SCIENCE CHINA
Information Sciences

• Supplementary File •

A memristor model for high-precision dynamic conductance and its application in neuromorphic computing

Jinhong HU[1], Siyuan SHEN[1], Wen LI[1], Ai CHEN[1], Jiening WU[1], Zhenqi LIU[1], Rui YUAN[1*], Shukai DUAN[1,2,3,4,5*] & Lidan WANG[1,2,3,4,5*]

[1]*College of Artificial Intelligence, Southwest University, Chongqing, 400715*
[2]*National & Local Joint Engineering Research Center of Intelligent Transmission and Control Technology, Chongqing, 400715, China*
[3]*Chongqing Key Laboratory of Brain-inspired Computing and Intelligent Chips, Chongqing, 400715, China*
[4]*Key Laboratory of Luminescence Analysis and Molecular Sensing (Southwest University), Ministry of Education, Chongqing, 400715, China*
[5]*Chongqing Key Laboratory of Brain-inspired Computing and Intelligent Chips, Chongqing, 400715, China*

Appendix A Memristor device fabrication and testing

The memristor device structure presented in this paper is a Ti/Pt/Ta_2O_5/Ta/Pt cross-array with a multilayer stack, Figure A1 shows a SEM image of the device. Ta_2O_5 devices were constructed on SiO_2/Si substrates through the following process. Initially, the adhesion layer and bottom electrode were patterned via lithography, with Ti (10 nm) and Pt (30 nm) deposited by magnetron sputtering and subsequently stripped off. Next, the functional layer was patterned using photolithography, and Ta_2O_5 (30 nm) was deposited via magnetron sputtering and stripped. Finally, the top electrode was patterned through photolithography again, and Ta (10 nm) along with Pt (30 nm) were deposited by magnetron sputtering before being stripped off.

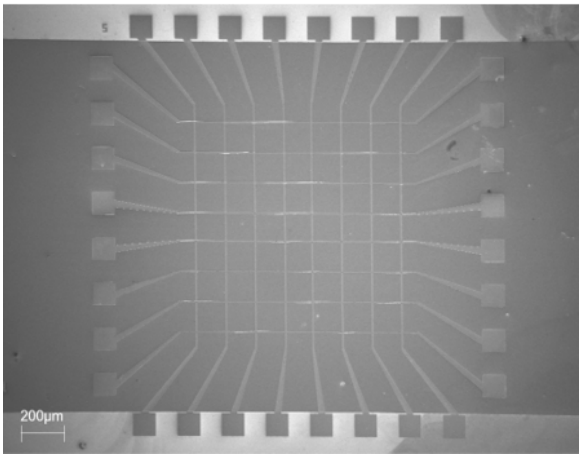

Figure A1 SEM image.

Devices with this structure are capable of stable voltage-current cycling without the need for the forming process. Figure A2(b) and Figure A2(c) shows the voltage-time and current-time curves during the test, with set threshold voltage at 0.47 V and reset threshold voltage at -0.40 V. Its resistance capable of transitioning from a low resistance state of 820Ω to a high resistance state of 49kΩ. This transition results in a switching ratio that exceeds 50. During the voltage-current cycling tests, the memristor device under test was first secured on a precision probe station to ensure stable positioning throughout the measurement. High-precision probes were then used to contact the top and bottom electrodes of the device, establishing reliable electrical connections. Subsequently, a semiconductor parameter analyzer (KEYSIGHT B1500A) was employed to apply a stable voltage signal while simultaneously monitoring the device current in real time, allowing evaluation of its voltage-current characteristics and cycling stability. For further measurements of voltage-time and current-time curves, the semiconductor parameter analyzer served as the signal source, with the device connected to a high-precision digital oscilloscope (RIGOL MSO8064) to record the dynamic response during voltage application.

* Corresponding author (email: yuanruiswu@swu.edu.cn, duansk@swu.edu.cn, ldwang@swu.edu.cn)

