Robust Focal-Plane Analog Processing Hardware for Dynamic Texture Segmentation

Jorge Fernández-Berni, Ricardo Carmona-Galán
Institute of Microelectronics of Seville (IMSE-CNM)
Consejo Superior de Investigaciones Científicas y Universidad de Sevilla
C/ Américo Vespucio s/n, 41092, Seville, Spain
Email: berni@imse.cnm.es

***Abstract*—Cellular Nonlinear Networks (CNN) establish a theoretical framework in which programmable focal-plane image processing arrays can be developed. The conventional support for its analog programmability in VLSI is the implementation of transconductor-based multiplication of the input, output and state variables times the corresponding template elements. However, some distributions of weights can be greatly affected by the intrinsic nonidealities of the physical implementation. This is exactly the case when implementing linear diffusion within a transconductor-based CNN implementation. In this paper we propose an alternative implementation: a resistive grid based on MOSFETs operating in the triode region to realize linear diffusion of the input image, considered as the initial state of the network. In addition, these MOS-resistors can be employed as switches in order to sub-divide the image into bins, sized to track features on the appropriate scale. Thus, by simply controlling the size of the binning and for how long the pixel voltages will diffuse, it will be possible to segment and track dynamic textures along an image flow. Each frame of the flow is described by a smaller image in which each pixel represents the energy of the corresponding image bin, once the non-relevant spatial frequency components have been filtered out. We will demonstrate that the resulting low-resolution representation of the scene is very robust to the different sources of nonidealities in a standard CMOS technology.**

I. Introduction

Cellular Nonlinear Networks (CNN) [1] define a framework for analog parallel array processing based on elementary processing units interacting only with a finite neighborhood. The interest in CNNs arises mainly from two facts. Firstly, many tasks related to the processing of images can be defined in terms of operations among neighborg pixels and thereby directly mappable onto a CNN. And secondly, the local scope of the interactions makes them simpler for VLSI implementation compared to more general fully-connected neural networks. Thus it has leaded to numerous physical implementations of focal-plane image processing arrays based on the CNN framework. These arrays are meant to carry out low-level operations over images in a massively parallel and efficient way. Within such operations, the implementation of linear diffusion turns into essential. It is an ubiquitous tool in image processing and vision algorithms. Its application field ranges from obtaining representations of a scene at different scales [2] to simply reducing the effects caused by noise. Nonidealities intrinsically associated with the physical implementation of the interconnection weights can however render the implementation of linear diffusion in transconductor-based CNNs impractical [3]. An alternative is the use of a resistive grid. The VLSI implementation of true resistors is, however, bulky, specially if we are interested in performing time-controlled diffusion. Instead MOS transistors can be used as resistors, in spite of their nonlinearities. By carefully choosing the signal range and an adequate design of the geometry of the transistors, a wide range of resistive grids can be emulated with moderate accuracy [4]. Besides, by controlling the gate voltage, it is possible to determine the processing realized by the grid.

In this paper we present a reconfigurable focal-plane processor based on performing CNN processing through diffusion dynamics in image blocks of programmable size. The primitive block employed to implement the connections is a MOSFET operating in ohmic region. Though relatively simple, the type of filtering that we are realizing on each image frame turns into a very powerful aid in order to segment dynamic textures. A dynamic texture (DT) is a spatially-repetitive time-varying visual pattern whose temporal variation presents certain stationarity [5]. They are very common in natural scenes, e. g. smoke or a flock of birds. However, both spatial and temporal extents of a DT cannot be determined in advance. This is why segmentation at early image processing stages can help to real-time tracking of phenomena manifested in the form of visual DTs. Therefore, our circuit takes advantage of their spatial repeatability. Both in the size of the image bins in which the full frame is divided, and in the duration of the linear diffusion –that is related to the spatial cut-frequency of the spatial Gaussian low-pass filter implemented–, we can encode the distinctive frequency signature of the targeted texture. Notice that, in the real circuit, this is achieved by selecting the interconnecting MOS-resistors that will be turned on, and controlling the amount of time they will be effectivelly on. Once the filtering is done, the energy of each pixel is locally computed and the energy of the bin is obtained by charge redistribution with the help of another transistor grid. The result is a reduced representation of the scene in which each pixel contains the energy of the corresponding bin. It permits to focus only on those regions where this energy meets certain conditions. We will finally show simulations on the robustness of the VLSI implementation of the array against mismatch and temperature drift.

