HAL
open science

A Cellular Analog Network for MRF-based Video Motion Detection

Franck Luthon, Daniela Dragomirescu

To cite this version:

Franck Luthon, Daniela Dragomirescu. A Cellular Analog Network for MRF-based Video Motion Detection. IEEE Transactions on Circuits and Systems Part 1 Fundamental Theory and Applications, 1999, 46 (2), pp.281-293. hal-00408662

HAL Id: hal-00408662

https://hal.science/hal-00408662v1

Submitted on 31 Jul 2009

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.

A Cellular Analog Network for MRF-based Video Motion Detection

Franck Luthon and Daniela Dragomirescu, *Student Member, IEEE*

Abstract— **The implementation of a visual motion detection algorithm on an analog network is presented. The algorithm is based on Markov random field (MRF) modeling. Robust motion detection is achieved by using a spatiotemporal neighborhood for modeling pixel interactions. Not only are the moving edges detected, but also the inner part of moving regions. Moreover, the motion echo is eliminated. For hardware implementation, the algorithm is mapped onto a resistive network incorporating, at each pixel-node, light acquisition and on-chip processing. Elementary Markov cells are made of resistors and non-linear current sources driven by observations, derived from luminance information acquired by photoreceptors. Electrical simulations of the cell are reported. Experimental results with synthetic and real-world image sequences exhibit the performance of the network. For circuit design, the switched-current technique is used. The circuit is intended for submicronic complementary metal-oxide-semiconductor (CMOS) technology.**

Keywords— **Image sequences, Markov random fields (MRF's), motion detection, motion chip, smart sensor.**

I. Introduction

THE design of smart sensors for visual motion analysis is of interest for the following applications: telesurveillance; videoconferencing; videophone; compression; and computer vision. Motion analysis in image sequences mainly consists of four stages [1]: *detection* of temporal changes; *estimation* of optical flow or velocity; *segmentation* of different moving regions; and *interpretation* of the scene content. Here, we focus on robust motion detection. By robust motion detection we mean a process in between detection and segmentation of moving areas, i.e., obtention of a binary field accurately representing the projection (or mask) of moving objects in the image plane. Assuming a static camera and little variation of the scene lighting, motion information is derived from temporal differences in the intensity function I. To recover complete masks of moving objects, this basic information must be processed in order to reduce the influence of acquisition noise, to cancel the echo of motion, and to fill in the interior of masks (Fig. 1). For that purpose, a regularization technique is used, based on Markov random field (MRF) modeling and energy minimization via the iterated conditional modes (ICM) relaxation algorithm. But computation cost is high and software implementation is incompatible with real-time processing.

Manuscript received xxx; revised September 25, 1998.

F. Luthon is with the Signal and Image Laboratory, Grenoble Polytechnic National Institute, Grenoble, France (e-mail: Franck.Luthon@inpg.fr).

D. Dragomirescu was with the Bucharest Polytechnic University, Bucharest, Romania. She is now with the System Analysis and Architecture Laboratory, National Center for Scientific Research, Toulouse, France.

Publisher Item Identifier S 1057-7122(99)01579-2.

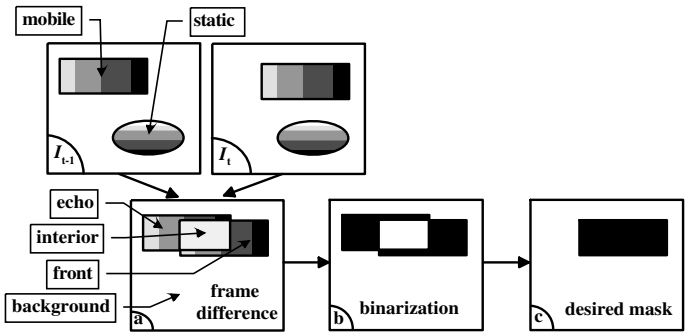

Fig. 1. Robust motion detection process illustrated by two consecutive images I_{t-1}, I_t of a synthetic sequence, representing a mobile rectangle and a static ellipse. (a) Observation: temporal changes. (b) Binarization: initial binary map. (c) Ideal output: motion mask.

Hardware implementation on special-purpose vision circuits (ASIC's) gives the best computation and communication efficiency. The main advantages of analog ASIC's are processing speed, low power consumption, small hardware size, and high pixel density, while the drawbacks are limited computational precision and flexibility, development times and financial cost. Two approaches exist for the design of smart sensors [2]: computational direction and biological models. Here, a computational approach is adopted. Interestingly, however, starting from a mathematical viewpoint (i.e., statistical regularization based on MRF) and mapping the algorithm onto an analog network results in a circuit which resembles, in many respects, a biologically inspired cellular neural nonlinear network (CNN) [3].

A brief survey of the literature reveals some interesting points. First, most approaches for motion-chip design are biologically inspired. They use either gradient-based schemes or correlation schemes to compute global image motion [4], [5], [6]. Second, from the application viewpoint, a clear distinction should be made between motion detection, estimation, or segmentation and between static and mobile camera. Moreover, robustness with respect to noise and to the number, size, and speed of mobile objects is of major concern. To our knowledge, no chip exists that recovers the complete masks of moving regions. Most circuits either detect moving edges only (motion transitions) or estimate the global velocity in the image plane. Many so-called motion chips are based on the detection on excited cells of positive and negative pulses in temporal derivatives. They actually compute basic temporal differences [7]. A few circuits report sparse or dense optical flow, but no segmentation is available (cf. optical flow computation chip with biological approach [8]). The first instance of a robust

optical flow computation chip is presented in [2]. A motion detection-and-estimation algorithm that is an amalgamation of biologically and computationally based models is implemented on a smart sensor. Third, many authors put emphasis on the effectiveness of adopting a current-mode approach (i.e., representing signals as currents instead of voltages) for the design of building blocks in analog very large scale integration (VLSI) neural systems, although they do not agree about the best MOS transistor operating mode: the subthreshold region (translinear circuits) [6], linear region [9], or saturation region.

The context in which we place our work is restricted to robust motion detection (i.e., recovery of moving masks without velocity estimation) in the case of a static camera. The algorithm is based on a computational (nonbiological) method, and an analog implementation is presented which uses a current-mode approach. The paper is organized as follows. Section II presents the mathematical model used for robust motion detection and the electrical analogy found for mapping the algorithm onto an analog network. Section III deals with circuit architecture. In section IV, the circuit design for VLSI implementation is detailed. Electrical simulations exhibit the cell behavior. In section V, the network is simulated and motion detection results with both synthetic and real-world image sequences are reported.

II. MRF Framework with Electrical Modeling

A. Observations, Labels, and Neighborhood Structure

Motion detection aims at labeling each pixel of a video sequence according to moving and static areas in consecutive images. Usual assumptions are a static camera and almost constant illumination of the scene. Thus, the raw observation o_s at pixel (or *site*) $s = (i, j, t)$ is given by temporal changes[1] in the intensity function $o_s = |I(i, j, t) - I(i, j, t-1)|$ where (i, j) are the discrete spatial coordinates of pixel s and t is the discrete time instant. The motion label l_s to be given to site s may take one of the two values

$$l_s = \begin{cases} m \equiv \text{“1”} & \text{if } s \in \texttt{moving} \text{ area}, \\ b \equiv \text{“0”} & \text{if } s \in \text{static } \texttt{background}. \end{cases} \tag{1}$$

Let $l = \{l_s, s \in S\}$ (respectively $o = \{o_s, s \in S\}$) be one realization of label field L (respectively observation field O) on image S at time[2] t. The random field L is assumed to have MRF properties, i.e., a Gibbs distribution and a neighborhood structure. The spatiotemporal neighborhood η_s of site s is given in Fig. 2.

