

Homeostatic Dendritic Neuron Based on Co-Integrated Volatile and Non-Volatile Memristors for Neuromorphic Processing

Corresponding Author: Professor Yuchao Yang

This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

The manuscript introduces, for the first time, a hardware implementation that co-integrates spiking neurons with intrinsic homeostatic regulation and dendritic delay pathways using memristive devices. The use of volatile VO₂ for neuron circuits and non-volatile RRAM for programmable delays represents a novel device-level approach to emulating complex neuronal behaviors. The authors present a functioning prototype: a 32×18 RRAM array integrated with VO₂ neuron circuits on a custom PCB. VO₂ devices show stable switching over 500 cycles, and RRAM cells reliably support multi-level conductance states. The neuron circuit achieves a compact footprint (~76.8 μm²) and low energy consumption (~19.3 pJ/spike), indicating strong engineering optimization. The system is evaluated on two distinct datasets. On the CWRU fault diagnosis task, the architecture achieves ~92.9% accuracy—a 5.84% improvement over baseline. On the GSC keyword spotting task, dendritic circuits alone yield a ~1.56% gain, while combining dendrites with homeostatic neurons results in ~86.6% accuracy (+3.5%). The inclusion of an ablation study across four network variants strengthens the attribution of performance gains. The architecture supports sparse, event-driven processing, with real-time inference managed via FPGA coordination. This hybrid pipeline demonstrates potential for edge computing applications. The manuscript is well-organized, with clear figures and thorough explanations. Mathematical modeling and comparisons to existing neuromorphic designs enhance the technical depth. Supplementary materials provide additional device data and experimental details.

While the new neuromorphic architecture presented in this paper shows promise, there are several key concerns that need to be addressed:

1. The architecture is conceptually scalable, but the physical demonstration is limited in scale. The largest network tested in hardware (for fault diagnosis) has 32 neurons in the hidden layer and 10 output classes, and the RRAM array used (576 devices) could store at most a few hundred weights or delay elements. The speech recognition task with a 120–200–35 network was, as I understand, largely evaluated in simulation with the hardware parameters, then partially mapped to the hardware for inference. The authors do use the phrase “system-level simulations” when reporting the SNN accuracy, which suggests that not all layers or neurons were physically instantiated at full scale. The reliance on an FPGA and PC for data preprocessing and coordination, while practical, means this is not a fully self-contained neuromorphic chip. As a result, claims about “unprecedented scalability” and bridging to real-world edge computing need to be taken cautiously. The current hardware is an impressive proof-of-concept, but additional evidence would be needed to show that this approach can be scaled up to large networks (with thousands of neurons/synapses) on a single chip without relying on external control logic. I would encourage the authors to clarify the distinction between what was simulated vs. physically implemented, and perhaps temper statements about scalability to match the demonstrated network sizes.

2. Gained by the dendritic structures, especially on the speech dataset, are relatively small (+1.5–3.5%). While any improvement in accuracy for a given model complexity is useful, the practical significance could be questioned. For instance, 1–3% gains might fall within the variance of different runs or might be achievable through other means (e.g., additional training epochs or slightly larger networks) in SNNs. The authors should comment on the statistical significance of these improvements – e.g., were the results averaged over multiple trials or a single train/test split? It would also strengthen the paper to compare against an adaptive LIF (ALIF) neuron baseline. The homeostatic neuron is said to “extend the adaptive threshold behavior of classical ALIF neurons”, yet the comparisons in the experiments were only against non-adaptive LIF neurons. An ALIF neuron model can already provide some firing rate adaptation; how much of the 5.8% accuracy gain (on CWRU) might an ALIF have achieved? The authors claim “significant advantages” over existing ALIF models (mentioning faster convergence and accuracy), but no direct empirical comparison was shown in the results. This is

a missed opportunity to validate the specific advantage of homeostasis vs. simpler adaptation. I recommend the authors include or discuss a baseline with ALIF neurons in the revised manuscript to convincingly demonstrate that the VO2-based homeostatic mechanism offers more than what a standard adaptive threshold would. Otherwise, the novelty of the neuron model might be called into question.