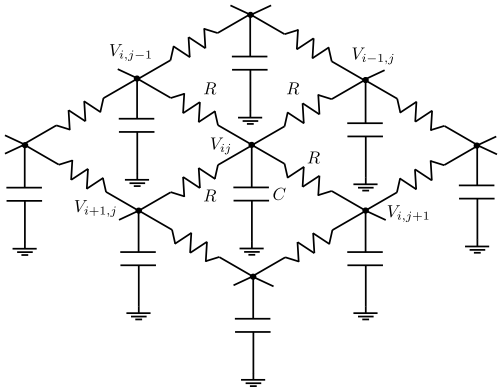

Fig. 1. Resistive grid performing linear diffusion

II. Scene representation

One of the main functionalities of a focal-plane analog array processor is to reduce the amount of information transmitted for subsequent –usually digital– processing stages. In other words, a simplified representation of the original raw scene must be delivered containing only the necessary data for the rest of the processing. In our case, the desired outcome is the segmentation of a certain dynamic texture. And the necessary data are on the frequency bands which define its signature. We therefore require spatial filtering analog hardware capable of extracting information about different bands of frequencies.

Consider the resistive grid depicted in Fig. 1. In order to determine the spatial filtering which performs, let the initial voltage at the capacitor of every node be the value of the corresponding pixel. If we permit the network to evolve from this initial state, the equation satisfied at each node inside the network is:

$$\tau \frac{dV_{ij}}{dt} = -4V_{ij} + V_{i+1,j} + V_{i-1,j} + V_{i,j+1} + V_{i,j-1} \qquad (1)$$

where $\tau = RC$. Applying the DFT to this equation we obtain:

$$\tau \frac{d\hat{V}_{uv}}{dt} = -4\hat{V}_{uv} + e^{\frac{2\pi iu}{M}}\hat{V}_{uv} + e^{\frac{-2\pi iu}{M}}\hat{V}_{uv} + \\ + e^{\frac{2\pi iv}{N}}\hat{V}_{uv} + e^{\frac{-2\pi iv}{N}}\hat{V}_{uv} \qquad (2)$$

where we have considered an array whose size is $M \times N$ pixels. Eq. (2) can be rewritten as:

$$\tau \frac{d\hat{V}_{uv}}{dt} = -4\left[\sin^2\left(\frac{\pi u}{M}\right) + \sin^2\left(\frac{\pi v}{N}\right)\right]\hat{V}_{uv} \qquad (3)$$

and solving now in the time domain we obtain:

$$\hat{H}_{uv}(t) = \frac{\hat{V}_{uv}(t)}{\hat{V}_{uv}(0)} = e^{-\frac{4t}{\tau}\left[\sin^2\left(\frac{\pi u}{M}\right)+\sin^2\left(\frac{\pi v}{N}\right)\right]} \qquad (4)$$

where $\hat{V}_{uv}(0)$ represents the DFT of the image defined by the initial voltages at the capacitors and $\hat{V}_{uv}(t)$ is the DFT of the image defined by those same node voltages after a certain time interval t since the network started to evolve at time instant $t_0 = 0$. Thus Eq. (4) describes the filtering process undergone by the initial image as the network evolves. It is an approximation of the lowpass isotropic filtering performed by an ideal continous-plane diffusion process [6]. For VLSI implementation, MOS transistors can substitute resistors one by one. It permits to obtain larger resistances with less area than resistors made with polysilicon or diffusion strips[1]. Besides, it is possible to adjust the lowpass filtering implemented by controlling the ratio t/τ through the gate voltage. It turns the transistors into switches with voltage-controlled resistance. A resolution between 6 and 7 bits with respect to an ideal resistive grid can be achieved within a wide signal range despite nonlinearities and mismatch in the transistors [4].