According to [10], maximizing the *a posteriori* probability of label field realization l, given observation field realization o, is equivalent to minimizing an energy or cost function U consisting of two terms $U = U_m(l) + U_a(o, l)$.

[1]The absolute value is taken in order to be insensitive to the sign of contrast between mobile and static regions.

[2]Whenever needed, a temporal subscript will be explicitly added in the notations (e.g., o_{t+1} for observation field o at time $t+1$).

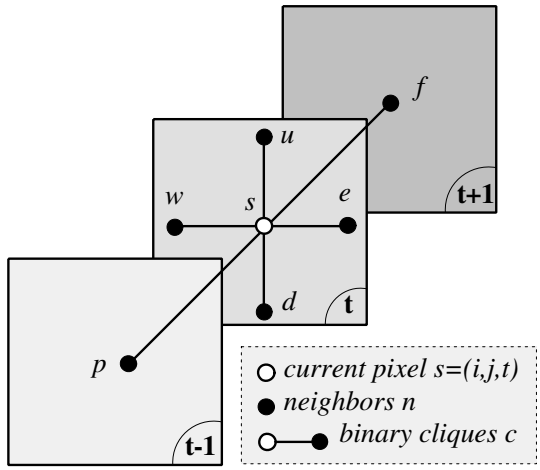

Fig. 2. Spatiotemporal neighborhood η_s with binary cliques (u, d, e, w, p, f standing respectively for up, down, east, west, past, and future neighbor).

B. Energy Functions

The model energy $U_m(l)$, embedding *a priori* modeling of spatiotemporal interactions between sites, is a regularization term similar to smoothness constraints classically used to solve ill-posed problems. Let C be the set of all binary cliques $c = (s, n)$ constituting the neighborhoods where $n \in \eta_s$ stands for any neighbor of s, spatial or temporal. Then, $U_m(l)$ is given by

$$U_m(l) = \sum_{c \in C} V_c(l_s, l_n) \quad \text{with} \quad V_c(l_s, l_n) = \beta_c (l_s - l_n)^2. \tag{2}$$

$V_c(l_s, l_n)$ is an elementary potential function defined on each clique $c = (s, n)$. β_c stands for a generic parameter that may take one of three positive values β_s, β_p and β_f depending on the type of clique (spatial, past or future). These model parameters may be chosen experimentally, or estimated with an expectation-maximization (EM) algorithm. The expression of V_c favors spatiotemporal homogeneity since a configuration with two identical neighboring labels does not increase energy U_m, while U_m increases of a quantity proportional to β_c in the case of two different labels. Moreover, the quadratic expression in (2) is compatible with electrical implementation, as shown below. More sophisticated expressions could be used to comply with robust estimators. Some of them are also implementable in hardware (e.g., antibump function in [5]).

The attachment energy $U_a(o, l)$ is a link-to-data energy, reflecting the good match (attachment) between labels and observations. Its expression is given by [11]

$$U_a(o, l) = \frac{1}{2\sigma^2} \sum_{s \in S} [o_s - \Psi(l_s)]^2 \tag{3}$$

where σ^2 is the observation variance that may be estimated online or only from the first image pair of the sequence. Ψ is a function of labels, the role of which is to model typical configurations that may arise in the observation field. For

motion detection, Ψ may be defined as[3]

$$\Psi(l_s) = \begin{cases} 0 & \text{if} \quad l_s = l_p = b \quad \text{(background)} \\ \alpha_1 > 0 & \text{if} \quad l_s = l_p = m \quad \text{(motion)} \\ \alpha_2 >> \alpha_1 & \text{if} \quad l_s \neq l_p \quad \text{(transition)} \end{cases} \tag{4}$$

where α_1 and α_2 are positive parameters that may be predefined or estimated online [11].

C. Software Energy Minimization via ICM Relaxation

The algorithm[4] flow-chart is given in Fig. 3. Because of

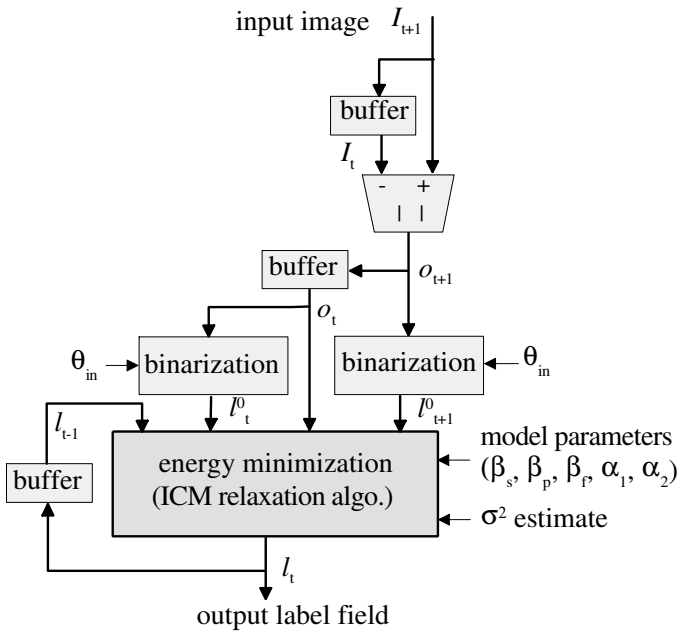

Fig. 3. Algorithm block diagram.

the spatiotemporal neighborhood chosen for motion detection (Fig. 2), three label fields are required for relaxation to start: past, present and future. Therefore, the algorithm implies a one image delay since the future image at time $t+1$ is required to get the label field corresponding to time t.

The minimum energy may be computed using either stochastic relaxation algorithms, like simulated annealing [10], or deterministic algorithms, such as iterated conditional modes (ICM) [12]. Here, the iterative method ICM is used because of its lower computation cost. But it is suboptimal in the sense that it may get stuck in the first encountered local minimum of the energy function. Therefore, label field initialization is crucial in order to be sure to converge to the global minimum. At time t past label field l_{t-1} is already available as the result of previous relaxation. However, present and future label fields must be initialized. This is done by binarization of observation fields o_t, o_{t+1} (simple comparison to a threshold θ_{in} or likelihood test methods), yielding initial field estimates l^0_t, l^0_{t+1}. Then, ICM relaxation runs over field L and field L at time t is scanned pixel per pixel. At each site, a local decision is taken. The label given to a site is the one inducing the lower energy within its neighborhood. This relaxation iterates until convergence is achieved, based on a relative energy decrease criterion (typically $\Delta U/U < 0.01\%$). The processing rate is typically of 1 image/s on a Sparc-10 workstation for images of size 128×128.

Instead of software implementation via ICM for finding the global minimum solution, hardware implementation on an analog VLSI network is an highly efficient alternative in achieving real-time processing. The basic principle consists of allowing an electrical network find the minimum energy configuration by relaxing to its state of minimal power dissipation[5].