3. The dendritic circuit design, while innovative, introduces additional complexity into the neuron model. The paper itself notes that without careful design, the use of RRAM-based delay lines can lead to undesirable effects such as multiple spurious spikes and disrupted timing in deeper network layers. The authors had to engineer a pulse peak-holding circuit and additional transistors to ensure each dendritic input yields only one spike at the neuron. This mitigation is sensible, but it also highlights that the system required non-trivial tuning to function correctly. My concern is how robust such a design is under process variations or device non-idealities. Memristive devices (both RRAM and VO2) are known to suffer from cycle-to-cycle and device-to-device variability. The manuscript would benefit from a discussion (or data, if available) on how sensitive the network accuracy is to variations in the RRAM conductance states or VO2 threshold voltages. The authors do show a distribution of conductance levels for the RRAM array, implying 8-level programming is robust across many cells. However, during operation, noise or drift could affect spike timing – was any error-correcting or calibration mechanism needed to maintain consistent delays? Additionally, the endurance of the memristors might be relevant if on-line learning or frequent reprogramming of weights is expected (though in these experiments, weights were presumably written once after training). While these issues do not detract from the concept, they could impact reproducibility and long-term reliability, and should be acknowledged.

4. The network training was conducted in software (with custom neuron models), and then the parameters were transferred to hardware for inference. One concern is the potential mismatch between the software model and hardware behavior. The authors derived equations (1)–(3) to model the VO2 neuron dynamics and used those in simulation. It's important that these models accurately capture the analog circuit behavior; otherwise, a network trained on the model might perform sub-optimally on the real hardware. Did the authors verify the hardware outputs against the simulation (e.g., by measuring the confusion matrix or accuracy directly from the hardware runs)? The workflow in Fig. 5e implies a full forward and backward pass simulation. I suspect the final accuracies (92.88%, 86.64%) were obtained from the simulation model rather than purely measured from hardware, due to the logistical challenge of measuring thousands of classifications on the custom board. Clarifying this in the paper would be helpful. If the hardware was only used to evaluate a subset (like the neuron and dendrite response, energy per spike, etc.), the authors should make clear which results are simulated vs. measured. This is crucial for readers who might attempt to reproduce or build on this work – they need to know how much can be expected to work in physical memristors versus what was idealized.

5. While the paper is mostly well-written, a few points of clarity could be improved. The description of the dendritic circuit and its modification (Section on Fig. 4h–4j) is quite dense and could confuse readers not intimately familiar with analog circuit design. The authors might consider simplifying that explanation or moving some details to supplementary material. Moreover, there are minor editorial issues to fix: for example, on page 15 (Fig. 4 description) it reads “During readout, the current charges the membrane potential...”, which contains typos (“readout” for readout, “membrane” for membrane). These do not affect the science but polishing them will improve professionalism. Another minor issue is inconsistent notation (e.g., in equations the use of variables could be better defined in the text). Ensuring all acronyms (like STFT, GSC) are defined when first used would also aid readers. None of these are major problems, but they should be addressed in revision for clarity and precision.

Reviewer #2

(Remarks to the Author)

This manuscript presents a novel hardware implementations of neuromorphic neuron models leveraging VO2 devices. First the authors describe a basic LIF neuron implementation that leverages the inherent switching behavior of a VO2 memristor to emulate spiking behavior. The authors then incorporate a dynamic feedback loop that allows the neuron to self-regulate its firing behavior, referred to as the homeostatic neuron. Finally, the authors use a RRAM array to mimic the temporal delays of biological dendrites (between synapse and soma). The authors argue that the additional temporal complexity introduced by the homeostatic and dendritic mechanisms offer superior temporal signal processing, using a temporal classification task on an industrial defect detection dataset (CWRU, the Case Western Reserve University bearing fault dataset) and a speech recognition task (using GSC, the Google Speech Commands dataset).