One of the most interesting aspects of the filtering performed by a resistive grid is that information about any band of spatial frequencies with approximately the same norm can be extracted by stopping the diffusion at two different time instants. Nevertheless, we need a procedure for summarizing the filtering process and thereby obtaining a simplified representation of the scene. And such a procedure is the focal-plane computation of the energy. Let $V_{ij}(t)$ be the voltages at the nodes of a $M \times N$ MOSFET-based resistive grid after a certain interval of diffusion t. The total energy is defined as:

$$E(t) = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} |V_{ij}(t)|^2 = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} |\hat{V}_{uv}(t)|^2 \qquad (5)$$

Eq. (5) along with Eq. (4) mean that the amount of energy that remains in the image accounts for the filtering undergone during the diffusion. The total charge in the whole capacitor array is conserved, but, naturally, the system evolves towards the less energetic configuration. Thus the energy at each time instant is a measure of the evolution of the diffusion process. The longer t the less $E(t)$. The energy lost between two consecutive points in time during the difussion corresponds to that of the spatial frequencies filtered. Notice that changing the reference level for the amplitude of the pixels does not have an effect outside of the DC component. A constant value added to every pixel does not eliminate nor modify any of the spatial frequency components already present, apart from that at the origin of the Fourier space.

III. Focal-plane hardware

A DT can appear into a scene at any location. We therefore need flexible focal-plane hardware capable of extracting the desired information in different zones of the scene. Moreover, as the scale of the DT can also vary, the size of the zones where the signature is searched must be programmable. That is to say, the hardware which we are looking for must carry out a programmable image binning. Within every bin, a

[1]For example, in a standard 0.35μm CMOS process, a PMOS transistor with dimensions 0.4/1 achieves an equivalent resistance of 85kΩ. To reach the same resistance with p-type diffusion, a strip of around 600 squares would be necessary.

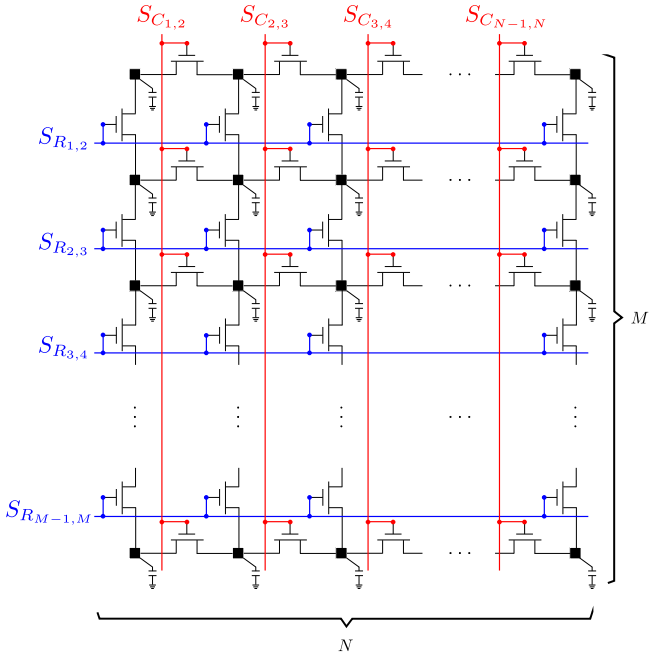

Fig. 2. Hardware structure for programmable image binning and filtering

controlled diffusion process and the subsequent computation of the energy will lead to the targeted simplified representation.

Consider the circuit depicted in Fig. 2. It consists of a $M \times N$ grid where the value of each pixel is stored in a capacitor. Each capacitor is 4-connected to the neighboring capacitors by means of MOS transistors. Both the capacitors and the transistors are nominally identical throughout the grid. The gate voltage of each transistor is controlled by the corresponding selection signal $S_{R_{i,i+1}}$ or $S_{C_{j,j+1}}$ which respectively connects the i/j-th row/column with the $i+1/j+1$-th row/column. When selected, the MOSFETs are biased in the ohmic region, behaving as resistors connecting two nodes. Otherwise the MOS transistors are off, establishing the boundary of a bin. Thus a particular distribution in the set of row and column selection signals determines the size and amount of bins in which the image plane is divided. Finally, once the image plane division is defined, a charge diffusion process is performed within the bin whenever the corresponding control signals remain selected. By deselecting them, the diffusion is stopped.

