RTI: 19990218 6.06b/W (May 26 2000) Ed: S. Gayathri Pgn: Prabha.G

Real-Time Imaging 7, 000–000 (2001)
doi:10.1006/rtim.1999.0218, available online at http://www.idealibrary.com on IDEAL®

A One-Dimensional Analog VLSI Implementation for Nonlinear Real-Time Signal Preprocessing

Reconstruction of given noisy data is an ill-posed problem and a computationally intensive task. Nonlinear regularization techniques are used to find a unique solution under certain constraints. In our contribution we present a parallel mixed-signal architecture which solves this nonlinear problem with in microseconds. By connecting all parallel cells in a circular manner it is possible to process noisy data vectors of infinite length. This is achieved by virtually shifting the nonlinear adaptive filter kernel over the noisy data vector. Additionally, we focus on the interaction between theory, discretization, numerical simulations, macro-modeling, and analog VLSI implementation for a theoretically well understood class of computer vision in an exemplary and paradigmatic way. A one-dimensional (1D) experimental chip has been fabricated using 0.8 μm CMOS technology. On-chip measurements are shown to agree with results from numerical simulations. Results from applying the 1D chip to nonlinear smoothing of two-dimensional image data will also be given correspondence.

K. Wiehler[1], J. Heers[2], C. Schnörr[3], H.S. Stiehl[2] and R.-R. Grigat[1]

[1] *Technical University of Hamburg-Harburg, Image Processing Group, Germany*
[2] *University of Hamburg, Cognitive Systems Group, Germany*
[3] *University of Mannheim, Institute for Image Processing, Pattern Recognition, and Computer Graphics, Germany*

Introduction

Overview

An important research topic in computer vision is the variational approach to nonlinear preprocessing (adaptive smoothing) of images. The goal is to overcome current techniques which are based on weak heuristics and hardly manageable process parameters. Every progress in this research has immediate positive consequences for solving real-life applications of computer vision.

Adaptive smoothing of images means computation of smooth regions and preservation of intensity transitions using raw image data. Since local computer vision algorithms for segmentation of regions and detection of transitions behave complementarily, the necessity of integration of both processes into one approach is obvious. A recent formulation of this problem is an optimization scheme in the context of variational calculus, a view which subsumes conventional approaches in computer vision (e.g., "split-and-merge").

Corresponding author: Klaus Wiehler, TU Hamburg-Harburg, 4-08/1, Harburger Schloβstraβe 20, D-21071 Hamburg, Germany. Tel: +49 (0)40 42878-3536/3125; Fax: ext. 2911; E-mail: k.wiehler@tuhh.de

1077-2014/01/000000 + 00 $35.00/0

Since all approaches from this class result in iterative solutions of sparse nonlinear equations, analog VLSI implementations allow for real-time application, i.e. at (video) frame rates.

The most powerful, in the sense of general, variational approach was suggested by Mumford and Shah [1]. The theory of this model (e.g. properties of solutions) are subject of an intensive mathematical research and, with focus on efficient algorithms, a lot of work has to be done. A drawback of this continuously formulated approach and its discretization is that there is no continuous dependence of the results and the input data. Furthermore, as yet, an analog VLSI implementation seems to be impossible to achieve.

A class of variational approaches which are less powerful but have globally favorable properties was proposed by Schnörr [2] (see the *Problems statement* section). As a special case this class subsumes a continuously formulated approach which resembles an approach by Harris *et al.* [3], suggested for an analog implementation, but avoids some of its drawbacks (e.g. dependence on coordinates, non-controllable smoothing behavior). Furthermore, our underlying mathematical model allows for the design of approaches given arbitrary data types (color, motion, etc.).

Analog VLSI technology experiences a remarkable renaissance in the context of hardware implementation of CNNs, opto-electrical systems (vision chips), and micro systems. Advantages are, for example, time- and amplitude-continuous processing or the possibility of massive parallelism with a current-dynamic range of seven decades. Main problems of an analog implementation are the device imperfections (mismatch) of MOS components, the storage of analog signals, the stability of feedback circuits, and the CAD support for the design of circuits with higher complexity. The mismatch of MOS components can be handled, as far as possible, with symmetrical component layout and furthermore with an individual adjustment of single circuit units. Storage of analog signals can be achieved for short periods with on-chip capacities, for medium periods with amplitude-quantization along with refreshing, and for long periods with isolated charge-packets in floating gates. Up to now, the automation of the design of complex analog circuits is not entirely possible. For the hierarchical design macro-models are created, nevertheless the designer has to know details about the model structure and physical parasitic effects. At this point, the analog design differs from digital design and resides for the time being in the domain of electronic engineers. For the implementation of resistive nonlinear networks one has to carefully consider the monotone behavior of nonlinear characteristics, which is mandatory for stability since otherwise global oscillation may occur. This condition of monotonicity should also not be violated in a parasitic or dynamic manner by mismatch of circuit units and represents consequently a non-trivial boundary condition for the circuit designer.

Related work

Low-level feature extraction and image smoothing are key issues in image processing and computer vision. Variational approaches [1, 4–6] provide a mathematically concise problem formulation (cf. the survey [7]) being superior to *ad-hoc* smoothing schemes. A common problem with these approaches, however, is their computational complexity from an optimization point-of-view. Stochastic optimization [4] is not feasible for typical image sizes, and deterministic annealing procedures [5, 6] cannot guarantee to obtain a "good" local minimum. Therefore, the use of *non-quadratic* but *convex* functionals has been advocated to simplify image smoothing from a computational viewpoint [2,8,9]. Furthermore, although being much simpler, convex functionals nevertheless provide reasonable approximations (cf. [10]) to the prototypical but mathematically and computationally sophisticated variational smoothing approach of Mumford and Shah [1]. Despite of efficient digital implementations of numerical schemes for solving variational equations [11], the computational effort prohibits applications for which frame-rate performance is required. As an alternative, analog hardware concepts allow to map classes of early vision problems onto time-continuous high-speed analog circuitry by the direct use of voltage, current and charge relationships of the physical devices [12].

The behavioral characteristic of electronic circuits as well as the solution of variational equations can be described by nonlinear differential equations [2]. An analog circuit can solve the equations in fractions of a second. The speed is limited only by the unavoidable parasitic capacitors and the finite power dissipation [13]. This is the major motivation for the design of analog VLSI networks suitable for solving such early vision problems.

Inspired by biology, Koch and Poggio [14,15] investigated the relationship between mathematically stated regularization schemes and analog networks.

Kirchhoff's current and voltage laws, which represent conservation and continuity restrictions satisfied by each network component, solves for the regularization. They derived a direct relationship between the generalized energy of a nonlinear resistor (*co-content*) and the theoretical regularization expression. From this viewpoint, the properties of the electronic devices can be analyzed regarding their effects on the solution. Mathematically well defined characteristics like convexity, consistency, and uniqueness can be controlled directly by the features of the used electronic devices. Therefore, it is possible to describe prerequisites on the circuit level which were originally defined on higher mathematical levels.

For the realization of those networks, substantial work has been carried out by Mead *et al.* [12, 16, 17] by utilizing sub-threshold standard MOS technology for massively-parallel analog signal processing. Good examples are optical motion sensors, silicon retinas, etc. By including optical sensors (CMOS-compatible photo diodes and transistors) vision systems can be implemented on one single chip. For current successful implementations see, for example, [18].

Build upon Mead's fundamental research, Harris [3, 19] implemented nonlinear resistive networks for early vision tasks. Utilizing resistive fuse elements he implemented the non-convex approach of Blake and Zissermann [5] (weak membrane model). He achieved a robust solution by a graduated convexity mechanism which is inherent to his resistive fuse circuit. The so-called *tiny-tanh* network is a further development of Sivilotti's [20] nonlinear saturating resistor. By introducing a high-gain positive feedback, the network resembles a discrete TV-norm realization [21]. Nevertheless, the implementation lacks both an efficient input-output mechanism and the possibility for controlling the regularization parameters.

Most recent implementations of resistive fuse algorithms utilizing floating-gate MOS-transistors for both, controlling the transfer characteristic and long-term error compensation, can be found in [22].

Also, in the last decade a lot of research in the field of massively-parallel hardware has been triggered by the Cellular Neural Network (CNN) theory introduced by Chua and Yang [23, 24]. A CNN is defined on a time-continuous, discrete grid and consists of an uniform array of analog nonlinear dynamic computing cells. They are interconnected in a local neighborhood and contain no learning scheme or adaptation mechanism and therefore are best suited for direct analog implementation. Discretized formulations of the regularization equation proposed in this paper (see Theory section) can be restated in the CNN framework [25] as well. The model has to be altered to the hardware-friendly *full-range* model [26] using nonlinear feedback templates. Nevertheless the CNN framework is defined on a discrete grid and therefore limited in its applications.