D. Electrical Analogy

The idea of mapping the algorithm on an analog network stems from a close study of the equation underlying the state of minimal energy. Let us use the following explicit notations, corresponding to current pixel $s = (i, j, t)$: $o_{ij} = o_s$, $l_{ij} = l_s$, $p_{ij} = l_p$ for past label and $f_{ij} = l_f$ for future label. Then, denoting $K = 1/2\sigma^2$, the total energy U may be rewritten as

$$\begin{aligned} U \quad = \quad & \sum_{i,j} \beta_s (l_{ij} - l_{i+1,j})^2 + \beta_s (l_{ij} - l_{i-1,j})^2 \\ & + \beta_s (l_{ij} - l_{i,j+1})^2 + \beta_s (l_{ij} - l_{i,j-1})^2 + \beta_f (l_{ij} - f_{ij})^2 \\ & + \beta_p (l_{ij} - p_{ij})^2 + K \left[o_{ij} - \Psi(l_{ij}) \right]^2 . \end{aligned} \tag{5}$$

For analog implementation, we must suppose that labels l_{ij} may take continuous values[6] in the range b to m (assuming $m > b$), instead of binary values in the set $\{b; m\}$. The minimum of U with respect to all l_{ij} corresponds to null partial derivatives

$$\begin{aligned} \forall ij \ \frac{\partial U}{\partial l_{ij}} = 0 \quad \Leftrightarrow \quad & \beta_s \nabla^2 l_{ij} + \beta_f (l_{ij} - f_{ij}) + \beta_p (l_{ij} - p_{ij}) \\ & - K \frac{\partial \Psi}{\partial l_{ij}} \left[o_{ij} - \Psi(l_{ij}) \right] = 0 \end{aligned} \tag{6}$$

where $\nabla^2 l_{ij} = 4 l_{ij} - l_{i+1,j} - l_{i-1,j} - l_{i,j+1} - l_{i,j-1}$ is a discrete five-point approximation of the spatial Laplacian of l_{ij}. Now, an electrical analogy with Kirchhoff's laws may be done by identifying each term in (6) to a current converging to node ij. Labels correspond to electrical potentials and label differences to voltages. Parameters β_c

[3]This expression relies on the basic assumption that an object, when moving, generates in the image plane high-intensity variations in transition areas (front and echo areas), and smaller variations in sliding areas (i.e., inner parts) of moving objects (Fig. 1).

[4]Compared to the original two-pass algorithm proposed by Bouthémy [11], the modified algorithm proposed here leads to a one-pass algorithm, suitable for analog hardware implementation on a network made of a single processing layer, with the same performance with regard to the quality of motion detection results.

[5]In contrast to computer implementation for which initialization l^0_t of the present label field is required, initialization of electrical potentials is not necessary in analog hardware implementation since the circuit will relax to its state of minimal power dissipation, whatever its initial conditions are.

[6]To recover the final binary labels relevant for motion detection, a simple comparison to a threshold θ_{out} applied to these continuous labels is required.

represent conductances. The last term in (6) is a current source depending on observation o_{ij} and on attachment function Ψ. Its role is similar to the attachment energy $U_a(o,l)$ (i.e., observation fit). Moreover, it should be easily implemented on an analog circuit (which means that a linear function of l_{ij} is required for Ψ). Bearing these constraints in mind, and inspired by (4), there are many possible candidates for Ψ. The simplest form would be $\Psi(l_{ij}) = Al_{ij}$ with $A =$ constant. However, to properly cancel motion echo, the following expression is proposed: $\Psi(l_{ij}) = A_{ij}(l_{ij} - p_{ij})$ where $A_{ij} = \alpha_2(p_{ij} - f_{ij}) - \alpha_1$ with $\alpha_2 > \alpha_1 > 0$. Replacing Ψ and its derivative in (6), and introducing a leakage capacitance specific of the circuit dynamics yields, for all ij,

$$C\frac{\partial l_{ij}}{\partial t} = \beta_s \nabla^2 l_{ij} + \beta_f(l_{ij} - f_{ij}) + \beta_a(l_{ij} - p_{ij}) - KA_{ij}o_{ij} \quad (7)$$

with $\beta_a = \beta_p + KA_{ij}^2$. At electrical equilibrium, the left member of (7) tends to zero, corresponding to the minimum energy solution. Thus, this linear system of differential equations may be solved via an analog resistive network as implemented below.

III. Circuit Architecture

A. Elementary Markov Cell

The cell shown in Fig. 4 contains six resistors (four spatial resistors R_s, two temporal resistors R_f, R_p) and two voltage-controlled current sources J_1 and J_2. The four resistors R_s are connected to the four nearest neighboring cells. Past and future neighbors are actually not physical but, rather, are virtual neighbors in the sense that they do not correspond to actual electrical cells as spatial neighbors do. They only represent side information entering the physical nodes. Theoretically, resistor R_p, at least, should be adjustable since it depends on σ^2 and on ij past and future labels. In practical implementation, however, one can choose a fixed value since the temporal dependency of R_p can be compensated with a proper choice of α_2 and α_1. Moreover, generator J_2 is connected to $V_{cc}/2$ in order to be able to generate current in both directions, depending on the sign of $(f_{ij} - p_{ij})$, while J_1 is connected to V_{cc} since it always injects current into the node (o_{ij} is always ≥ 0).

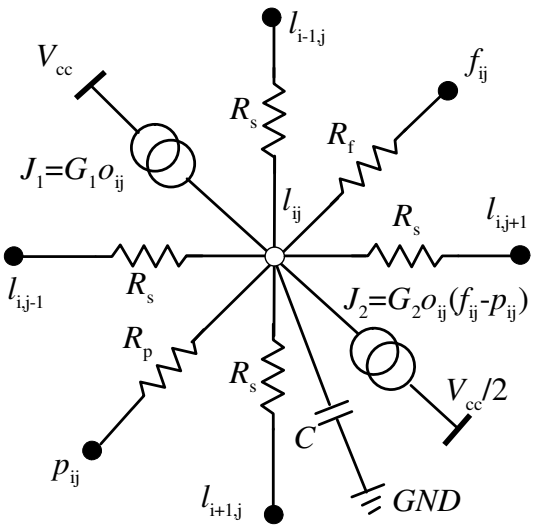

Fig. 4. Elementary Markov cell with $R_s = 1/\beta_s$, $R_f = 1/\beta_f$, $R_p = 1/\beta_a$, $G_1 = K\alpha_1$, and $G_2 = K\alpha_2$.

The elementary cell of Fig. 4 is in charge of a local electrical processing at node ij. After all the required command voltages (o_{ij}, f_{ij}, p_{ij}) are set up onto the cell, one lets the cell relax. When there is no motion information ($o_{ij} = 0$), current sources are inactive and electrical potential l_{ij} is only influenced by its neighbors (spatiotemporal smoothing due to resistors). When motion information is present ($o_{ij} \neq 0$), current sources are active and a current is injected into (respectively, extracted from) node ij, depending on the sign of $(f_{ij} - p_{ij})$, i.e., on the type of transition (static-to-mobile, mobile-to-static, etc.)[7].

Current sources J_1, J_2 ensure the good fit to observed data. J_1 is responsible for recovering the inner part of moving masks, while J_2 handles transition areas (front and echo). Let ΔJ be the sum of currents delivered by the two sources. Four typical cases may occur[8], corresponding, respectively, to background, interior, echo, and front areas (Fig. 1).

