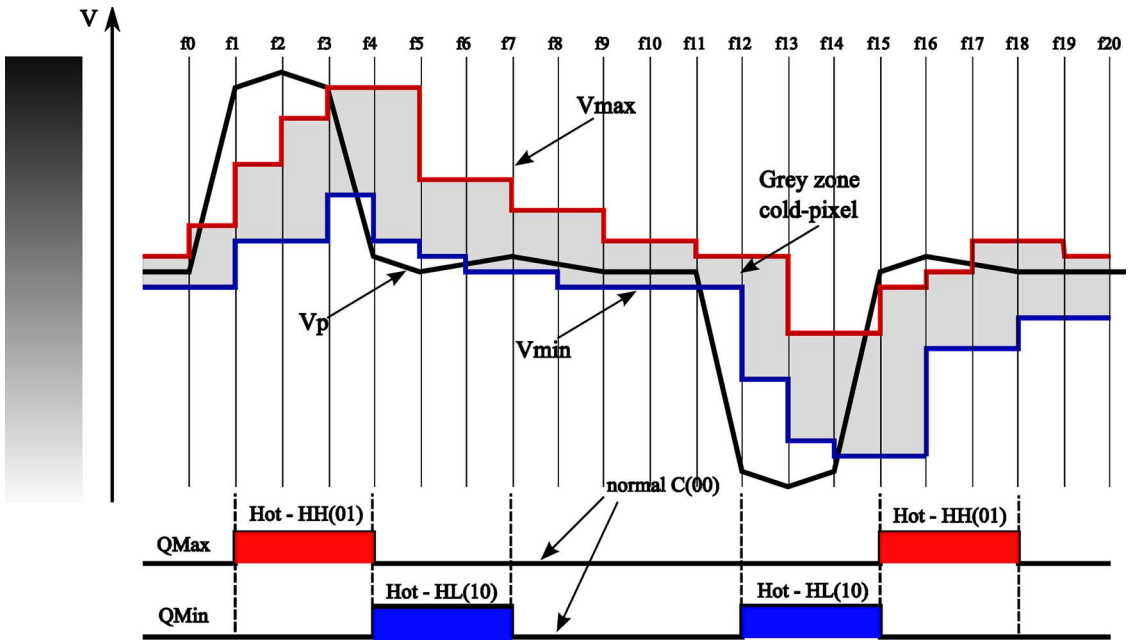


Fig. 2. Pixel-level dynamic background subtraction. The horizontal axis is divided in 20 frames to show how the computation evolves along time. The number of frames (20) does not refer to a real case. The current signal is acquired by the photodiode; $V_{\max}$ represents the upper-bound while $V_{\max}$ is the lower-bound.

We may now describe the basic analog signal processing that has been implemented at the pixel-level. Fig. 2 shows a voltage diagram of the adaptive dynamic background subtraction embedded in the pixel. V_P is the current value of the pixel, sampled at the end of the integration time and acquired at each frame; $V_{\max}$ and $V_{\min}$ are two threshold voltages, changing from frame to frame, according with the signal dynamics. Together they define a grey-zone, corresponding to the gray-colored area in Fig. 2, inside which the current signal V_P is recognized as normal, i.e., having normal "background" behavior with respect to its past values. Whenever V_P goes above $V_{\max}$, the behavior of the signal is considered anomalous. Under this condition, the pixel is classified as *hot* relative to the upper bound (V_{Max}), and the signal Q_{Max} is asserted. At the same time, $V_{\max}$ quickly starts following V_P to absorb the unexpected variation, thus implementing the running average with the desired learning rate. The unexpected variation is absorbed when $V_P < V_{\max}$, so that the system recovers the highest sensitivity of the pixel with respect to potential positive voltage variations. The situation is dual for the lower bound: a hot pixel is detected whenever $V_P < V_{\min}$, Q_{Min} is asserted and the thresholds start following the signal with different time constants. Therefore, depending on the pixel activity, the width of the grey-zone changes from frame to frame, adapting the sensitivity to the specific working conditions and detecting alert situations (outside the grey zone) with large reliability. The larger the grey-zone, the larger the pixel signal change that is considered normal. The algorithm, which is graphically described in Fig. 2, is formalized in (6), (7), (8) and (9):

Upper Bound:

$$\begin{aligned}\text{Hot High (HH)} \rightarrow V_{P_i} > V_{\mathrm{Max}_i} \rightarrow V_{\mathrm{Max}_{i+1}} \\ = V_{\mathrm{Max}_i} + \alpha_H \left(V_{P_i} - V_{\mathrm{Max}_i}\right); \quad (6)\end{aligned}$$

$$\begin{aligned}\text{Cold High (CH)} \rightarrow V_{P_i} \leq V_{\mathrm{Max}_i} \rightarrow V_{\mathrm{Max}_{i+1}} \\ = V_{\mathrm{Max}_i} + \alpha_C \left(V_{P_i} - V_{\mathrm{Max}_i}\right); \quad (7)\end{aligned}$$

Lower Bound:

$$\begin{aligned}\text{Hot Low (HL)} \rightarrow V_{P_i} < V_{\mathrm{Min}_i} \rightarrow V_{\mathrm{Min}_{i+1}} \\ = V_{\mathrm{Min}_i} + \alpha_H \left(V_{P_i} - V_{\mathrm{Min}_i}\right); \quad (8)\end{aligned}$$

$$\begin{aligned}\text{Cold Low (CL)} \rightarrow V_{P_i} \geq V_{\mathrm{Min}_i} \rightarrow V_{\mathrm{Min}_{i+1}} \\ = V_{\mathrm{Min}_i} + \alpha_C \left(V_{P_i} - V_{\mathrm{Min}_i}\right). \quad (9)\end{aligned}$$

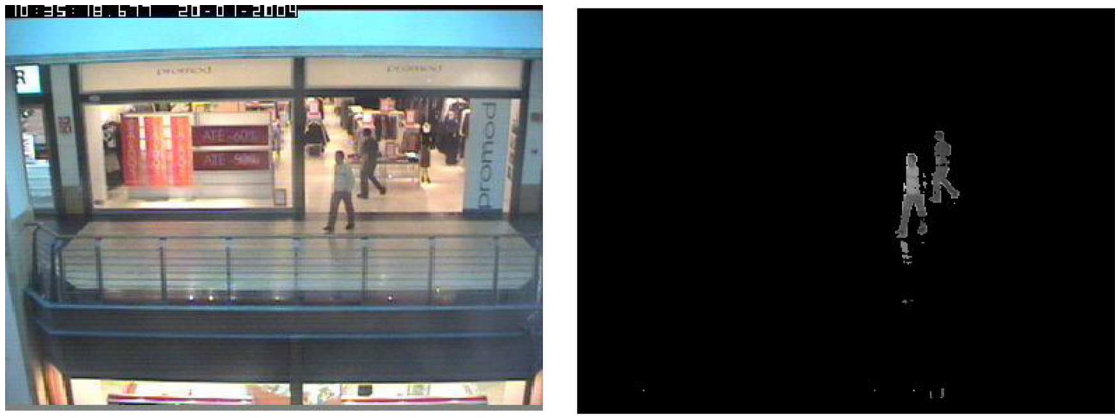
Equations (6) and (8) describe the two hot-pixel conditions. In these cases, the filters work with the fastest time constant (α_H). Under cold-pixel conditions, in (7) and (9), the filters adopt the slowest one (α_C).

In order to validate its overall performance, the algorithm was simulated on a standard dataset, consisting of a number of video clips, recorded within the CAVIAR project [20], acting out the different scenarios of interest, as shown in Fig. 3. These include people walking alone, meeting with others, window shopping, entering and exiting shops, fighting and passing out and last, but not least, leaving a package in a public place.

The grey-level pixels in the right image are those recognized as hot-pixels. Therefore, they are transparent with respect to the original image. The black pixels are those recognized as normal background and do not bring any information. Although this is a simple scenario, the number of hot-pixels covers about 10% of the total image resolution, turning into a significant reduction of data to be sent off-chip.

IV. Pixel Implementation

The main challenge in the implementation is how to embed a low pass filter with programmable time constant in a pixel.



a) b)

Fig. 3. Algorithm simulated on a CAVIAR dataset. (a) original image; (b) image filtered with the presented algorithm for adaptive dynamic background subtraction. The visible parts of the image are those where a hot-pixel has been detected.

