Switched-Capacitor Networks for Scale-Space Generation

M. Suárez, V.M. Brea, D. Cabello
Centro de Investigación en Tecnologías de la Información (CITIUS)
University of Santiago de Compostela
Santiago de Compostela, Spain
Email:manuel.suarez.cambre@usc.es

F. Pozas-Flores, R. Carmona-Galán
A. Rodríguez-Vázquez
Instituto de Microelectrónica de Sevilla (IMSE-CNM)
CSIC-Universidad de Sevilla, Spain

***Abstract*—This work examines the parallel and the bilinear implementations as two different switched-capacitor network topologies for scale-space filtering. The paper assesses the validity of both topologies as scale-space generators in focal-plane processing through object detection with a known scale- and rotation-invariant feature extractor as SIFT.**

***Keywords*—switched-capacitor networks, scale-space theory, Gaussian Filtering, diffusion, SIFT.**

I. INTRODUCTION

In scale-space theory signals are represented at several scales, each conveying different details of the original signal. Every new scale (σ_i) is the result of a smoothing operator on the former scale, either the original signal itself (scale σ_0) or a former smoothed version (scale σ_{i-1}) [1]. Consistent scale-spaces are generated when the structure or details of an image do not increase with the scale parameter σ, i.e., the number of local extrema cannot increase with σ.

Babaud et al. show the uniqueness of the Gaussian function as the only valid smoothing operator in 1D signals [2]. In 2D signals, there is not any operator that guarantees that the number of local extrema does not increase with σ. Nevertheless, it is also shown that in continuous 2D signals, the Gaussian kernel is the operator that minimizes the appearance of new details with increasing smoothing [3]. In discrete 2D signals, e.g. images, the solution to the discretized version of the diffusion equation is a correct smoothing operator for scale-space generation [4].

RC networks implement naturally the discretized version of the diffusion equation; hence they provide valid scale-spaces. Reference [5] is a recent example of an RC network for focal-plane processing. Therein the different σ levels were defined by sampling the transient evolution of the network. Switched-capacitor networks are another alternative for an RC grid [6]. In this case, the network evolves across a series of stationary states obtained by charge redistribution among capacitors. This allows for an accurate control of the network evolution through the number of clock cycles. Besides, different network topologies and switching rate offer a great flexibility for focal plane processing.

This work addresses two different switched-capacitor network topologies for scale-space filtering. The paper examines their drawbacks and benefits, and it also assesses their performance in focal plane-processing as scale-space generators with object detection with the SIFT algorithm [7].

II. SWITCHED-CAPACITOR IMPLEMENTATIONS

We propose two different switched-capacitor topologies for an RC network that implements the discretized version of the diffusion equation, that is, scale-space filtering. Fig. 1 displays such two solutions along with a conventional RC approach for two nodes of a 1D network. In both cases the resistor joining the two capacitors is replaced with either, one or two capacitors and their corresponding switches, yielding the so-called parallel (Forward-Euler) implementation, named as 1C (Fig. 1.b) throughout this paper, and a variant of the bilinear approach, named hereafter as 2C (Fig. 1.c) switched capacitor networks [8]. Switches are controlled by two non-overlapping signals, ϕ_1 and ϕ_2. A cycle n is complete after both signals are set to HI. The voltage at any cycle is the result of charge redistribution between two capacitors. Thus, at any cycle any voltage of interest is a settled voltage. This is true as long as ϕ_1 or ϕ_2 are long enough, or equivalently, when the resistance of the switch and the exchange capacitor C_E are small enough. Next two subsections go through the two switched-capacitor network topologies studied in this paper for the particular case of a 1D network with two nodes. This analysis gives valuable information for a subsequent extension to 2D.

A. 1C Network Analysis in 1D

In a charge redistribution system with two capacitors the charge conservation principle is stated by Eq.(1).

$$\Delta Q_1(t) = -\Delta Q_2(t) \Rightarrow \Delta V_1(t) = -\frac{C_2}{C_1}\Delta V_2(t) \qquad (1)$$

This means that the charge amount lost by one of the capacitors is gained by the other one. When both capacitors have the same value, the final voltage at both capacitors settles down to the average value of the initial voltages.