1. `Static-to-static pixel`: $o_{ij} = 0$ and $p_{ij} = f_{ij} = 0$. Then current sources do not work: $\Delta J = 0$.
2. `Mobile-to-mobile pixel`: o_{ij} small and $p_{ij} = 1$
   - With `adequate input threshold` (θ_{in} low): $f_{ij} = 1$. Then generator J_1 only is active: $\Delta J = G_1 o_{ij} > 0$. A small current is injected into node ij, increasing slightly its electrical potential. This allows to fill in the inner part of moving masks.
   - With `inadequate input threshold` (θ_{in} too high): $f_{ij} = 0$. Both generators are active: $\Delta J = (G_1 - G_2)o_{ij} < 0$ since $G_2 > G_1$. A small current is extracted from node ij, decreasing slightly its electrical potential. However, since past neighbor is labeled as mobile and observation is small, this influence is not critical and does not break mask homogeneity.
3. `Mobile-to-static transition`: o_{ij} large and $p_{ij} = 1, f_{ij} = 0$. $\Delta J = (G_1 - G_2)o_{ij} \ll 0$. A large current is pumped out of the node, decreasing drastically its electrical potential. This allows to cancel the echo area.
4. `Static-to-mobile transition`: o_{ij} large and $p_{ij} = 0, f_{ij} = 1$. $\Delta J = (G_1 + G_2)o_{ij} \gg 0$. A large current is pumped into the node, increasing drastically its electrical potential. This allows to enhance the moving front area.

When steady state is reached after relaxation, the electrical potential l_{ij} is output to a comparator, giving a binary value corresponding to the final motion label at pixel ij. This Markov cell has the desired behavior for recovering the

[7] $(f_{ij} - p_{ij})$ may take only three values: 0, +1 or −1. Indeed, p_{ij} and f_{ij} take binary values in the set {"0" $\equiv GND$; "1" $\equiv V_{cc}$} since they are obtained after thresholding electrical potentials. Hence, we have $(f_{ij} - p_{ij}) = sgn(f_{ij} - p_{ij})$.

[8] We suppose that motion detection was correctly performed at the previous instant (i.e., past label p_{ij} has the right value).

complete motion masks, and it eliminates isolated points due to acquisition noise, thanks to the resistive smoothing. This clearly outperforms a simple frame difference.

B. Complete Cell

The elementary Markov cell needs some circuitry around it in order to work properly when integrated in a focal plane network. The complete cell (Fig. 5) must include a photoreceptor plus the required circuits to compute o_{ij}, f_{ij}, and p_{ij} to give the desired binary output, i.e., static or mobile labeling, and must be interconnected to other cells and to line and column shift registers (or address decoders) for outputting results on a video monitor.

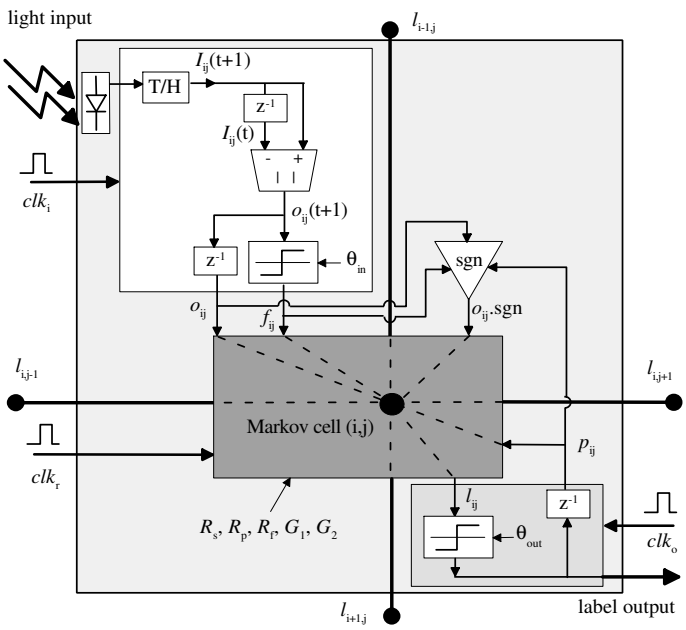

Fig. 5. Complete cell with acquisition (photosensor), preprocessing (computation of o_{ij}, f_{ij}), processing (Markov cell relaxation), and postprocessing (binarization, and memory output latch to get p_{ij}). The preprocessing stage with the track and hold (T/H) circuit is depicted inside the white box.

Light intensity at pixel ij is acquired continuously by a photosensor. An analog T/H circuit samples the input current at a rate corresponding to standard video rate, e.g., 30 images/s (clock clk_i). Observation and future label are computed by elementary circuits (delay memories, absolute value of difference, and comparator). This constitutes the preprocessing stage.

The processing stage (relaxation) is embedded in the Markov cell itself (dark box in the middle of Fig. 5). Once command voltages are settled, clock clk_r triggers the network relaxation, allowing all Markov cells to relax in parallel to their state of minimal power dissipation.

The postprocessing stage consists of a comparator and a memory output latch. When steady state is reached after network relaxation, clock clk_o triggers output binarization and storage of final label at the same rate as input sampling. A simple threshold θ_{out}, applied on the electrical potential, gives the desired binary label. The binary output is also stored in memory to give p_{ij} for the next clock trigger clk_r. The timing diagram is given in Fig. 6.

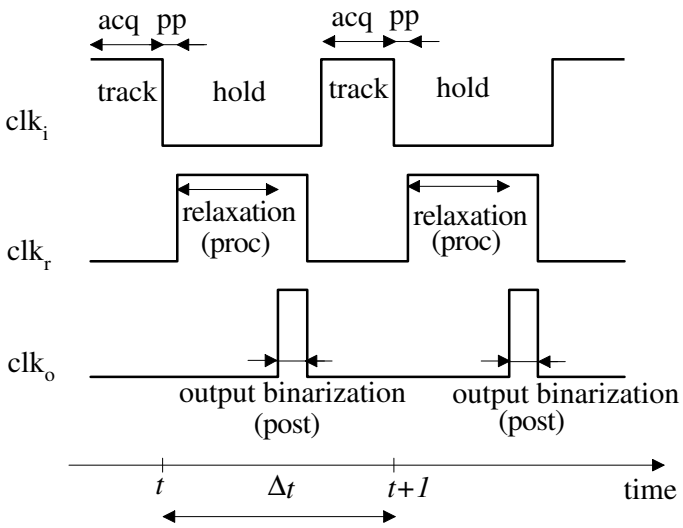

Fig. 6. Timing diagram ($\Delta t = 33ms$). *acq*, *pp*, *proc*, and *post* stand respectively for acquisition, preprocessing, processing, and postprocessing.

C. Network Architecture

The network architecture is given in Fig. 7. In addition to clocks clk_i, clk_r, and clk_o, two other clocks, clk_v and clk_h, are required to trigger the transfers of cell binary outputs into vertical and horizontal shift registers, with subsequent serial transmission to a video sequencer for visualization on a monitor.

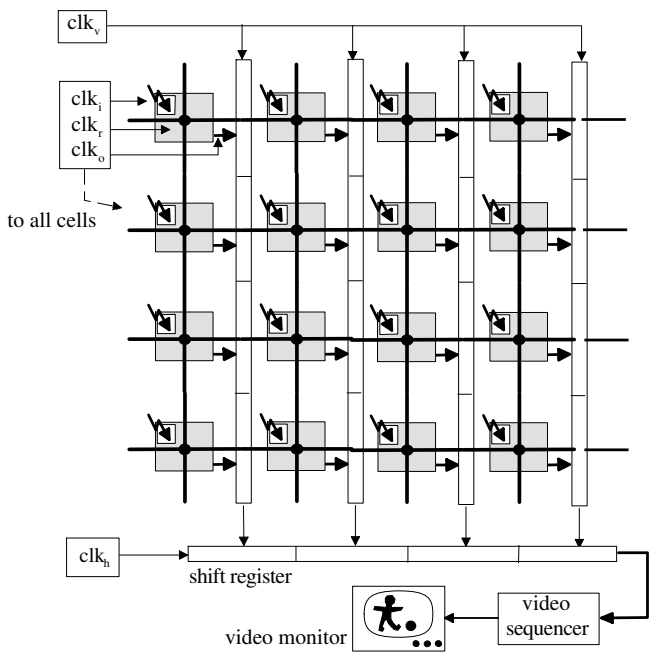

Fig. 7. Network architecture.