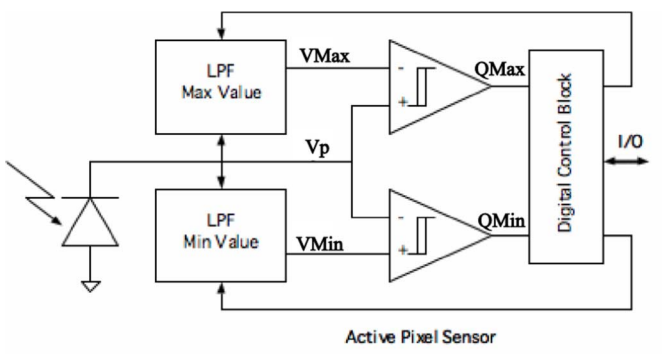


Fig. 4. Basic block diagram of the proposed pixel. Two Low-Pass Filters (LPF) are required to generate the Upper and Lower Boundary signals ($V_{\rm Max}, V_{\rm Min}$), that are compared to the current signal V_P by means of two comparators. The output of the comparators is masked by a digital circuitry providing the proper control outputs to tune the LPFs. The Digital Control Block is placed at the periphery of the imager and not embedded into the pixel.

Although the presented algorithm looks rather simple and naturally oriented to a VLSI design, it is difficult to be implemented in CMOS without compromising the pixel characteristics. As shown in Fig. 4, it requires two programmable low-pass filters (LPF) and two comparators to be embedded in each pixel. The two LPFs generate the two threshold voltages $V_{\rm Max}$ and $V_{\rm Min}$, which define the boundary conditions of the current signal V_P. The two comparators check where the current signal is with respect to these thresholds and encode this position into a 2-bit code (BMax, BMin). The generation of the two threshold voltages, as described in Fig. 2, requires that the two LPFs change their frequency response according with the binary code (BMax, BMin). This is the reason for the two feedback loops in Fig. 4. The pixel schematic is shown in Fig. 5. The photodiode is buffered by the source follower M3, which is loaded by the current source M1 and turned ON and OFF by M2, through the global phase V_{p_clk}. This minimizes the operating duty-cycle together with the dc-current of the buffer. The current signal V_P feeds two Switched-Capacitors Low-Pass Filters (SC-LPF1, SC-LPF2) providing the upper-bound and lower-bound thresholds ($V_{\rm Max}, V_{\rm Min}$), respectively. Two clocked comparators (CMP1, CMP2) compare V_P with $V_{\rm Max}$ and $V_{\rm Min}$, generating the two output bits ($Q_{\rm Max}, Q_{\rm Min}$), which are used to code the four possible states of the pixel, as shown in Table I. Although status 4 is forbidden under ideal operating conditions, it has to be taken into account in particular under the sensor learning phase, after the system power-on. In this case, right after powering the sensor, the two voltage thresholds $V_{\rm Max}$, $V_{\rm Min}$ start from unpredictable values, allowing the forbidden status to be set (Table I). The SC-LPFs have to be operated for a certain number of frames (typically 50–100 frames) so that the two threshold voltages can properly track the current signal, suppressing any hot-pixel. It has to be pointed out that the learning phase should be run under a static scene.

Three additional analog signals ($V_P, V_{\rm Max}, V_{\rm Min}$) are available in the pixel, carried by three bit-lines (BVp, BVMax, BVMin). They are activated at the pixel selection and multiplexed at the periphery of the array. Three unity-gain OpAmps are used to buffer the signals before they are delivered off-chip. Although the use of these signals turn into a significant transistor overhead for the pixel and large power consumption for the OpAmps, they are very important both for the test and debug phases as well as for the sensor learning phase. In a more engineered version, only the current signal V_P should be preserved, being fundamental for the sensor installation procedures. In this case, four transistors and two bit-lines could be spared, with benefit for the pixel pitch.

A. Switched-Capacitors Programmable Low-Pass Filter

A first-order SC-Low Pass Filter can be implemented in CMOS by means of two MOS switches and two capacitors (Fig. 6) [21]. Its transfer function in the z-domain is:

$$H(z) = \frac{z^{-1}}{1 + k - kz^{-1}} \tag{10}$$

where $k = C_2/C_1$. In our case $k = 220$ fF$/150$ fF $\simeq 1.5$. Assuming that the clock frequency of the filter is much larger than the frequency of the photo-generated signal ($\omega_{PH} \ll \omega_C$) and using the linear transform $z = 1 + sT$, where $T = 1/$fps is

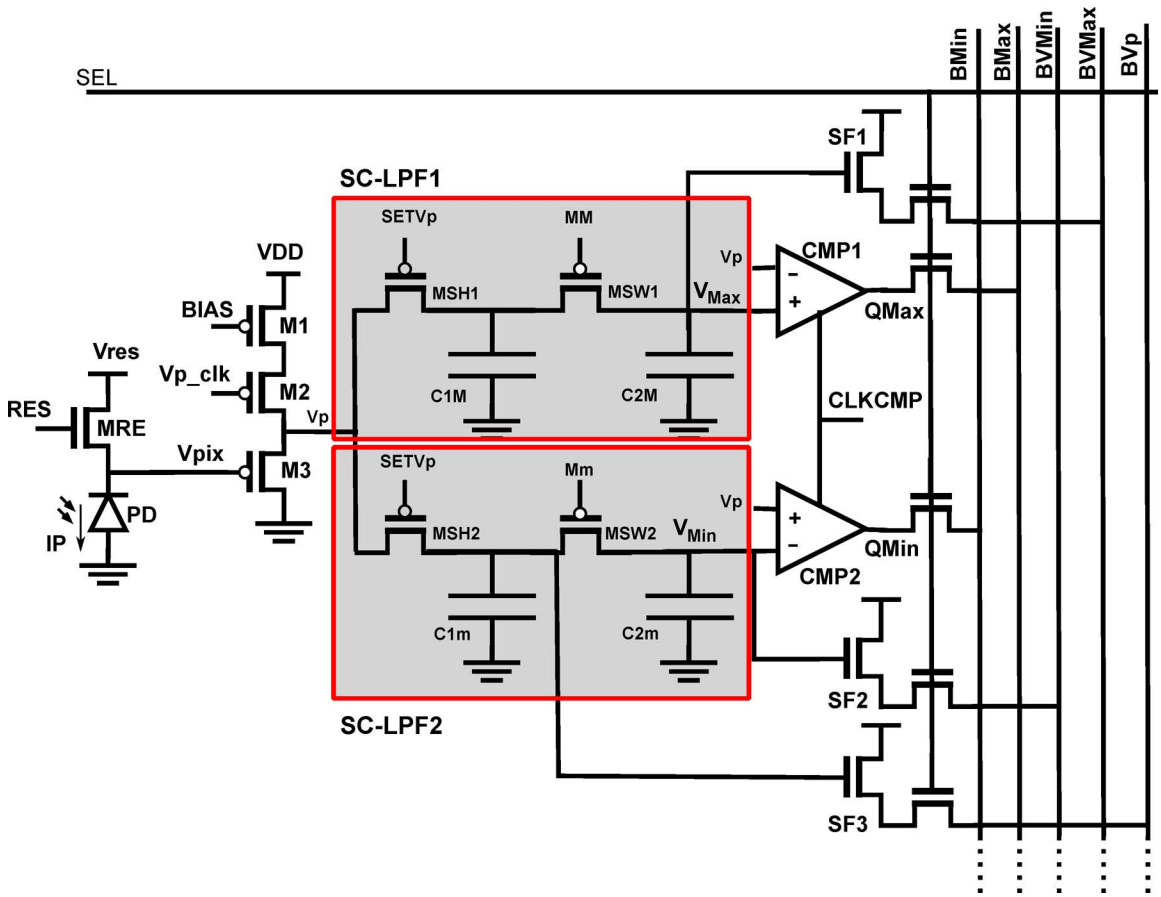


Fig. 5. Schematic of the pixel. The photodiode PD is buffered with a PMOS source-follower which is activated by V_{P_CLK} only at the end of the integration time to minimize the dc power consumption. The two SC-LPFs are fed by the current photo-generated signal V_P and generate the two voltage thresholds $(V_{\mathrm{Max}}, V_{\mathrm{Min}})$.

