Second Generation Programmable Artificial Retina

Fabrice Paillet
CTA/GIP
16bis Av. Prieur de la Côte d'Or, 94114 Arcueil Cedex, FRANCE
paillet@etca.fr

Damien Mercier
ENSTA/LEI
32, Bd Victor, 75015 Paris, FRANCE
mercier@etca.fr

Thierry M. Bernard

Abstract — Lodging a general enough digital processing element (PE) in each pixel of a CMOS imager aims at getting a programmable artificial retina (PAR) that can support fast, compact, low power and low cost vision. Using original architecture and circuit techniques, we have designed and operated a 128×128 PAR capable of grey-level image processing and pattern recognition with not even 50 transistors per PE.

1. INTRODUCTION

As year 2000 approaches, the progress in VLSI CMOS technology has a growing influence on the techniques used for image capture and processing. It is known for long [1] that any reverse-biased unsheltered diode in a standard CMOS circuit generates a reverse photonic current with fair efficiency and accuracy when submitted to visible radiation. But, while pixel size in imaging arrays remains lower-bounded to at least several microns due to hard optical limitations, transistor size is now heading down toward the tenth of a micron. Incidentally, the depth of depleted regions, which conditions the proportion of photogenerated charges collected, may become too shallow, thus requiring less obvious charge collection techniques [2].

Taking advantage of the new deal for image capture purposes, specific CMOS readout circuit techniques have been developed [3,4] that associate a photodiode with a few MOS transistors in each pixel for faithful off-array signal transmission. With submicronic transistors, they imply a minor degradation of pixel size or fill factor. The resulting CMOS imagers have recently matched the standards of mainstream CCD imagers [2]. They feature an obvious cost advantage since, unlike CCD imagers, they can be produced using technologies developed for microprocessors or RAMs. But even more important for cost reduction are the suppression of the external CCD clock driver component and the possibility to settle an A/D conversion module on chip: this is typical of the system-on-chip trend. Cost is indeed a critical issue in the rapidly growing consumer market related to digital imaging. Power consumption, which is reduced by about an order of magnitude, is also an important issue for battery-powered appliances or bus-powered computer peripherals.

What happens today with image capture will concern image processing tomorrow as artificial vision enters everyone's home and car. But where a few "analog" transistors fit now using back-the-wave technologies, there will be space for a hundred or so with nowadays most aggressive process such as 0,18 - 0,1µm (especially if transistor size can be kept minimal because of a digital use). This raises the following question: what to do with a hundred transistors or so in each pixel if their presence has minor influence on the imaging array dimensions or, in other words, if they can be settled there for free?

In this paper, we show an example of what can be already done with about 50 transistors per pixel. We have used them to process and analyse images where they are sensed. Our prime design criterion has been to get the best possible algorithmic versatility without neglecting power consumption [5]. We have actually designed, tested and operated a 128×128 CMOS imager in which each pixel contains a tiny digital processing element (PE). It is the largest ever reported in the literature, though it has been fabricated in a standard and quite outdated 0,8µm CMOS process. We have called it PVLSAR2.2, standing for Programmable and Versatile Large Size Artificial Retina. The present paper provides an overall presentation of the circuit from design to operation.

2. PVLSAR2.2

As indicated by its version number, PVLSAR2.2 is not the first of its kind. A low cost 18×18 version has been used for first investigations (see Fig. 4b). PVLSAR2.2 also shares some fundamental principles with an older retina of ours [6], but with several novel features including grey-level and multiresolution processing, improved communication network, memory virtualisation...