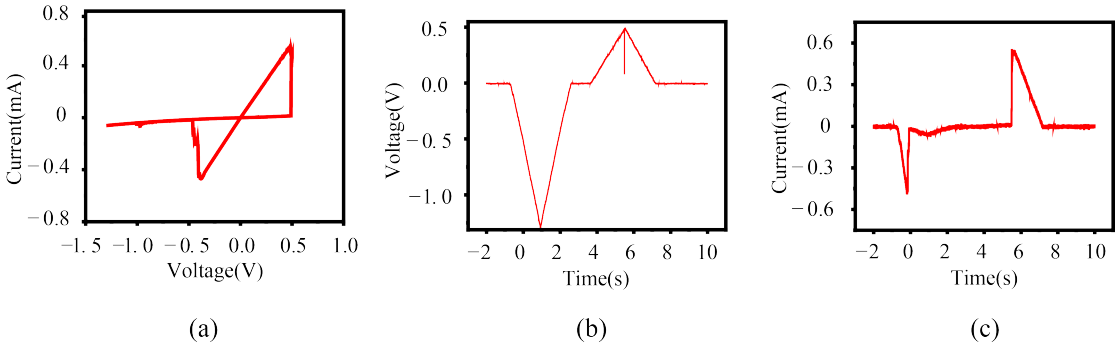

Figure A2 (Color online) (a) The I-V curve of the fabricated device; (b) Voltage-time curve of applied voltage during testing; (c) The corresponding response current I-t curve.

Appendix B Parameter extraction for memristor models

Based on the experimentally measured I-V hysteresis loops, we first perform parameter extraction for the device's conduction behavior under stable resistance states. For the high-conductance state (HCS), the curve segment after a stable SET operation is selected. Using nonlinear least-squares regression, the optimal fitting parameters are determined as $g_{max} = 1.21 \times 10^{-3}$ and $g_{max,flu} = -3.22 \times 10^{-5}$. Similarly, for the low-conductance state (LCS), the I-V segment following the RESET operation is analyzed. Given the significant nonlinear fluctuations in this region, a higher-order polynomial fit is employed to ensure modeling accuracy. The resulting key parameters are the baseline conductance $g_{min} = 1.65 \times 10^{-5}$ and the fluctuation coefficients $g_{min,flu1} = -2.95 \times 10^{-5}$ and $g_{min,flu2} = 8.34 \times 10^{-5}$.

The threshold voltages, V_p and V_n, are calibrated based on the critical switching points of the device, physically corresponding to the extrema of the current change rate ($\Delta I/\Delta V$). As illustrated in Figure B1(a), a prominent peak appears in the differential conductance curve at 0.47 V during the forward voltage sweep. This indicates that the growth rate of the conductive filaments reaches its maximum, leading to a rapid transition from the high-resistance state (HRS) to the low-resistance state (LRS); thus, the forward threshold voltage is defined as $V_p = 0.47$V. Correspondingly, during the reverse sweep shown in Figure B1(b), the peak of $\Delta I/\Delta V$ occurs at -0.4V, marking the dominant moment of the filament dissolution process. To maintain the positive definiteness of the model parameters, the absolute value of this voltage is taken as the reverse threshold parameter, i.e., $V_n = 0.4$V.

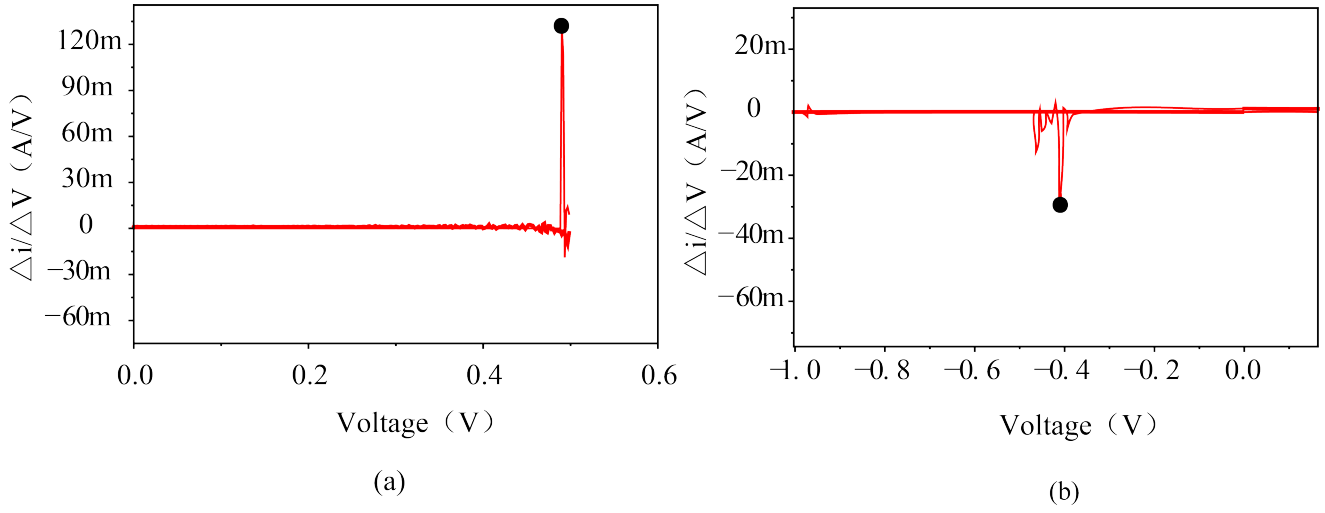

Figure B1 (Color online) (a) Forward current change rate curve; (b) Reverse current change rate curve.