Various CMOS VLSI implementations of CNNs have been reported [27]. They can be discriminated by fixed [28] or variable [29, 30] templates, current [26] or voltage mode [31], full [32] or limited [33] CNN-model, on-chip [34] or off-chip sensors. They are referenced here in detail as a starting point for further investigations in the field of analog VLSI systems for signal processing.

Contribution and organization

In our paper, we present a massively-parallel VLSI hardware implementation of a nonlinear smoothing approach (one-dimensional (1D) case, 32 nodes) for applications in computer vision as well as a special architecture for processing 1D data streams. By analog VLSI technology, limits in performance of iterative solutions on digital computers have been overcome. In particular, the interaction between theory, discretization, numerical simulations, macro-modeling, and analog VLSI implementation for a theoretically well understood class of computer vision methods have been investigated in an exemplary and paradigmatic way (see also Figure 1). Characteristic of our work is the thorough validation on all levels of description and implementation.

In the second section we propose a convex variational approach for adaptive smoothing and sketch a 1D FEM discretization. We focus on the nonlinear characteristic transfer function whose analog implementation proved to be the hardest task for an analog circuit design. The feedback of circuit-mismatch for the case of the characteristic transfer function on the mathematical model is described in the third section (*Transfer function constraints*). Detailed investigation results in constraints for analog circuit design. Then (in *Analog cells*) we follow on with a brief review of technology imperfections in the context of analog VLSI circuits. Based on this, we present a constraint-driven top-down design for the core analog cell used for the 1D implementation. In

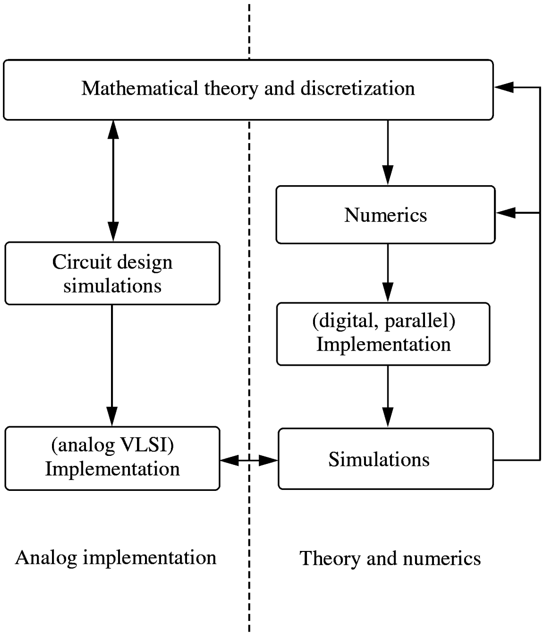
Mathematical theory and discretization
Numerics
Circuit design simulations
(digital, parallel) Implementation
(analog VLSI) Implementation
Simulations
Analog implementation
Theory and numerics

Figure 1. Development scheme of an analog VLSI-chip for adaptive smoothing.

the fourth section a systolic architecture tailored to time-continuous analog signal processing is described in detail. The results achieved with an 0.8 μm CMOS implementation of the so-called dynamic circular network (dCN) are shown in the fifth section. We conclude our paper in the final section.

Theory

Problem statement

A convex variational approach for adaptive smoothing was proposed by Schnörr [2] which results in the unique minimizer u of the *strict* convex functional

$$J(v) = \frac{1}{2}\int_{\Omega}\left\{(v-g)^2 + \lambda(|\nabla v|)\right\}dx, \qquad (1)$$

for a given data g and the non-quadratic function

$$\lambda(t) = \begin{cases} \lambda_h^2 t^2, & 0 \le t \le c_\rho. \\ \lambda_l^2 t^2 + (\lambda_h^2 - \lambda_l^2)c_\rho(2t - c_\rho), & 0 < c_\rho \le t. \end{cases} \qquad (2)$$

The approach comprises essentially two parameters: λ_h determines the degree of smoothing, whereas c_ρ controls the adaption to signal variations. For the latter, the smoothing process becomes anisotropic at loci with high gradients and gradually stops at such loci along the direction of the image gradient. Hence, essential signal structures are preserved. Finally, λ_l is a small positive constant ($\lambda_l = 0.1$, for example), the meaning of which is not relevant for the investigation presented here.

The global minimizer of Eqn (1) is the unique solution to the following variational equation [2]:

$$L(u,v) = \int_{\Omega}(u-g)v + \rho(|\nabla u|)\nabla u \nabla v \; dx = 0, \;\; \forall v, \qquad (3)$$

where the function $\rho(t) := \lambda'(t)/(2t)$ (see Figure 2) characterizes the adaptive smoothing process (1).

Partial integration shows that, at least formally, u may be considered as the steady-state solution of the nonlinear diffusion equation [2, 35]:

$$\frac{du}{dt} = \nabla \cdot (\rho(|\nabla u|)\nabla u) - (u-g), \qquad (4)$$

with Neumann boundary conditions.

Discretization

For the discretization of variational problems, FEM is the first choice. FEM can be applied in a mechanistic way, boundary conditions are incorporated automatically, and the discrete approach obtained is consistent in the sense that, under certain conditions [2], discrete solutions u_h to (3), for example, converge to the continuous solution u for vanishing mesh width. In this way discrete versions of our approach allow to maintain favorable properties of the underlying continuous problem formulation, like the rotational invariance of

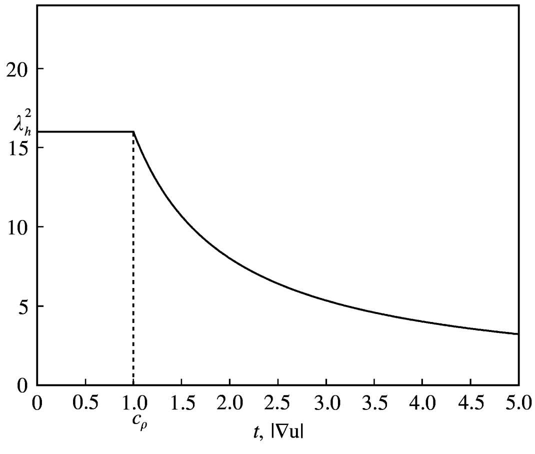

Figure 2. The characteristic function $\rho(t)$.

the smoothness term, for example. For a sound introduction to FEM we refer to, e.g. [36].

The basic idea of the Finite Element Method is the restriction of optimization problems to finite-dimensional subspaces. Let $\{\phi,\ldots,\phi_n\}$ be basis functions of a finite-dimensional subspace $\mathscr{H}_h \subset \mathscr{H}$. Then, the restriction of (3) to $\mathscr{H}_h$ reads:

$$L(u_h, \phi_i) = 0, \quad \forall i = 0, \ldots, n, \tag{5}$$

with minimizer $u_h \in \text{span}\{\phi_o,\ldots,\phi_n\}$. If we define the isomorphism

$$I : \mathbf{R}^n \rightarrow \mathscr{H}_h; \quad \mathbf{u} \rightarrow \sum_j u_j \phi_j, \tag{6}$$

and the mappings

$$\mathbf{L}_i(\mathbf{u}) := L(I(\mathbf{u}), \phi_i), \quad \mathbf{u} = (u_0, \ldots, u_n)^T, \tag{7}$$

then the solution of (5) is equivalent to the solution of the nonlinear system:

$$\mathbf{L}(\mathbf{u}) = 0, \quad \mathbf{L} = (\mathbf{L}_0, \ldots, \mathbf{L_n})^{\mathbf{T}} \tag{8}$$

As mentioned above, we know from FEM theory that the solutions u_h converges to the solution u of (3), if the formal discretization parameter h approaches zero [2, 36].

Now we apply the FEM to the case of 1D discrete signal data, with a constant sampling rate of 1. Thus, our domain reads $\Omega = [0,n]$. The first step is to interpolate the signal data in a linear manner. Next, we assign to each sample x_i a basis function ϕ_i which is uniquely defined by the following conditions:

$\phi_i(x)$ is linear within each interval $[x_i, x_{i+1}]$.

