Image Processing by a Programmable Grid Comprising Quantum Dots and Memristors

Yalcin Yilmaz and Pinaki Mazumder, *Fellow, IEEE*

***Abstract*—Real-time vision systems require computationally intensive tasks which often benefit greatly from fast and accurate feature extractions. Resistive grid-based analog structures have been shown to perform these tasks with high accuracy and added advantages of compact area, noise immunity, and lower power consumption compared to their digital counterparts. However, these are static structures and can only perform one type of image processing task. In this paper, an analog programmable memristive grid-based architecture capable of performing various real-time image processing tasks such as edge and line detections is presented. The unit cell structure employs 3-D confined resonant tunneling diodes that are called quantum dots in this paper for signal amplification and latching, and these dots are interconnected between neighboring cells through nonvolatile continuously variable resistive elements that are more popularly known as memristors. A method to program memristive connections is introduced and verified through circuit simulations. Various diffusion characteristics, edge detection, and line detection tasks have been demonstrated through simulations using a 2-D array of the proposed cell structure and analytical models have been provided.**

***Index Terms*—Cellular neural networks, image processing, memristors, resonant tunneling devices (RTDs).**

I. Introduction

FEATURE extraction is a fundamental task in vision systems as extracted features provide bases for correlation. In digital general purpose processors, many image processing applications require an immense number of operations per second, albeit these applications do not require floating point accuracy [11]. Use of fast, simple, and relatively accurate extraction systems in vision machines directly reduces the processing time and required iterations. The main processor element can thereby rely on the reduced dataset that provides quality information on the extracted features for decision making.

Inherent parallel processing capabilities of cellular nonlinear network (CNN)-based architectures make them an efficient platform for various image processing tasks [1], [2]. Real-time operation provides fast processing times, and local connections provide simplicity, scalability, and power efficiency for VLSI implementations [15]. Therefore, much effort has been put into developing novel methods and finding adequate CNN templates to perform detail extraction tasks in vision systems, such as edge detection [16]–[18] which benefit greatly from immense parallelism and computational efficiency.

Resistive grid-based architectures are shown to provide simple yet efficient ways to perform many image processing tasks and motion detection, and they are simple forms of CNNs [1]. Additional advantages including compact area, noise immunity, and lower power consumption compared to digital computation structures, make them attractive for researchers. They are also relatively insensitive to mismatches in component values in VLSI chips [12]. However, most of the resistive grid-based architectures in the literature are static application specific structures and do not have the functional flexibility of their digital counterparts. Therefore, novel methods and devices should be introduced in these architectures to achieve functional versatility.

Resonant tunneling diodes (RTDs) have been employed in many applications including various CNN architectures due to their negative differential resistance (NDR) and fast switching characteristics. In [9], RTDs have been introduced as variable resistors to introduce versatility and compactness to CNN unit cells. In [10], a CNN architecture employing RTDs is investigated for its operation and it is shown that RTDs support fast settling times for various image processing applications.

Memristors have recently attracted significant attention in various applications after Hewlett–Packard research labs revealed that memristance can be observed in nanoscale thin film devices [6]. The significance of these devices arises from the fact that they can retain their resistive state even when power is turned OFF, displaying nonvolatility and they might enable scaling beyond CMOS technology limits. The variable resistance characteristics of these devices are proposed to be utilized in ultradense crossbar memories [13], configurable logic applications, and as synaptic connections in neuromorphic architectures [14]. They have also been used for carrying out image processing tasks which benefit from their nonlinearity and adaptive characteristics [19]. Most of these applications could benefit from the use of memristors more if these devices show properties of long-term stability of resistive states and little or no degradation of these states when the values stored in these devices are read. Fabrication results reported in [22] indicate the observation of diode-like behavior in amorphous Silicon (a-Si) devices which are undisturbed when the voltages across the devices are below a certain threshold and can retain their states more than 4 years at room temperature.

In this paper, a memristive grid-based architecture which improves the velocity tuned filter architecture proposed by our

Manuscript received December 14, 2012; revised March 25, 2013; accepted May 1, 2013. Date of publication May 15, 2013; date of current version November 6, 2013. This work was supported in part by the National Science Foundation under Grant CCF 1017143. The review of this paper was arranged by Associate Editor M. R. Stan.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: yalciny@umich.edu; mazum@eecs.umich.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNANO.2013.2263153

group [4] is presented. It is demonstrated that when memristive connections are incorporated, various diffusion characteristics are obtained, and the proposed architecture can be programmed for different image processing applications such as edge detection and line detection providing flexible analog processing environment that can perform various tasks. In addition RTDs are utilized to provide high speed signal detection and amplification. A method to program memristive connections in four directions is also proposed and demonstrated.

In Section II, information on RTDs and memristors are provided and the proposed unit cell structure is introduced. In Section III, a method to program memristive connections in array configuration is demonstrated. Analytical bases for edge detection and line detection operations are provided in Section IV, and simulations verifying these operations are presented in Section V.

II. Proposed Architecture

A. Memristor Model

Memristor is the fourth fundamental circuit element which relates charge with magnetic flux as described by Chua in [20]. HP research labs revealed in [6] that two terminal thin-film-based devices can exhibit memristive behavior.

As laid out in [6], these devices can be modeled as a combination of two series variable resistors, with one of the resistors having a high-dopant concentration, thus having low resistance and the other having a low-dopant concentration, thus having high resistance. Application of a voltage across the terminals of the memristive device triggers dopant drift. Depending on the voltage polarity, the width of the doped region can increase or decrease. As the width of the doped region increases, conductance increases and as it decreases, conductance decreases.

Therefore, the total resistance of the memristor can be expressed as

$$R = \frac{w}{D} R_{\mathrm{ON}} + \left(1 - \frac{w}{D}\right) R_{\mathrm{OFF}} \tag{1}$$

where w is the width of the doped region, D is the total length of the thin film, R_{ON} is the lowest resistance when $w = D$, and R_{OFF} is the highest resistance when $w = 0$. When the current is passed through the device, the width of the doped region, w changes. The rate of change of w with time is

$$\frac{dw(t)}{dt} = \mu_v \frac{R_{\mathrm{ON}}}{D} i(t) \tag{2}$$

where μ_v is the dopant mobility and $i(t)$ is the current passing through the device.