Parameters A_p and A_n are determined by equations (B1) and (B2), respectively:

$$A_p = \frac{g_{pk,p}}{g_{max} - g_{min}}, \tag{B1}$$

$$A_n = \frac{g_{pk,n}}{g_{max} - g_{min}}. \tag{B2}$$

In the description of the model's dynamic characteristics, the parameters $g_{pk,n}$ and $g_{pk,p}$ play a crucial role by quantifying the extreme transient switching kinetics during the RESET and SET processes, respectively. $g_{pk,n}$ is defined as the minimum value of the time derivative of conductance ($\Delta g/\Delta t$) during the transition from the low-resistance state (LRS) to the high-resistance state (HRS), which physically corresponds to the moment of the most intense conductive filament rupture and the most rapid resistance increase.

Conversely, $g_{pk,p}$ represents the maximum rate of conductance change during the HRS-to-LRS transition, characterizing the peak response speed at the instant of conduction channel reconnection. Here, g_{max} and g_{min} serve as the physical saturation boundaries, defining the dynamic range of conductance evolution. These parameters are extracted through differential analysis of the transient conductance curves. As shown in Figure B2(a) and (b), the characteristic peaks at the switching instants are clearly observable by plotting the evolution of the conductance change rate over time. By direct measurement of these extrema, we obtain $g_{pk,n} = -13.618$ and $g_{pk,p} = 19.815$. The parameters α_p and α_n are assigned as empirical values during the extraction process; in this study, they are set to $\alpha_p = 1$ and $\alpha_n = 0.5$, respectively.

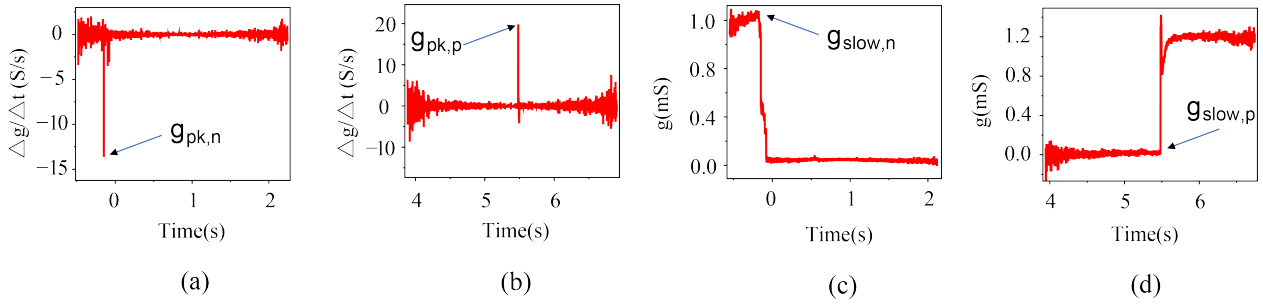

Figure B2 (Color online) (a) Time evolution curve of the rate of change of conductivity during the transition from the low-resistance state to the high-resistance state; (b) Time evolution curve of the rate of change of conductivity during the transition from the high-resistance state to the low-resistance state. (c) $g - t$ curve from low-resistance state to high-resistance state; (d) $g - t$ time curve from high-resistance state to low-resistance state.

The boundary values x_p and x_n are determined by equations (B3) and (B4):

$$x_p = \frac{g_{slow,p} - g_{min}}{g_{max} - g_{min}}, \tag{B3}$$

$$x_n = \frac{g_{slow,n} - g_{min}}{g_{max} - g_{min}}. \tag{B4}$$

The model introduces two parameters, $g_{slow,n}$ and $g_{slow,p}$, to calibrate the critical conductance values corresponding to the maximum switching rates during the dynamic transition processes. Physically, these parameters define the device state at the moment of the most intense conductive filament growth or rupture, serving as crucial anchors connecting steady-state boundaries with transient kinetics. These parameters can be extracted from the conductance-time (g-t) transient response curves shown in Figure B2(c) and (d). By observing the curve profiles, the characteristic conductance values for the RESET and SET processes can be directly identified as $g_{slow,n} = 1.05 \times 10^{-3}$ and $g_{slow,p} = 3.30899 \times 10^{-5}$, respectively. However, in practical parameter extraction, precisely locating these critical points through visual inspection is challenging due to the smooth nonlinear gradients in the g-t transition regions. To address this, an auxiliary confirmation method is employed: first, the exact time of the maximum rate of change (t_{peak}), corresponding to the extrema of $\Delta g/\Delta t$, is identified from the conductance change rate curves (Figure B2(a) and (b)). Subsequently, this t_{peak} is mapped back to the g-t evolution curve to determine the corresponding conductance value. By leveraging the sensitivity of differential signals to state changes, this method significantly enhances the objectivity and accuracy of the $g_{slow,n}$ and $g_{slow,p}$ values.