$\phi_i(x) = 1$ at sample x_i,

$\phi_i(x) = 0$ at every sample $x_j \neq x_i$

Eqn (7) now reads:

$$\mathbf{L}_i(\mathbf{u}) = \sum_k (u_k - g_k) \int_\Omega \phi_k \phi_i \, dx +$$

$$\sum_k u_k \int_\Omega \rho(|\nabla u_h|) \nabla \phi_k \cdot \nabla \phi_i \, dx. \tag{9}$$

The integral terms in (9) vanish for all $|i-k|>1$. The remaining integral terms can be computed analytically to obtain a sparse system of nonlinear equations in terms of the sample variables $\mathbf{u}$. Evaluating (9), we get for $\Omega = [0, n]$

$$\begin{aligned}
\mathbf{L}_0(u_0, u_1) =& \frac{1}{3}(2u_0 + u_1) - \frac{1}{3}(2g_0 + g_1) \\
& - \rho(|u_1 - u_0|)(u_1 - u_0) \\
\mathbf{L}_n(u_{n-1}, u_n) =& \frac{1}{3}(u_{n-1} + 2u_n) - \\
& \frac{1}{3}(g_{n-1} + 2g_n) + \rho(|u_n - u_{n-1}|)(u_n - u_{n-1}) \\
\mathbf{L}_i(u_{i-1} \cdot u_i, u_{i+1}) =& \frac{1}{6}(u_{i-1} + 4u_i + u_{i+1}) \\
& - \frac{1}{6}(g_{i-1} + 4g_i + g_{i+1}) + \rho(|u_i - u_{i-1}|) \\
& (u_i - u_{i-1}) - \rho(|u_{i+1} - u_i|)(u_{i+1} - u_i).
\end{aligned} \tag{10}$$

The discretization of (4) reads

$$\begin{aligned}
\frac{du_0}{dt} =& \frac{1}{3}(2g_0 + g_1) - \frac{1}{3}(2u_0 + u_1) - f_{nl}(u_1 - u_0) \\
\frac{du_n}{dt} =& \frac{1}{3}(g_{n-1} + 2g_n) - \frac{1}{3}(u_{n-1} + 2u_n) + f_{nl}(u_n - u_{n-1}) \\
\frac{du_i}{dt} =& \frac{1}{6}(g_{i-1} + 4g_i + g_{i+1}) - \frac{1}{6}(u_{i-1} + 4u_i + u_{i+1}) \\
& + f_{nl}(u_{i-1} - u_i) + f_{nl}(u_{i+1} - u_i),
\end{aligned} \tag{11}$$

with the transfer function $f_{nl}(t) = \rho(|t|)t$ (see Figure 3).

For solving (8) in case of 2 or 3 dimensions on digital computers, we have developed different efficient numerical schemes [11]. Although efficient implementation on parallel computers is possible, the computational effort

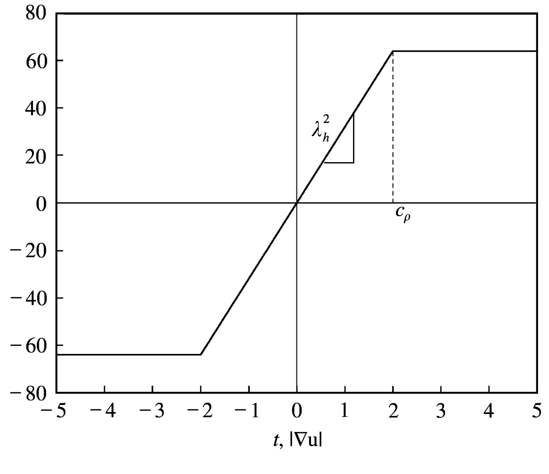

Figure 3. The 1D transfer function $f_{nl}(t)$.

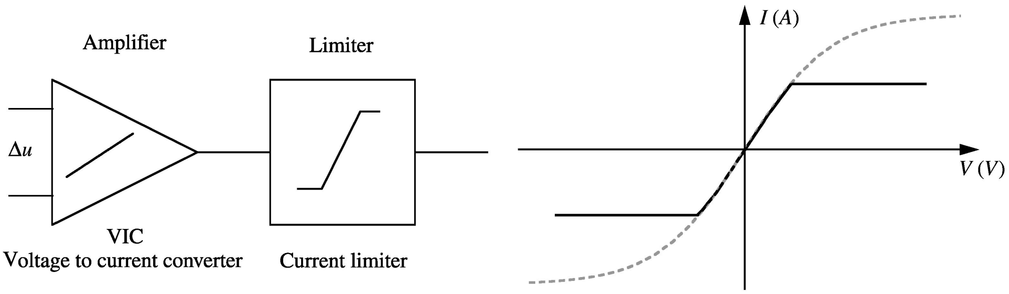

Figure 4. Model (left) and behavior (right) of the approximate transfer function $f_w(t)$.

prohibits applications for which (near) real-time performance is required. As an alternative, analog hardware concepts allow to map Eqn (8) onto time-continuous high-speed analog circuitry (see next section).

A Constraint-Driven Design

Transfer function constraints

In this section, we report on modeling the variational Eqn (11) with focus on the circuit implementation. Due to the limited accuracy of analog circuits (see *Analog cells* section) a proper analysis of deviations from the theoretical case is necessary. Since the nonlinear part of (1) dominates the image smoothing behavior, we emphasize on the modeling of the transfer function $f_{nl}(t)$.

The nonlinear transfer function $f_{nl}(t)$ is implemented by a simple amplifer-limiter circuit (see Figure 4) which is specified as follows:

$$f_w(t)\begin{cases} \lambda_h^2 t, & |t| < c_\rho \\ sign(t) c_\rho \lambda_h^2, & |t| \geq c_\rho. \end{cases} \tag{12}$$

The approximate transfer function (12) corresponds to (2) for the case of $\lambda_l = 0$. Due to the real circuit behavior, the transfer function is monotonously increasing. Our experiments have shown, that small variations of the shape of transfer function $f_{nl}(t)$ do not decisively influence the results of the image smoothing process.

However, the results are more sensitive to offset errors and inaccuracies of the smoothing parameters λ_h and c_ρ in the transfer function (see Figure 5). These errors, which are caused by imperfections of the underlying semiconductor manufacturing process, can be categorized further as follows (see also Table 1):

(1) the input offset error, which is mainly caused by threshold voltage deviations of the input differential pair;
(2) the output error, which is caused by current mirror mismatch in the output current limiter;
(3) errors in the saturation level (also caused by current mirror mismatch); and
(4) gain errors of the input stage.

Regarding the analog implementation we expect two types of errors; on the hand, global (or systematic) errors, which affect all nodes in the same way, and on the other hand, local (or stochastic) errors, which affect each node individually (for a review on error in analog circuits see next section).

A thorough investigation of all errors, and also of the interaction between them, has been carried out* which we summarize in Table 2. In the first row, limits for the global error δ are given. The second row lists the limits for the local error σ (normally distributed offsets and parameters with standard deviation σ). It can be seen that the influence of local error is stronger than the influence of the global error.

Analog cells

As described in the previous section, imperfections of the underlying micro electronic devices may cause a serious degradation of the results. Especially for analog cells, the exact performance cannot be predicted and

*Details of our investigations can be found in the Diploma thesis by Ralf Wö (1997) *Störungsanalyse eines eindimensinalen nichtlinearen Relaxationsnetzwerkes zur Signalverarbeitung*. University of Hamburg, Departments of Informatics.

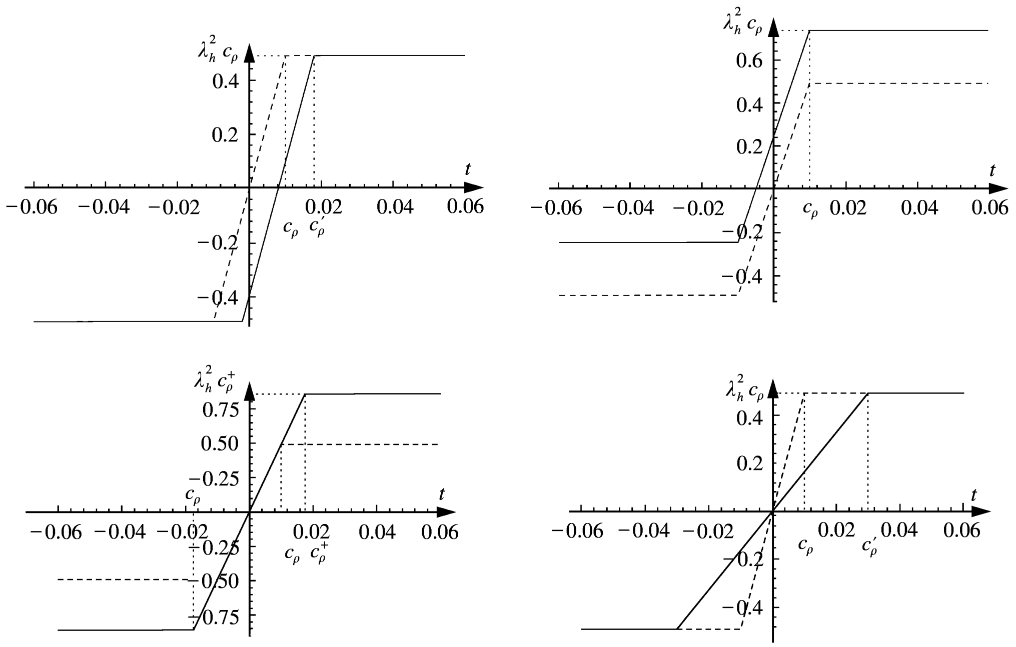

Figure 5. Transfer function deviations: X-offset (top left), Y-offset (top right), c_ρ-offset (bottom left) and λ_h^2-offset with $c_\rho \cdot \lambda_h^2 = \text{const}$ (bottom right).

thus, the design process must consider technology imperfections in an early stage.