The above model presented by HP labs is a linear model and does not account for the nonlinearities that are present in most fabricated devices due to second-order boundary effects seen at thin film edges. The movement of the boundary between doped and undoped regions is greatly hindered when the width of the doped region approaches device limits (i.e., $w = 0$ or $w =$ D) [21]. After including the boundary effects, the model expression becomes

$$\frac{dw(t)}{dt} = \mu_v \frac{R_{\mathrm{ON}}}{D} i(t) f(x) \tag{3}$$

where $f(x)$ is the window function modeling the nonlinear dopant drift. This function is an estimation of nonlinearity and depends on the specific device behavior. A sample function is provided in [21].

The actual switching characteristics, namely switching delay, of memristors depend on material properties, device dimensions, and biasing voltage.

In the proposed architecture, memristive connections based on this model are adopted to provide programmability for the realization of different characteristics including isotropic, anisotropic symmetrical, and asymmetrical diffusion in grid architecture giving way to various spatiotemporal filter implementations.

A SPICE model [7] based on this memristor model is adapted to carry out circuit simulations.

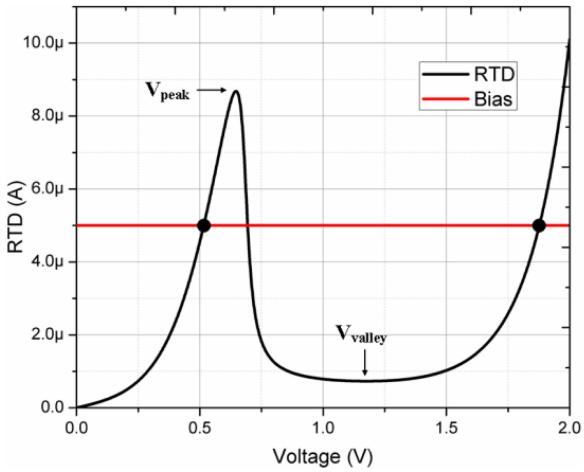

Fig. 1. RTD $I-V$ curve. The red line corresponds to the bias level. The intersections of the two lines represent the stable operating points.

B. Resonant Tunneling Diode Model and Biasing

RTDs have been employed in many circuit applications utilizing their fundamental characteristic of NDR. NDR implies that for certain range, the increase in applied voltage across an NDR device will result in decreased current through it, indicating increased resistance with increased voltage.

RTD conductance is determined by two mechanisms: the first mechanism is resonant tunneling, which provides the NDR characteristic, and the other mechanism is diode conduction.

The NDR property of the RTD I-V characteristics is shown in Fig. 1 utilizing the physics-based model laid out in [8]. The RTD current $J_{\mathrm{RTD}}(V)$ is given by

$$J_1(V) = \frac{qm * kT\Gamma}{4\pi^2 \hbar^3} \mathrm{in} \left(\frac{1 + e^{(E_F - E_r + n_1 qV/2)/kT}}{1 + e^{(E_F - E_r - n_1 qV/2)/kT}} \right) * \left(\frac{\pi}{2} + \arctan\left(\frac{E_r - n_1 qV/2}{\Gamma/2} \right) \right) \tag{4}$$

$$J_2(V) = H\left(e^{n_2 qV/kT} - 1\right) \tag{5}$$

$$J_{\mathrm{RTD}}(V) = J_1(V) + J_2(V) \tag{6}$$

where $J_1(V)$ is the current due to resonant tunneling and $J_2(V)$ is the diode conduction current. E_F is the Fermi energy, E_r is the resonant level energy, Γ is the resonant width, n_1 and n_2 are empirical model parameters. $q, m^*, k, T, \hbar$ are electron charge, effective mass, Boltzmann constant, absolute temperature, and reduced Planck constant, respectively. V is the voltage across the device.

The main advantage of the NDR characteristic becomes apparent when RTD is biased with a static current source. If current magnitude of the source is selected such that it intersects RTD I-V curve in three places as shown in Fig. 1, two stable voltage points are obtained. This result indicates that for the same amount of current passing through RTD, the voltage across it can take two stable values which correspond to the lowest and highest voltage intersection points. RTD does not stabilize in the middle intersection point, since any small disturbance causes it to switch to one of the outer intersection points.

The bistable characteristic of this structure can be utilized to build voltage level detectors since any voltage below switching threshold results in stabilizing in the low state, and any voltage above threshold results in stabilizing in high state. RTD switching threshold can be approximated as

$$V\mathrm{th}_{\mathrm{RTD}} = \frac{V_{\mathrm{peak}} + V_{\mathrm{valley}}}{2} \tag{7}$$

where V_{peak} and V_{valley} are the peak and the valley voltages of the RTD, respectively.

When used in the detection mode, as the system starts all the RTDs are biased to the low-voltage state, and a controlled disturbance toward a higher voltage results in the RTDs stabilizing at the higher stable level, allowing the detection and locking of the signal state.

C. Unit Cell Structure

Fig. 2(a) shows the proposed unit cell structure. It is composed of memristors to provide resistive connections to neighboring cells, diodes to introduce unidirectionality to these connections, and RTDs to detect and latch signal levels. The proposed cell has an input node denoted by $I_{n,m}$, a center node $C_{n,m}$, and an output node $O_{n,m}$. The input is driven by voltage signals that correspond to the pixel intensity level which can be generated by a photodetector.

Four memristors are connected to the center nodes of the unit cell and its neighbors, making the center node voltage a function of the center node voltages of the neighboring cells. Resistances of memristor connections determine how much neighbors' center voltages contribute to the center voltage of the cell. Series diodes allow current in one direction separating how outputs of the two neighboring cells affect each other. The output node is isolated from the center node by a diode providing a voltage barrier equal to the diode threshold. RTDs enable detection and latching of output signals. When biased with a current source, RTDs initially settle at the lower stable voltage. When the voltage level on the center node goes above detection threshold, RTDs settle at the higher stable voltage. Two stable states provide a binary output. The detection threshold is equal to the sum of the diode threshold and the switching threshold of the RTD.