A. Global Architectural Choices

Within the last decade, some remarkable analog circuit techniques have been developed for image pre-processing with a few tens transistors per pixel [7]. But they are dedicated, whereas we look for versatility which requires programmability. CNNs (Cellular Non-linear Networks) are an attempt to get "analog programmability" with one to two hundreds transistors per pixel [8]. But CNN architectures do not scale well, because of precision and power consumption problems. In PVLSAR2.2, the PE (Processing Element) is mostly digital, actually boolean from the computing viewpoint. Note that this implies local A/D conversion which means all PEs should be (and are) exactly the same; otherwise image capture would be corrupted with fixed pattern noise. The PE control signals are distributed through...
metal wires along rows (line or column) of pixels. Each wire controls all PEs of its row, which is typical of a SIMD (Single Instruction Multiple Data) control mode. PVLSAR2.2 is indeed an SIMD array processor with local optical input. It is actually more than that, as the end of section 2C will explain.

Generally speaking, the part of a processor that directly acts on data is called the datapath. When the datapath control signals are too numerous, instruction decoding is useful to decrease control bandwidth. Does it make sense in the pixel? As a case study, let us consider the exemplary PE presented in [9]: it is a pure datapath containing about 110 transistors controlled by about 40 signals. Without instruction decoding [9], the pixel layout is dominated by wiring, which is unsatisfactory. As only one control signal is changed at a time, instruction decoding could reduce the number of row wires to 6. However, transistor cost would be at least 120 — using a tree structure (about 80 nMOS and 40 pMOS) — thus doubling PE transistor count, while the power consumption for driving the row control wires would triple. Though intermediate solutions are possible, as well as time-multiplexed ones [10], the lesson we learn from this case is that avoiding any instruction decoding is really worth some effort. Section 2B explains how we have managed to do so while maintaining a good balance between wiring and logic. Anyway, the result is that the PVLSAR2.2 PE is a pure datapath. And so is the whole PE array, with binary images as elementary data.

To separate design problems and ease testability, the PE control signals are directly connected to external pins through combs of buffers. So the PVLSAR2.2 chip itself is a pure datapath with control signal generation entirely at the charge of external components. This is obviously a bad solution from energy and system viewpoints. An on-chip global instruction decoder/sequencer beside the array would be the first basic step to lower chip-to-chip communication on a commercial version. But PVLSAR2.2 was only meant to investigate the most original part: the PE array. However, we view it as a hard Intellectual Property to be used in conjunction with a processor core and some communication peripheral to form a vision system on one chip.

B. Data Storage and Communication

In order to process images, the PE has to handle a certain number of binary data, either image data or intermediate variables used for computations. The PE architecture is characterized by the maximal number of binary data that can coexist and be exploited for computations in the PE. We call it the memory capacity of the PE and note it M.

Due to parasitic photocurrents in the substrate of a retina chip (it is directly exposed to light), it is safer to use static rather than dynamic memory to hold binary data. So each bit achieves static data storage. Let S stand for an impulse on clock S. Starting from dynamic data storage M1=M2=S=0, sequence \{S!M2!\} shifts data to the right. (b) symbolic representation of the former: circle = SSBR, arrow = pass transistor. (c) Shifting structure used for PVLSAR2.2 and illustrated for a 3x3 PE array. There are 5 SSBRs per PE: R0 to R4. SSBRs R1 (resp. R2) are linked together to form a northward (resp. southward) shift-register. These two shift-registers moving in opposite directions are connected to each other on both sides of the array to form a ring shift-register controlled by clock D (see Fig. 2). The same goes with SSBRs R3 and R4 on the east-west axis. All other control signals are internal to the PE. Intra-PE communication is performed by swappers (ring shift-registers only linking two SSBRs), R1-R2 and R3-R4 swappers, which are respectively activated by \{E!M1!\} and \{E!M2!\}, allow to turn the ring shift-registers over, thus easily reversing rotation. With computation operators settled between R2 and R3, the whole structure proves to be an efficient NEWS interconnection network [11] while only needing 5 control signals (see Fig. 2).
section (the case studied was based on standard static RAM cells).

+ Using shift registers gives the opportunity to merge together the select/read/write circuitry associated to binary registers and the NEWS (North-East-West-South) interconnection network needed for exchanging data with nearest neighbors, thus saving transistors.