In analog circuit design the resulting error can be categorized as shown in Figure 6. Systematic errors are caused by different operating points of the various transistors (e.g. a simple current mirror copies the current in an exact way only if the source drain voltage is equal for both). They can be reduced either by carefully choosing the circuit topology and/or transistor parameters or by adding auxiliary transistors which keep the substantial transistors in a constant bias condition (cascode transistors). These errors can be simulated very accurately with standard simulation tools. Therefore, their impact on the results can be predicted by appropriate back annotation into high level simulation tools (see preceding section). For the application of adaptive signal smoothing the most important systematic errors are the offset error and the saturation level of the nonlinear transfer characteristic (see Figure 5). The error should be as small as possible and independent of the data input and output.

Table 1. Types of errors for the amplification-limiter circuit

Circuit errors	Transfer functions deviations
Input offset error	X-offset
Output offset error	Y-offset
Output error of the limiter	c_ρ-deviation
Amplification error in the 1st stage	λ_h^2-deviation, with $\lambda_h^2 c_\rho = \text{const.}$

Random errors are caused by technology imperfections [37]. Due to this random influence even equal transistors in the schematic behave slightly different in their corresponding realization (mismatch). Moreover one can discriminate between intra- and inter-die variations. The first one takes only the transistors on one chip into account whereas in the latter case the variations between transistors on different chips are considered. For most applications of analog computational circuits, the performance is determined by the interaction of matched neighboring transistors (differential pairs, current mirrors, translinear loops, etc.). The accuracy is reduced by the mismatch of these devices. Therefore, the intra-die variation is the most important error source. The larger errors due to inter-die variations have to be small enough to guarantee sufficient bias condition on all chips (*yield*) for a proper performance. For the case of cascading multiple analog chips an appropriate compensation method has to be implemented [38].

Table 2. Necessary conditions for reasonable segmentation results. δ denotes the global error and σ denotes the local error, i.e. the standard deviation of normally distributed offsets and parameters

	X-offset	Y-offset	c_ρ-offset	λ_h^2-offset
Global error	$\delta \leq 0.25\ c_\rho$	uncritical	$\delta \leq 0.5\ c_\rho$	$\delta \leq 0.5\ c_\rho$
Local error	$\sigma \leq 0.2\ c_\rho$	$\sigma \leq 0.4\ c_\rho \lambda_h^2$	$\sigma \leq 0.25\ c_\rho$	$\sigma \leq 0.25\ \lambda_h^2$

To reduce the impact of the imperfections, different levels of the design hierarchy have to be taken into consideration: (i) algorithmic level; (ii) schematic level; and (iii) layout level. The more abstract the design level the higher the possibility for essential reductions.

First, regarding the algorithm level, the system has to be insensitive regarding the mismatch, e.g. this can be achieved by reducing the input output value range (increasing signal to noise ratio, e.g. CNN full range model [26]). In our case the algorithm is continuously dependent on the regularization parameters and therefore inherently insensitive to slight variations of the transfer function characteristic (see second section).

On the schematic level, the type of signal representation (voltage, current, pulses) has to be considered carefully. The choice depends on either the preferred input and output signals and the mandatory signal transformations. Furthermore, the operation mode, weak or strong inversion, has to be taken into account. Strong inversion circuits offer better accuracy and higher speed compared to circuits in weak inversion mode [39]. Nevertheless, the latter ones have a bipolar-like transfer characteristic which could be used to build efficient circuits for special applications (translinear circuits [40]).

On the layout level careful placement and routing techniques have to be used to reduce mismatch as far as possible [41].

By introducing the physical normalization values I_{unit}, V_{unit} and $g_{unit} = I_{unit}/V_{unit}$ for the current, voltage and transconductance respectively, the system's Eqn (11) can be mapped to a voltage/current relationship:

$$C\dot{v}_{u_i} = g_{unit}\frac{1}{6}\left[(v_{g_{i-1}} + 4v_{g_i} + v_{g_{i+1}} - (v_{u_{i-1}} + 4v_{u_i} + v_{u_{i+1}}\right] + f_{nl}^{ota}(v_{u_{i-1}} - v_{u_i}) + f_{nl}^{fota}(v_{u_{i+1}} - v_{u_i}). \quad (13)$$

Where $v_{g_i} = V_{unit}\, g_{ii}$ and $v_{u_i} = V_{unit}\, u_i$ are the corresponding voltage values of the input and state variables used in the implementation. f_{nl}^{ota} corresponds to f_{nl} in Eqn (11) taking a voltage as an input and a current as an output signal. C is the capacitor which determines the time constant τ of the system: $\tau \approx Cg_{unit}$.

The key element of the nonlinear relaxation scheme is the controllable piece-wise linear transfer function f_{nl}^{ota} in the above equation. The circuitry for this element should have minimum offset and saturation value errors. The smaller these variations the larger the range of the relaxation parameter can be made (previous section). Considering Eqn (12) again, an architecture based on operational transconductors has been chosen (Figure 7; the switches are used for the reconfiguration as described in the next section). Input and output signals are voltages while the summation as well as the nonlinear operator are realized with currents.

The circuit used for the nonlinearity is depicted in Figure 8. It is a standard OTA followed by a current limiter. The systematic errors are further reduced by introducing cascode transistors in the output stage. The random offset value of the circuit is dominated by the offset voltage error of the input differential stage. For low variations, the signal to bias ratio is an important

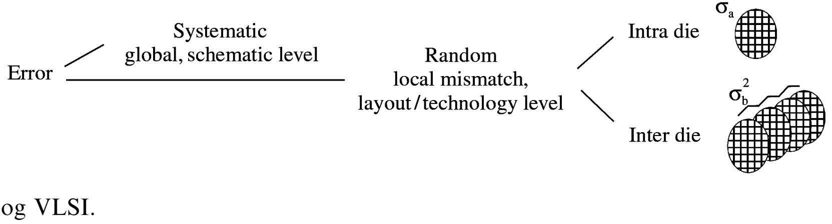

Figure 6. Errors in analog VLSI.

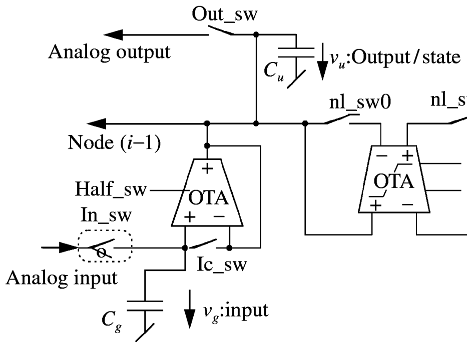

Figure 7. Core cell architecture, all switches, except, ic_sw, are closed for normal operation.

figure in circuit design [30]. Since the bias current of the limiter is in the range of the saturation current itself, the variation of the saturation current is as low as possible. Furthermore the layout was done after careful consideration of the matching topology in the circuit. All circuits operate in strong inversion. The current levels are in the range of micro amperes which results in relaxation time constants of microseconds for the used capacitors (≈ 1 pF).

Based on an analytical error analysis, the simple core circuits were optimized regarding their minimum possible variations [42]. With these results, area and power consumption can be calculated for the specification of the application. The result of a Monte-Carlo simulation for the used nonlinear circuit is depicted in Figure 9. The left and right histograms show the offset voltage error before (at point A in the schematic in Figure 8) and after the current limiter (at the output node). As expected from theory the error is dominated by the input stage. With these results and the constraints given by the high-level analysis of the algorithm the impact of mismatch can be predicted accurately for the system level.

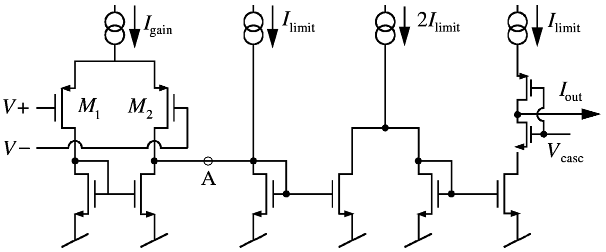

Figure 8. Non-linear circuit (only one output branch is depicted).

Analog architecture—dynamic Circular Network (*dCN*)

Based on the 1D discretization of the convex functional (11) an architecture has been developed. The design has been driven by required range of the parameter space (c_ρ and λ_h^2) and in addition relies on a perturbation analysis indicating the minimum accuracy necessary to achieve satisfying performance (see previous section).