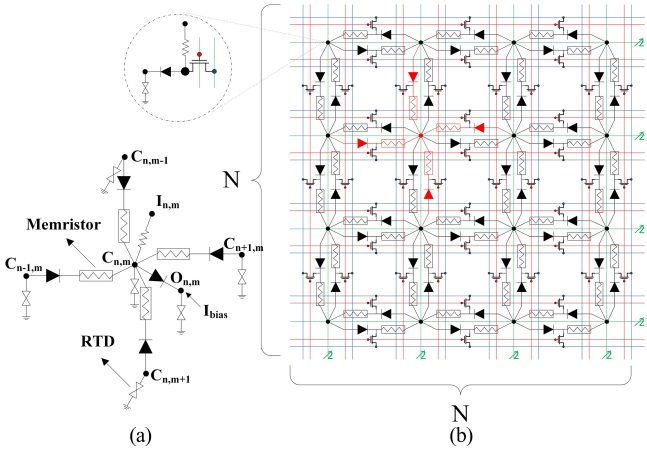

Fig. 2. (a) Proposed memristive cell. (b) Top view of the memristive processing array. A unit cell is highlighted in red. Red and green lines denote programming connections to access transistors.

Fig. 2(b) shows unit cells connected in a 2-D array fashion. A top view for a 4×4 sample processing array is provided to show the neighboring connections. The unit cell shown in Fig. 2(a) is highlighted in red. In order to program certain functionalities in the array, the memristor resistances need to be altered. The Green lines in Fig. 2(b) indicate the programming connections to the cells. Each green connection denotes programming-enable signal and voltage driver connections. Access transistors are used to isolate the connections during normal array operations. Programming connections can also be made to share the same connections as cell inputs, thus reducing number of access transistors if the input resistances are designed to be small at the expense of increased programming time or increased programming voltages due to voltage drop across the input resistor. Connections shown in red and blue as well as access transistors are needed during cell erase to bypass reverse-biased diodes. During erase operation voltage polarity across the memristor is reversed.

III. Programming Memristive Connections

To be able to implement different processing tasks in the same array, we need a procedure to program the resistances of the memristive connections.

Fig. 3 shows the programming flow for an $N \times N$ array. Programming is performed in four directions (left-to-right, right-to-left, top-to-bottom, and bottom-to-top) one direction at a time. Programming of the whole array is completed in four passes across the array in different directions to change the resistances of memristors in these directions. While a pass is being made in one direction, the resistances are set in a column by column fashion. Programming in this fashion reduces the total required time significantly compared to programming every memristors in the array individually. The duration and voltage amplitude of

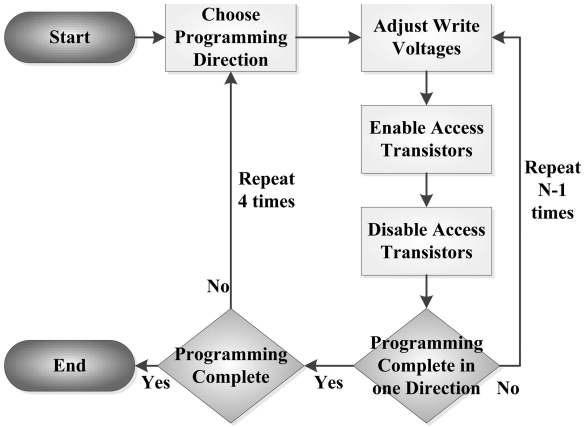

Fig. 3. Array programming flow.

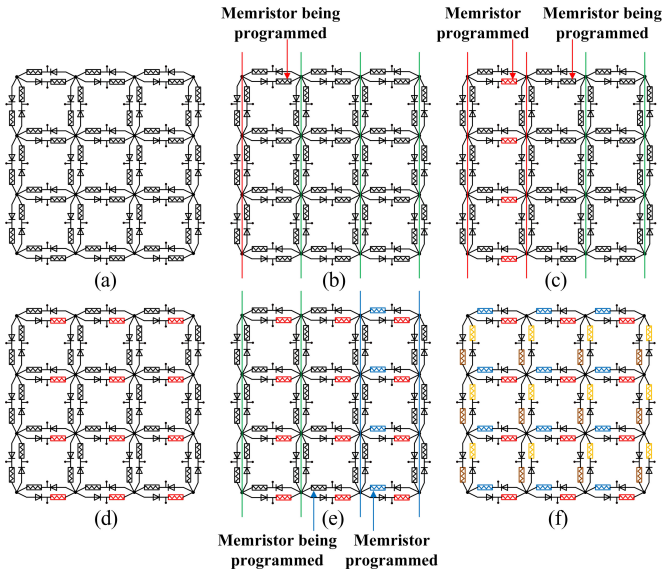

Fig. 4. Programming in one direction. The green lines indicate the low-voltage level (0 V); other colors indicate altered high voltage levels. The different colors of memristors indicate different final resistances. (a) Array in initial state. (b) Programming started in left to right direction. (c) Programming of first column of memristors completed in left to right direction. (d) Programming of all memristors completed in left to right direction. (e) Programming of first column of memristors completed in right to left direction. (f) Memristor array after all memristors are programmed in all directions.

the write pulses determine the resistance to be stored in memristors.

Within one direction, same voltage amplitudes and pulse durations are used. However, pulse characteristics can be changed in different directions to program different resistances, hence to program different functionalities to the array.

A sample programming operation in the left to right direction is shown in Fig. 4. All the memristors in the array are initially at the low-resistance state. The programming begins by setting the first column write voltage to high (indicated with a red line) and the remaining columns (indicated with a green line) to low (0 V in our implementation).