TABLE I
PIXEL STATUS. QMAX, QMIN ARE USED TO CODE THE STATUS OF THE PIXEL

Status	Event Detection	V_{Max}	V_{Min}	QMax	QMin
1	Cold-pixel Max, Cold-pixel Min	CH	CL	0	0
2	Hot-pixel Max, Cold-pixel Min	HH	CL	1	0
3	Cold-pixel Max, Hot-pixel Min	CH	HL	0	1
4	Hot-pixel Max, Hot-pixel Min	HH	HL	1	1

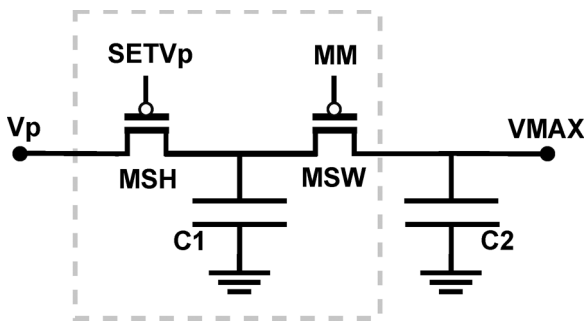


Fig. 6. Schematic of the Switched-Capacitors Low-Pass Filter (SC-LPF) implemented in the pixel.

the frame period of the sensor, typically ranging from 10 to 30 frame/s, the filter transfer function expressed in the continuous time domain s is

$$H(s) = \frac{1}{1 + s \cdot \tau_0} = \frac{1}{1 + s \cdot T(1+k)}. \tag{11}$$

Therefore, the time constant of the filter is:

$$\tau_0 = T(1+k) = T(1 + C_2/C_1). \tag{12}$$

Because the filter is fed by the signal V_P, which is available only once per frame, the highest clock frequency of the filter is equal to the frame-rate $(\mathrm{fps} = 1/T)$. C_1 is always pre-charged with V_P at every frame by clocking SETV_P once a frame. By activating the second phase MM only once every n frames, the filter can be slowed-down by an arbitrary value, as shown in (13), where n is an integer multiplying the filter's time constant τ_0:

$$H(s) = \frac{1}{1 + s \cdot n \cdot \tau_0} = \frac{1}{1 + s \cdot n \cdot T(1+k)}. \tag{13}$$

In order to execute the proposed algorithm, the SC-LPF has to be switched between two values of time response $(n_H < n_C)$; under a hot-pixel detection, the related filter has to have a fast response (n_H), while under cold-pixel the filter has to have a slow response (n_L). This can be set though a proper control of

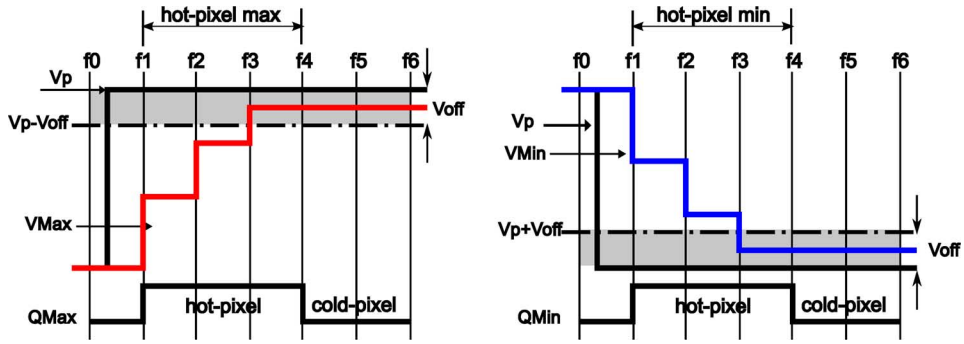


Fig. 7. Comparator built-in offset to enhance noise immunity when the current signal is near the thresholds.

the phase MM. Referring to Table I, for instance, in case of a "Hot-pixel Max" $(\mathrm{QMax} = H)$, the LPF V_{Max} should be fast (n_H), while under "Cold-pixel Max" $(\mathrm{QMax} = \mathrm{L})$ it has to be slowed-down (n_C). Although this is not a fine-tuning technique, its main advantage is that it is fully digital and does only require simple control logic to be integrated at the imager periphery.

The two SC-LPFs are tuned independently according to the status of the pixel (see Table I).

B. Noise Immunity

Under steady-state conditions when the value of the pixel does not change, the two threshold voltages $(V_{\mathrm{Max}}, V_{\mathrm{Min}})$ converge toward the current value V_P of the pixel, reducing the extent of the safe grey-area, and thus achieving the largest pixel sensitivity. An overly large sensitivity may result in spurious hot-pixel detection due to noise affecting the signals. For this reason, we have implemented *safe-zones* around the pixel value delimited by an offset voltage V_{off} that determines a minimum width of the grey-area, as shown in Fig. 7. Mathematically, as long as the following conditions are satisfied:

$$V_P - V_{\mathrm{off}} < V_{\mathrm{Max}} < V_P; \tag{14}$$

$$V_P < V_{\mathrm{Min}} < V_P + V_{\mathrm{off}}; \tag{15}$$

no hot-pixel will be detected. The width of the safe-zone is defined by properly unbalancing the differential input pairs of the clocked comparators CMP1 and CMP2 used in Fig. 5, whose schematic is shown in Fig. 8 [22]. In our case, an offset voltage $V_{\mathrm{off}} \simeq 50$ mV has been intentionally set in the two comparators.

V. Sensor Architecture

Fig. 9 shows the basic block diagram of the vision sensor. It consists of an array of 64×64 pixels, a 64-bit ROW DECODER, sequentially selecting the rows of the imager, and a 64-stage UPDATE REGISTER, delivering binary and analog data to the output of the chip and providing the control logic for the SC-LPFs of the pixels. Each pixel of the generic column j is connected with 2 binary bit-lines (BMaxj, BMinj) detecting "Hot-pixel Max" and "Hot-pixel Min" respectively, and 3 analog bit-lines $(V_{pj}, V_{\mathrm{Max}j}, V_{\mathrm{Min}j})$ delivering the analog signals of the photodiode, and the analog thresholds $V_{\mathrm{Max}j}$ and $V_{\mathrm{Min}j}$ respectively.

The basic operating function of the sensor is the following. At the end of the integration time, each pixel compares the current

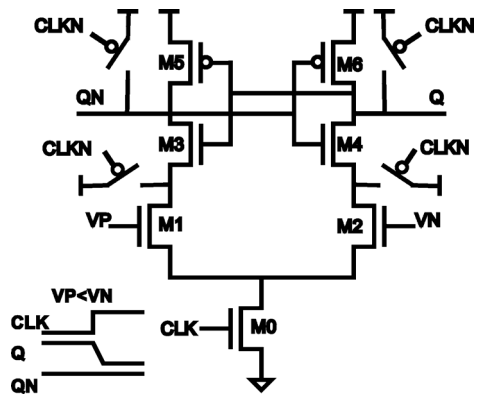


Fig. 8. Schematic of the clocked voltage comparator.

value V_P with the two thresholds $(V_{\mathrm{Max}}, V_{\mathrm{Min}})$ and computes the 2 bits QMax and QMin coding the status of the pixel. Then, the ROW DECODER sequentially selects the 64 rows of the array. Each pixel P(i, j) of the selected row places the values of QMax and QMin on the two bit-lines [BMaxj, BMinj], uploading them to the j-th cell of the UPDATE REGISTER, which has two functions: a) to control the time constant of the filter according to the status of the pixel; and b) to provide a binary and/or analog data readout of the selected row.

The filter control works as follows. Assume that P(i,j) detects a hot-pixel with respect to V_{Max}, i.e., a light-to-dark intensity change. During the readout phase, the two bit-lines [BMaxj, BMinj] are set to [1, 0]. The bit-lines are connected to the asynchronous set (S) of the flip-flop pair of the j-th, cell forcing $[\mathrm{QMax}j, \mathrm{QMin}j] = [1, 0]$. An UPDATE pulse is then provided, masked by [QMaxj, QMinj] through the AND gate. This turns into $[\mathrm{MM}j, \mathrm{Mm}j] = [\mathrm{UPDATE}, 0]$ which directly drive the two clocks of the selected SC-filters:

$$MM_j = UPDATE \cdot QMax_j; \tag{16}$$

$$Mm_j = UPDATE \cdot QMin_j. \tag{17}$$

After the UPDATE phase, data of the i-th row are read out and a new row is selected.