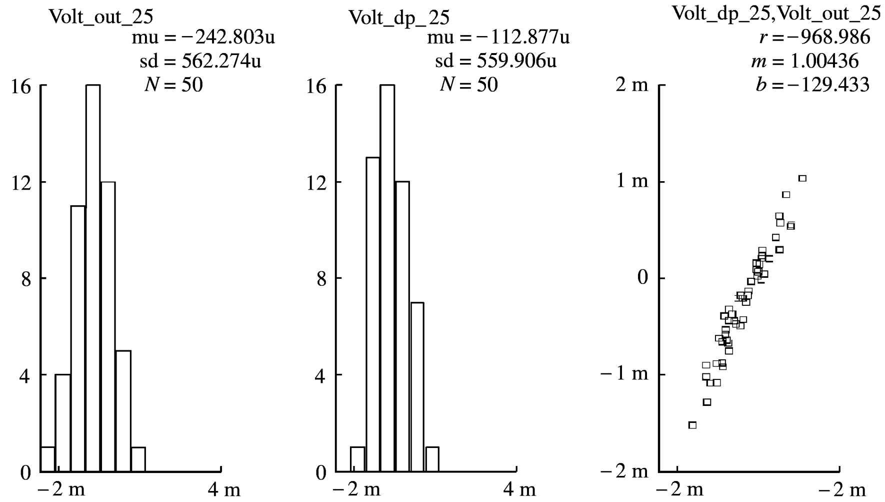

Figure 9. Monte-Carlo simulation results. Left: offset voltage error before the current limiter, middle: error after the current limiter, right: correlation between them.

A dynamic circular architecture for processing infinite signals

A major problem in designing analog circuits for signal processing is the interfacing to the signal source and the subsequent (digital) processing unit. When the signal is generated off-chip, the signal must be fed into the parallel structure of the analog unit. Usually this is done by simply writing the input to the isolated cells. During this time all cells act as memory elements, and afterwards the array "computes" the result within microseconds which is subsequently read out (e.g. [30]). Such a design yields time a rather poor computational performance since the effective computation time is low in comparison to the time used for data input/output. Another problem arises when the signal vector is larger than the chip cell array. In this case the signal vector has to be broken apart into overlapping blocks which have to be processed sequentially. As a consequence, the parts have to be merged in a further post-processing step [43].

Our solution to these problems is based on a dynamic reconfigurable circular architecture which is shifted along the data vector at a very high rate [44]. For the 1D case the neighborhood $\mathcal{N}_r$(i) of the cell C_i within a radius r on a grid $\mathcal{D}$ is defined by (Figure 10(a)).

$$\mathcal{N}_r(i) = \{C_j | \min(\langle i-j\rangle n, \langle j-i\rangle N) \le r, j = 0, \ldots, N-1, j \ne i\}\ i = 0, \ldots, N-1 \quad (14)$$

$\langle\cdot\rangle_N$ is the modulo N operation.

The shift mechanism of the network first replaces the static cell links with programmable switches and, second, allows each cell to perform in different working modes (see (16)–(18) below).

Let u_i denote the state variable of the cell C_i. Following Eqn (11) the dynamic behavior of the cell is defined by

$$\frac{\partial u_i}{\partial t} = \widetilde{L}(u_i, \mathcal{N}(i), \widetilde{g}_i). \quad (15)$$

$\mathcal{N}(i)$ is the local neighborhood modulo the network boundaries as defined in (14). The terms remaining constant during relaxation for node C_i are denoted by $\widetilde{g}_i$ for inner cells $\widetilde{g}_i\prime$ and for boundary cells. To operate the network in the dynamic manner as described above, each cell can be programmed to satisfy one of the following conditions (all index calculations are modulo N):

$$\begin{matrix}\text{fully connected cell}\\ \text{left (right) boundary cell with}\end{matrix} : \frac{\partial u_i}{\partial t} = \widetilde{L}(u_{i-1}, u_i, u_{i+1}, \widetilde{g}_i) \quad (16)$$

$$\begin{matrix}\text{vs. Neumann (zero flux) cond.}\\ \text{constant boundary cell with}\end{matrix} : \frac{\partial u_i}{\partial t} = \widetilde{L}(u_i, u_{i+1(i-1)}, \tilde{g}'_i) \quad (17)$$

$$\text{Dirichlet (fixed) condition} : \frac{\partial u_i}{\partial t} = 0. \quad (18)$$

Furthermore, a cell can be disconnected from the active network for calibration purposes. No long-term

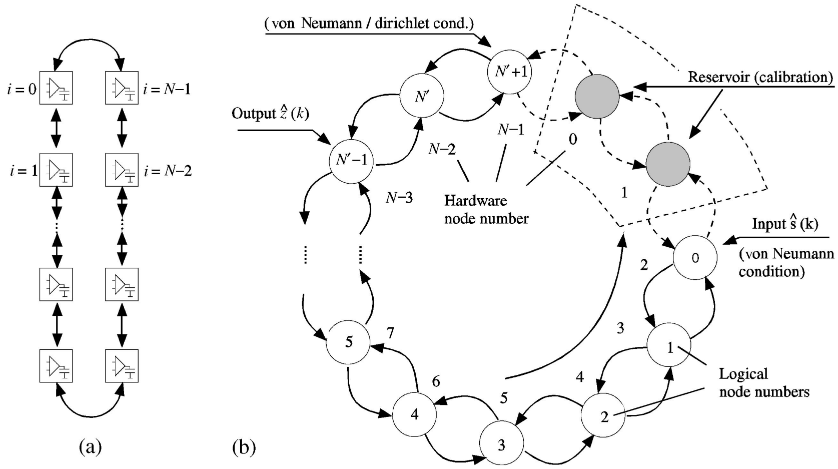

Figure 10. (a) 1D circular network; (b) dynamic circular loop: the network is reconfigured while the analog state variables stay resident in their corresponding cells. Part of the network can be disconnected from the active network for (e.g. offset voltage) calibration purposes.

analog storage devices are necessary since the calibration process is repeated periodically.

The operation of a 1D dynamic network is sketched in Figure 10(b). The analog values reside in the corresponding cell while only the connections of input, output, and boundary cells are reconfigured. The analog network processes the data while it is dynamically reconfigured. Moreover, the reconfiguration process is simple because only a maximum number of six cells have to be accessed per cycle.

When no processing is performed, the network acts like an analog shift register with a delay of N' multiplied by the sample time period. The effective length N' is the distance between the input and output node. It depends on the size of the calibration pool and available boundary conditions (see Figure 10(b)). For optimal Dirichlet boundary condition $\left(\frac{\partial u_i}{\partial t}=0\right)$ at the end of the network, the effective length N' can be chosen as large as possible since there will be no reflection from that boundary node.

Signal propagation

The architecture depicted in Figure 10 can be seen as a systolic architecture. In difference to conventional systolic systems where the data is "pumped" through the computational units, in this approach the connections between the cells are altered to satisfy the systolic behavior. In the following paragraphs the process of reconfiguration is reviewed in an analytical way. For simplicity the logical network length N' is set to N, the number of cells.

Let u_i^k denote an index operation modulo N in the kth step: $u_i^k = u_{\langle i-k\rangle}$ with i the logical node number and $\langle i-k\rangle$ the hardware node number (see Figure 10(b)). By applying Eqns (16)–(18) to the simplified 1D discrete case of Eqn (11), the system equations can be formulated as follows

$$\dot{u}_i^k = \frac{1}{2}(g_i^k - u_i^k) + f_{nl}(|u_{i+1}^k - u_i^k|) \quad : i = 0 \qquad (19)$$

$$\begin{aligned}\dot{u}_i^k = {} & (g_i^k - u_i^k) + f_{nl}(|u_{i-1}^k - u_i^k|) \\ & + f_{nl}(|u_{i+1}^k - u_i^k|) \quad : i = 1..(N-2)\end{aligned} \qquad (20)$$

$$\dot{u}_i^k = 0 \quad : i = N-1, \qquad (21)$$

for $kT<t\leq(k+1)T$ while the cell inputs are $g_i^k = \hat{s}(k-i), \forall i = 0,\ldots,N-1$, and the initial values $(t = kT)$ are given by the preexisting intermediate results

$$u_0^k(t = kT) = \hat{s}(kT) \ : i = 0 \qquad (22)$$

$$u_i^k(t = kT) = u_{i-1}^{k-1}(t = kT) \ : i = 1..(N-1). \qquad (23)$$

$\hat{\mathrm{s}}^{(k)}$ is the input signal and T the sample time. The signal is virtually shifted over the network since the input signal g and the initial conditions (23) remain constant, respectively $(\langle i-1-(k-1)\rangle_N = \langle i-k\rangle_N)$. The output can be extracted at each processing step by retrieving the state value of the Dirichlet boundary node $\hat{z}(k) = u_{N-1}^k$.