In this configuration the first column of memristive connections observe a nonzero voltage difference across, whereas the remaining connections observe zero-voltage difference. In

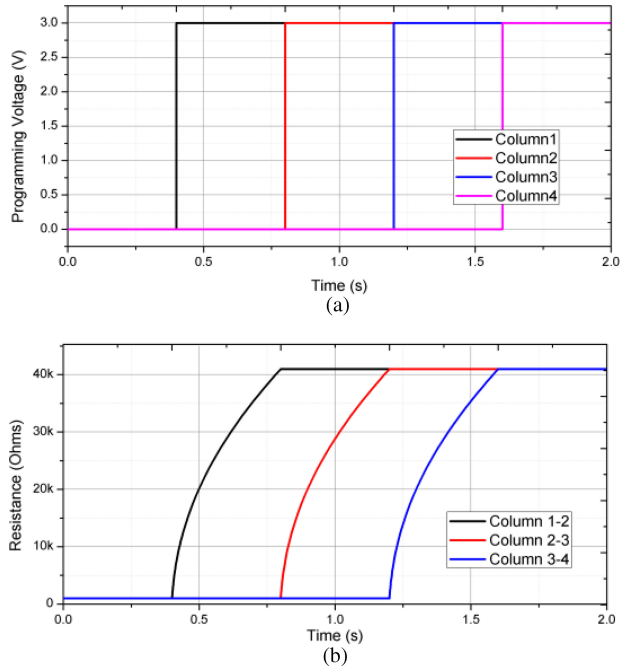

Fig. 5. Programming in one direction in a 4 × 4 array. (a) Programming voltages. (b) Memristor resistances in the same row.

the first column, only half of the memristive connections are programmed due to the fact that half of the series diodes are forward-biased conducting high currents, and the other half are reverse biased.

Once the resistances of the first column memristors reach the desired level, the second column write voltage is set to high, making the voltage difference across these memristors zero, thus stopping their programming. The rise of the voltage levels on the second column in turn causes the voltage difference across the next column memristors to be nonzero. Once these memristors reach the desired resistance, the next column's voltage is raised. This process is repeated until all the memristors in the selected direction are programmed.

When programming in the selected direction is completed, another direction is selected and the same process is repeated in this new direction. The use of different voltages or change of voltage raise-durations result in programming of different resistances in this direction.

In Fig. 5(a), sample left-to-right direction programming voltages to the memristor array are shown. As described earlier, write voltages are applied per column basis. Voltage levels are increased with same time intervals. Fig. 5(b) shows how the resistances of the memristive connections change. The programming scheme succeeds in tuning all the memristors in the same direction to the same resistive state.

In order for our proposed method to be feasible, two critical requirements must be met: the first requirement is that memristors should be able to be programmed even when there is a forward-biased diode connected in series. The second requirement is that the resistive state of the memristor should not change or should change negligibly when there is a reverse-biased diode connected in series.

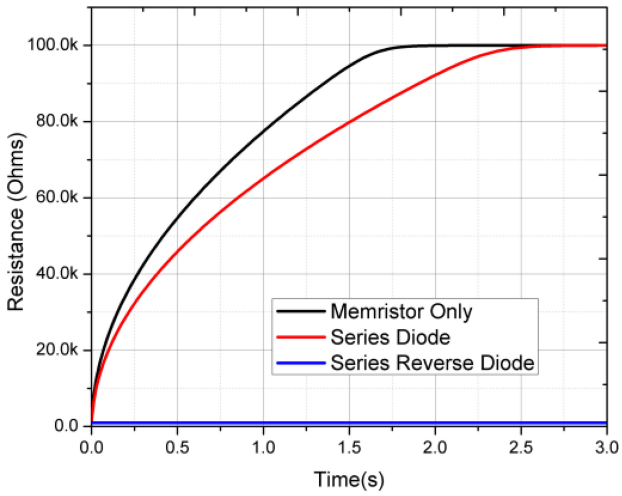

Fig. 6. Memristive connections under different bias conditions.

Fig. 6 shows effect of having a series diode with memristor while performing a programming operation. The results indicate that having a forward-biased series diode with memristor causes the memristor to be programmed to a lower resistance than when programmed with no series diode. This reduction in resistance can be compensated for by increasing programming time or voltage amplitude. This result indicates that it is still possible to program memristive connections with series resistance. A series reverse-biased diode causes no significant change in the resistance of the memristor during programming, effectively shielding it from the high voltage bias. This property enables programming connections in opposite directions possible, which is crucial in our proposed scheme.

IV. Analytical Modeling

A. Edge Detection

Edge detection provides physical information about object boundaries in processed images and is a fundamental feature extraction task in vision systems. An edge is located at the transition points between two different intensity levels.

The memristive grid provides diffusion characteristics that can be adjusted by controlling the resistances. These characteristics combined together with bistable RTD biasing can be used to implement various image processing tasks including edge detection. When all the memristors are programmed to the same resistance, the grid shows symmetric diffusion properties that can be applied to detect edges or contours of an input image.

An edge exists whenever a low input is neighbored by a high input, since an edge is defined where the discontinuity between the input voltages occur.

A simplified analysis on 1-D connection (see Fig. 7) is carried out to show that this structure can be used for edge detection. In the figure, $I_{n,m}$ is the input voltage level, $C_{n,m}$ is the center node voltage level, and $O_{n,m}$ is the output voltage level for the nth node.

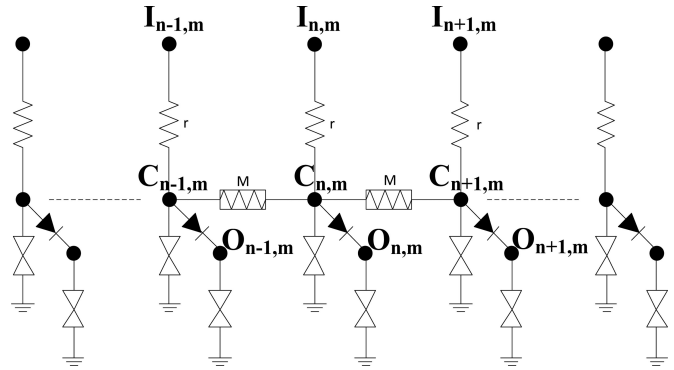

Fig. 7. Proposed circuitry in 1D case.