- With shift-registers, there will be data moved that do not need it (limited between twice and four times too many for the structure of Fig. 1c), thus wasting energy.

- Control is complex because all data movements are interdependent.

To get the largest M, the above advantages are determining. In particular, local instruction decoding is avoided. Control is actually based on a multiphase clock system. The detailed transistor schematics of the PE is presented on Fig. 2. Only 42 (=8.4xM) transistors and 5 clocks are used for data storage and communication (both intra- and inter-PE). Besides, more SSBRs could be added without changing this figure. As far as the drawbacks are concerned, Fig. 1c significantly improves [11] over that used in [6]. Besides, regarding power consumption, the massive parallelism available in such an array processor makes it acceptable to operate below maximal clock frequency, thus allowing low voltage operation [5] (see section 2F). SSBRs would loose much of their interest if it were not possible to connect them through single nMOS transistors used a pass gates. The danger with such a solution is signal degradation when transmitting a logical one. To avoid this problem, two different power supplies are used: VddPE for inverters in the PE array, and VddCD with a higher voltage for clock drivers at the boundary. Actually, signal degradation is acceptable provided that negligible static short-circuit is triggered in inverters. Eventually, the required difference between VddCD and VddPE is only a matter of body effect and noise margin (with equal nMOS and pMOS thresholds) and about half a volt is enough.

**C. Data Processing**

With binary data, computing operators are limited to COPY, AND-OR, XOR and their complemented versions. The various operators supported by the PVLSAR2.2 PE are obtained by sequences of impulses applied on the clock signals controlling the PE. As a general notation, D! stands for an impulse on clock D. In the following, sequences always start with all clocks set to 0 (inactive) and with consistent inverters in each SSBR (as if they had just left the static memory state M1=M2=1). Consistency is actually enforced each time there is an impulse on M1 or M2. It is necessary to enforce the static memory state often enough, to prevent data from being corrupted, because of the parasitic photocurrents in the substrate (as exposed in the previous section) that tend to slowly discharge the small capacitances that hold data at the input of inverters while in dynamic state (M1=0 and/or M2=0).

The central computing structure of the PVLSAR2.2 PE is the unique transistor controlled by clock C on Fig. 2. It performs charge sharing between the input of two inverters respectively belonging to registers R2 and R3. With equal capacitances bearing initial logical values A and B (see Fig. 2), charge sharing produces an analog voltage representing m(A,B), the average between A and B. Since the AND, OR and XOR operators are symmetrical functions, A.B, A+B and A⊕B can be computed from m(A,B) by comparing it to different thresholds. Asymmetric inverters are used for this purpose (see Fig. 2). Inverters built with minimal size transistors naturally feature a low threshold. To get an inverter with a high threshold, we have used an additional transistor with a dedicated static analog control signal T (see Fig. 2). So the capacitance balance between the two inverter inputs is obtained at small area cost (using minimal size transistors) and in a voltage and process insensitive fashion (a requirement for an Intellectual Property). Sequence \{C!M2!M1!\} yields A.B in R3 while keeping A in R2, whereas \{C!M1!M2!\} yields A+B in R2 while keeping B in R3. Sequence \{C!D!M1!M2!\} simultaneously yields A+B in R2 and A.B in the R3 register of the east neighbor (it is easy