The UPDATE REGISTER also embeds a 64×3 analog-channel multiplexer consisting of a column decoder which sequentially selects the analog bit-lines of one of the 64 pixels in the row. It provides 3 analog serial outputs $(V_P, V_{\mathrm{Max}}, V_{\mathrm{Min}})$ which are buffered by three unity-gain OpAmps and delivered

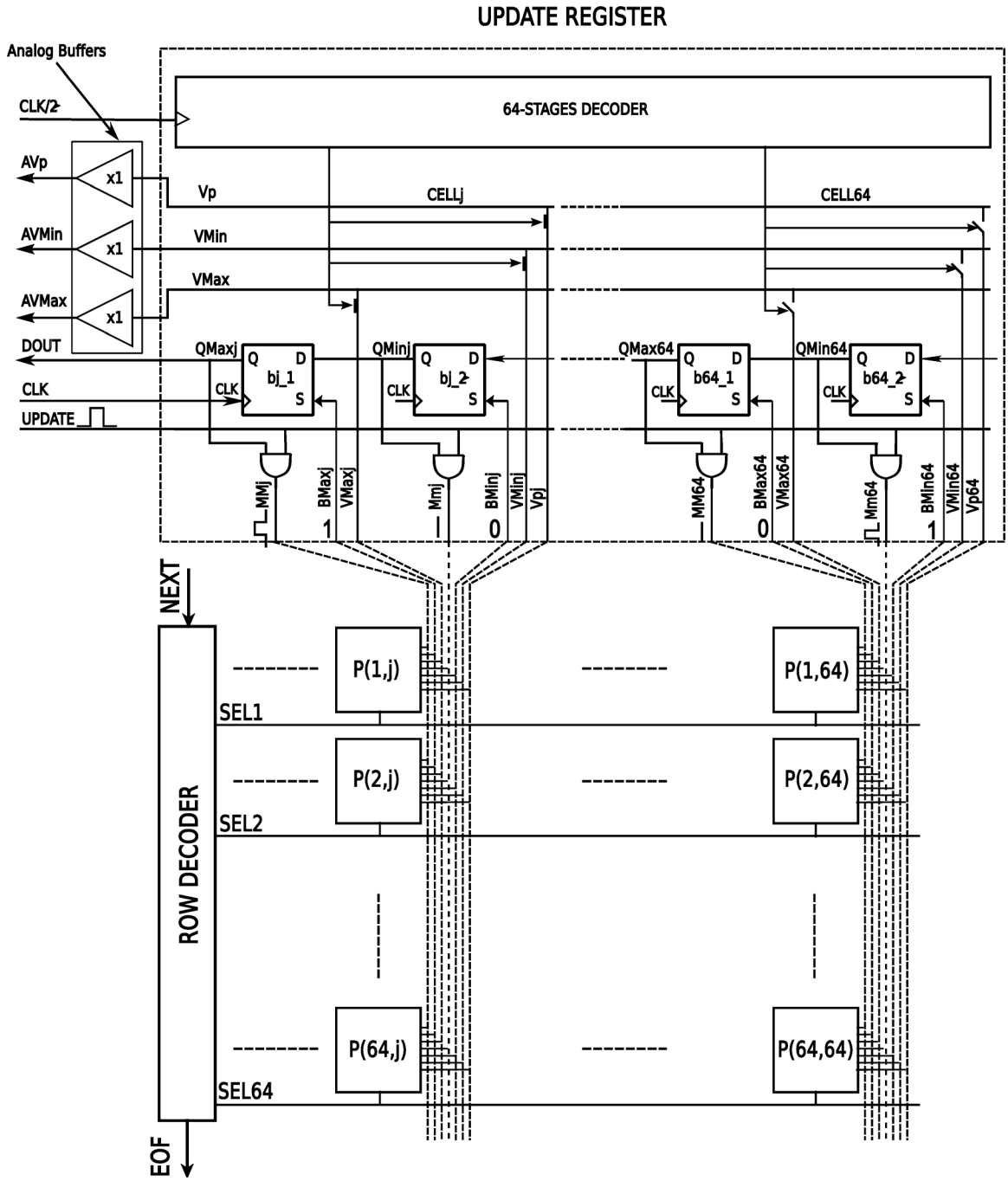


Fig. 9. Block diagram of the 64×64 pixels sensor architecture with detailed schematic of the UPDATE REGISTER and its connections to the selected i-th row (SELi) of the imager. The D-Type flip-flops can be set (S) by the bit-lines. Before shifting out data to the output of the chip (DOUT) by applying an external clock (CLK), an UPDATE pulse is provided, which is masked by the bit-line values (BMj, Bmj) before arriving to the filters of the selected pixels (MMj, Mmj).

synchronously with the binary data (DOUT). The multiplexer is usually enabled during the sensor test and calibration. These signals allow the current signal V_P of each pixel to be monitored together with the activities of the two analog memories $V_{\rm Max}$ and $V_{\rm Min}$ during the algorithm execution.

The interface of the sensor is fully digital, with the exception of the three analog signals devoted to the chip debug and testing. Fig. 10 shows the timing diagram of the main operating phases of the sensor. After the Integration Time (Ti), the signal is sampled onto the first capacitors of the SC-filters (C1M and C1m of Fig. 5) by turning ON the buffer $(\mathrm{Vp\_clk} = 0)$ and pulsing SETVp. The two comparators are then clocked, comparing V_P with $V_{\rm Max}$ and $V_{\rm Min}$. Binary data (QMax, QMin) are now stored into each pixel and the sensor is ready for the readout phase. START selects the first row of the imager and stores the values of the bit-lines (BMaxj, BMinj) into the UPDATE REGISTER. At the rising edge of EOR (End Of Row), data are ready to be read out. Before reading out, the filters (SC-LPF1, SC-LPF2) of the pixels of the selected row have to be updated, by pulsing UPDATE, according with the corresponding values of (QMax, QMin). Now, data can be read out serially through DOUT, providing 64×2 clock pulses on CLK. The next row is selected

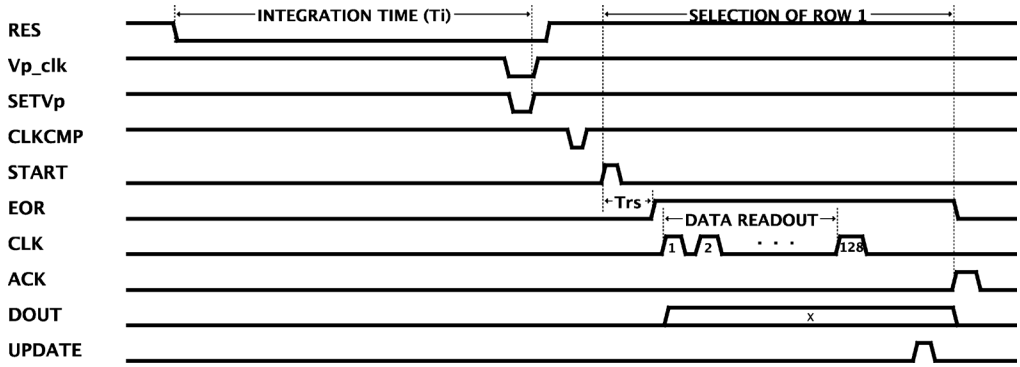


Fig. 10. Timing diagram of the sensor including Integration Time (Ti), data readout of the first row and filters update.

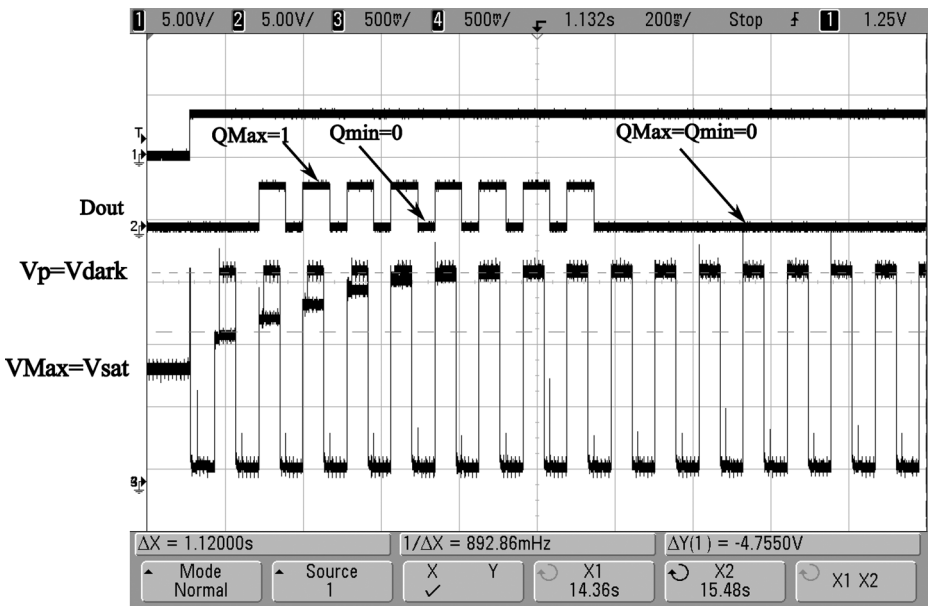


Fig. 11. Pixel response on electrical stimulus simulating a sudden light-to-dark change. The current signal is set to its highest level, $V_P = V_{\rm dark}$, while $V_{\rm Max}$ is pre-charged to the photodiode saturation level $V_{\rm Max} = V_{\rm sat}$, thus forcing a Hot-pixel Max condition.

by pulsing ACK. This operation is executed 63 times during the array raster-scan readout phase, until EOF (End Of Frame) is asserted.