We assume that there is an edge between inputs $I_{n,m}$ and $I_{n+1,m}$. Thus, $I_{n,m}$ and input before it are high and $I_{n+1,m}$ and inputs after it are low (0 V for this analysis purposes).

We start our analysis by applying Kirchoff's Current law to nodes $C_{n,m}$ and $C_{n+1,m}$ to obtain the node voltages. Initially, the effects of the nearest neighborhood are ignored

$$\frac{I_{n,m}-C_{n,m}}{r}=\frac{C_{n,m}-C_{n+1,m}}{M}+\frac{C_{n,m}}{R_{\mathrm{RTD1}}} \tag{8}$$

where r is the input resistance, M is the resistance of the memristor, R_{RTD1} is the effective resistance of center node RTD, and $\frac{C_{n,m}}{R_{\mathrm{RTD1}}}$ indicates current through RTD. The current branch through the diode to the output nodes is also ignored, since the diode current is order of magnitude less until the center node voltage reaches the switching threshold at the output node. When the RTD current equation is inserted, the equation becomes

$$\begin{aligned}\frac{\left(I_{(n,m)}-C_{(n,m)}\right)}{r}&=\frac{\left(C_{(n,m)}-C_{(n+1,m)}\right)}{M}\\&+\frac{(qm*kT\Gamma)}{(4\pi^2\hbar^3)}\ln\left(\frac{\left(1+e^{\frac{(E_F-E_r+n_1qC_{n,m}/2)}{kT}}\right)}{\left(1+e^{\frac{(E_F-E_r+n_1qC_{n,m}/2)}{kT}}\right)}\right)\\&*\left(\frac{\pi}{2}+\arctan\right)\left(\frac{(E_r-n_1qC_{n,m}/2)}{\Gamma/2}\right)\\&+H\left(e^{n_2qC_{n,m}/kT}-1\right).\end{aligned} \tag{9}$$

Similarly, at node $C_{n+1,m}$:

$$\frac{C_{n,m}-C_{n+1,m}}{M}=\frac{C_{n+1,m}}{r}+\frac{C_{n+1,m}}{R_{\mathrm{RTD2}}} \tag{10}$$

$$\begin{aligned}\frac{C_{n,m}-C_{n+1,m}}{M}&=\frac{C_{n+1,m}}{r}\\&+\frac{(qm*kT\Gamma)}{(4\pi^2\hbar^3)}\ln\left(\frac{\left(1+e^{\frac{E_F-E_r+n_1qC_{n+1,m}/2}{kT}}\right)}{\left(1+e^{\frac{E_F-E_r+n_1qC_{n+1,m}/2}{kT}}\right)}\right)\\&*\left(\frac{\pi}{2}+\arctan\right)\left(\frac{E_r-n_1qC_{n+1,m}/2}{\Gamma/2}\right)\\&+H\left(e^{n_2qC_{n,m}/kT}-1\right).\end{aligned} \tag{11}$$

R_{RTD2} is the effective resistance of output node RTD. Equations (8) and (9) can be evaluated numerically to obtain the intermediate node voltages $C_{n,m}$ and $C_{n+1,m}$. Designing what these voltages will be is essential to obtain edge detection functionality. When there is an edge, the center node voltage $C_{n,m}$ should rise to disturb the detection node RTD.

Parameters should be picked such that

$$C_{n,m} > V_{\mathrm{threshold}} \text{ when } I_{n,m} = V_{\mathrm{high}}$$
$$C_{n+1,m} < V_{\mathrm{threshold}} \text{ when } I_{n+1,m} = V_{\mathrm{low}}(0\ \mathrm{V})$$
$$V_{\mathrm{threshold}} = V_d + V_{\mathrm{RTD}}$$

where V_d is the diode threshold, and V_{RTD} is the switching threshold of the RTD. In this way, the output $O_{n,m}$ will switch to high stable point, indicating there is an edge and $O_{n+1,m}$ will remain at low stable point.

If the effects of the nearing neigbors are considered, one can see that since $I_{n+2,m}$ is also low, the actual voltage on the node $C_{n+1,m}$ will be lower than the above calculated value, hence not violating the condition $C_{n+1,m} < V_{\mathrm{threshold}}$, but instead further helping to meet it. Similarly, $I_{n+1,m}$ helps node $C_{n,m}$ to be higher than $V_{\mathrm{threshold}}$.

In 2-D case, the state equation of the diffusion circuitry can be obtained as

$$\frac{I_{n,m} - C_{n,m}}{r} + \frac{C_{n-1,m} - C_{n,m}}{M} + \frac{C_{n,m-1} - C_{n,m}}{M} + \frac{C_{n+1,m} - C_{n,m}}{M} + \frac{C_{n,m+1} - C_{n,m}}{M} - I_{\mathrm{RTD}} = c\frac{ds_{n,m}}{dt} \tag{12}$$

where

$$I_{\mathrm{RTD}} = \frac{C_{n,m}}{R_{\mathrm{RTD}}} \tag{13}$$

and c is the parasitic capacitance of the RTD. Assuming RTD has a finite resistance, replacing (12) in (11)

$$I_{n,m} = C_{n,m} = \left(1 + \frac{4r}{M} + \frac{r}{R_{\mathrm{RTD}}}\right) - \frac{r}{M}(C_{n-1,m} + C_{n,m-1} + C_{n+1,m} + C_{n,m+1}) + rc\frac{dC_{n,m}}{dt}. \tag{14}$$

Taking Fourier transform, the transfer function is

$$H(f_m, f_n, f_t) = \frac{S(f_m, f_n, f_t)}{E(f_m, f_n, f_t)} = \frac{1}{\left(1 + \frac{4r}{M} + \frac{r}{R_{\mathrm{RTD}}}\right) - \frac{2r}{M}(\cos(2\pi f_m) + \cos(2\pi f_n)) + rc2\pi i f_t}. \tag{14}$$

As the RTD $I-V$ curve indicates, it acts as a positive variable resistor, indicating that the real part of the denominator of the transfer function is always positive.