The extra capacitor C_E and switches ϕ_1 and ϕ_2 for the 1C network shown in Fig. 1.b isolate capacitors C_1 and C_2 from one another, allowing for a succession of settled voltages at C_1 (C_2) provided by the charge redistribution between C_1 (C_2)

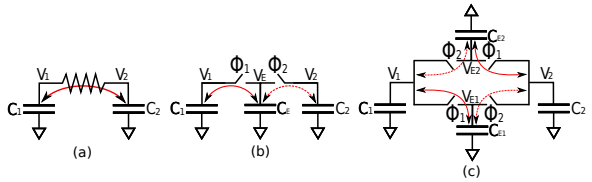

Fig. 1. Different circuit realizations for an RC network (two nodes shown).

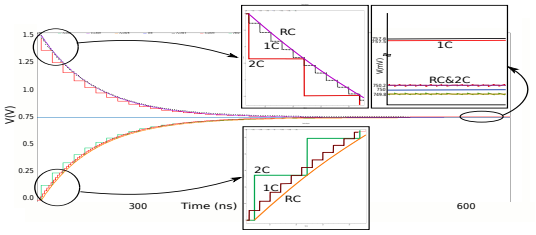

Fig. 2. Evolution of two nodes for the differnt RC networks of Fig (1).

and C_E. The voltage evolution at C_1 and C_2 is an exponential decay (rising) towards an intermediate level between the two initial voltages at C_1 and C_2 with a time constant given by the network, which in the case of the 1C 1D network is expressed as Eq.(2).

$$T_{1C\_network} = \frac{1}{C_E f}\frac{C_1 C_2}{C_1 + C_2} \tag{2}$$

f is the frequency of the clock comprising the two non-overlapping signals ϕ_1 and ϕ_2.

In focal-plane processing an adequate diffusion or smoothing process among pixels with any switched-capacitor network is only possible if the initial voltage at the exchange capacitors, C_E in the case of Fig. 1.b, is set to either $V_{C_1}(t_0)$ or $V_{C_2}(t_0)$. Any other initial voltages at the exchange capacitors would lead to an erroneous result.

If we use the first ϕ_1 cycle to initialize C_E, the first complete charge redistribution between C_1 and C_2 would be ready during the second ϕ_1 cycle ($\phi_1{}^2$). At this time, the voltages at C_1 and C_2 are calculated with Eq. (3) and Eq. (4).

$$V_2(\phi_2^1) = V_2(t_0) + \frac{C_E}{C_2 + C_E}[V_1(t_0) - V_2(t_0)] \tag{3}$$

$$V_1(\phi_1^2) = V_1(t_0) - \frac{C_2}{C_1}\frac{C_E}{C_2 + C_E}[V_1(t_0) - V_2(t_0)] + \xi \tag{4}$$

It is apparent that Eq. (1) does not hold for capacitors C_1 and C_2. This is due to the charge stored in the exchange capacitor C_E, causing a signal-dependent error ξ given by Eq. (5). Although this can be minimized with $C_E << C_1$, C_2, it is a cumulative error throughout the diffusion cycles. This error should be added to those derived from an actual implementation on an IC. The nominal error given by Eq. (5) might make that a scale-space generated by the 1C switched-capacitor network lead to an erroneous processing in a feature extractor like SIFT in image processing.

$$\xi = \frac{C_2 C_E}{C_1(C_2 + C_E)(C_1 + C_E)}[V_1(t_0) - V_2(t_0)] \tag{5}$$

B. 2C Network Analysis in 1D

Similarly to 1C, the voltage evolution at C_1 and C_2 for a 2C switched-capacitor network follows an exponential decay (rising). With initial conditions $V_{E1}(t_0) = V_1(t_0)$ and $V_{E2}(t_0) = V_2(t_0)$, the voltages at the end of the first complete diffusion ($\phi_1{}^2$) are those listed in Eq. (6) and Eq. (7).

$$V_1(\phi_1^2) = V_1(t_0) - \frac{C_{E2}}{C_1 + C_{E2}}[V_1(t_0) - V_2(t_0)] \tag{6}$$

$$V_2(\phi_2^1) = V_2(t_0) + \frac{C_{E1}}{C_2 + C_{E1}}[V_1(t_0) - V_2(t_0)] \tag{7}$$

If C_1=C_2=C and $C_{E1} = C_{E2} = C_E$, Eq. (8) holds.

$$\Delta V_1 = -\frac{C_E}{C + C_E}[V_1(t_0) - V_2(t_0)] = -\Delta V_2 \tag{8}$$

The 2C network evolves with a time constant given by Eq. (9), (where $C_1 = C_2 = C$, $C_{E1} = C_{E2} = C_E$), which is half the value of that of the 1C network (Eq. (2)) under the same conditions.

$$T_{2C\_network} = \frac{1}{C_E f}\frac{C}{4} \tag{9}$$

The most important issue is that, differently from the 1C network, the 2C network provides a proper smoothing operator, suitable for scale-space generation. This is stated by Eq. (8) which meets the condition stated by Eq. (1).