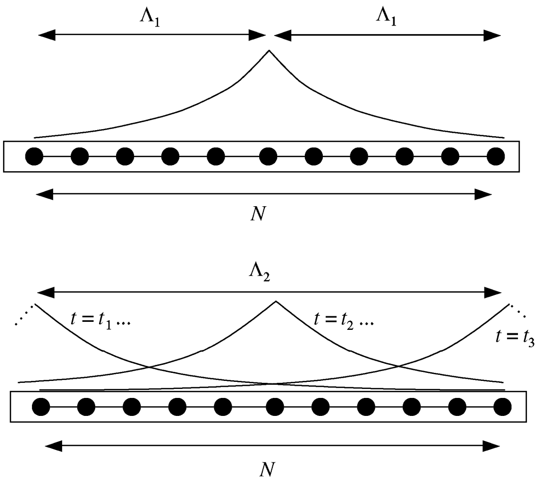

Figure 11. Influence radius Λ vs. the network size N (Λ_1 parallel mode, Λ_2 circular mode), t_2 after $N/2$, t_3 after N cycles.

Furthermore an "area of influence" is defined in a neighborhood of the cell $C_j : i-\Lambda<j<i+\Lambda$. Λ describes the maximum distance between two mutually influencing cells. The value of this crucial distance is

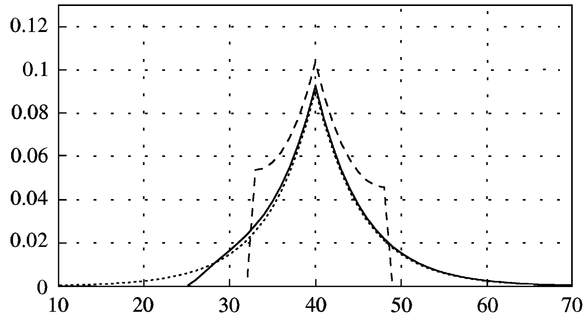

Figure 12. Linear diffusion results for different architectures; as an input signal a Dirac-impulse has been used. —circular (N = 16); – –parallel (N = 16); – – –parallel (N = 200).

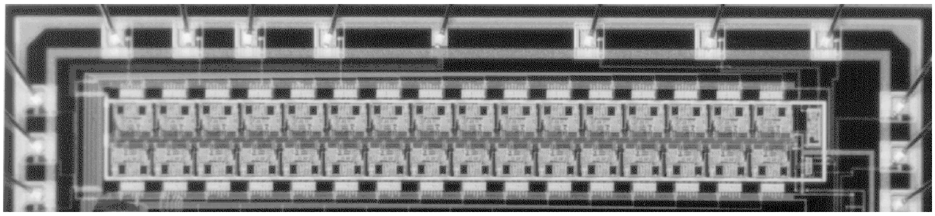

Figure 13. Layout photograph of the chip: 32 cells are performing a non-linear filtering of analog signals of infinite length (fabricated in 0.8 μm CMOS, Austria Microsystems/AMS).

determined by the characteristic space constant (e.g. diffusion constant) for the underlying signal processing task. As a matter of fact Λ is also limited by the size of the network and the required or achievable accuracy of the used analog circuit.

In a conventional parallel processing mode the diffusion constant has to be chosen to satisfy $\Lambda_p < N/2$, with N being the number of parallel working nodes. This is the minimum condition for a correct result at the center position. For the dynamic circular mode Λ can be doubled ($\Lambda_c < N$) since N is the size of memory in the network (Figure 11). Of course this holds only for systems where the time constant τ is smaller than the sample time T. For analog systems, this can be guaranteed by the extremely fast settling time of the network.

Figure 12 shows the impulse response for different network architectures. The dotted line is the correct propagation for a linear diffusion ($\Lambda \approx 30$) computed on a 1D network of 200 nodes operating in parallel. The dashed line is valid for a parallel network of only 16 nodes, whereas the solid one holds for a network of the same size operating in the dynamic circular manner. It can be seen that the difference in impulse response

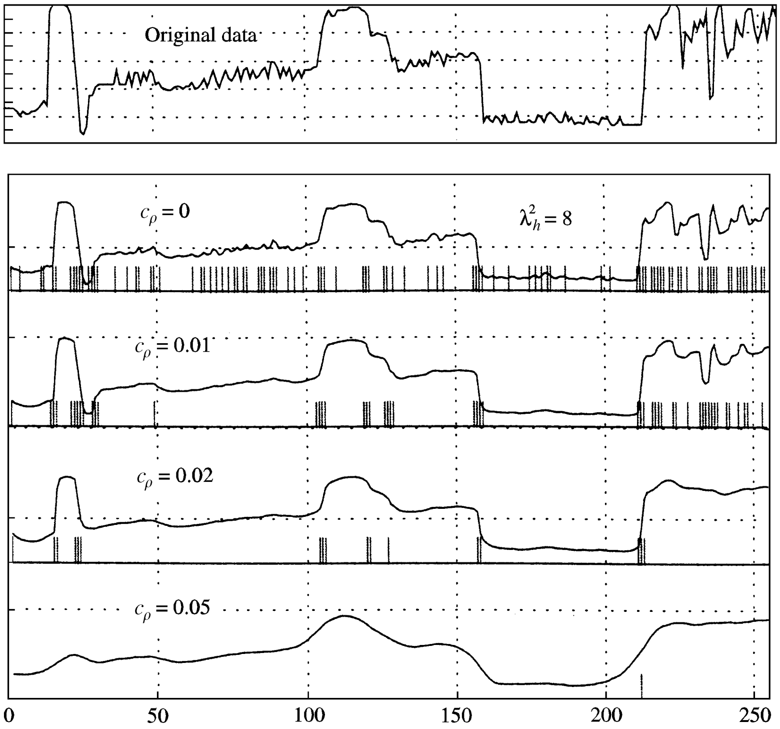

Figure 14. Measurement results for different values of c_p. At such loci where the smoothing process adaptively stops vertical lines are plotted.

between loop-mode and the correct propagation result is much smaller than between the parallel and ideal mode. Due to the fixed value boundary on the right, the memory length is extended to infinity resulting in an asymmetric impulse response.

1D 0.8 μm CMOS realization and 1D/2D experiments

Based upon the algorithm and architecture described in the previous sections, a 1D prototype consisting of 32 identical analog cells was fabricated (see Figure 13). The major challenge was the design of the nonlinear function which has to be very accurate to guarantee minimum impact of transistor mismatch (see *Analog cells* section). The dynamic routing resources are included by MOS switches which are controlled by a digital unit ([44], Figure 7).

Measured results for different values of c_ρ are depicted in Figure 14. c_ρ ranges from 0.01 to 0.05 for an approximate constant value of λ_h^2. It can be seen that for smaller values of c_ρ the salient signal structure is preserved while for increasing c_ρ the process degenerates to a linear diffusion.

The measured signal can be compared to the simulation results computed by solving the nonlinear equation for $c_\rho = 0.01$ and $\lambda_h^2 = 8$ in Figure 15. The difference between these signal vectors is mainly due to offset voltage deviations between neighboring cells. Whereas in smooth areas the difference is as low as the offset voltage itself (a few millivolts), in steep portions of the signal the difference can be large since the adaptive behavior is slightly different.

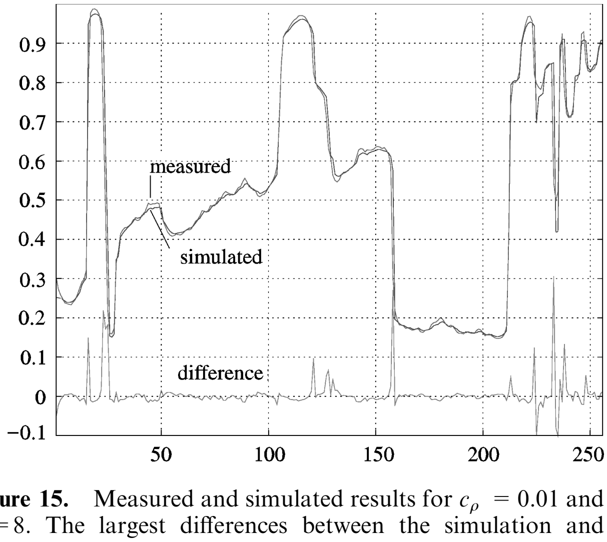

Figure 15. Measured and simulated results for c_ρ = 0.01 and $\lambda_h^2 = 8$. The largest differences between the simulation and measured data is located in steep signal portions due to the adaptive behavior.