Neuromorphic implementations of spiking neural networks have been experiencing a resurgence and this paper focuses on processing of temporally-varying signals, and area where I think neuromorphic processing has a lot to offer. I wasn't entirely convinced by the authors' argument that these new neuron models have concrete advantages to offer, however. The authors mention a 5.84% accuracy improvement in defect detection for homeostatic neurons and a 1.56% accuracy improvement in the speech recognition for neurons with dendritic structures (relative to baseline models with LIF neurons). While these are promising results, the authors do not document what accuracies are produced by conventional or state-of-the-art approaches. Accuracy rates exceeding 98-99% have been reported in the literature for both datasets. While these results do not leverage neuromorphic approaches, it seems like the authors might need a more compelling application — the reported results (92.88% for CWRU and <90% for GSC) don't really seem viable by themselves. Maybe if the authors could provide some energy-efficiency comparison?

Also, could the authors comment on the size/physical footprint for these circuits? It seems like the additional temporal complexity will require a bigger physical footprint. While the authors show an accuracy improvement over a baseline LIF-based network, understanding the trade-off space would improve the broad interest of this manuscript.

Minor:

- I could have used a table that collected all the benchmark results so that performance of different models (LIF, homeostatic, dendritic, homeostatic+dendritic) could be compared with more conventional approaches on the different tasks/datasets.
- page 15: "readout" is misspelled
- page 15: I think "result" should be "results"

Reviewer #3

(Remarks to the Author)

The manuscript reports on the utilization of VO₂-based devices to implement homeostatic neurons, and on a combination of RRAM and VO₂ devices to implement dendritic branches. The authors build a system consisting of a single VO₂ memristor device and a small 1T1R conventional array connected through an FPGA. System-level simulations of the proposed SNN, incorporating homeostatic neurons and dendritic structures, are tested on the CWRU dataset and for speech recognition tasks. Results are promising; however, there is a lack of comparison with state-of-the-art approaches, making it difficult to assess the benefits of the proposed architecture. While I find the presented method and conclusions valuable, I believe the authors need to more clearly articulate the novelty of their work and substantiate it with more detailed experimental results. Additionally, certain aspects of the manuscript require supplementary data and explanations to aid reader comprehension.

1. The idea of using a single Threshold Switching (TS) device with a capacitor — the TS/C structure — to implement a spiking neuron is not novel, and no comparison with previously published works is provided. The only apparent novelty in Fig. 2 is that the input is a current rather than a voltage; however, the advantages of this choice are not explained.
2. From my understanding, the neuron parameters in the proposed design are fixed, determined by the values of the resistances and capacitors. In contrast, biological neural networks typically contain neurons with heterogeneous parameters that can adapt through homeostatic plasticity rules. What the authors refer to as "homeostatic plasticity" in Fig. 2g–i appears to be simply a change in output frequency when the input signal (current, in this case) is varied. In neuroscience, neuron homeostatic plasticity refers to a set of mechanisms that maintain stable activity levels over time, despite changes in input or network dynamics. It acts as a self-regulating process to prevent neural circuits from becoming either overactive or underactive.
3. The circuit proposed in Fig. 3 implements excitatory/inhibitory behavior; however, this is different from homeostatic plasticity. Excitatory/inhibitory behavior describes the instantaneous effect of an input on a neuron, whereas homeostatic plasticity is a long-term adaptive process that adjusts the excitatory/inhibitory balance to keep firing rates within a healthy range. Implementing true homeostatic plasticity would require some form of non-volatile memory to store parameter changes over the long term.
4. It also appears that the results shown in Fig. 3 are purely simulation-based. If this is the case, it should be explicitly stated in the manuscript.
5. Additionally, the proposed neuron (both in Fig. 2 and Fig. 3) can only integrate positive currents. While the target application described here does not require integration of negative currents, high-precision spiking neural networks generally require handling both positive and negative weights. This means neurons should be able to integrate both excitatory and inhibitory currents. Without this capability, the proposed design may face limitations in broader SNN implementations.
6. A comparison with purely CMOS-based solutions would also be important to better illustrate the advantages of the proposed approach.
7. Regarding dendritic behavior, my understanding is that it requires the ability to modulate both the synaptic weight and the delay for each synapse [1]. If I have understood correctly, the proposed circuit uses non-volatile RRAM to modulate delays — but how are the weights modulated? Furthermore, what is the range of time constants achievable with your design? In many realistic models, time constants on the order of milliseconds or longer are required.
8. The proposed circuit requires the integration of VO₂ memristors, conventional non-volatile memristors, and CMOS circuitry. However, the feasibility and cost implications of co-integrating VO₂ memristors with conventional non-volatile memristors are not discussed.
9. The authors claim: "The dendritic computation is achieved through hybrid analog circuits incorporating VO₂ devices and CMOS amplifiers, facilitating programmable temporal dynamics and nonlinear integration." Can you confirm whether these hybrid circuits have actually been designed and tested, and whether VO₂ devices have been co-integrated on CMOS circuits? It is not clear what is simulated and what has actually been fabricated and tested.
10. Is the RRAM array capable of performing analog in-memory computing, or is it simply a conventional RRAM array? Can you provide basic characterization of the array, such as multi-level programming capability and variability?
11. If I understand correctly, in the system-level SNN simulations you have only one delay per dendrite, and you train this delay. This implies that models with dendrites have more trainable parameters. Can you comment on this point and compare