C. Simulations

The differences between the 1C and the 2C approaches are illustrated with electrical simulations. Fig. 2 plots the voltages at C_1 and C_2 for a network of just two nodes. Switches and capacitors are implemented as ideal elements. Capacitor values have been adjusted to have the same network time constant in the 1C and the 2C solutions. We also show the response of a conventional RC network. As seen in the zoom-in at the upper right-hand side corner of Fig. 2, the settled voltages from the 1C approach differ from the expected value. The error level amounts to $\sim 7mV/1.5V$, between 7 and 8 bits, for $C_E/C = 0.1$. The designer should check the error level caused by the 1C network when providing scale-space in the corresponding algorithm and if this is still supported by the application.

D. 2D Extension

Here we analyze how the 2C switched-capacitor network performs in 2D, i.e. in focal-plane processing. Fig. 3 depicts the case of a 2D 4-connected network, where the hardware for a pixel is enclosed in a dashed-line. It is easy to show that the voltage at pixel ij after the first complete diffusion cycle is determined by Eq. (10).

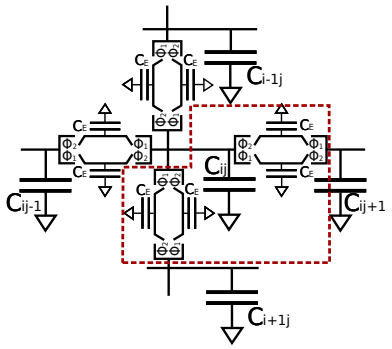

Fig. 3. 2C switched-capacitor implementation for a 2D 4-connected network.

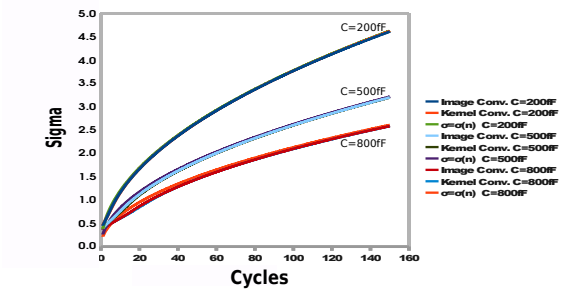

Fig. 4. $\sigma = \sigma(n)$ plots for different C/C_E ratios.

$$V_{ij}(t) = V_{ij}(t_0) + [V_{i-1j}(t_0) + V_{i+1j}(t_0) + V_{ij-1}(t_0) + V_{ij+1}(t_0) - 4V_{ij}(t_0)] \frac{\frac{C_E}{C}}{1 + 4\frac{C_E}{C}} \tag{10}$$

It is also easy to see that the result of a Gaussian kernel on an image takes the form of Eq. (11).

$$V_{ij}(t) = V_{ij}(t_0) + [V_{i-1j}(t_0) + V_{i+1j}(t_0) + V_{ij-1}(t_0) + V_{ij+1}(t_0) - 4V_{ij}(t_0)] \frac{e^{-\frac{1}{2\sigma^2}}}{1 + 4e^{-\frac{1}{2\sigma^2}}} \tag{11}$$

Eq. (12) is σ for the first cycle of diffusion, σ_0.

$$\sigma_0 = \left(2ln\frac{C}{C_E}\right)^{-1/2} \tag{12}$$

The relation σ-number of cycles, $\sigma = \sigma(n)$, can be inferred from the approximation of the continuous-time by the discrete-time case (Eq. (10)). Such an approach is obtained by dividing Eq. (10) by the clock period (T_{CLK}) and making it infinitely short, resulting in the network time constant given by Eq. (13).