![Fig. 2. Transistor schematics of the PE, constructed around 5 SSBRs: R0 to R4 (as on Fig. 1c). P and T are analog voltages while the other control signals are digital clocks. Here are the different functions with the hardware resources they use (t stands for transistors): data storage (M1, M2, 30t), communication (R, E, D, 12t), computation (C, X, T, 4t), capture and ADC (P, X, 3t). The total cost is 49 transistors, 9 minimal width wires for control and two larger wires for ground and power supply. The size of all transistors is small, generally minimal.](image-url)
then to get it back). The point is to further compute $A \oplus B$ with the help of an inverter but without the need of a third register [12], thus achieving a better use of the PE memory capacity $M$. This is all the more important as $M$ is small. It can be crucial for temporal processing, e.g. motion analysis, which needs to store results from the past. A short example that also illustrates this, is to compare the largest counter that can be implemented in the PE: PVLSAR2.2 holds 5 bits of memory and a 4 bit Gray code counter is implemented with the fifth bit used to store the current binary value; This means 1 bit is "wasted". In [12], the PE holds 8 bits of memory but the maximum number of bits effectively used to count is 6, 2 bits are needed for the computation itself and thus are "wasted". By the way, the whole XOR computation eventually yields $A \oplus B$ and $A.B$, which are the well-known sum and carry terms appearing in bit-serial addition of digital numbers.

To invert the content of a register such as R2, the most area-saving technique we have found is to borrow an inverter from the neighbor register R1. This is achieved by using the two pass transistors controlled by clock X between R1 and R2 on Fig. 2. Sequence $\{X!R!M1!R!M1!\}$ inverts R2 while keeping other data unchanged. Notice that $\{R!M1!\}$ corresponds to a swap of the data in registers R0 and R1. This shows a side effect, common in PVLSAR2.2, of using control signals at their best (the same signal for two non-intersecting shift registers or for another functionality). Using only the sequence $\{X!R!M1!\}$ both invert R2 and swap registers R0 and R1.

Last but not least, when $X=M1=1$, a ring oscillator with 3 inverters is obtained. The random bit generator created is a useful facility for implementing stochastic image processing algorithms. It has been operated and actually provides visually random images. Finally, the PE is well fitted for boolean computations, including bit-serial and stochastic ones, while just using 2 digital control signals (C and X), one bias voltage $T$, and 4 transistors. Only a few computing sequences have been mentioned. There are many others for which registers R2 and R3 are not the only ones involved.

Digital processing amounts to computing boolean functions. The disjunctive normal form and the ring-sum-expansion are two well-known representations of boolean functions that allow their computation with only two binary registers — in other words, borrowed from Turing theory, with a space complexity of 2 — provided that all binary variables are readable elsewhere. But in that case, computing time — or time complexity — may depend exponentially on the number of variables, with likely unacceptable combinatorial explosion. In that case, increasing space complexity enough often allows to dramatically reduce time complexity and bring it back within reasonable limits. This is usually needed for operators that pertain to the linear approach of image processing, whereas mathematical morphology operators are much less demanding. When space complexity has to be increased, the memory capacity of the PE will often be insufficient. To cope with this problem under stringent area constraints, we have enhanced the basic SIMD architecture of PVLSAR2.2 with a clustering facility. PEs may be periodically clustered by groups of 2 or 4, so that all registers in the cluster can be shared. Clustering actually provides virtual memory at the expense of spatial parallelism: in a cluster, the output values for each pixel are computed sequentially. Clustering is simply obtained by using different control signals for odd and even lines or columns, without any extra area cost for the PE. This scheme could obviously be extended to get larger clusters, provided some adequate on-chip control structure. It would also be useful for multiresolution image processing.

### D. Data I/O

In any retina, data input is almost exclusively limited to image capture. Apart from that, PVLSAR2.2, as any array processor, is fitted with edge conditions facilities to be able to input zeros or ones, as required, when shifting binary images for processing purposes. Also, arbitrary images may be input for test purposes.