The hardware employed to calculate the energy of the bins is very simple. It consists of a charge-redistribution grid like that of Fig. 2 linked pixel to pixel with the grid performing the controlled diffusion. For this new grid, all the capacitors must be pre-charged to a reference voltage V_{REF}. The link between both grids is realized by means of the circuit depicted in Fig. 3, where C_P represents the sensing capacitor storing the pixel value and C_E the corresponding capacitor pre-charged to V_{REF}. Once the diffusion has been stopped, the switch S_E in all the pixels is switched on during a fixed period of time T_E. The final voltage at C_E is:

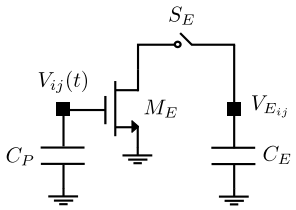

Fig. 3. In-pixel circuit for the computation of the image energy

$$V_{E_{ij}} = V_{REF} - \frac{T_E}{C_E}\beta[V_{ij}(t) - V_{th}]^2 \tag{6}$$

We are assuming that all the transistors M_E, operating in saturation, are nominally identical. Deviations occur from pixel to pixel due to mismatch in the threshold voltage (V_{th}), the transconductance parameter (β), and the body-effect constant (γ, not appearing explicitly in this equation). Being area dependent effects, transistors M_E are tailored to control the resulting error in the computation. Also, mobility degradation contributes to the deviation from the behaviour depicted in Eq. (6). The useful signal range will be limited by this. When S_E is switched back off in all the pixels after time T_E, a total charge redistribution takes place at the nodes $V_{E_{ij}}$ with the same binning scheme as for the controlled diffusion. It results in averaging the pixel energy at each bin:

$$\bar{V_{E_{kl}}} = V_{REF} - \frac{\beta T_E}{WHC_E} \sum_{i=kW}^{kW+W-1} \sum_{j=lH}^{lH+H-1} [V_{ij}(t) - V_{th}]^2 \tag{7}$$

where the indexes k and l identify the bin and $W \times H$ represents its size in pixels. This voltage is proportional to the total energy of the pixels of that bin t seconds after the diffusion started. As referred before, the offset introduced by V_{th} does not affect any spatial frequency other than the DC component. Finally, in order to achieve the reduced representation of the scene, only one pixel out of every bin needs to be read as all the capacitors within the bin will be at the same voltage $\bar{V_{E_{kl}}}$.

IV. Simulation Results

Consider the circuit depicted in Fig. 4. It corresponds to an elementary cell inside an array implementing the previously described hardware. Note that each MOS-resistor is shared by the corresponding neighbor cell. The used models of the transistors belong to a standard 0.35μm CMOS 3.3V process. Note that the sensing capacitance C_P in Fig. 3 is implemented by a MOS-based capacitor. This capacitor, along with the four nominally identical transistors which connect it to its neighbors, are designed to emulate a resistive grid with $\tau = 85ns$ for typical mean conditions. It is possible to obtain the same value of τ by reducing the dimensions of the NMOS transistor implementing C_P and increasing the length of the PMOS transistors emulating a certain resistance. It would mean less area. However, it also introduces too many charge errors coming from the PMOS transistors on