The terms $w_p(x, x_p)$ and $w_n(x, x_p, x_n)$ represent the window functions, which are determined by equations (B5) and (B6):

$$w_p(x, x_p) = \frac{\beta_p(1 - x)}{1 - x_p}, \tag{B5}$$

$$w_n(x, x_p, x_n) = \frac{\beta_n x}{1 - x_n + x_p}, \tag{B6}$$

The parameters β_p and β_n serve as buffering factors that optimize the I-V characteristics during the resistance transition phase by modulating the window functions. Specifically, through the adjustments in equations (B5) and (B6), they effectively mitigate abrupt changes in the I-V curves during switching, resulting in smoother and more continuous transitions. This modulation helps suppress sudden current fluctuations and instabilities during resistance state changes. These factors are determined empirically; in this model, they are set to $\beta_p = 0.5$ and $\beta_n = 1$.

Appendix C Model comparison

To validate the advantages of this memristor model, we compared it with Yakopcic's model, two widely used generalized memristor models, and two recent models. The verification was conducted using the I-V curves obtained from the experiments shown in Figure A2(a). The specific parameters for the six model configurations included in the comparison are shown in Table C1.

To quantitatively evaluate the fitting accuracy of various memristor models against experimental data, the Root Mean Square Error

Table C1 Parameter settings for the proposed model and other comparative models.

Model Name	Parameters and Values
Proposed Model	$g_{max} = 1.21 \times 10^{-3}$, $g_{max,flu} = -3.22 \times 10^{-5}$, $g_{min} = 1.65 \times 10^{-5}$, $g_{min,flu1} = -2.95 \times 10^{-5}$, $g_{min,flu2} = 8.34 \times 10^{-5}$, $V_p = 0.47$, $V_n = -0.40$, $A_p = 16601.87$, $A_n = -11409.76$, $x_p = 0.0139$, $x_n = 0.8659$, $\alpha_p = 1$, $\alpha_n = 0.5$, $\beta_p = 0.5$, $\beta_n = 1$
Yakopcic's [1]	$g_{max} = 1.21 \times 10^{-3}$, $g_{min} = -3.3 \times 10^{-3}$, $b = -6.57 \times 10^{-3}$, $V_p = 0.47$, $V_n = -0.40$, $A_p = 48256.15$, $A_n = -358.68$, $x_p = 0.0139$, $x_n = 0.8659$
Ion Concentration [2]	$\gamma = 8 \times 10^{-6}$, $\delta = 1$, $\alpha = 1.15 \times 10^{-2}$, $\beta = 0.1$, $\mu_1 = 30$, $\mu_2 = 30$, $k = 5 \times 10^{-4}$
VTEAM [3]	$R_{on} = 820$, $R_{off} = 4.7 \times 10^4$, $\omega_{on} = 0$, $\omega_{off} = 25 \times 10^{-9}$, $k_{on} = -0.018$, $k_{off} = 10^{-5}$, $\alpha_{on} = 18$, $\alpha_{off} = 2$, $V_{on} = -0.47$, $V_{off} = 0.4$
B_{mod} [4]	$k_1 = 7.23 \times 10^3$, $k_2 = -121$, $m = 3$, $v_{thr} = 0.47$, $G_{on} = 1.21 \times 10^{-3}$, $G_{off} = 1.646 \times 10^{-5}$, $x_0 = 0.034$
B_{6m} [5]	$R_{on} = 820$, $R_{off} = 4.7 \times 10^4$, $k = 15000$, $m = -520.1$, $v_{thr} = 0.47$, $\eta = 1$, $x_0 = 0.027$

(RMSE) is introduced as the core performance metric. The RMSE effectively measures the degree of deviation between simulation results and measured data; its mathematical expression is given in equation (C1):

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\left(\frac{\sum_{i=1}^{N}(V_{\mathrm{MODEL},i} - V_{\mathrm{ref},i})^2}{\bar{V}_{\mathrm{ref}}^2} + \frac{\sum_{i=1}^{N}(I_{\mathrm{MODEL},i} - I_{\mathrm{ref},i})^2}{\bar{I}_{\mathrm{ref}}^2}\right)}, \tag{C1}$$