VI. Experimental Results

Various tests have been carried out on the fabricated chip to measure and verify the basic sensor functionality. Of particular interest is the filter time response. To measure it, we have induced an abrupt signal variation V_P on the photodiode to cross the thresholds $V_{\rm Max}$ and $V_{\rm Min}$, respectively, as shown in Figs. 11 and 12. Both filters have been clocked at every frame thus making them work at their fastest time response. This operating mode has been implemented by pulsing the phases MM and Mm once at every frame ($n = 1$ in (13)), thus making the two filters quickly react to any hot-pixel event. When setting $n = 1$, the two filters are fast enough to cover the target application, which requires time constants ranging between 30–300 frames, which means from 2 s to 20 s in case the sensor works at 15 frame/s.

HH Test (Fig. 11)-Hot-pixel Max

Initial conditions: $V_{\rm Max} = V_{\rm sat}; V_P = V_{\rm dark} \rightarrow V_P > V_{\rm Max}$

The pixel takes 8 frames ($Q_{\rm Max} = 1$) to absorb the anomalous signal change.

HL Test (Fig. 12)-Hot-pixel Min

Initial conditions: $V_{\rm Min} = V_{\rm dark}; V_P = V_{\rm sat} \rightarrow V_P < V_{\rm Min}$

The pixel takes 5 frames ($Q_{\rm Min} = 1$) to absorb the anomalous signal change.

It is possible to note that the two filters exhibit different time response necessary to absorb an hot-pixel event (8 frames and 5 frames for HH and HL respectively). This means that the filters need to be tuned according with these characterisitcs. Therefore, assuming the desired time response to be 80 frames for both Hot-pixel Max and Hot-pixel Min, the value $n = 10$ for the Hot-pixel Max has to be chosen, turning into a time constant of $10 \times 8 = 80$ frames, while for Hot-pixel Min, $n = 16$ turning into a time constant of $16 \times 5 = 80$ frames. The different behavior of HH and HL with respect to the algorithm is probably due to the coupling effects, in particular the clock of the comparators which are fed by floating capacitors. Despite the pixel layout symmetry, the signals $V_P, V_{\rm Max}, V_{\rm Min}$ are not equally affected by the clock coupling. This behavior can be observed in Fig. 13, where the imager was operated in the dark while the

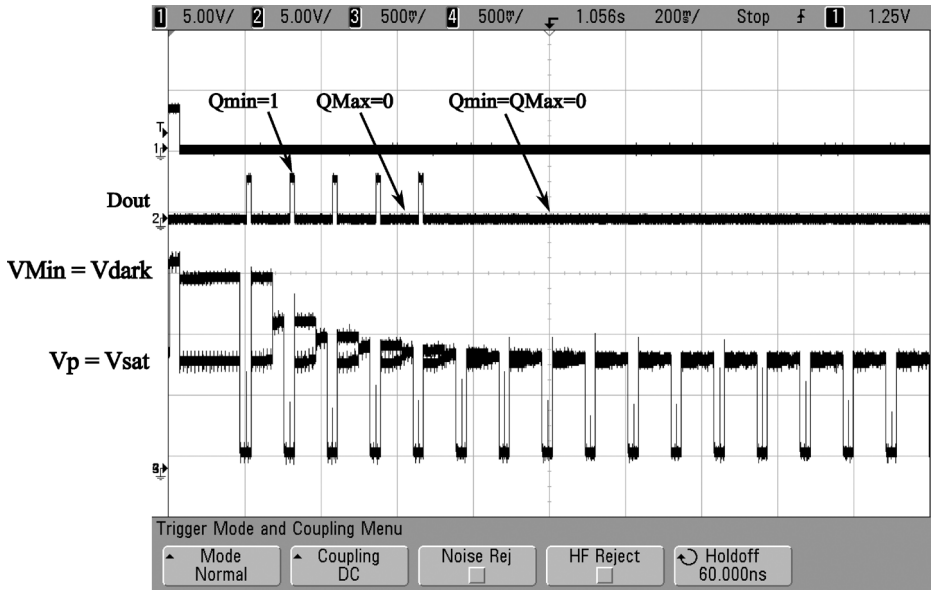


Fig. 12. Pixel response on electrical stimulus simulating a sudden dark-to-light change. V_{Min} is set to its highest level, $V_{\mathrm{Min}} = V_{\mathrm{dark}}$, while the current signal V_P is pre-charged to the photodiode saturation level V_P=V_{sat}, thus forcing a Hot-pixel Min condition.

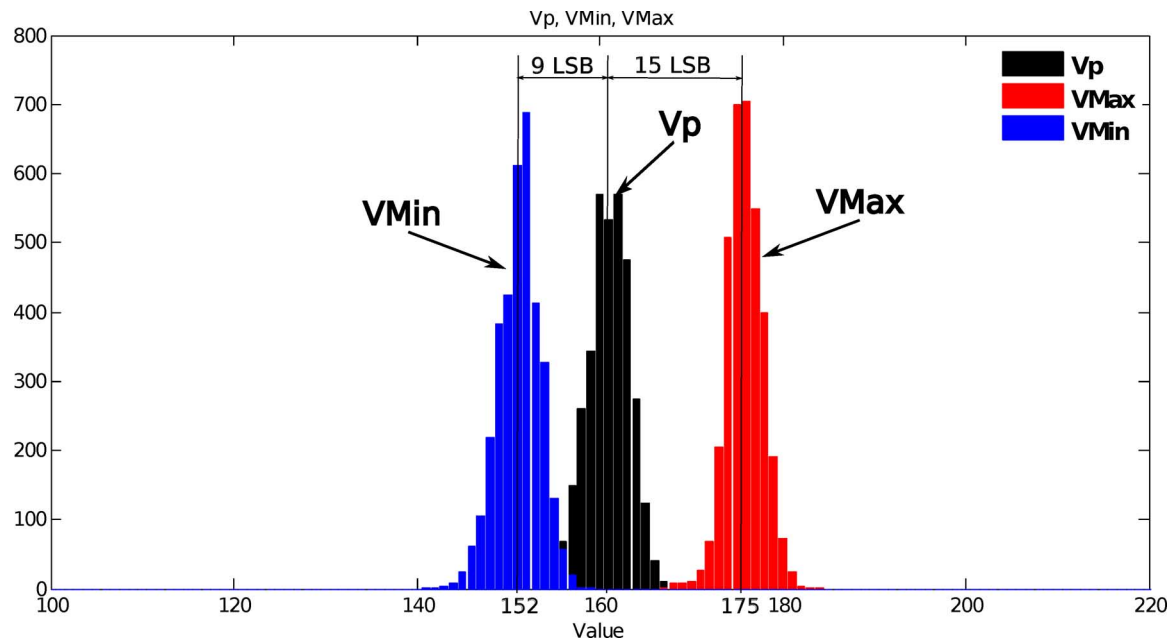


Fig. 13. Histogram of the three analog voltages ($V_P, V_{\mathrm{Max}}, V_{\mathrm{Min}}$) referred to a uniformly illuminated grey-level image. The two thresholds have been settled in order to absorb all the hot-pixels, turning into a black output image. The distance of VMax and VMin with respect to Vp is defined by the built-in offset of the pixel comparators (CMP1, CMP2). The three beans are centered on code 152 (V_{Min}), 161 (V_P) and 175 (V_{Max}) respectively.

photodiodes were intentionally reset to an intermediate value (VR), thus simulating a uniform grey-level image (161/255). Values of V_{Max} and V_{Min} of the entire array have been measured and plotted together with V_P. Experimental results show that V_P is effectively placed between the two threshold ($V_{\mathrm{Max}} = 175, V_{\mathrm{Min}} = 152$). However, V_P is not exactly in the middle of the range, so that the sensitivity level for hot pixel detection is higher for the lower bound (9 LSB) than for the upper bound (14 LSB). We must point out that these values refer to a black image, i.e., with no active hot-pixel. If an increase in sensitivity is required, we have to expect a certain number of spurious hot-pixels. This is due to the coupling effect of the clocked comparators and to the fact that both V_{Max} and V_{Min} drift toward VDD, because of the junction leakage current of the two PMOS transistors MSW1 and MSW2 working as switches (Fig. 5). Fig. 14 shows measurement results of a pixel under abrupt light changes. After a positive signal variation of V_P, a hot-pixel is detected ($\mathrm{QMax} = 1$) and V_{Max} quickly tries to absorb the anomalous change. This takes about 15 frames, while V_{Min} slowly reaches the final value of V_P in more than 70 frames. The pixel behaves the same in case of $\mathrm{QMin} = 1$. As already mentioned, a certain amount of mismatch can be observed, which is probably due to the leakage current in the memories and coupling effects during the comparator operation. Typically, the ex-