$$\tau = \frac{C + 4C_E}{C_E} T_{CLK} \tag{13}$$

In [5] the relation $\sigma = \sigma(\tau)$ is given. In our case, this is formulated as Eq. (14), where the time has been expressed as a multiple of T_{CLK}, $t = n.T_{CLK}$, with n being an integer number.

$$\sigma = \sqrt{2\frac{t}{\tau}} = \sqrt{\frac{2nC_E}{4C_E + C}} \tag{14}$$

σ can also be found empirically by comparison with Gaussian kernels of well-defined σ levels. The algorithm searches for the σ level that produces the least RMSE. This approach can be done in two ways: 1) image comparison- comparing the images from a convolution of Gaussian kernels on a given image with the evolution of the 2C network across different clock cycles and 2) kernels comparison- comparing a Gaussian kernel of a well-defined σ-level with the recursive n-th convolution of the kernel of σ_0, (Eq. (12)), derived from the 2C network. Fig. 4 plots the curves for both methods along with the σ predicted by Eq. (14) for three different C/C_E ratios, with C being set to 200 fF, 500 fF and 800 fF, and C_E=20 fF. The first thing to note is that σ has a large range of variation. This is particularly true for low C values. It must be said here that it is very hard to know in advance the most suited σ levels for a given application, although $\sigma = \sigma(n)$ is needed irrespective of how it was achieved. It should also be said that Eq. (14) suits better the experimental methods for low C/C_E ratios.

III. ASSESSMENT: SIFT-BASED OBJECT DETECTION

This section assesses the validity of the switched capacitor networks described herein for object detection with the SIFT algorithm [7]. The left-hand side of Fig. 5 shows an example of a 240 x 320 image on which we run the SIFT algorithm. We derive the number of keypoints for a set of seven images from synthetic transformations, in this case, a combination of scale changes with rotations, (Fig. 5 displays an example of one of such rotations).

Fig. 6 displays the results of the SIFT implementation available at [9]. The left-hand side figure contains the number of keypoints in every image. The horizontal blueline is the number of keypoints in the original image. The green line shows the number of keypoints on every transformed image. The target is object detection, so we are interested in the number of matched points between images, shown on the right-hand side of the figure. We depict both the outcome of the software implementation available at [9] and the results from generating the scale-spaces with 1C and 2C switched-capacitor networks with 4-neighborhood for C=200 fF and $C_E = 20$ fF. In this example, σ was obtained empirically from the $\sigma = \sigma(n)$ graph of Fig. 4 with the method of comparing image convolutions described above. As expected, the SIFT outcome of the 2C solution is closer to the software-based implementation than that of the 1C network. Nevertheless, the 1C implementation still produces a high number of points, so that up to this point the 1C network should not be discarded as hardware solution, especially with high C/C_E ratios, although

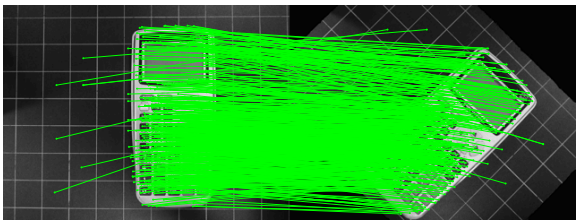

Fig. 5. An example of object detection with SIFT the algorithm.

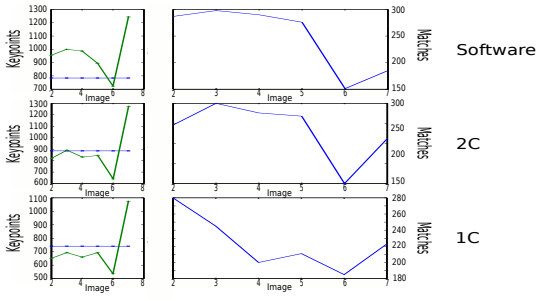

Fig. 6. Outcome of the SIFT algorithm for the image of Fig. 5 under several scale- and rotation-transformations for the software, and the 1C and 2C switched-capacitor networks.

a detailed study of errors from an actual IC approach is still to be made.