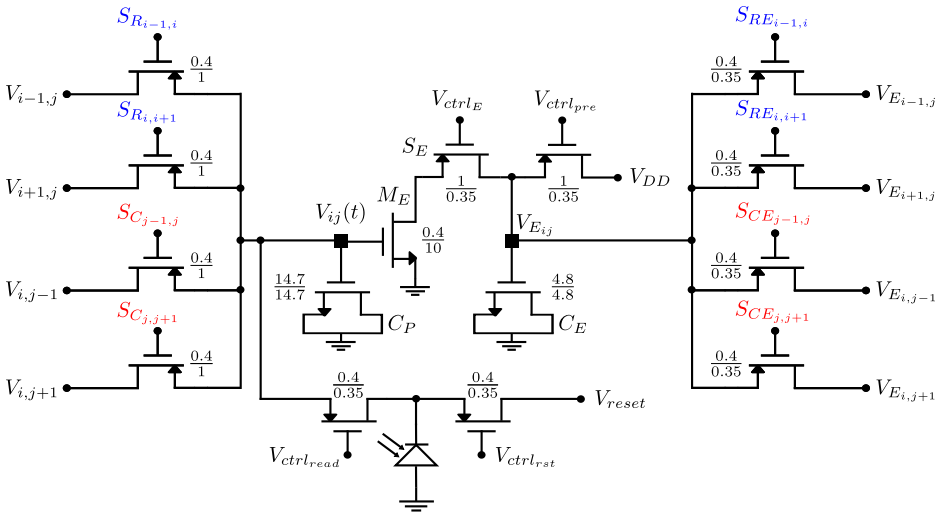

Fig. 4. Elementary cell of the simulated array

switching to control the diffusion. The signal range for this MOS-based resistive grid is $[1.5V, 2.5V]$. It keeps the RMSE with respect to the equivalent ideal resistive grid below 0.75% for a diffusion process applied to a grayscale Lena image [4]. Regarding the capacitance C_E in Fig. 3, it is also implemented by a MOS-based capacitor. Its design is determined by the dimensions of the transistor M_E, which must be long enough to reduce short-channel effects as much as possible, and by the period of time T_E. In our case, we choose $T_E = 20ns$, which is a feasible requirement for the control signal V_{ctrl_E}. In addition to the switch S_E, we have introduced the precharging switch connected to the reference voltage V_{DD} and controlled by the signal $V_{ctrl_{pre}}$. In this way, by sweeping the signal range of V_{ij}, that is, all the possible values of the pixels, and computing for every case the energy at $V_{E_{ij}}$, the parameters in Eq. (6) can be estimated. Specifically, by applying a least square fitting for typical mean conditions to $V_{DD} - V_{E_{ij}}$, with $V_{DD} = 3.3V$, a RMSE less than 0.8% can be obtained for the energy computation of the pixel values. The signal range for $V_{E_{ij}}$ is $[2.16V, 3.02V]$. Finally, this node is connected to its neighbors through PMOS transistors. It implements the charge redistribution grid for energy computation. Control signals $S_{RE_{i,i+1}}$ and $S_{CE_{j,j+1}}$ establish the binning scheme in this grid, like their counterparts $S_{R_{i,i+1}}$ and $S_{C_{j,j+1}}$ in the grid for controlled diffusion. The dimensions of the PMOS transistors are the minimum possible.

At this point, it can be demonstrated that a focal-plane array composed of elementary cells like that of Fig. 4 is very robust to parameter variations, mismatch and temperature dependences associated to the transistors in a standard CMOS process. Consider that the image plane is divided into bins whose size is 8×8 px. Let us also assume that the initial values of the bin pixels are those of the first pattern depicted in Fig. 5, which does not contain any distinctive frequency component. This initial state will be referred to as B_0. Our objective is the segmentation at this scale of two different textures, namely DT_1 and DT_2, whose characteristic patterns are depicted in Fig. 5. DT_1 corresponds to a signature with significant content at wave number indexes $(u, v) = (4, 0)$ and $(u, v) = (0, 4)$. DT_2 corresponds to a signature with significant content at $(u, v) = (1, 0)$ and $(u, v) = (0, 1)$. In order to perform the segmentation of these patterns we are going to compute three values of energy for every bin of each image: the total energy E_T, that is, the initial energy without filtering, the energy E_{t_1} after a certain diffusion duration $t = t_1$, and the energy of the DC component E_{DC}. The time instant t_1 must be set in such a way that the presence of the patterns can be distinguished. To this end, let us define t_1 as the time instant at which the energy associated to the wave number with index $(u, v) = (2, 0)$, or equivalently $(u, v) = (0, 2)$, has been filtered a 50%. According to Eq. (4), this condition means:

$$|\hat{H}_{2,0}(t_1)|^2 = e^{\frac{-8t_1}{\tau}\sin^2(\frac{2\pi}{8})} = 0.5 \tag{8}$$

which is translated into $t_1 = 0.17\tau$. With this filtering, the energy of the components at $(u, v) = (4, 0)$ and $(u, v) = (0, 4)$ has been reduced a 75% approximately while the energy of the components at $(u, v) = (1, 0)$ and $(u, v) = (0, 1)$ has been reduced only a 20% approximately. It leads to the following definitions:

$$P_H = 100\frac{E_T - E_{t_1}}{E_T - E_{DC}}\% \tag{9}$$

which is an estimation of the percentage of frequency components higher than $(u, v) = (2, 0)$ with respect to the energy of all the components other than the DC component and:

$$P_L = 100\frac{E_{t_1} - E_{DC}}{E_T - E_{DC}}\% \tag{10}$$

which defines an estimation of the percentage of frequency components lower than $(u, v) = (2, 0)$ with respect to, again, the energy of all the components other than the DC component.

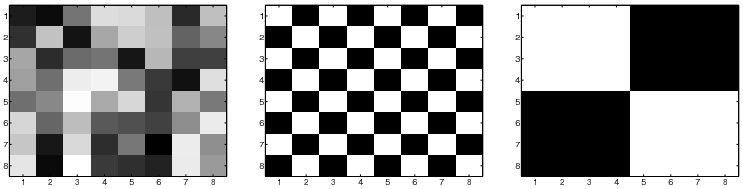

Fig. 5. Patterns considered in a bin whose size is 8×8 px.

The main contributor to the total energy of a set of pixels is normally the DC component by far. That is the reason why the previous definitions are relative to the total energy without this component. It permits to obtain meaningful values of P_H and P_L. In this way, we extract from simulation E_T, E_{t_1} and E_{DC} for the patterns B_0, DT_1 and DT_2. Keep in mind that any of the nodes $V_{E_{ij}}$ belonging to the corresponding bin could be read out once computed some of these values of energy, as previously mentioned. Subsequently, P_H and P_L are calculated off-line by a digital processor. Notice that the size of the resulting images from the processing is much more manageable. Finally, it is important to remark that energy will be expressed in the results as $V_{REF} - V_{E_{ij}}$ according to Eq. (7), with $V_{REF} = V_{DD} = 3.3V$. In Fig. 6, the evolution of the voltages $V_{E_{ij}}$ (upper part of the plot) and V_{ij} (lower part of the plot) for the different computations performed over the characteristic pattern of DT_1 is depicted. This evolution shows how the capacitor C_E at every cell is precharged before computing the initial energy of the pixels. Then a diffusion step is carried out in order to subsequently compute E_{t_1}. Finally, after a complete diffusion, E_{DC} is computed. Simulations for all the corners of the CMOS process have been realized. The results are summarized in Table I. Notice that, according to the previously mentioned errors associated to the controlled diffusion and energy computation, an A/D converter with an equivalent resolution between 6 and 7 bits would suffice for V_{ij} and $V_{E_{ij}}$. For such a converter, the values of energy for B_0 in all the corners would be barely distinguishable. It means that this pattern does not contain any significant frequency component for the bands considered. In this way, the values of P_H and P_L for B_0, although included for completeness, are meaningless. However, when it comes to the characteristic pattern of DT_1 and DT_2, the values of energy can be perfectly distinguished from the described converter. Specifically, the differences between E_T and E_{t_1} for DT_1 and between E_{t_1} and E_{DC} for DT_2 clearly highlight the presence of the respective pattern within the bin. In short, it is possible to detect the presence of the patterns considered within any bin by simply monitoring the corresponding values of E_T, E_{t_1} and E_{DC}. Finally, to emphasize the robustness of the processing, 30 Monte-Carlo simulations at different temperatures have been realized. Typical mean conditions were used, introducing independent Gaussian deviations of W, L, μ_0 and t_{ox} with $\sigma = 10\%$. These deviations generate in turn deviations in the crucial parameters V_{th}, γ and β [7] which directly affect the previously described equations defining the processing. The results are summarized in Table II. It can be observed that the mean values of the different parameters keep the same tendency than in Table I. Concerning the standard deviations, notice that for the case of the DT_1 and DT_2 patterns the deviations in P_H and P_L do not prevent from detecting their presence within the bin. However, for the B_0 pattern, the closeness of the energy values implies abrupt changes in P_H and P_L. Thus its standard deviation is very significant and they are again calculated for completeness. Analogous simulations intended to segment three different patterns with a size of bin of 32×32 pixels confirm all the described results. In fact, the accuracy of the processing is even greater. Note that the larger the size of the bin the less the influence that strong deviations in a cell have in the rest of them within the bin. On averaging the energy among more cells, the effect of wrong values in the result is attenuated. Finally, it is interesting to remark that the pixel values in the patterns DT1 and DT2 correspond to the maximum and minimum within the prescribed signal range. By reducing the relative amplitude of these values, the energy of the corresponding pattern is also reduced, making more difficult its segmentation. The sensitivity of the processing will thus depend not only on its resolution but also on the nature of the background from which pattern is to be segmented. For example, the segmentation against a background where only a DC component exists will be easier than against a background whose spectral content already includes certain energy at the spatial frequency of the targeted patterns.