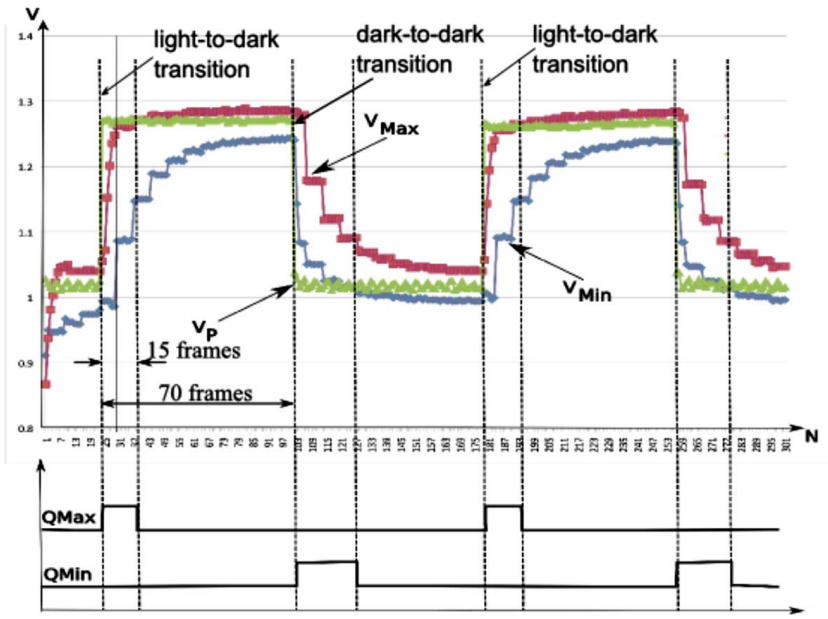


Fig. 14. Pixel response vs abrupt light changes (V_P). Under hot-pixel condition ($\mathrm{QMax} = 1$) V_{Max} quickly absorbs the anomalous variation, while V_{Min} slowly reaches the final value of V_P.

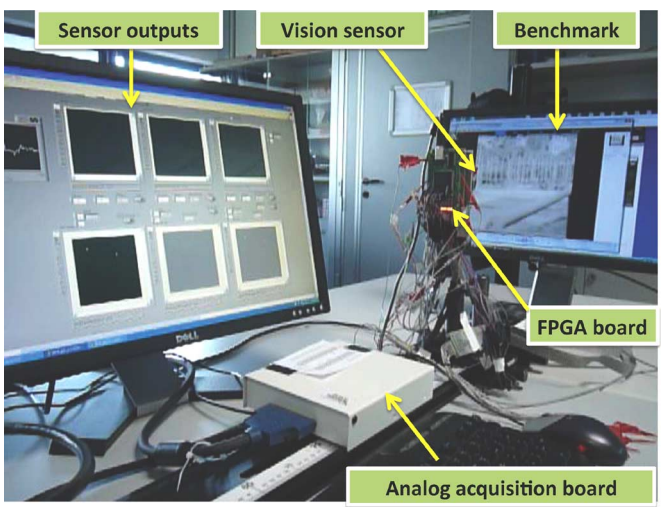


Fig. 15. Photograph of the measurement setup used to debug and test the sensor functionalities. The sensor prototype consists of two stacked boards. The vision chip with optics is placed on the front one, while the FPGA is placed on the rear, driving the vision sensor and interfacing the prototype with a PC, through the USB. In order to test the algorithm execution, a movie has been used as benchmark, continuously running on a monitor and acquired by the vision sensor. The analog outputs, generated by the chip, have been acquired through an analog/digital acquisition board and displayed on the monitor (left).

pected number of hot pixels is kept relatively constant at about $1/100$ of the total number of pixels. Thus a pixel is hot approximately every 100 frames. This keeps the sensitivity of the sensor constantly at its maximum. Fig. 15 shows the experimental setup used to test the functionality of the chip. The sensor looks at a monitor which repeatedly shows a movie (Benchmark) with moving objects. The vision sensor is directly interfaced to an FPGA board, providing the timing to the chip. Binary and analog outputs are acquired by a mixed analog/digital Acquisition Board and visualized on a monitor (Sensor Outputs) in order to evaluate the analog ($V_P, V_{\mathrm{Max}}, V_{\mathrm{Min}}$) and digital information ($Q_{\mathrm{Max}} \oplus Q_{\mathrm{Min}}, Q_{\mathrm{Max}}, Q_{\mathrm{Min}}$) delivered by the sensor. Fig. 16 shows an example of the sensor operating function, emphasizing the hot-pixel compensation of the adaptive algorithm. The hand moves periodically in front of the imager. The motion is intentionally slow with respect to the algorithm response time. At the beginning, the hand is still and no hot-pixel is detected. When the hand starts moving, a large amount of hot-pixels is asserted (Fig. 16(d)). The analog memories try to reduce their gap with respect to the signal V_P. Therefore, V_{Max} (Fig. 16(b)) increases from frame to frame, becoming darker and darker, while V_{Min} (Fig. 16(c)) decreased, becoming lighter and lighter. In this way, the number of hot-pixels slowly decreases until a black image is reached.

Table II lists the main chip specifications. The total chip power consumption is 66 μW, where 33 μW are consumed by the vision sensor, while another 30 μW are burned by the BIAS block, generating the temperature compensated global bias voltage for the buffers of the pixels. This block belongs to a library cell, provided by the silicon foundry, and is not designed for low-power applications. Moreover, its power consumption does not scale with the imager resolution. This is the reason why we did not take it into account in the sensor power consumption as well as in the pixel power consumption and in the computing power, listed in Table II. The main source of power consumption is due to the pixel embedded voltage buffer M1–M3 of Fig. 5, which sinks about 1 μA. In order to minimize its activity, the source-follower is turned on only at the end of the integration time to sample the current signal V_P onto the two capacitors C_{1M} and C_{1m}. This operation is executed in 10 μs and could be further reduced, after which the photodiode is not used anymore. Fig. 17 shows the chip micrograph together with pixel implementation details. The layout of the two SC-filters is symmetric with respect to the

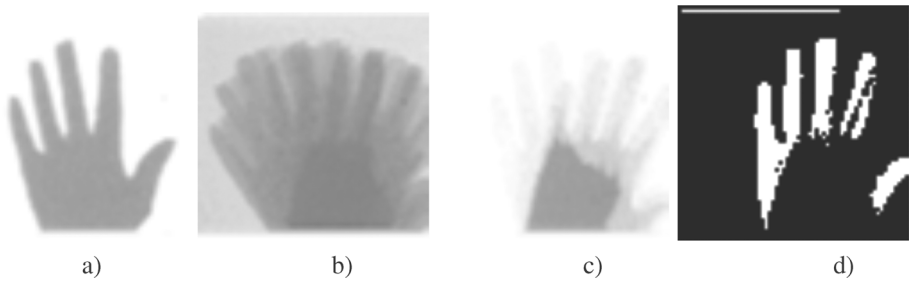


Fig. 16. Working example of a moving hand. (a) Current analog signal V_P, acquired by the sensor; (b) image stored in the $V_{\rm Max}$ analog memory; (c) image stored in the $V_{\rm Min}$ analog memory; (d) binary image delivered off-chip ($Q_{\rm Max} \oplus Q_{\rm Min}$).

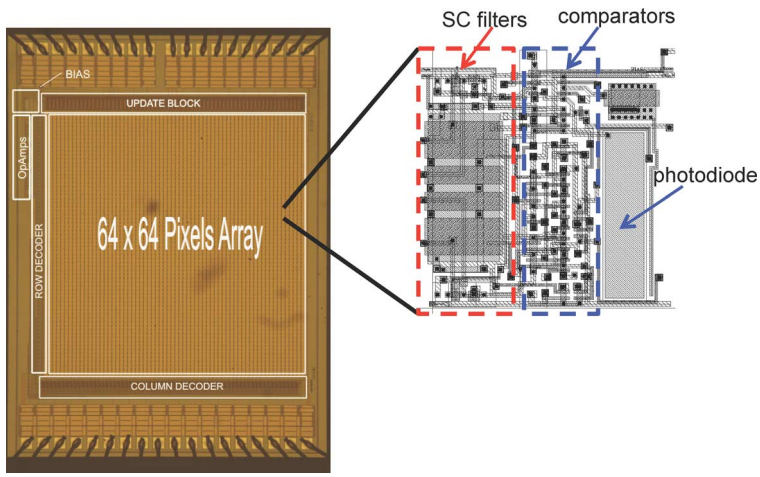


Fig. 17. Micrograph of the chip. In the top-left part the BIAS block, providing the bias voltage to the pixels, and the three unity gain analog buffers (OpAmps) delivering the analog outputs of the pixels during the chip debug phase. In the blow-up, layout of the pixel with embedded processing.