Finally, we also plot the $recall\ vs.\ 1-precision$ curves. The precision is defined as $p = \#tp/(\#tp + \#fp)$, and the recall as $r = \#tp/(\#tp + \#fn)$, with $\#tp$ being the number of true positives, $\#fp$ the number of false positives and $\#fn$ the number of false negatives. $\#tp + \#fp$ is the number of matches, shown as overlapping points in Fig. 5. The number of matches is calculated by comparing the descriptor vectors of two keypoints. If the difference of modules of such vectors is below (above) a certain threshold (th), the corresponding pair of keypoints is regarded as a match. A match becomes true positive when it also complies with the location condition; otherwise it is a false positive. The location condition can be checked easily in a synthetic (known) transformation as that of Fig. 5, as it is possible to track the keypoints from the original to the transformed image; thus it is also possible to calculate the number of false negatives.

Fig. 7 displays the $recall\ vs.\ 1-precision$ curves for Fig. 5, where the original image is rotated 45^o. The graph is achieved by changing th, the threshold for the comparison of two descriptor vectors, for the software solution and the 1C and 2C networks as scale-space generators. It is seen as the 2C network follows more closely the software solution. Again, C= 200 fF and C_E = 20 fF were used, and σ was obtained empirically from the graph of Fig. 4 with the method of comparing image convolutions described above.

IV. OUTLOOK AND CONCLUSIONS

This work has studied two different topologies of switched-capacitor networks as RC grids for scale-space generation in focal-plane processing. The difference between the two

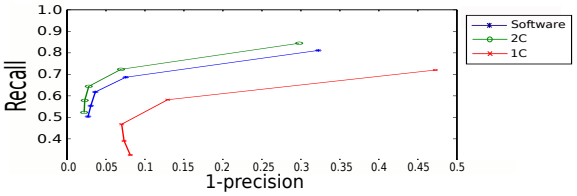

Fig. 7. Recall vs. 1-precision plot for the image of Fig. 5.

topologies lies in the resistor realization; in the so-called 1C approach the resistor joining two nodes is implemented with one switched-capacitor, in the 2C solution, such a resistor is made with two switched-capacitors in parallel. We have seen that in 1D the 2C network is an adequate smoothing operator. On the contrary, the 1C mode generates a signal-dependent error. This error can be minimized with high ratios between the state capacitor (C) and the capacitor joining two nodes (R), which would lead to trade-offs on an IC. We have also assessed the performance of both solutions in focal-plane processing through object detection with a scale- and rotation-invariant feature extractor, the SIFT algorithm, that uses scale-space filtering. We have seen that the 2C solution clearly outperforms the 1C approach, giving an outcome closer to a software-based implementation. All these studies have been conducted under ideal conditions, in the near future; an analysis of errors derived from an actual implementation will be made with a view to a future IC.

ACKNOWLEDGMENT

This work was supported by Xunta de Galicia and MICINN (Spain) under the contracts 10PXI206037PR and TEC2009-12686. The work of F. Pozas, R. Carmona and A. Rodríguez-Vázquez is partially supported by ONR (USA) through grant N000141110312, and MICINN (Spain) through project TEC2009-11812, that is co-funded by the European Regional Development Fund.

REFERENCES

[1] A.P. Witkin, "Scale space filtering", Proceedings of International Joint Conference on Artificial Intelligence, Karlsruhe, 1983.
[2] Jean Babaud et al., "Uniqueness of the Gaussian kernel for scale-space filtering", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 1, pp. 26-33, 1986.
[3] J.J. Koenderink and A.J. van Doorn, "The structure of images", Biology Cybernetics, vol. 50, pp. 363-370, 1984.
[4] Tony Lindeberg, "Scale-space for discrete signals", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 3, pp. 234-254, March 1990.
[5] J. Fernández-Berni, R. Carmona-Galán, and Luis Carranza-González, "FLIP-Q: A QCIF resolution focal-plane array for low-power image processing", IEEE Journal of Solid-State Circuits, Vol. 46, no. 3, pp. 669-680, March 2011.
[6] Christopher B. Umminger and Charles G. Sodini, "Switched capacitor networks for focal plane image processing systems", IEEE Transactions on Circuits and Systems for Video Technology, vol. 2, no. 4, pp. 392-400, December 1992.
[7] D. Lowe, "Distinctive image features from scale-invariant keypoints", IJCV 60 (2) (2004) 91-110.
[8] Mark N. Seidel, "Switched-capacitor networks for image processing: analysis, synthesis, response bounding, and implementation", MIT, PhD, 1994.
[9] http://www.vlfeat.org/∼vedaldi/code/sift.html.