V. Conclusions

A focal-plane analog parallel processing array has been presented. It is intended to segment dynamic textures in an image flow. The segmentation is performed by two simple MOS-based resistive grids linked node by node with a transistor. The processing carried out by the array, based on the diffusion operation, is massively parallel and power-efficient, suitable for vision systems with strict power budgets. Finally, the robustness of the VLSI implementation of the array has been demonstrated even for extreme temperatures and mismatch.

Acknowledgment

This work is funded by CICE/JA (2006-TIC-2352) and MICINN (TEC 2006-15722). The authors thank the Office of Naval Research for granting Mr. Fernández-Berni's registration and hotel expenses during the conference.

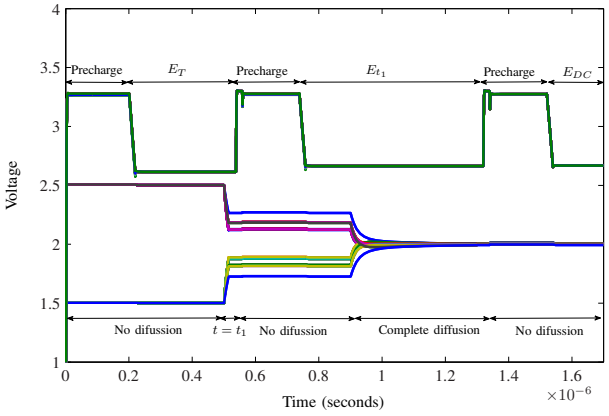

Fig. 6. Evolution of $V_{E_{ij}}$ (upper part) and V_{ij} (lower part) for the different computations performed over the characteristic pattern of DT_1

REFERENCES

[1] L. Chua and T. Roska, "The CNN paradigm," *IEEE TCAS-I*, vol. 40, no. 3, pp. 147–156, 1993.

[2] T. Lindeberg, "Feature detection with automatic scale selection," *Int. J. Computer Vision*, vol. 30, no. 2, pp. 79–116, 1998.

[3] J. Fernández-Berni and R. Carmona-Galán, "On the implementation of linear diffusion in transconductance-based cellular nonlinear networks," *Int. J. Circuit Theory and Apps.*, vol. 37, no. 4, pp. 543–567, 2009.

[4] ——, "Accurate design of a MOS-based resistive network for time-controlled diffusion filtering," in *ECCTD*, Antalya, Turkey, 2009, pp. 683–686.

[5] R. Nelson and R. Polana, "Qualitative recognition of motion using temporal texture," *CVGIP: Image Understanding*, vol. 56, no. 1, pp. 78–89, 1992.