IV. VLSI Implementation

A. Switched-Current Mode for Preprocessing Stage

For hardware implementation, one may choose among three approaches. The most usual way is to handle information as *voltages* (classical technology using operational amplifiers (op-amps)). Another choice is the technique of *switched capacitors*, where information is represented as electrical charges. This usually requires linear precision capacitors fabricated with additional process step over standard MOS technology.

An attractive alternative for smart-pixel CNN implementation is to work in *current-mode* [13]. The technique of *switched currents* (SI) is well suited for the preprocessing stage. The SI-concept was introduced by Hughes *et al.* [14] for the design of analog systems in standard MOS technology for sampled-data signal processing. Here, the sampling is temporal, corresponding to standard video rate of 30 images/s, i.e., $\Delta t = 33ms$ between two consecutive images. The main advantages of SI technique are low-voltage operation, high speed, no need for precise capacitors or op-amps.

The basic building block of the SI technique is the dynamic current mirror, which is controlled by a switching transistor driven by a clock signal Clk. When Clk is set, the circuit acts as a classical inverting current amplifier. When Clk is reset, the circuit has a memory effect. The output current keeps the value it had at the switching instant. This circuit thus plays the role of a basic current T/H circuit, i.e., an analog current memory.

A keystone circuit for the SI technique is the analog delay memory (z^{-1}). The basic dynamic current mirror cannot be used as such to implement a delay memory, mainly because of the *switch clock feedthrough* effect between signals and clocks. In order to reduce the influence of clock feedthrough, a current cancellation mirror is used, based on the idea of Song *et al.* [15]. Many other solutions for improved dynamic current mirrors are reported in the literature, e.g., [16]. This remains an active field of research.

The circuit in charge of computing observation $o_{ij}(t+1)$ is given in Fig. 8. Its relative complexity (20 transistors) is due to a design effort made for reducing the dead zone arising for low output currents. SPICE analysis shows the behavior of this circuit (Fig. 9). The insensitivity to small currents is favorable to the processing since it acts as a nonlinear filter applied on observations by lowering the influence of too small currents that are not significant of actual intensity changes induced by motion.

Since the observation computed by the AVD circuit is represented as a current, the input thresholding circuit (θ_{in}) required for computing f_{ij} works in current mode. Thus, a current comparator is needed for the preprocessing stage. One can use circuits such as the one proposed in [17], although a simpler circuit was used here.

Fig. 10 reports SPICE simulation of the whole preprocessing stage in the case of a sinusoidal variation of input light intensity. Electrical signals, corresponding to observation and future label, are of sufficient quality for our purpose despite the peaks observable on other signals involved

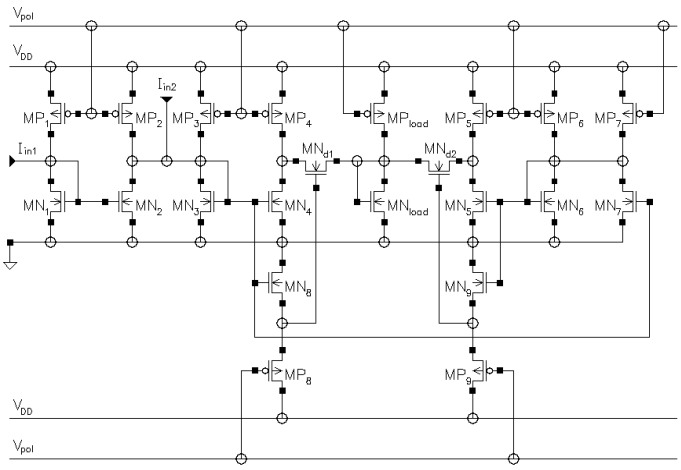

Fig. 8. Circuit schematics of the absolute value of difference (AVD) circuit. I_{in1} and I_{in2} are the two input currents. MN_{load}, MP_{load} is the output load where the resulting observation is available.

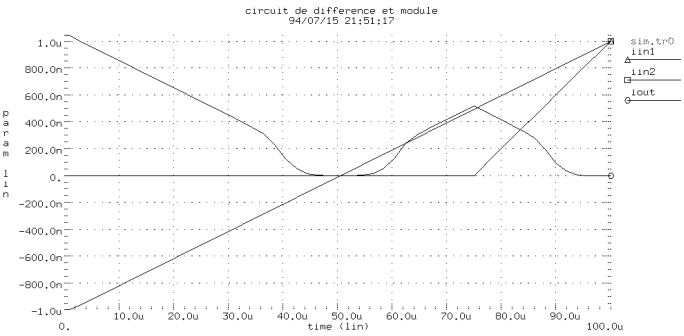

Fig. 9. SPICE simulation of the AVD circuit. I_{in1} and I_{in2} are the two input currents. I_{out} is the AVD output current. Note the dead zone effect arising when current difference is lower than $300nA$.

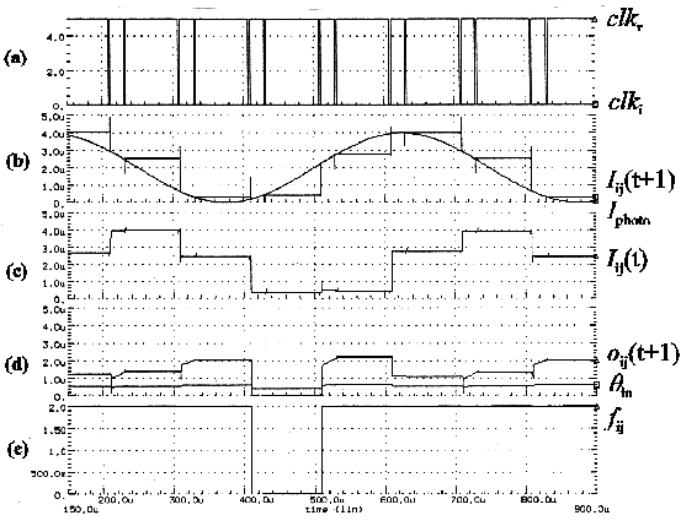

Fig. 10. SPICE simulation of preprocessing stage in the case of a sinusoidal variation of input current. (a) Clocks clk_i and clk_r. (b) Input current I_{photo} undergoing a sinusoidal variation for simulation purposes, and output of the T/H circuit $I_{ij}(t+1)$. (c) Delayed current $I_{ij}(t)$. (d) Output of AVD circuit $o_{ij}(t+1)$ and threshold level θ_{in} used for binarization. (e) Future label f_{ij}.

in preprocessing stage.

A voltage-controlled current inverter (sgn) is required to get a signed value of o_{ij}; the sign $0, \pm 1$ depending on $(f_{ij} - p_{ij})$. It might be implemented either with current mirrors or op-amps (design not detailed here).

Concerning the photoreceptor, either a phototransistor or a photodiode may be used [13]. An important issue is light adaptation [5].

B. Postprocessing Circuitry

The building blocks required for the design of the postprocessing stage work in classical voltage-mode, and have already been addressed in the literature.