In the error calculation equation, N represents the total number of samples used for statistical evaluation, covering both the experimental I-V data and the model-generated simulation results. To ensure that the point-to-point mapping during error calculation is physically meaningful, the discrepancy in sampling rates between experimental and simulated data must be addressed. Typically, the extremely small computation steps in the simulation environment result in a significantly larger volume of model data compared to the points captured by experimental equipment. Consequently, this study implements a linear uniform downsampling strategy on the high-density model data. Rather than simple truncation, this strategy extracts characteristic points from the model output at equal intervals based on the experimental data indices, thereby minimizing data redundancy while preserving the nonlinear evolution characteristics of the I-V curves in the transition regions. In the error evaluation function (C1), V_{MODEL} and I_{MODEL} are the aligned simulation voltage and current vectors, while V_{ref} and I_{ref} serve as the experimental reference values. The terms $\bar{V}_{ref}$ and $\bar{I}_{ref}$ denote the Euclidean norms of the reference voltage and current, introduced to eliminate the influence of physical dimensions and numerical magnitudes on error weighting. In calculating the fitting errors, we first evaluated the global RMSE for each model.However, for memristors, drastic transitions in resistance states typically occur over extremely short timescales, accompanied by significant current fluctuations. Traditional global error evaluation is often biased by the high proportion of data points in the stable high- and low-resistance regions, which can mask the true performance of the model in critical switching zones. To address this, we introduce an event-driven centric local sampling strategy to precisely quantify the capture accuracy during key transition processes. First, a differential discrimination method is used to locate the dynamic singularities where forward SET and reverse RESET occur in the simulation sequence. Subsequently, taking this instantaneous moment as the central axis, 62 characteristic sampling points are extended symmetrically in both directions to construct a high-dynamic evaluation window containing 125 data points. This window fully covers the entire process of conductive filament nucleation, growth, and breakdown (or rupture). The results demonstrate that our model achieves an RMSE of less than 3% within this transition interval. This significant accuracy advantage not only quantitatively proves the numerical stability of the model but also qualitatively validates that the carrier transport equations and conductive filament evolution dynamics constructed in this work can more realistically reproduce the internal microscopic physical evolution of the device, establishing a highly reliable foundation for subsequent applications.

Appendix D LTspice realization of the model

To extend the application of our model to more scenarios, we have also implemented it in LTspice, the specific implementation code is provided in Table D1. The principle is shown in Figure D1(a).

After constructing the LTspice circuit, we compared its I-V characteristics with the experimental data, as shown in Figure D1(b). The current response after voltage application exhibited only minor deviations from the test data, indicating that the SPICE circuit has successfully implemented our model.

Appendix E Application in neuromorphic computing

In this section, the dataset used is a publicly available Chinese human voice dataset for digits 0-9 (https://gitcode.com/open-source-toolkit/b2012?utm_source=tools_gitcode&index=top&type=card&&isLogin=1).

The dataset is segmented into two distinct components: the training set and the test set. The training set consists of 500 samples

Table D1 The LTspice sub-circuit code for the proposed model.

SPICE Subcircuit Description

```
.subckt memristor TE BE X
.param gmax=0.00121 gmaxflu=-0.00003224 gmin=0.0000164644
 gminflu1=-0.00002951 gminflu2=0.00008343 Vp=0.47 Vn=0.4
 Ap=16601.87342 An=11409.75585 xp=0.013933257 xn=0.865945004
 x0=0 eta=1 alphap=1 alphan=0.5 betap=0.5 betan=1
.func wp(V) betap*(1-V)/(1-xp)
.func wn(V) betan*V/(1-xn+xp)
.func Gp(V) IF(V > Vp, Ap*(exp(V)-exp(Vp)), 0)
.func Gn(V) IF(V< -Vn , -An*(exp(-V)-exp(Vn)),0)
.func G(V) (Gp(V) + Gn(V))
.func Fp(V2) IF(V2 >= xp, alphap*exp(-(V2-xp))*wp(V2),1)
.func Fn(V2) IF(V2 <= (1-xn),alphan*exp(V2+xn-1)*wn(V2),1)
.func F(V1, V2) IF(eta*V1 >= 0,Fp(V2),Fn(V2))
.func IVRel(V1,V2) (gmax*V1+gmaxflu*V1*V1)*V2+
 (gmin*V1+gminflu1*V1*V1+gminflu2*V1*V1*V1)*(1-V2)
C X 0 {1}
.ic V(X)=x0
Gx 0 X value={eta*F(V(TE,BE),V(X,0))*G(V(TE,BE))}
G TE BE value={IVRel(V(TE,BE),V(X,0))}
.ends memristor
```