[6] B. Jahne, H. Hauβecker, and P. Geiβler, *Handbook of Computer Vision and Applications*. Academic Press, 1999, vol. 2, ch. 4.

[7] P. Drennan and C. McAndrew, "Understanding MOSFET mismatch for analog design," *IEEE J. Solid-State Circ.*, vol. 38, no. 3, pp. 450–456, 2003.

Corner	*Bin*	E_T	E_{t_1}	E_{DC}	P_H	P_L
tm ($\tau = 85ns$)	B_0	0.6468V	0.6321V	0.6264V	72.06%	27.94%
	DT_1	0.6853V	0.6339V	0.6316V	**95.74%**	**4.26%**
	DT_2	0.6898V	0.67V	0.6333V	**35.11%**	**64.89%**
wo ($\tau = 107ns$)	B_0	0.8501V	0.838V	0.8209V	41.43%	58.57%
	DT_1	0.8949V	0.8351V	0.8296V	**91.62%**	**8.38%**
	DT_2	0.8947V	0.8716V	0.8406V	**42.63%**	**57.37%**
wz ($\tau = 60ns$)	B_0	0.4957V	0.4854V	0.4778V	57.54%	42.46%
	DT_1	0.5346V	0.4908V	0.4839V	**83.8%**	**16.2%**
	DT_2	0.5351V	0.5233V	0.4847V	**23.40%**	**76.6%**
ws ($\tau = 148ns$)	B_0	0.3754V	0.3672V	0.3603V	54.3%	45.7%
	DT_1	0.4084V	0.3703V	0.3645V	**86.69%**	**13.31%**
	DT_2	0.408V	0.3967V	0.3683V	**28.3%**	**71.7%**
wp ($\tau = 49ns$)	B_0	1.0608V	1.0438V	1.0266V	49.7%	50.3%
	DT_1	1.1V	1.0423V	1.0423V	**100%**	**0%**
	DT_2	1.1001V	1.0844V	1.0381V	**25.27%**	**74.73%**

TABLE I
RESULTS OF SIMULATION FOR THE DIFFERENT CORNERS OF THE CMOS PROCESS

Temperature ($^\circ C$)	*Bin*	$\bar{E}_T(\sigma)$	$\bar{E}_{t_1}(\sigma)$	$\bar{E}_{DC}(\sigma)$	$\bar{P}_H(\sigma)$	$\bar{P}_L(\sigma)$
0	B_0	0.7364V (0.0235V)	0.7231V (0.023V)	0.7142V (0.0225V)	60.99% (30.98%)	39.01% (30.98%)
	DT_1	0.7885V (0.0305V)	0.7319V (0.028V)	0.7237V (0.0272V)	**87.3% (9.06%)**	**12.7% (9.06%)**
	DT_2	0.7939V (0.0311V)	0.7767V (0.0284V)	0.7296V (0.0231V)	**26.77% (9.29%)**	**73.23% (9.29%)**
27	B_0	0.6502V (0.0204V)	0.6394V (0.0201V)	0.6316V (0.02V)	61.84% (17.91%)	38.16% (17.91%)
	DT_1	0.6951V (0.0277V)	0.6471V (0.0247V)	0.6405V (0.0237V)	**87.82% (5.49%)**	**12.18% (5.49%)**
	DT_2	0.6941V (0.0262V)	0.6804V (0.0251V)	0.6421V (0.02V)	**26.32% (9.61%)**	**73.68% (9.61%)**
85	B_0	0.5169V (0.0173V)	0.5097V (0.0167V)	0.504V (0.0165V)	61.33% (35.5%)	38.66% (35.5%)
	DT_1	0.5522V (0.0225V)	0.5166V (0.0201V)	0.5110V (0.0196V)	**86.47% (6.76%)**	**13.53% (6.76%)**
	DT_2	0.5517V (0.0204V)	0.5423V (0.0199V)	0.5129V (0.0169V)	**24.12% (7.53%)**	**75.88% (7.53%)**

TABLE II
RESULTS OF 30 MONTE-CARLO SIMULATIONS FOR TYPICAL MEAN CONDITIONS AT DIFFERENT TEMPERATURES