An output thresholding circuit is required to compare the steady-state electrical potential l_{ij} with a fixed reference voltage θ_{out} for binarization after relaxation. This output comparator working in voltage mode can be implemented with a classical transductance amplifier followed by a comparator (op-amp). An output latch (z^{-1}) is also required to get p_{ij}. For outputting binary labels one may choose among various solutions. Either shift-registers (made of flip flops), as proposed here, or classical memory address-decoding (requiring only one transistor at each cell output, which is less expensive in terms of area) may be used.

C. Markov Cell Design and Simulation

Voltage-controlled current sources J_1 and J_2 may be realized with a single MOS transistor that consumes little area.

Markov cell resistors (especially R_p) should be adjustable in order to be able to tune model parameters β_c according to acquisition and lighting conditions. Moreover, floating resistors are needed, and they should be as symmetric as possible to get an isotropic behavior. A simple solution consists in using the channel of a MOS transistor working in the linear region (Fig. 11). The aspect ratio W/L of

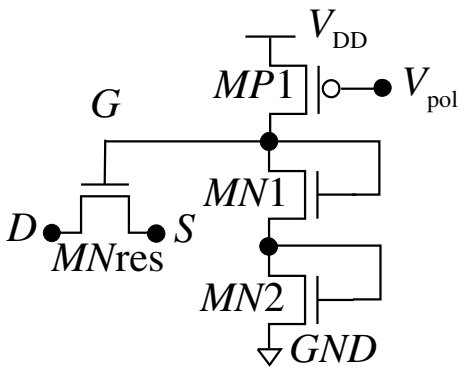

Fig. 11. Adjustable and floating MOS resistor. Circuit schematics. Transistors MP_1, MN_1, and MN_2 constitute the polarization circuit. MN_{res} is the floating adjustable resistor.

MN_{res} must be small enough to have suitable resistances (typically $100k\Omega$). Therefore, we design snake-like MOS resistors (cf. layout). The resistor value is adjusted by the grid potential V_G, which is driven by the command voltage V_{pol}. A single polarization circuit can be used for the four spatial resistors R_s in each cell, which spares nine transistors per cell. Electrical simulations (HDL-A simulator[9]) for three different polarizations of the resistor circuit are given in Fig. 12, when V_{DS} is varied in the range $[-0.5V; 0.5V]$. One obtains nonlinear resistors. Interestingly enough, however, is the fact that this nonlinearity is favorable for motion detection, as explained below.

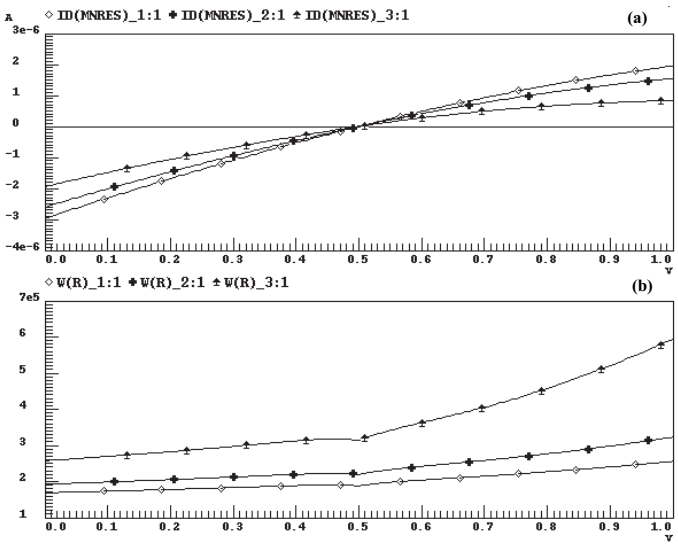

Fig. 12. Electrical simulation of the adjustable MOS resistor for three different polarizations V_{pol}: $2.18V$ ($\diamond$); $2.28V$ ($+$); $2.38V$ ($\spadesuit$). Source voltage applied to MN_{res} is constant ($V_S = 0.5V$) while V_D is varied in the range $[0; 1V]$. (a) Drain current in transistor MN_{res} as a function of V_D. (b) Resistance variation as a function of V_D.

Electrical simulation of the Markov cell is given in Fig. 13. For these simulations, all command voltages were computed externally and fed into the cell description-files. All spatial neighbors were set arbitrarily to potential $V_{cc}/2$ in order to inhibit spatial context so that only the temporal behavior is emphasized. A voltage V_{obs} is used to represent the observation signal o_{ij} that controls the two current sources. Current sources are ideal. Fig. 13(a) shows the current generated by J_2 as a function of V_{obs} in the four cases that can arise: 1) for static-to-mobile pixel (curve $*$), current increases; 2) for mobile-to-static pixel (curve $+$), current decreases; and 3) in the two other cases, current is zero. Fig. 13(b) shows the evolution of current J_1 which increases linearly with observation. Fig. 13(c) shows the variation of electrical potential at node ij for a Markov cell made of ideal resistors[10]. Fig. 13(d) shows electrical simulations obtained for a cell made of resistors designed with MOS transistors. They are in good agreement with the theoretical behavior depicted in Fig. 13(c). The nonlin-

[9]HDL-A and Eldo are modeling and simulation tools, developed by ANACAD Electrical Engineering Software, for analog and mixed-signal simulation with VHDL-based language and SPICE syntax.

[10]Note that for static-to-static transition (curve $\diamond$), the simulation reports a slightly increasing potential, whereas it should stay constant. This is not a bad behavior of the cell, but only results from the fact that V_{obs} is linearly increased for simulation purposes. In a real situation, however, a static pixel gives a null observation $V_{obs} = 0$. Hence, l_{ij} would actually remain constant.

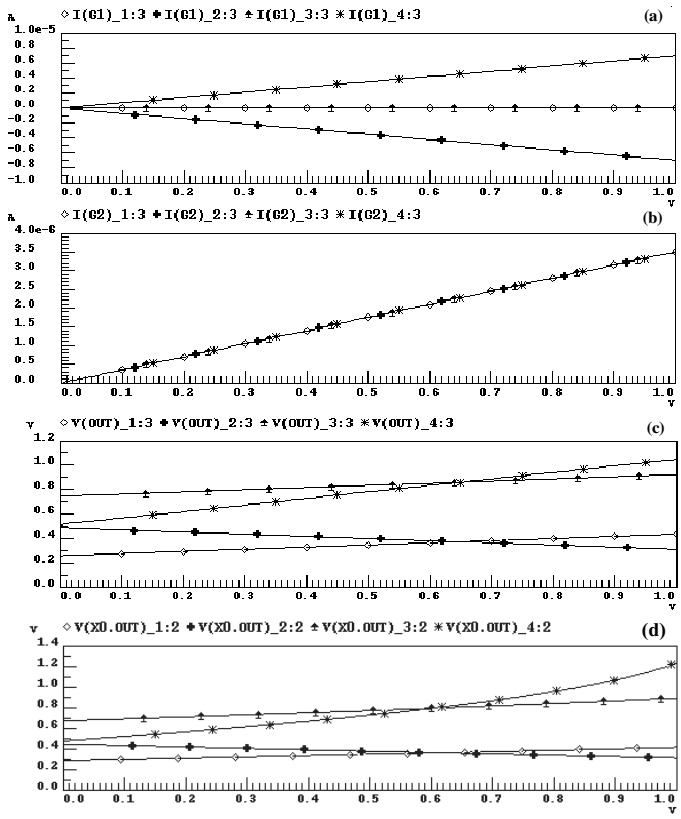

Fig. 13. Electrical simulation of Markov cell. Curve symbols correspond to the four typical cases: static-to-static (◇); mobile-to-static (+); mobile-to-mobile (♠); and static-to-mobile (∗). Observation voltage $V_{obs} \equiv o_{ij}$ is varied linearly in the range $[0; 1V]$ for simulation purpose. (a) Current generated by J_2 as a function of V_{obs}. (b) Current generated by J_1. (c) Electrical potential $V_{out} \equiv l_{ij}$ for a cell made of ideal resistors (Eldo simulator). (d) Idem, but for a cell made of real MOS resistors (HDL-A simulator).

earity is particularly visible in the case of static-to-mobile transition (curve ∗). But instead of being a drawback, it acts favorably for enhancing the front area of a moving object, which is a very interesting behavior. Indeed, it acts as a kind of smooth line process [8].