TABLE II
MAIN SENSOR CHARACTERISTICS

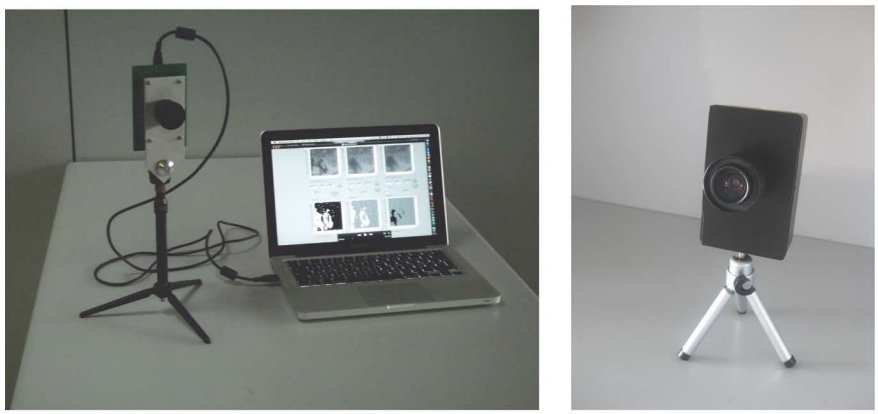
Parameter	Value
Technology	CMOS 0.35 μm 2P-3M
Array Size	64×64 pixels
Pixel Size	$26\ \mu m \times 26\ \mu m$
Number of transistors/pixel	45
Dynamic range	$52dB$
V_P range	750mV
Fill Factor	12%
Supply Voltage	3.3V
Total Power Consumption at the vision sensor with BIAS when running at 13fps	66 μW
Power Consumption at the vision sensor without BIAS when running at 13fps	33 μW
Pixel Power Consumption per frame when running at 13fps	$620pW/pixel.frame$
Vision sensor Computing Performance without BIAS circuit	42 $GOPS/W$
Vision sensor Computing Density without BIAS circuit	4 $GOPS/mm^2$

horizontal axis. The switches are placed at the top and bottom, while the NMOS capacitors are in the middle.

A picture of the vision system prototype is shown in Fig. 18. The system is interfaced directly to a PC through a USB connection. A graphical user interface allows data to be visualized and provides a way to change some operating parameters of the sensor such as the integration time and the algorithm learning rate (α_H, α_C).

VII. COMPARISONS WITH OTHER SENSOR ARCHITECTURES

Although no standard metrics have been defined yet to compare vision sensors performance, it is worth listing the specifications of the most significant sensors performing similar functionalities such as spatial and temporal contrast. Table III lists a number of sensors with comparable resolution and CMOS process. We observe that power consumption varies significantly between the different solutions. The presented sensor together with [23] are those that exhibit the lowest power consumption, which is a factor of about 250 lower with respect to [18]. The fill factor is relatively low with respect to other implemented chips [18], [23], mostly because of the large area occupied by the four capacitors of the two SC-LPFs (C1M, C2M, C1m, C2M). The overall performance of the sensor can be estimated according to the executed operation at the operating frame-rate. In our case, the operations are analog and are mainly performed at pixel-level, as described in (6),



a) b)

Fig. 18. Vision system prototype. (a) the chip is interfaced with an FPGA, providing the proper timing and supporting the USB link with the PC. A graphic interface allows the user to set few sensor's parameters: integration time and filter's learning rates (α_H, α_C); (b) final version of the sensor demo.

TABLE III
SUMMARY AND COMPARISON OF CHIP CHARACTERISTICS

	Ruedi [16]	Zaghlioul [12]	Mallik [17]	Lichtsteiner [18]	Costas-Santos [19]	Gottardi [23]	This work
Functionality	spatial temporal contrast	spatial temporal contrast	spatial temporal contrast	temporal change	Async. temporal contrast	spatial temporal contrast	temporal contrast
Array size	128×128	96×60	90×90	128×128	32×32	128×64	64×64
Fab. process	$0.5\mu m$	$0.35\mu m$	$0.5\mu m$	$0.35\mu m$	$0.35\mu m$	$0.35\mu m$	$0.35\mu m$
Pixel size (μm)	69×69	34×40	25×25	40×40	58×56	26×26.5	26×26
N. of transistors	50T	38T	6T+2cap	26T	104T+1cap	45T	45T
Fill factor	9%	14%	17%	8.1%	3%	20%	12%
Supply Voltage	3.3V	3.3V	3V	3.3V	-	3.3V	3.3V
Power cons.	$300mW$	$62.7mW$	$4.2mW$	$30mW$	$10mW$	$100\mu W$	$33\mu W$
Operating range	120dB	45dB	51dB	120dB	-	100dB	52dB
Frame rate	60 to 500 fps	10M events/s	30 fps	$100\mu s$ 2M events/s	1.6 Meps	50 fps	13 fps
FOM (nW/frame.pixel)	305	63	17	15	6.5	0.244	0.620

TABLE IV
COMPUTING PERFORMANCE

	SPE 2003 [24]	ACE16K 2004 [25]	SCAMP 2005 [26]	SRVC 2009 [27]	ASPA 2011 [28]	This Work
Technology	$0.35\mu m$	$0.35\mu m$	$0.35\mu m$	$0.35\mu m$	$0.35\mu m$	$0.35\mu m$
Processing Type	digital	analog	analog	binary	digital	analog
Array Size (pixels)	64 x 64	128 x 128	128 x 128	16 x 16	19 x 22	64 x 64
Pixel Size (μm^2)	67.4 x 67.4	75.7 x 77.3	49.35 x 49.35	30 x 40	100 x 117	26 x 26
Memory/ Pixel	24 bits	2 an reg, 4 bin	9 an reg	4 bits	64 bits	4 analog
Total Power	-	$2.9W$	$240mW$	$8.72mW$	$26.4mW$	$33\mu W$
P_A $MOPS/mm^2$	343	3800	512	94	205.2	4000
P_E $GOPS/W$	-	180	85.3	24.4	38	42

(7), (8) and (9). As listed in Table IV, the sensor performs about 4 GOPS/mm^2 and 42 GOPS/W at 13 fps. These values are in line with the most significant vision chips featuring image processing techniques that have been published in the last years.

VIII. CONCLUSION

We have presented a 64×64 pixel, ultra-low power vision sensor embedding pixel-level analog dynamic background subtraction as the basic image filtering for a scene interpretation algorithm. Each pixel delivers 2 bits detecting an anomalous signal behavior (hot-pixel) with respect to its past history. The algorithm uses two SC-Filters/pixel, changing their frequency response through a fully digital control. Each filter can be tuned according with the pixel status. The sensor delivers a $64 \times 64 \times 2$—bit image where the asserted pixels are those detecting a potential alert situation. This image is the input of the high-level algorithm for scene interpretation, which is executed by an external processor. The sensor exhibits a pixel size which is in-line with those of similar vision sensors. Some undesired effects have been observed and reported in Section VI. They are mainly due to the coupling effects of CLKCMP on the three analog signals $(V_P, V_{\mathrm{Max}}, V_{\mathrm{Min}})$, which are stored onto high-impedance nodes. In fact, the clocked comparators strongly affect the analog signals although a careful symmetric layout has been adopted. Moreover, the built-in offset in the

comparators differential pair cannot be changed, turning into a limitation in the capabilities of the sensor to adapt itself to the different application scenarios. Alternative design solutions would be more desirable, in order to be able to change this value upon request.

ACKNOWLEDGMENT

The authors would like to thank Marco De Nicola for his support in the board design and interface.

REFERENCES

[1] B. Cho, A. Krymski, and E. R. Fossum, "A 1.5-mV 550-μ w 176 $\times$ 144 autonomous CMOS active pixel image sensor," *IEEE Trans. Electron Devices*, vol. 50, no. 1, pp. 96–105, Jan. 2003.

[2] K. Kagawa, S. Shishido, M. Nunoshita, and J. Ohta, "A 3.6 pW/frame. pixel 1.35 V PWM CMOS imager with dynamic pixel readout and no static bias current," in *IEEE ISSCC Dig. Tech. Papers*, 2008, pp. 54–55.

[3] N. Massari, M. Gottardi, and S. A. Jawed, "A 100 μw 64 $\times$ 128 pixels contrast-based asynchronous binary vision sensor for wireless sensor networs," in *IEEE ISSCC Dig. Tech. Papers*, 2008, pp. 588–589.