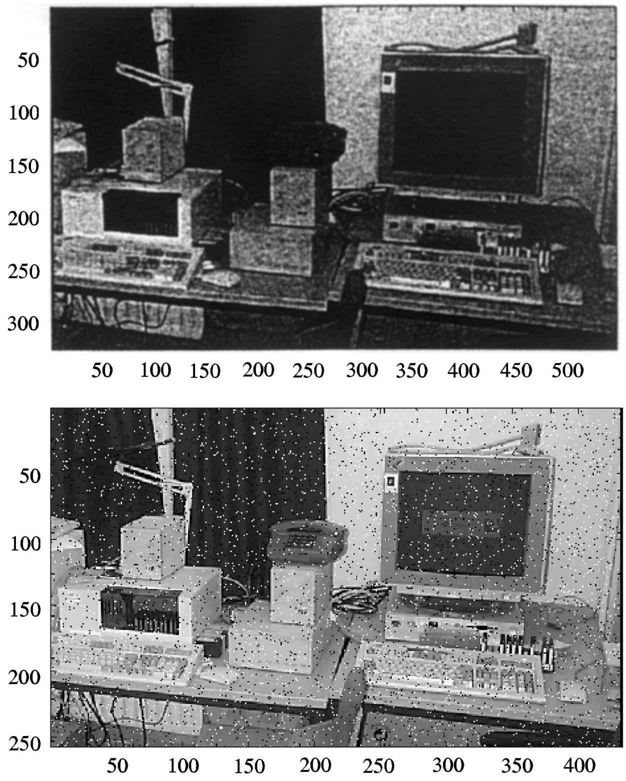

Figure 16. A real-world gray-value image corrupted with noise (top: gaussian noise σ = 10%, bottom: salt and pepper noise 5%).

In further experiments an overall random error of about $\sigma = 2.5$ mV (standard deviation of the voltage differences between input and output of band limited signals) was measured. This first experimental design can successfully and efficiently perform computationally intensive nonlinear smoothing of 1D signals.

In addition to the 1D experiments, we carried out several experiments with 2D gray-value images (see Figure 16, [45]). The goal of these experiments was to investigate from a practical perspective to which extent even only the 1D chip with its reduced design complexity is capable of satisfactorily smoothing 2D image data in a nonlinear fashion. In this case, the image data was fed into the 1D chip in a consecutive row-by-row and column-by-column manner (note that for this input

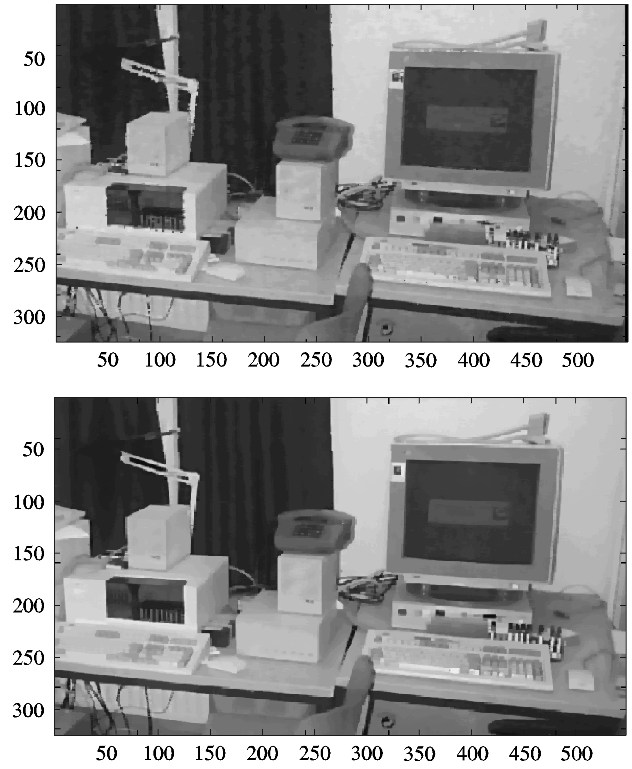

Figure 17. Smoothed image using the 1D analog nonlinear relaxation network (see text).

scheme the result is not fully consistent with the 2D theory since only an orientation-constrained dual nonlinear diffusion along orthogonal data vectors can be achieved). The promising results shown in Figures 17 and 18 demonstrate the validity of this pragmatic approach since adaptive nonlinear smoothing as well as preservation of contrast edges can be achieved. For this experiment, the partial derivatives u_x and u_y have been computed via simple finite differences whereas the threshold has been set to values slightly above the theoretical value of c_ρ such as to reduce the influence of the inevitable noise of the analog components. Our current experience with the 1D experimental chip indicates a peak performance of 500 kHz/pixel (equivalent to 130 ms/image). Since this performance is not limited by the relaxation time of the 1D network of cells itself but solely by the rather weak performance of the currently used read-out circuitry, it can be even further improved in the future through appropriate design measures. Eventually the experiments showed that the 1D chip has the potential of a powerful but cheap image preprocessing device.

Conclusion and further work

An efficient analog 1D hardware implementation of a nonlinear diffusion algorithm has been proposed and fabricated. Substantial problems (e.g. input/output of the data, limited number of cells) in the design of massively-parallel analog hardware have been overcome. The analog network cells are connected in a circular structure. Due to the dynamic reconfiguration of the connections a nonlinear adaptive filter kernel can be shifted virtually over a signal vector of “infinite” length. The boundary conditions have to be configured as von Neumann (zero flux) or Dirichlet (fixed) boundaries. Furthermore, our design has the potential for calibrating cells, that are not in the active part of the network, without disrupting the data stream. The storage duration is constant for all analog signal samples. Consequently, linear systematic error effects resulting from leakage currents can be compensated as well. A prototype containing a 1D nonlinear network has been fabricated in 0.8 μm CMOS and is fully functional. The chip can also be used for 2D image smoothing provided that the result of successive 1D relaxation along the rows and columns of the image is acceptable for a given task.

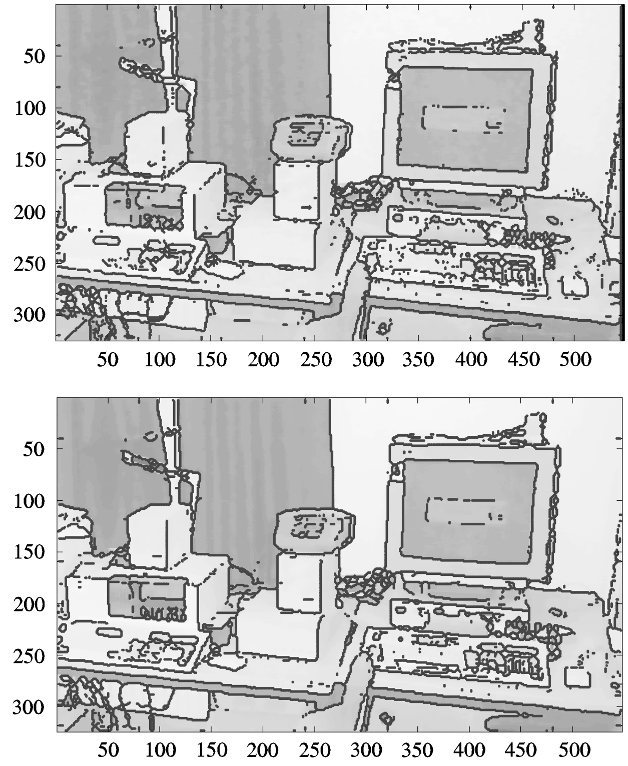

Figure 18. Detected edges as signal transitions where the smoothing process adaptively stopped.

Further work will focus on a 2D realization being fully consistent with the multi-dimensional theory. In addition, effective error reduction methods and efficient architectures for higher dimensions have to be investigated in order to exploit the potential of parallel analog hardware for computer vision.