D. Layout and Electrical Characteristics

A layout of the complete cell was designed and a netlist was extracted, with ES2[11] $1.2\mu m$ CMOS technology. The cell contains about 150 transistors. Cell size is about $220\mu m \times 220\mu m$, corresponding to a density of 25 cells/mm^2. With a $0.5\mu m$ technology the size would be about $150\mu m \times 150\mu m$. In Fig. 14 one can see the photoreceptor (upper left), the six snake-like MOS resistors implementing R_s, R_p, R_f (in dark in the middle), the capacitors for current analog master-slave delay-memories (in dark at the bottom), and all the interconnections required (30% of the whole surface). This layout reveals one major problem that has yet to be addressed. The fill factor (photosensitive area/total cell size) is only 10%. It is an important issue since it governs the achievable spatial resolution of

[11] European Silicon Structures.

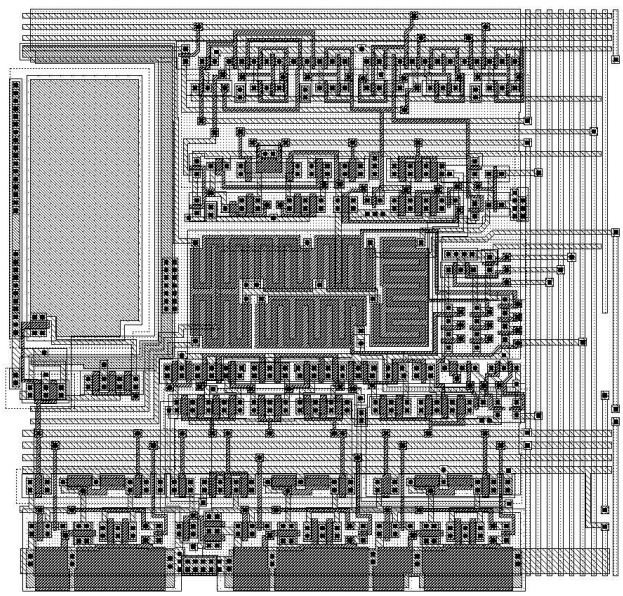

Fig. 14. Complete cell layout (ES2 ECPD12 technology).

the chip.

The power dissipation for a network of size 64×64 is evaluated to be about $150mW$. Resistor values are in the range $100k\Omega$ to $1M\Omega$. The order of magnitude for currents is about $1\mu A$ with $V_{cc} = 1V$. Relaxation duration is typically less than $1\mu s$, so that real-time motion detection is achievable.

V. Experimental Results

Electrical simulations of the network were performed with Eldo and HDL-A. Since the complete chip with photoreceptors is not yet fabricated, gray-level images stored in computer memory are used. Pixels are 8-bit coded. Command voltages (o_{ij}, f_{ij}) are precomputed externally and entered into the Markov cell description files. Hence, only the electrical behavior of network-interconnected Markov cells is simulated. Concerning parameter setting, threshold values $\theta_{in}, \theta_{out}$, given in the captions, refer to the 256 gray-level dynamic range of images. They are adjusted depending on the sequence (lighting conditions, noise, etc.). For a proper working of the network, the condition $R_f \leq R_p < R_s$ is required. R_s and R_f values are constant for all sequences. Typical values are $R_s = 400k\Omega, R_f = 200k\Omega, R_p = 220k\Omega$. G_1 and G_2 are the transconductances of current sources J_1 and J_2. Their order of magnitude is $\sim 10^{-5}$ Siemens. An important feature is the scalability of all parameters, that can be chosen according to the maximum power dissipation acceptable in the network.

A synthetic sequence of eight images is shown on the upper row of Fig. 15. Image size, and hence network size, is 16×16. The sequence contains two dark squares moving on a noisy background. The squares have a nonuniform intensity (linear variation of gray level). The second row represents observation fields where transition areas (front and echo) are visible. In the inner part of moving objects

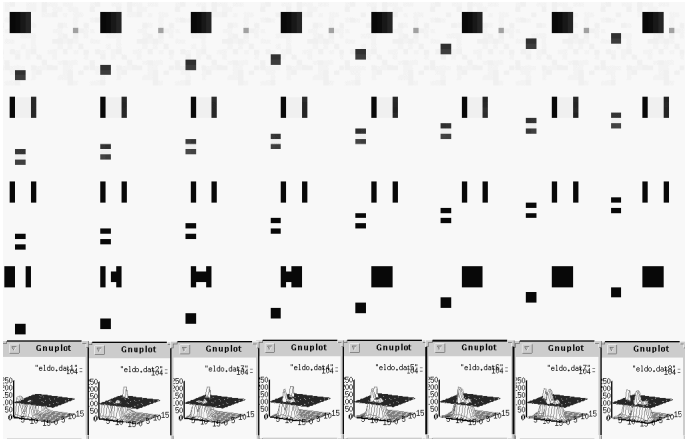

Fig. 15. *From top to bottom*: (a) Synthetic image sequence. (b) Sequence of observations. (c) Future label fields. (d) Final motion masks. (e) Three-dimensional maps of electrical potentials; the dark horizontal plane indicates the output threshold level θ_{out}. Note that observation and future fields are shifted of one image to the left with respect to image sequence and mask sequence. Parameter set $\theta_{in} = 11, \theta_{out} = 104$, $R_s = 400k\Omega$, $R_f = 200k\Omega$, $R_p = 206k\Omega$, $G_1 = 950.10^{-8} Siemens$, and $G_2 = 2015.10^{-8} Siemens$.

(sliding areas) observations are small but not zero (light gray barely visible). The coarse estimate of future label fields (binary maps) is given in third row. Because of a too high input threshold θ_{in}, only transition areas are retained and inner parts are eliminated. They are restored, however, by the relaxation process, as shown in the fourth row, representing the moving masks obtained after thresholding steady-state electrical potentials of the relaxed network. The masks are perfectly extracted after the fourth image[12]. The echo is canceled, and the inner part of the masks is filled in. The lower row shows the three-dimensional maps of electrical potentials. Nodes having a high electrical potential correspond to mobile pixels, while nodes at a potential near the ground voltage correspond to static pixels.

Five images of a real-world sequence are shown in the upper row of Fig. 16. This street sequence contains a white car moving in a street as seen from the top of a building. Other cars in the background are parked. This sequence was acquired with a three-CCD (charge coupled device) camera (only the 8-bit green component is processed). Image size, and hence network size, is 64×64. The second row shows the observation fields, where echo is clearly visible. The third row gives the future label fields, with enlarged masks because of the echo phenomenon. The bottom row of Fig. 16 gives the moving masks detected by the network after relaxation. Although the car is moving rapidly (passing the field of view within eight images) and is poorly-textured (uniform white roof and hood), it is fairly well detected.

The circuit is insensitive to contrast: dark or light moving objects are equally well detected. Other simulations not reported here prove that the network also detects slow

[12]The algorithm adapts after a few images since no past information is available at the beginning of the sequence.