[4] D. Kim, Z. Fu, J. H. Park, and E. Culurciello, "A 1-mw CMOS temporal-difference AER sensor for wireless sensor networks," *IEEE Trans. Electron Devices*, vol. 56, no. 11, pp. 2586–2593, Nov. 2009.

[5] S. Hanson, Z. Foo, D. Blaauw, and D. Sylvester, "A 0.5 V sub-microwatt CMOS image sensor with pulse-width modulation read-out," *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 759–767, Apr. 2010.

[6] N. Massari, M. De Nicola, N. Cottini, and M. Gottardi, "A 64 $\times$ 64 pixels 30 μW vision sensor with binary data compression," in *Proc. IEEE SENSORS 2010*, Nov. 2010, pp. 118–121.

[7] C. S. M. Law and A. Bermak, "A novel asynchronous pixel for an energy harvesting CMOS image sensor," *IEEE Trans. VLSI*, vol. 19, no. 1, pp. 118–129, Jan. 2011.

[8] N. Cottini, L. Gasparini, M. D. Nicola, N. Massari, and M. Gottardi, "A CMOS ultra-low power vision sensor with image compression and embedded event-driven energy-management," *IEEE Trans. Circuits and Systems—Emerging Technologies*, vol. 1, no. 2, pp. 1–10, Oct. 2011.

[9] J. Choi, S. Park, J. Cho, and E. Yoon, "A 0.5 v 4.95 μW 11.8 fps PWM CMOS imager with 82 dB dynamic range and 0.055% fixed-pattern noise," in *IEEE ISSCC Dig. Tech. Papers*, 2012, pp. 114–115.

[10] M.-T. Chung and C.-C. Hsieh, "A 1.36 μw adaptive CMOS image sensor with reconfigurable modes of operation from available energy/illumination for distributed wireless sensor network," in *IEEE ISSCC Dig. Tech. Papers*, 2012, pp. 112–113.

[11] S. U. Ay, "A 1.32 pW/frame.pixel 1.2 V CMOS energy-harvesting and imaging (EHI) APS imager," in *IEEE ISSCC Dig. Tech. Papers*, 2011, pp. 116–117.

[12] K. A. Zaghlioul and K. Boahen, "Optic nerve signals in a neuromorphic chip II: Testing and results," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 4, pp. 667–675, Apr. 2004.

[13] N. Cottini, M. Gottardi, N. Massari, R. Passerone, and Z. Smilansky, "A 32 μW 42 GOPS/W 64 $\times$ 64 pixels vision sensor with dynamic background subtraction for scene interpretation," in *Proc. Int. Symp. Low Power Electronics and Design*, Jul. 30–Aug. 1 2012, pp. 315–320.

[14] Z. Smilansky, "Miniature Autonomous Agents for Scene Interpretation," U.S. patent 7,489,802 B2,, Feb. 10, 2009.

[15] L. Gasparini, M. D. Nicola, N. Massari, and M. Gottardi, "A micro-power asynchronous contrast-based vision sensor working on contrast," in *IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 1040–1043.

[16] P.-F. Ruedi, P. Heim, F. Kaess, E. Grenet, F. Heitger, P.-Y. Burgi, S. Syger, and P. Nussbaum, "A 128 $\times$ 128 pixel 120 dB dynamic-range vision sensor chip for image contrast and orientation extraction," *IEEE J. Solid-State Circuits*, vol. 38, no. 12, pp. 2325–2333, Dec. 2003.

[17] U. Mallik, M. Clapp, E. Choi, G. Chauwenberghs, and R. Etienne-Cummings, "Temporal change threshold detection imager," in *IEEE ISSCC Dig. Tech. Papers*, 2005, pp. 362–363.

[18] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128 $\times$ 128 120 dB 30 mW asynchronous vision sensor that responds to relative intensity change," in *IEEE ISSCC Dig. Tech. Papers*, 2006, pp. 25–26.

[19] J. Costas-Santos, T. Serrano-Gotarredona, R. Serrano-Gotarredona, and B. Linares-Barranco, "A spatial contrast retina with on-chip calibration for neuromorphic spike-based AER vision systems," *IEEE Trans. Circuits Syst. I*, vol. 54, no. 7, pp. 1444–1458, Jul. 2007.

[20] [Online]. Available: http://homepage.inf.ed.ac.uk/rbf/caviar/

[21] P. Allen, "Switched-capacitors filters and their applications," 1984.

[22] Y.-T. Wang and B. Razavi, "An 8-bit 150-MHz CMOS A/D converter," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 308–317, Mar. 2000.

[23] M. Gottardi, N. Massari, and S. A. Jawed, "A 100 μw 128 $\times$ 64 pixels contrast-based asynchronous binary sensor for sensor network applications," *IEEE J. Solid-State Circuits*, vol. 44, no. 5, pp. 1582–1592, May 2009.

[24] T. Komuro, I. Ishii, M. Ishikawa, and A. Yoshida, "A digital vision chip specialized for high-speed target tracking," *IEEE Trans. Electron Devices*, vol. 50, no. 1, pp. 191–199, 2003.

[25] G. Linan *et al.*, "A 1000 fps at 128 $\times$ 128 vision processor with 8-bit digitized I/O," *IEEE J. Solid-State Circuits*, vol. 39, no. 7, pp. 1044–1055, 2004.

[26] P. Dudek and P. Hicks, "A general-purpose processor-per-pixel analogue SIMD vision chip," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 52, no. 1, pp. 13–20, 2005, Change.

[27] M. Wei, L. Qingyu, Z. Wancheng, and W. Nan-Jian, "A programmable SIMD vision chip for real-time vision applications," *IEEE J. Solid-State Circuits*, vol. 43, no. 6, pp. 1470–1479, 2008, Change.

[28] A. Lopich and P. Dudek, "A SIMD cellular processor array vision chip with asynchronous processing capabilites," *IEEE Trans. Circuits Syst. I*, vol. 58, no. 10, pp. 2420–2431, Oct. 2011.

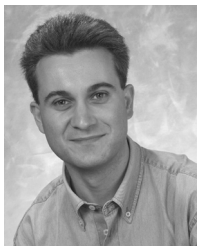
Nicola Cottini received the B.Sc. degree and the M.Sc. degree in telecommunication engineering from the University of Trento, Trento, Italy, in 2005 and 2008, respectively. He is currently working towards the Ph.D. degree at Fondazione Bruno Kessler, Trento, on the design of low-power CMOS vision sensors and systems with event-driven capabilities. His research interests also include the development of hardware-oriented low-energy vision algorithms for people monitoring and tracking.

Massimo Gottardi (M'96) received the Laurea degree in electrical engineering from the University of Bologna, Italy, in 1987.

In the same year, he joined the Integrated Optical Sensors Group of the Center for Scientific and Technological Research (ITC-irst), Trento (I), where he was initially involved in the design and characterization of CCD and CCD/CMOS optical sensors with analog early processing, in collaboration with Harvard University, Cambridge (MA). Since 1993 he has been involved in the design of CMOS integrated optical sensors. His research interests are mainly in the design of ultra-low power vision sensors for energy-autonomous applications and energy-aware, hardware-oriented image processing algorithms and architectures.

Nicola Massari (M'08) was born in Venezia, Italy, in 1973. He received the Laurea degree in electronics engineering from the University of Padova, Italy, in 1999.

Since 2000, he has been involved with the Center for Scientific and Technological Research (ITC-irst) of Trento, Italy, now Fondazione Bruno Kessler (FBK) as a Research Associate in the Smart Optical sensor and Interfaces (SOI) group. His research interests are in the field of CMOS integrated optical sensors with on-chip processing and low-power consumption, and ROIC design for sensors.



Roberto Passerone (M'04) received the M.S. and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, in 1997 and 2004, respectively.

He is an Assistant Professor at the Department of Information Engineering and Computer Science at the University of Trento, Italy. Before joining the University of Trento, he was Research Scientist at Cadence Design Systems.

Prof. Passerone has published numerous research papers on international conferences and journals in the area of design methods for systems and integrated circuits, formal models and design methodologies for embedded systems, with particular attention to image processing and wireless sensor networks.

Zeev Smilansky received the doctorate degree in pure mathematics from a joint program from the Hebrew University of Jerusalem and University College, London. He holds a Master's degree in pure mathematics (*cum laude*) and Bachelor's degree in mathematics and physics (*cum laude*) from the Hebrew University of Jerusalem.

He has over 25 years' experience in developing complex systems that involve mathematical algorithms, software, optics, electronics, and mechanics. He was responsible for development of novel color systems for the world's first digital scanner at Scitex Corporation. He founded the algorithm department at Orbotech, which he headed for five years. In 2006 he founded Emza Visual Sense, a startup devoted to developing autonomous visual sensors. He has authored 15 patents and is widely published in his various fields of expertise.