B. Line Detection

Introducing anisotropy in the vertical and horizontal directions in the memristive grid allows the implementation of line detection. In order to detect lines, the center node voltages should be made a weaker function of the neighboring cells' center node voltages in one direction and a stronger function in the other.

For example, high resistance in the vertical, and low resistance in the horizontal direction limits the effects of the neighboring cells in the vertical direction and enables diffusion in the horizontal direction, which means the detection of lines in the horizontal direction.

Low resistance in the vertical and high resistance in the horizontal direction limits the effects of the neighboring cells in the horizontal direction and enables diffusion in the vertical direction, which means the detection of vertical lines.

In this case, the diffusion network state equation becomes

$$I_{n,m} = C_{n,m}\left(1 + \frac{2r}{M_{\mathrm{high}}} + \frac{2r}{M_{\mathrm{low}}} + \frac{r}{R_{\mathrm{RTD}}}\right) - \frac{r}{M_{\mathrm{high}}}(C_{n-1,m} + C_{n+1,m}) - \frac{r}{M_{\mathrm{low}}}(C_{n,m-1} + C_{n,m+1}) + rc\frac{dC_{n,m}}{dt} \tag{16}$$

where M_{high} is the resistance of the memristor when programmed to high, and M_{low} is the resistance of the memristor when programmed to low. The state equation is symmetrical for vertical and horizontal line detection cases.

V. Simulation Results

Simulation results verifying various diffusion configurations and demonstrating edge detection and line detection operations are presented in this section. Simulations are carried out on a 64 $\times$ 64 array. RTDs based on device characteristics shown in Fig. 1 as well as the memristor model from [7] are used. Simulations are carried out with nominal parameters to provide proof of concept. However, studies carried out in [12] show that resistive grid-based architectures are variation tolerant and can operate with nonoptimal values. Therefore, our proposed architecture is expected to be variation tolerant. For example, for the edge detection case, this tolerance depends on how much margin is left between the designed high/low voltages and the threshold.

The proposed memristive grid can support the different diffusion characteristics mentioned above. These characteristics are important in many image processing applications. Anisotropic diffusion can be used for edge extraction applications [5], anisotropic symmetrical diffusion characteristics can be used for line detection, and anisotropic asymmetrical diffusion characteristics can be used for motion detection [3].

Fig. 8 shows diffusion characteristics that can be realized in our proposed architecture. When all the memristors are programmed to the same resistance, isotropic diffusion [see Fig. 8(b)] is obtained. When horizontal memristors are programmed to low and vertical ones are programmed to high resistance, anisotropic symmetrical diffusion in horizontal direction [see Fig. 8(c)] is obtained. When the resistances of the horizontal and vertical memristors are programmed reverse with respect to the previous case, anisotropic symmetrical diffusion in vertical direction [see Fig. 8(d)] is achieved. Finally, when all

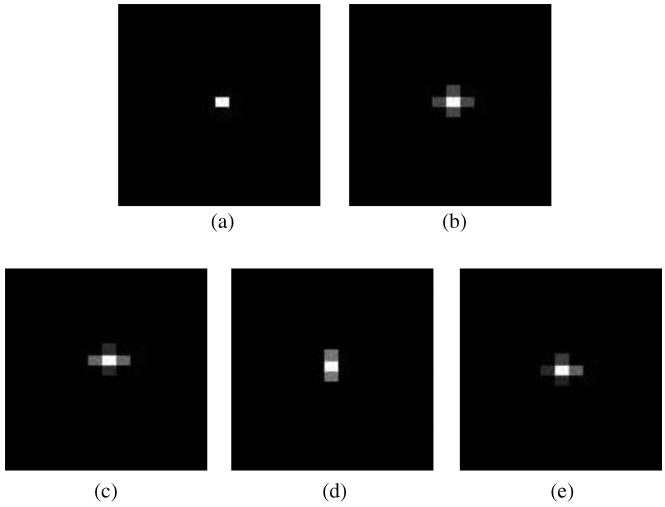

Fig. 8. Various diffusion characteristics that can be implemented in proposed architecture. (a) Input. (b) Isotropic Diffusion. (c) Anisotropic symmetrical diffusion in horizontal direction. (d) Anisotropic symmetrical diffusion in vertical direction. (e) Anisotropic asymmetrical diffusion.

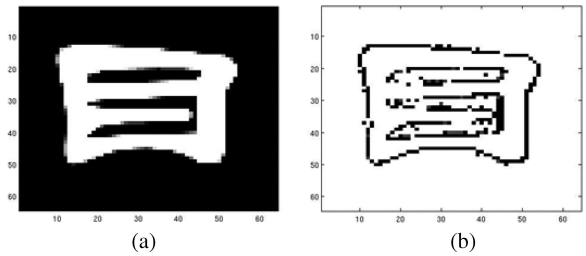

Fig. 9. (a) Edge detection sample input with irregular edges (b) Output result.

resistances are programmed to different resistances, anisotropic asymmetrical diffusion [see Fig. 8(e)] is achieved. These characteristics or combinations of them can be utilized to perform various vision tasks.

A. Edge Detection

Edge detection simulations are carried out on two types of input images: first, with irregular edges and intermediate pixel intensity values around the edges shown in Fig. 9; second, with regular edges and maximum pixel intensity difference around the edges shown in Fig. 10.

In edge detection mode, the proposed architecture is initiated with high and low input values corresponding to black and white pixels with scaled voltages in between corresponding to shades of gray. When the first input type shown in Fig. 9(a) is applied to the array, the edge pattern shown in Fig. 9(b) is observed at the $O_{n,m}$ nodes of the architecture. In Fig. 9(b), black lines correspond to high RTD voltage level and white lines correspond to low RTD voltage level on $O_{n,m}$ nodes.

The results of the above simulation suggest that edges are extracted with relative accuracy. In the regions with thicknesses of a few pixels or where the borders include shades of gray, discontinuities or jumps in the border lines are observed. However, the quality of the results can be improved by fine tuning grid resistance as well as RTD design parameters.