For test and algorithmic development purposes, PVLSAR2.2 is fitted with 16 binary output channels evenly distributed on the northern edge of the PE array. With the help of the shifting structure (see Fig. 1c), a 128x128 binary image is output in 5.6k cycles (225μs if PVLSAR2.2 is operated at 25MHz), without modifying any data — a cycle corresponds to one clock impulse. But a programmable artificial retina is actually not expected to output images. With the computing abilities presented in the previous section, more than low-level image processing can be performed, up to segmentation and pattern recognition, as illustrated on Fig. 5. This allows to reduce the output flow rate by possibly several orders of magnitude as only concentrated pieces of information remain to be extracted from the retina:

- global image descriptors can be obtained by measuring the current drawn on the power supply of the core array. Indeed, the charge sharing technique presented in the previous section triggers the same short-circuit current in each PE where $A \neq B$. The power supply comb is then exploited as an analog adder to provide image correlation (or sum) [13] as a universal primitive. A 12 bit precision should be achievable with 1μs long measurements.
- stability detection is needed for relaxation algorithms. For this, an image OR, which tells if there is at least a pixel set to one in an image, can be obtained from the above current measurements, with the help of a preliminary dilation operation on the image to make a single pixel detectable. There are alternative solutions. A line OR and a column OR have been respectively settled at the northern and eastern borders of the PE array. A
global OR can then be easily obtained by combining the line or column OR with a programmed 1D OR computation in the PEs, along the perpendicular direction.

- A list of co-ordinates is the natural data structure to describe a sparse image produced by a pattern recognition procedure. A few solutions have been devised to efficiently output such data [14], but it was conservatively chosen not to embed any of them in PVLSAR2.2’s PE. Still the line hardwired OR settled at the border of the array can be used for this purpose, allowing to output only non-empty lines.

E. Image Capture and A/D Conversion

Best quality image capture was not a goal of the PVLSAR2.2 project. Image sensing and analog to digital conversion are performed by a simple and compact 3-transistors circuit connecting a photodiode to the digital part of the PE. It is shown on the left part of Fig. 2. Let us roughly explain its operation as if nMOS transistors were totally off when in subthreshold mode (below \( V_n \)) and without accounting for the body effect. The photodiode is operated in integration mode. It is diode-connected to an external voltage source \( P \) which provides two levels: the reset level \( V_r \) or the antiblooming level \( V_a \), with \( \min(V_r, V_a) > V_n > 2V_n > V_a \). Let \( V_\phi \) be the photodiode voltage. At reset, \( V_\phi = V_r - V_n \). Then, \( P \) goes down to the antiblooming level \( V_a \) such that \( V_\phi \) cannot go below \( V_a - V_n \). Then \( V_\phi \) decreases according to the light flux on the photodiode. When \( V_\phi \) reaches \( V_n \), the ADC transistor turns too weak to load a zero in register \( R_0 \) (put in dynamic state) through a pass transistor controlled by \( X \): this can be checked quickly (in a few clock cycles) and repeatedly since \( V_\phi \) is unaffected by the operation. By checking frequently enough, it is possible to know when — in discrete time — \( V_\phi \) has reached \( V_n \). We call this event the photodiode commutation. Commutation time provides the pixel grey level. Absolute time may be counted in the PE. So was obtained Fig. 3a, using a 4-bit Gray code counter. But often [15], only relative time is important: only the differences between grey levels have to be known and precision is only needed close to 0. This allows to use a larger grey level dynamics in spite of the small PE memory capacity. So was obtained Fig. 3b.

F. Layout, Test and Operation of PVLSAR2.2

PVLSAR2.2 is a full custom circuit, designed and fabricated in a 0.8\( \mu \)m single poly, 2 metal layers CMOS process. The main characteristics in designing such PARs is to find a good trade-off between compactness, versatility, efficiency and power. The silicon area constraint was the major one for PVLSAR2.2, but may become less predominant against the others as CMOS process get deeper and deeper into sub-micron. Particularly due to the reduced features (only 2 metal levels…) of the technology used for PVLSAR2.2, power and control clock signals were a critical issue. Clock capacitance are rather big to drive: the wires run across the whole chip. Note that lodging clock repeaters among pixels is undesirable for photosensitivity matching purposes. Power rails distribution, also, had to be maximized in size and quantities as much as possible, to be able to manage efficiently the high and brief current peaks generated when all processors simultaneously ask for current. As we are aiming at larger size PARs, using more up-to-date technologies fitted with five to six metal layers will ease power distribution more than clock distribution, considering the above constraints.