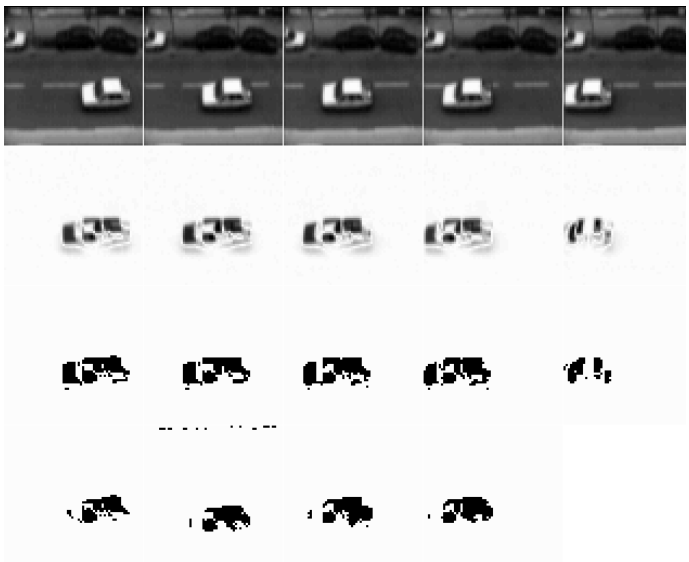

Fig. 16. *From top to bottom*: (a) Street sequence. (b) Sequence of observations at five time instants. (c) Future label fields. (d) Final motion masks. Note that observation and future fields are shifted of one image to the left with respect to image sequence and mask sequence. Parameter set $\theta_{in} = 30, \theta_{out} = 130$, $R_s = 400k\Omega$, $R_f = 200k\Omega$, $R_p = 216k\Omega$, $G_1 = 700.10^{-8} Siemens$, and $G_2 = 1400.10^{-8} Siemens$.

or fast moving objects, small objects, or objects in noisy sequences.

VI. Discussion

This article reports the design of a smart sensor for robust motion detection. Starting from a computational method, the analog implementation leads to what we call a cellular analog network (CAN), rather than CNN, in order to emphasis the fact that it is not biologically inspired. Still, the resulting circuit reveals analogy with biological neural networks. The proposed chip is a full vision system, coupling sensor and computation, that recovers accurately the complete masks of moving objects in the case of a static camera. The processing proved to be robust in various situations, such as contrast, noise, speed, number, and size of moving objects.

The counterpart of this smart low-level processing is a fill factor that is not yet satisfactory. Therefore, much work has still to be done, such as further investigation for the design of all building blocks in the circuit, study of submicronic technological aspects (interconnections, power consumption, surface, delay on clock-lines, etc.), and precise evaluation and validation of the circuit (influence of all parameters). Interesting is the fact that, similar to biological systems, accuracy is not a critical issue. We found that analog implementation even outperforms computer simulations, both for processing speed and detection quality.

Acknowledgment

The authors wish to thank V. G. Popescu and S. Popescu for their contribution to this work and J. Hérault for fruitful discussions.

REFERENCES

[1] A. Mitiche and P. Bouthémy, "Computation and analysis of image motion: A synopsis of current problems and methods," *International Journal of Computer Vision*, vol. 19, no. 1, pp. 29–55, 1996.

[2] R. Etienne-Cummings, J. Van der Spiegel, and P. Mueller, "A focal plane visual motion measurement sensor," *IEEE Trans. on Circuits and Systems-I*, vol. 44, no. 1, pp. 55–66, Jan. 1997.

[3] L. O. Chua and T. Roska, "The CNN paradigm," *IEEE Trans. on Circuits and Systems-I*, vol. 40, no. 3, pp. 147–156, Mar. 1993.

[4] T. Horiuchi, W. Bair, B. Bishofberger, A. Moore, C. Koch, and J. Lazzaro, "Computing motion using analog VLSI vision chips: An experimental comparison among different approaches," *International Journal of Computer Vision*, vol. 8, no. 3, pp. 203–216, 1992.

[5] T. Delbrück, "Silicon retina with correlation-based, velocity-tuned pixels," *IEEE Trans. on Neural Networks*, vol. 4, no. 3, pp. 529–541, May 1993.

[6] A. G. Andreou, R. C. Meitzler, K. Strohbehn, and K. A. Boahen, "Analog VLSI neuromorphic image acquisition and preprocessing systems," *Neural Networks*, vol. 8, no. 7/8, pp. 1323–1347, 1995.

[7] A. Simoni, G. Torelli, F. Maloberti, A. Sartori, S. E. Plevridis, and A. N. Birbas, "A single-chip optical sensor with analog memory for motion detection," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 7, pp. 800–806, July 1995.

[8] J. Hutchinson, C. Koch, J. Luo, and C. Mead, "Computing motion using analog and binary resistive networks," *Computer*, vol. 21, pp. 52–63, Mar. 1988.

[9] P. Kinget and M. S. J. Steyaert, "A programmable analog cellular neural network CMOS chip for high speed image processing," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 3, pp. 235–243, Mar. 1995.

[10] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, Nov. 1984.

[11] P. Bouthémy and P. Lalande, "Recovery of moving object masks in an image sequence using local spatiotemporal contextual information," *Optical Engineering*, vol. 32, no. 6, pp. 1205–1212, June 1993.

[12] J. Besag, "On the statistical analysis of dirty pictures," *J. R. Statist. Soc. B*, vol. 48, no. 3, pp. 259–302, 1986.

[13] S. Espejo, A. Rodriguez-Vazquez, R. Dominguez-Castro, J. L. Huertas, and E. Sanchez-Sinencio, "Smart-pixel cellular neural networks in analog current-mode CMOS technology," *IEEE Journal of Solid-State Circuits*, vol. 29, no. 8, pp. 895–905, Aug. 1994.

[14] J. B. Hughes and K. W. Moulding, "Switched-current signal processing for video frequencies and beyond," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 3, pp. 314–322, Mar. 1993.

[15] M. Song, Y. Lee, and W. Kim, "A clock feedthrough reduction circuit for switched-current systems," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 2, pp. 133–137, Feb. 1993.

[16] G. Wegmann and E. A. Vittoz, "Analysis and improvements of accurate dynamic current mirrors," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 3, pp. 699–706, June 1990.

[17] A. Rodriguez-Vazquez, R. Dominguez-Castro, F. Medeiro, and M. Delgado-Restituto, "High resolution CMOS current comparators: Design and applications to current-mode function generation," *Analog Integrated Circuits and Signal Processing*, vol. 7, pp. 149–165, Mar. 1995.

Franck Luthon was born in Rambouillet, France, in 1963. He received the Electronics Eng. degree and the Dr. Eng. degree from the Grenoble Polytechnic National Institute, Grenoble, France, in 1985 and 1988, respectively.

He has been an Assistant Professor at Grenoble Polytechnic National Institute since October 1990, where he is currently with the Signal and Image Laboratory. His research interests are in the area of motion analysis and spatiotemporal segmentation in image sequences, with particular emphasis on real-time implementations.

Dr. Luthon has been an Associate Editor for the French signal processing Journal *Traitement du Signal* since 1995.

Daniela Dragomirescu (S'97) was born in Bucharest, Romania, in 1972. She received the Electronics Eng. degree from Bucharest Polytechnic University, Bucharest, Romania, in 1996. She is currently a Ph.D. student at the System Analysis and Architecture Laboratory, National Center for Scientific Research, Toulouse, France.

She received a TEMPUS fellowship from the European Community in 1996.