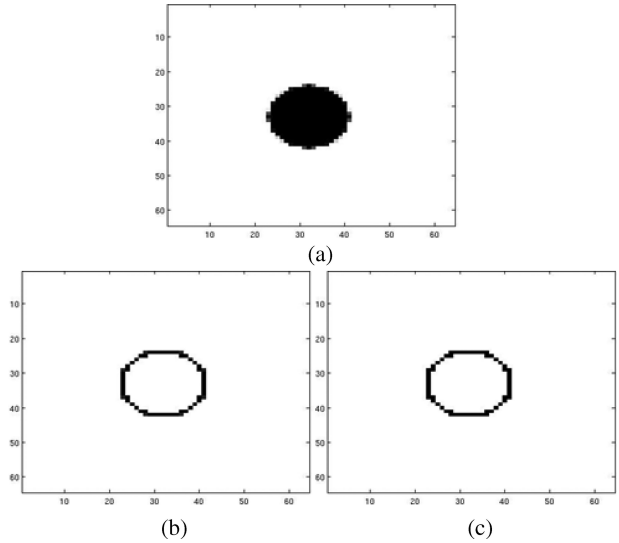

Fig. 10. (a) Edge detection sample 2 with regular edges (b) Output at 2.5 ns (c) Output at 100 ns.

The second set of sample filtering is provided in Fig. 10(a)–(c). The second type of image contains a circle with large continuous areas of the same color pixels. In this image, there is no region with several pixels thickness (except the outer line due to finite number of pixels available to represent a circle). The architecture is able to clearly outline the edges without any discontinuities.

Although the speed of operation can be tuned by scaling the current capabilities of the active devices, the above simulation shows that the results obtained after 2.5 ns into the operation are almost exactly the same as the results obtained after 100 ns.

Simulations on memristors using the model provided in [7] indicated that a read pulse of 100 ns duration and 2-V amplitude does not change the resistance of the memristor detectably, and a write process to the memristor usually takes in the range of a few seconds depending on the resistance to be encoded. Therefore, the architecture can perform the tuned operation repeatedly without detectably altering the tuning, thus minimizing (effectively eliminating) the need for a refresh operation.

The effect of component mismatch is more significant in input resistors (i.e., vertical resistors) compared to grid resistors (i.e., horizontal resistors) [12]. A set of simulation results are listed in Fig. 11(a)–(f), showing how the variation in input resistance changes edge detection results. Simulation results indicate that edges are detected less accurately when the input resistance deviates from the optimum value obtained through simulations. Exact resistance values depend on various circuit and device parameters such as voltage levels used, RTD and diode current characteristics. Therefore, the resistance variation is presented in percentages. Edge detection quality directly depends on the variation of the input resistance mainly due to two factors. The first factor is that input resistance changes the spatial frequency tuning of the architecture making it less sensitive to edges. The second factor is that large input resistances cause input voltage drops, thus putting the architecture off the operating region.

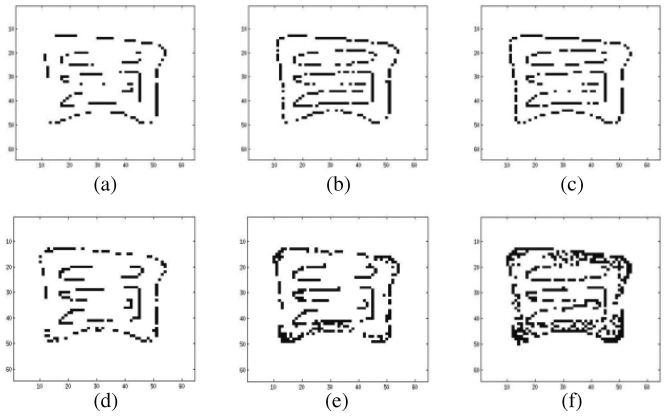

Fig. 11. Edge detection results with input resistance variation. (a) 50% resistance. (b) 75% resistance. (c) 100% resistance (nominal case). (d) 500% resistance. (e) 1000% resistance. (f) 2000% resistance.

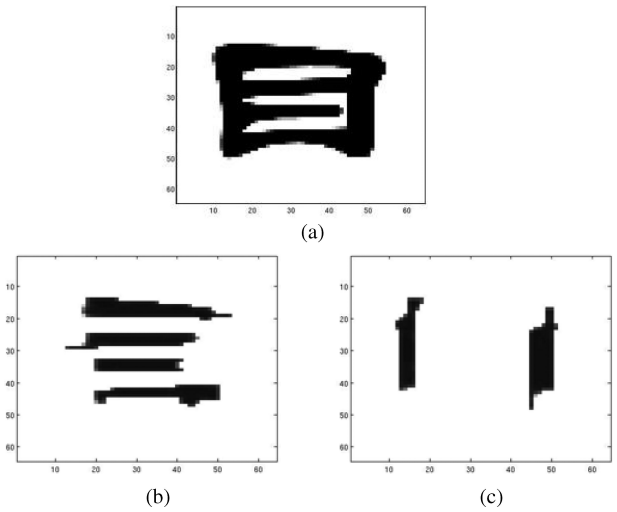

Fig. 12. (a) Line detection sample input. (b) Horizontal line detection. (c) Vertical line detection.

B. Line Detection

Fig. 12 shows line detection results. The plotted results are obtained 500 ns after system initialization.

When the image in Fig. 9(a) is considered, if we assume the black pixels represent the high voltages and white pixels represent the low voltages, the diffusion from high voltages to low voltages will result in the low voltages increasing and stabilizing at the higher voltage level. The results presented in Fig. 12 are inverted to clearly show the detected lines.

VI. Conclusion

An analog grid-based architecture incorporating memristor connections for programmability and RTDs for signal detection and latching is revealed. The architecture can be programmed to perform various image processing tasks. A method to change the resistive state of the memristors in an array configuration is also provided and demonstrated through circuit simulations.