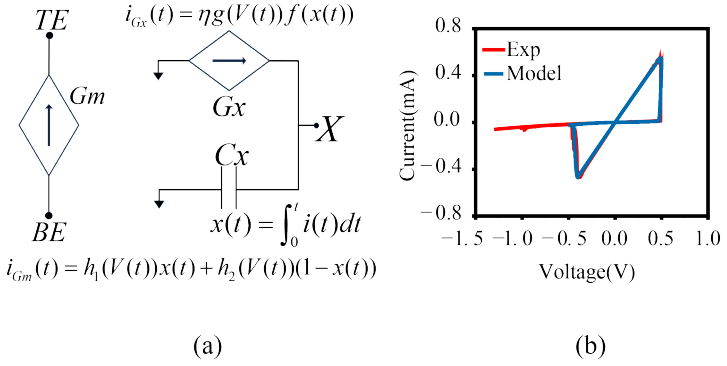

Figure D1 (Color online) (a) The model of SPICE function is illustrated, with the left half of the circuit representing the memristor model and the right half showing the acquisition of state variables; (b) The comparison between the results of LTspice simulation and the test data.

for each digit from 0 to 9, totaling 2500 samples, while the test set contains 50 samples for each digit, totaling 500 samples. In the classification task simulation, the memristor functions as an electronic synapse, physically storing the neural network weights via its non-volatile conductance states. Consequently, the weight updates rely on the LTP and LTD behaviors based on the memristor model, as shown in Figure E2(d). In our model, applying 40 consecutive positive pulses with a width of 100 ns and an amplitude of 0.471 V, immediately followed by 60 consecutive negative pulses with a width of 100 ns and an amplitude of -0.4004 V, induces LTP and LTD. The dataset needs to be preprocessed before classification. In this study, we employ mel frequency cepstral coefficients (MFCC) as the methodology for feature abstraction from speech signals. MFCC is a technique with extensive application in machine learning. It is a frequency-domain cepstral parameter derived from the Mel scale, which characterizes the nonlinear perception of frequency by the human ear.

During the simulation training, we apply a weight mapping method to map the training weights onto our memristor model, and update the weight values during backpropagation using the methods in equations (E2) and (E3). In this process, the training weights are initially quantized using uniform quantization

$$W_q = \begin{cases} W_n, & (w - W_n) < (W_{n+1} - w) \\ W_{n+1}, & (w - W_n) > (W_{n+1} - w), \end{cases} \tag{E1}$$

where W_q is the quantized weight and W_n and W_{n+1} are the boundary values of the small interval (W_n, W_{n+1}), respectively. If the weight value is relatively close to W_n, it is quantized to W_n; conversely, if it is closer to W_{n+1}, the weight is quantized to W_{n+1}. Using this quantization method, the training weights after quantization maintain a distribution that is roughly the same as the distribution

of the training weights before quantization. In our experiments, the 64-bit precision weights are mapped to 200 analog weights using the method in equation (E1). Each weight value is denoted by a pair of memristors, which store the positive and negative parts of the weight, respectively.

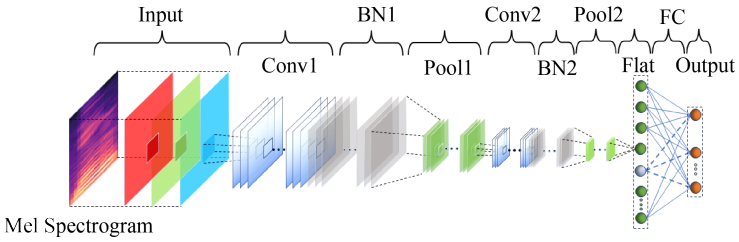

Figure E1 (Color online) Schematic diagram of the network structure used in the simulation experiment.