Each pixel fits within a 60\( \times \)60\( \mu \)m\(^2\) square. Total circuit area is 80mm\(^2\). The circuit has been tested and operated from both the parallel port of a PC at a clock frequency of 150kHz and through a PCI bus at 2Mhz. However, it has been designed to operate up to 100Mhz (Simple speed tests at 80Mhz with the 18\( \times \)18 retina were performed successfully). PVLSAR2.2 has proven fully functional. Such chips are easy to test functionally, because, as well as JTAG is used in sequential circuits for defect detection, all memory components in the chip are linked into shift registers.

Fig. 4a shows that the photodiode fill factor is about 30%: the whole pixel (photodiode + PE) is delimited by the dotted line zone; the photodiode itself corresponds to the area crossed by only 3 metal rails oriented in the upper left to the down right corner direction, as the rest of the PE is completely covered by the metal rails from its control.
signals, in the perpendicular direction. Computing circuits are so quite well protected from the light which could otherwise produce faults during computations. In 40ms (the usual video rate), a 5 lux illumination produces a 1V decrease of Vp. As expected, the fixed pattern noise performances of our capture and ADC circuit are poor: 48mV to cover 99% of the pixels, 13mV to cover 50%. So for a 1V decrease of Vp, 5-bit grey level dynamics is the best that can be obtained. Dark current is not significant: a 1V decrease of Vp in the dark takes 140s, which is equivalent to 1.4 milux illumination. This measure includes the influence of operating the retina during capture, since ten binary images were output to be displayed during this experiment. The minimum integration time is limited by the speed at which the P voltage source can be operated, about 1µs and the maximum, by the dark current to about 140s; thus the dynamic range is very high: about 163dB.

In order to minimise power consumption, the supply voltages have to be lowered as much as possible. Trouble occurs with the computing operators when VddPE goes below 2.1V, due to the high level threshold inverter that won’t operate correctly below this voltage. Table 1 gives power measurements for VddPE=2.2V and VddCD=3.3V (as a standard I/O voltage). This is a worst case value for PPE and P short circ., which should be divided by two to get the mean value. It appears that PVL SAR 2.2 would use about 1W with a clock frequency of 80MHz, which is very encouraging. We have actually managed to further decrease power consumption by using a peculiar subthreshold mode of operation for the PE [5], but at the expense of its computing abilities.

Several algorithms are now available on PVL SAR 2.2 ranging from low to mid-level vision, including motion detection, segmentation, shape recognition (see Fig. 5).

3. CONCLUSION

PVL SAR 2.2, of which we have reported here the design, test and operation is, with its 128x128 pixels resolution, the largest Programmable Artificial Retina (PAR) — a CMOS sensor which embeds a tiny processor in each pixel — ever, to our knowledge, though it was fabricated in an oldish 0.8µm technology. Its size, 80mm², was not in any way a technological upper limit and shows that large size PARs are achievable. We are convinced that, by using aggressive technologies, sizes such as 512x512 pixels are realistic, though compactness remains a major issue with power and clock distribution. Algorithms from low to mid-level vision have been successfully implemented. The SIMD nature of such circuits makes their performance independant from the size of the array, except for algorithms that require non-local data communications. However as size grows, defects in the chip manufacturing will become a real problem, for which finding a solution is not as straightforward as in non-computational sensor, due to constraints on space and sequentiality in SIMD computation. We are nevertheless convinced that what CMOS imagers are bringing today to digital imaging, PARs will bring it with even more acuity to artificial vision: reactivity, compactness, low power and low cost through system integration. PVL SAR 2.2 shows this is now a matter of only a few years.

4. REFERENCES