Analytical models characterizing edge detection and line detection configurations are discussed and simulation results are provided to verify functionality. The simulation data establish that the proposed architectural configuration incorporating programmable analog resistive elements can be reused to perform a wide gamut of image processing functions at extremely high speeds.

References

[1] B. E. Shi and L. O. Chua, "Resistive grid image filtering: input/output analysis via the CNN framework," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 39, no. 7, pp. 531–548, Jul. 1992.

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

[3] A. B. Torralba, "Analogue architectures for vision cellular neural networks and neuromorphic circuits," Ph.D. dissertation, Dept. Elect. Eng, Grenoble Inst. Technol., Grenoble, France, 1999.

[4] W. H. Lee and P. Mazumder, "Motion detection by quantum-dots-based velocity-tuned filter," *IEEE Trans. Nanotechnol.*, vol. 7, no. 3, pp. 355–362, May 2008.

[5] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[6] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, pp. 80–83, May 2008.

[7] Z. Biolek, D. Biolek, and V. Biolkova, "SPICE model of memristor with nonlinear dopant drift," *Radioengineering*, vol. 18, pp. 210–214, June 2009.

[8] J. N. Schulman, H. J. D. L. Santos, and D. H. Chow, "Physics based RTD current–voltage equation," *IEEE Electron Device Lett.*, vol. 17, no. 5, pp. 220–222, May 1996.

[9] M. Hanggi and L. O. Chua, "Cellular neural networks based on resonant tunneling diodes," *Int. J. Circuit Theory Appl.*, vol. 29, no. 5, pp. 487–504, Sep./Oct. 2001.

[10] P. Mazumder, S. R. Li, and I. Ebong, "Tunneling-based cellular nonlinear network architectures for image processing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 4, pp. 487–495, Apr. 2009..

[11] T. Roska, "Analogic CNN computing: Architectural, implementation, and algorithmic advances-a review," *Proc. IEEE 5th Int. Workshop Cellular Neural Networks Their Appl.*, pp. 3–10, Apr. 14–17, 1998.

[12] B. E. Shi, "The effect of mismatch in current versus voltage-mode resistive grids," *Int. J. Circuit Theory Appl.*, vol. 37, no. 1, pp. 53–65, Feb. 2009..

[13] Y. Yilmaz and P. Mazumder, "Threshold read method for multi-bit memristive crossbar memory," *Proc. Int. Symp. Electron. Syst. Des.*, pp. 217–222, Dec. 19–21, 2011.

[14] I. Ebong, D. Deshpande, Y. Yilmaz, and P. Mazumder, "Multi-purpose neuro-architecture with memristors," *Proc. IEEE 11th Conf. Nanotechnol.*, pp. 431–435, Aug. 15–18, 2011.

[15] H. Li, X. Liao, C. Li, H. Huang, and C. Li, "Edge detection of noisy images based on cellular neural networks," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 16, no. 9, pp. 3746–3759, Sep. 2011..

[16] J. Zhao, H. Wang, and D. Yu, "A new approach for edge detection of noisy image based on CNN," *Int. J. Circuit Theory Appl.*, vol. 31, no. 2, pp. 119–131, Feb. 2003..

[17] T. Yoshida, J. Kawata, T. Tada, A. Ushida, and J. Morimoto, "Edge detection method with CNN," *Proc. SICE Annu. Conf.*, vol. 2, pp. 1721–1724, Aug. 4–6, 2004.

[18] I. N. Aizenberg, "Processing of noisy and small-detailed gray-scale images using cellular neural networks," *J. Electr. Imaging*, vol. 6, no. 3, pp. 272–285, Jul. 1997.

[19] A. Gelencsér, T. Prodromakis, C. Toumazou, and T. Roska, "Biomimetic model of the outer plexiform layer by incorporating memristive devices," *Phys. Rev. E*, vol. 85, no. 4, pp. 041918-1–041918-10, Apr. 2012.

[20] L. O. Chua, "Memristor-the missing circuit element," *IEEE Trans. Circuit Theory*, vol. 18, no. 5, pp. 507–519, Sep. 1971.

[21] Y. N. Joglekar and S. J. Wolf, "The elusive memristor: Properties of basic electrical circuits," *Eur. J. Phys.*, vol. 30, pp. 661–675, 2009.

[22] K. Kim, S. H. Jo, S. Gaba, and W. Lu, "Nanoscale resistive memory with intrinsic diode characteristics and long endurance," *Appl. Phys. Lett.*, vol. 96, no. 5, pp. 053106-1–053106–3, Feb. 2010.

Yalcin Yilmaz received the B.S and M.S. degrees in electrical engineering from the University of Michigan, Ann Arbor, USA, in 2009 and 2011, respectively, where he is currently working toward the Ph.D. degree.

He is currently a Graduate Student Research Assistant in the Electrical Engineering and Computer Science Department at the University of Michigan, Ann Arbor, USA. His current research interests include the modeling, simulation, low-power digital and analog circuit designs for emerging technologies including spin-based devices, resonant tunneling diodes, and memristors.

Pinaki Mazumder (F'99) received the B.S. degree in electrical engineering from the Indian Institute of Science, Bangalore, Karnataka, and the B.Sc. Physics (Hons.) degree from Guwahati University, Guwahati, Assam, India, the M.S. degree in computer science from the University of Alberta, AB, Canada, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, USA, in 1988.,

He is currently, a Professor of electrical engineering and computer science at the University of Michigan, Ann Arbor, USA, where he has been teaching for the past 25 years. He spent 3 years at National Science Foundation serving as the lead Program Director of Emerging Models and Technologies Program in the CISE Directorate as well as leading the Quantum, Molecular and High Performance Simulation Program in the Engineering Directorate. He had worked for 6 years in industrial R&D laboratories which included AT&T Bell Laboratories in USA and Bharat Electronics, Ltd., in India. He spent his sabbatical at Stanford University, University of California at Berkeley, and NTT Center Research Laboratory in Japan. He has published more than 260 technical papers and 4 books on various aspects of VLSI technology.