Table E1 Hyperparameters for each layer in the network

Serial number	Layer type	Input dimension	Output dimension
1	Conv	72×72×3	70×70×12
Convolutional kernel 3×3×3×12, step size 1, no padding			
2	BN	70×70×12	70×70×12
3	Max-pooling	70×70×12	24×24×12
Pooling window 3×3 with a step size of 3			
4	Conv	24×24×12	22×22×18
Convolutional kernel 3×3×3×18, step size 1, no padding			
5	BN	22×22×18	22×22×18
6	Max-pooling	22×22×18	11×11×18
Pooling window 2×2 with a step size of 2			
7	Flat	11×11×18	2178×1
8	Dropout	2178×1	2178×1
Dropout rate: 0.25			
9	FC	2178×1	10×1
Output category: 10			

During the mapping process, the weight values are encoded as the number of pulses issued, and the final conductance value of the memristor is determined by the cumulative effect of these pulses. The final conductance value of the memristor depends on the number of pulses applied to it. The process of applying pulses to memristors is as follows: first, define the range of weight values and evenly divide it into n intervals, where n represents the total number of analog weight levels. In this study, the weight range is set between [-1, 1], with any weight exceeding this range replaced by the nearest boundary value. Starting from a weight of 0, labels are assigned to each interval, extending in both positive and negative directions. For example, the analog weight at 0 is labeled as 0, while the first interval in the positive direction is labeled as +1, representing the corresponding analog weight. The labels for all other intervals, whether in the positive or negative direction, follow the same principle. During conductance updates, a positive pulse is imposed to the memristor labeled +1. The number of pulses sent to each memristor, along with their polarity (positive or negative), depends on the size of the label and the sign of the corresponding analog weight.

The weight used in backpropagation for updating the conductance is

$$W_{update}^{n} = -\alpha\left(\frac{dE_n}{dW_{i,j}^{n}} + \frac{\beta}{2}\sigma\right) + M, \tag{E2}$$

where α is the learning rate, $\frac{\beta}{2}\sigma$ is the regularization parameter, and M is the computational parameter of the weight update process. In our experiment, the momentum

$$M_{i,j} = \begin{cases} -\alpha\left(\frac{dE_n}{dW_{i,j}^n} + \frac{\beta}{2}\sigma\right), & \left|W_{update}^{n-1}\right| > W_{threshold} \\ W_{update}^{n-1}, & \left|W_{update}^{n-1}\right| < W_{threshold}. \end{cases} \tag{E3}$$

The weight threshold $W_{threshold}$ is introduced to avoid frequent tiny adjustments of the weights in the memristor neural network.

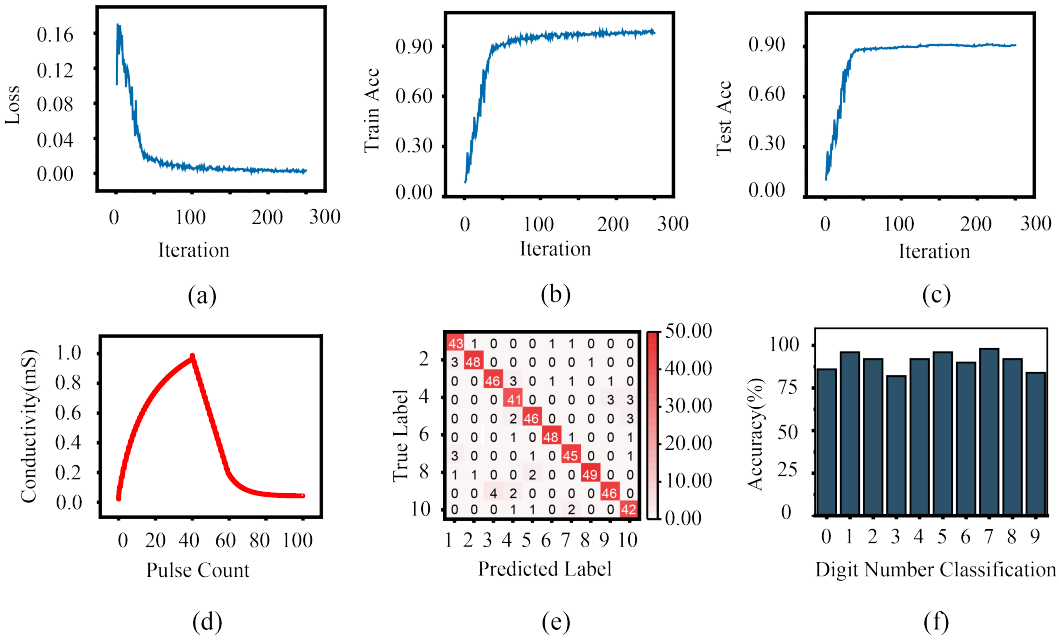

Figure E2 (Color online) (a) Loss; (b) Training accuracy; (c) Testing accuracy; (d) The LTP and LTD curves generated after applying continuous positive and negative pulses, where the positive pulse amplitude is 0.471V and the negative pulse amplitude is -0.4004V; (e) Confusion matrix; (f) The classification results for each digit from 0 to 9.