References

1. Mumford, D. & Shah, J. (1989) Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.* **42**: 577–685.
2. Schnörr, C. (1994) Unique reconstruction of piecewise-smooth images by minimizing strictly convex non-quadratic functionals. *J. of Math. Imag. Vision* **4**: 189–198.
3. Harris, J., Koch, C. & Luo, J. (1990) A two-dimensional analog VLSI circuit for detecting discontinuities in early vision. *Science* **248**: 1209–1211.
4. Geman, S. & Geman, D. (1984) Stochastic relaxation, Gibbs distributions and the Bayesain restoration of images. *IEEE Trans. Patt. Anal. Mach. Intell.* **6**(6): 721–741.
5. Blake, A. & Zisserman, A. (1987) *Visual Reconstruction*. MIT Press.
6. Geiger, D. & Yuille, A. (1991) A common framework for image segmentation. *Int. J. of Comp. Vision* **6**(3): 227–243.
7. More, J.-M. & Solimini, S. (1995) *Variational Methods in Image Segmentation*. Boston: Birkhäuser.
8. Stevenson, R.L., Schmitz, B.E. & Delp, E.J. (1994) Discontinuity preserving regularization of inverse problems. *IEEE Trans. Systems, Man and Cyb.* **24**(3): 455–469.
9. Li, S.Z., Huang, Y.H. & Fu, J. (1995) Convex energy functionals in the DA model. In *Proc. IEEE Int. Conf. Image Processing*.
10. Schnörr, C. (1998) A study of a convex variational approach for image segmentation and feature extraction. *J. of Math. Imag. Vision* **8**(3): 271–292.
11. Heers, J., Schnörr, C. & Stiehl, H.S. (1999) Parallel and globally convergent iterative numerical schemes for computationally efficient nonlinear variational image segmentation. Technical Report 283/99, University of Hamburg, Comp. Sci. Dept., AB KOGS.
12. Mead, C. (1989) *Analog VLSI and Neural Systems*. Addison-Wesley Computational and Neural Systems Series. Addison-Wesley Publishing Company.
13. Hosticka, B.J. (1985) Performance comparison of analog and digital circuits. *Proceedings of the IEEE* .
14. Poggio, T., Torre, V. & Koch, C. (1985) Computational vision and regularization theory. *Nature* **317**: 314–319.
15. Koch, C. & Poggio, T. (1985) Ill-posed problems in early vision: from computational theory to analogue networks. *Proceedings of the Royal Society London/B* **226**: 303–323.
16. Delbrueck, T. (1993) *Investigations of Visual Transduction and Motion Processing*. PhD thesis, California Institute of Technology Pasadena, California.
17. Mahowald, M. (1992) *VLSI Analog of Neuronal Visual Processing: A Synthesis of Form and Function*. PhD thesis, California Intitute of Technology Pasadena, California.
18. Schlüβler, J.U., Werner, J., Dohndorf, J., Koren, I. & Ramacher, U. (1996) Current mode implementation of a neural algorithm for image preprocessing. In *Proceedings of MicroNeuro*, pp. 107–110.
19. Harris, J., Koch, C. & Lou, J. (1989) Resistive fuses: Analog hardware for detecting discontinuities in early vision. In Mead, C. & Ismail, M. (eds), *Analog VLSI Implementation of Neural Systems*, Chp 2. Kluwer Academic Publishers, pp. 27–55.
20. Sivilotti, M.A., Mahowald, M.A. & Mead, C.A. (1987) Real-time visual computation using analog CMOS processing arrays. In *Stanford Conference on Very Large Scale Integration*. Cambridge M.A.: MIT Press.
21. Rudin, L.I., Osher, S. & Fatemi, E. (1992) Total variation based noise removal algorithms. *Physica D* **60**: 259–268.
22. Sawaji, T. Sakai, T., Nagai, H. & Matsumoto, T. (1998) A floating gate MOS implementation of resitive fuse. *Neural Computation* **10**(2): 485–498.
23. Chua, L.O. & Yang, L. (1988) Cellular neural networks: Applications. *IEEE Trans. on Circuits and Systems* **35**(10): 1273–1290.
24. Chua, L.O. & Yang, L. (1988) Cellular neural networks: Theory. *IEEE Trans. on Circuits and Systems* **35**(10): 1257–1272.
25. Schnörr, C., Stiehl, H.S. & Grigat, R.-R. (1996) On globally asymptotically stable continuous-time CNNs for adaptive smoothing of multidimensional signals. In *IEEE International Workshop on Cellular Neural Networks and their Applications*, Sevilla, pp. 351–356.
26. Rodríguez-Vàsquez, A., Espejo, S., Domíuez-Castro, R., Huertas, J.L. & Sànchez-Sinencio, E. (1993) Current-mode techniques for the implementation of continuous and discrete-time cellular neural networks. *IEEE Trans. on Circuits and Systems* **40**: 140–163.
27. Manganaro, G., Arena, P. & Fortuna, L. (1999) *Cellular Neural Networks*, Vol. 1, *Springer Series in Advanced Microelectronic*. Springer-Verlag.
28. Varrientos, J.E., Sánchez-Sinencio, E. & Ramírez-Angulo, J. (1993) A current-mode cellular neural network implementation. *IEEE Trans. on Circuits and Systems* **40**(3): 147–155.
29. Kinget, P. & Steyaert, M. (1995) A programmable analog cellular neural network CMOS chip for high speed image processing. *IEEE Journal of Solid-State Circuits* **30**(3): 235–243.
30. Kinget, P. & Steyaert, M. (1997) *Analog VLSI integration of massive parallel signal processing systems*. Kluwer Academic Publishers.
31. Nossek, J.A., Seiler, G., Roska, T. & Chua, L.O. (1992) Cellular neural networks: Theory and circuit design. *International Journal of Circuit Theory and Applications* **20**: 533–553.
32. Cruz, J.M. & Chua, L.O. (1998) A 16 × 16 cellular neural network universal chip: The first complete single-chip dynamic computer array with distributed memory and with gray-scale input-output. *Analog Integrated Circuits and Signal Processing* **15**: 227–237.
33. Piasio, A., Dawidwiziuk, A., Halonen, K. & Porra, V. (1997) Fast and compact 16 by 16 cellular neural network

implementation. *Analog Integrated Circuits and Signal Processing*.

34. Espejo, S., Rodríguez-Vásquez, A., Domínguez-Castro, R., Huertas, J.L. & Sánchez-Sinencio. (1994) Smart-pixel cellular neural networks in analog current-mode CMOS technology. *IEEE Journal of Solid-State Circuits*, **29**(8).
35. Sprengel, R. (1996) *Entwurf und Analyse nichtlinearer Diffusionsverfahren für die Bild-verarbeitung*. PhD thesis, Universität Hamburg.
36. Ciarlet, P.G. (1978) *The finite element method for elliptic problems*, Vol. 4, *Studies in mathematics and its applications*. Amsterdam: North-Holland Publishing.
37. Lovett, S.J., Welten, M., Mathewson, A. & Mason, B. (1998) Optimizing MOS transistor mismatch. *IEEE Journal of Solid-State Circuits* **33**(1): 147–149.
38. Domínguez-Castro, R., Espejo, S., Rodríguez-Vázquez, A. & Carmona, R. (1997) A 0.8 µm CMOS 2-D programmable mixed-signal focal-plane array processor with on-chip binary imaging and instructions storage. *IEEE Journal of Solid-State Circuits* **32**(7): 1013–1025.
39. Vittoz, E.A. (1994) Analog VLSI signal processing: Why, where, and how? *Analog Integrated Circuits and Signal Processing* **6**: 27–44.
40. Andreou, A.G. & Boahen, K.A. (1996) Translinear circuits in subthreshold MOS. *Analog Integrated Circuits and Signal Processing* **9**: 141–166.
41. Ismail, M. & Fiez, T. (1994) *Analog VLSI*, *Signal and Information Processing*. *Electrical and Computer Engineering Series*. McGraw-Hill.
42. Wiehler, K., Klingspohr, H. & Grigat, R.-R. Auswirkungen von Bauteil-störungen in der analogen Signalverarbeitung am Beispiel eines CMOS-Transkonduktanzverstärkers für zelluläre neuronale Netze. In *8. E.I.S. Workshop*, *Hamburg*, *Germany*, pp. 163–171. GMD National Research Center for Information, D-53754 Sankt Augustin.
43. Pineda de Gyvez, J., Wang, L. & Sanchez-Sinencio, E. (1996) Large-image CNN hard-ware processing using a time multiplexing scheme. In *IEEE International Workshop on Cellular Neural Networks and their Applications*, pp. 405–410.
44. Wiehler, K., Lembcke, R., Grigat, R.-R., Heers, J., Schnörr, C. & Stiehl, H.-S. (1998) Dynamic circular cellular networks for adaptive smoothing of multi-dimensional signals. In *IEEE International Workshop on Cellular Neural Networks and their Applications*, London, pp. 313–318.
45. Wiehler, K., Grigat, R.R., Heers, J., Schnörr, C. & Stiehl, H.S. (1998) Real-time Adaptive Smoothing with a 1-D Non-linear Relaxation Network in Analogue VLSI Technology. In *20. DAGM-Symposium Musterekennung*, pp. 341–348.

AUTHOR QUERY FORM

HARCOURT PUBLISHERS

JOURNAL TITLE: RTIM DATE: 5/12/2000
ARTICLE NO. : 19990218

Queries and / or remarks

Manuscript Page/line	Details required	Author's response
Refs 5, 12, 19, 27, 30, 41	Please provide city of publication.	
Ref 34	Provide initials of Sanchez-Sinecio, and pages or article.	
Refs 18, 33, 45, 13 and 9	More info required–vol, pages, or publisher and city.	
Page 8 (listings 3)	we have changed bevel to level, pl check.	
	Electronic files are not used for Fig. 16(a). Please check.	
	Figs. 11, 12, 13 & 14 are placed before reference. Please check.	