If the weight update variable is less than the threshold, it is used as the momentum of the weight update at the next iteration. When the predefined threshold is reached, the next momentum update will be determined by the accumulation of the current weight gradient and the regularization parameter. The adopted network architecture is shown in Figure E1, with detailed hyperparameters for each layer listed in Table E1.

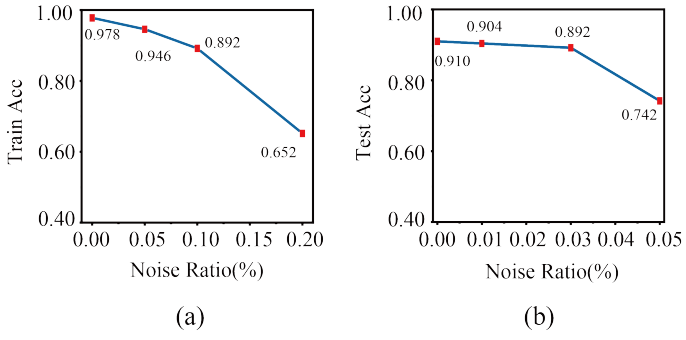

Figure E3 (Color online) (a) Training accuracy at read noise levels of 0%, 5%, 10%, and 20%; (b) Test accuracy at noise levels of 0%, 1%, 3%, and 5%.

As shown in Figure E2(a), Figure E2(b) and Figure E2(c), the simulation results indicate that the training accuracy reached 99.4%, demonstrating that the network effectively learned the features of the training dataset. The testing accuracy was as high as 91.6%, highlighting the strong generalizability of the network to unseen data. The confusion matrix is shown in Figure E2(e).

To further evaluate the performance of the model on individual digits, Figure E2(f) presents the recognition accuracy for each digit in the test set. The accuracy varied across digits, with the lowest being 82% and the highest reaching 98%. This variation can be attributed to factors such as the inherent difficulty in distinguishing certain digits and the sensitivity of the model to minor variations in the input data. Nevertheless, the overall recognition performance exhibited remarkable consistency, underscoring the robustness and

reliability of the proposed model in digit recognition tasks.

Given that memristors inevitably face inherent physical randomness in edge hardware deployments, we systematically evaluated the model in two critical stages: write noise during online training and read noise during offline inference. To address the programming variability caused by ion transport randomness during resistive memory's conductance update process, we introduced write noise following a zero-mean Gaussian distribution into the weight update step of backpropagation. We set standard deviation ratios of 5%, 10%, and 20% to simulate different operating conditions ranging from precise devices to highly variable ones. Experimental results are shown in Figure E3(a): At noise intensities of 5% and 10%, the network maintained accuracies of 94.6% and 89.2%, respectively. Even under extreme 20% noise conditions, the model converged to 65.2% accuracy, demonstrating the system's fundamental feature learning capability in harsh hardware environments. To account for potential steady-state effects such as conductance drift, thermal noise, and random telegraph noise (RTN) in devices, we incorporated read noise into the forward propagation calculations. Considering the lack of self-correction mechanisms during inference, the system exhibits heightened sensitivity to accuracy. Therefore, we tested noise levels of 1%, 3%, and 5%. The test results are shown in Figure E3(b). Under the aforementioned noise levels, the model achieved classification accuracies of 90.4%, 89.2%, and 74.2%, respectively. These results provide compelling evidence that our system maintains stable reasoning performance even under non-ideal conditions.

References

1 Yakopcic C, Taha T M, Mountain D J, et al. Memristor model optimization based on parameter extraction from device characterization data. IEEE Trans Comput Aided Des Integr Circuits Syst, 2019, 39(5): 1084–1095
2 Chang T, Jo S H, Kim K H, et al. Synaptic behaviors and modeling of a metal oxide memristive device. Appl Phys A, 2011, 102(4): 857–863
3 Kvatinsky S, Ramadan M, Friedman E G, et al. VTEAM: A general model for voltage-controlled memristors. IEEE Trans Circuits Syst II Express Briefs, 2015, 62(8): 786–790
4 Kirilov S. A simple memristor model for memory devices. 2024 13th Int Conf Mod Circuits Syst Technol (MOCAST), 2024: 1–4
5 Kirilov S, Mladenov V. LTSPICE Memristor Neuron with a Modified Transfer Function Based on Memristor Model with Parasitic Parameters. Electronics, 2025, 14(23): 4645
6 Hanifa R M, Isa K, Mohamad S. A review on speaker recognition: Technology and challenges. Comput Electr Eng, 2021, 90: 107005
7 Guo M J, Duan S K, Wang L D. Pulse coding off-chip learning algorithm for memristive artificial neural network. Chin Phys B, 2022, 31(7): 078702