it with approaches using non-trained delays?

12. The authors claim a 5.84% improvement over conventional LIF neurons, but to my understanding this refers to an LIF neuron designed by the authors, not to a state-of-the-art LIF neuron.

13. Overall, the comparison with state-of-the-art is missing, and the main advantages of the proposed approach — relative to classical SNNs and other dendrite-based implementations (see [1]–[3])— are not clearly established.

References:

- [1] D'Agostino, S., Moro, F., Torchet, T. et al. DenRAM: neuromorphic dendritic architecture with RRAM for efficient temporal processing with delays. *Nat Commun* 15, 3446 (2024).
- [2] Hammouamri, I., Khalfaoui-Hassani, I. & Masquelier, T. Learning delays in spiking neural networks using dilated convolutions with learnable spacings. *ICLR* (2024).
- [3] Sun, P., Eqlimi, E., Chua, Y., Devos, P. & Botteldooren, D. Adaptive axonal delays in feedforward spiking neural networks for accurate spoken word recognition. *ICASSP 2023*, 1–5 (IEEE, 2023).

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

This manuscript presents a co-integrated neuromorphic hardware system that combines VO₂-based volatile memristors with non-volatile RRAM devices to realize, in hardware, two biologically important yet rarely co-implemented functionalities: neuronal homeostatic regulation and dendritic delay-based temporal processing. Specifically, the authors demonstrate: (1) a VO₂-memristor-based homeostatic spiking neuron that autonomously restores its firing rate following excitatory or inhibitory perturbations, extending beyond conventional LIF and ALIF neuron models; (2) a programmable dendritic architecture employing multi-level RRAM conductance states to implement tunable temporal delays; and (3) a hardware-informed spiking neural network (SNN) framework that achieves statistically significant performance improvements on representative temporal tasks, including bearing fault diagnosis (CWRU, 92.14% ± 0.99%, outperforming both LIF and ALIF baselines) and speech recognition (GSC, 86.53% ± 0.18%), where dendritic delays consistently enhance accuracy. The system further demonstrates low-energy operation (~19 pJ per spike) and compact circuit footprints, underscoring its relevance for edge and neuromorphic computing applications. The revised manuscript incorporates ablation studies, repeated training runs, ALIF benchmarks, and robustness analyses, substantially strengthening the technical rigor and credibility of the results.

The work is of clear significance to the neuromorphic hardware community and is also relevant to related fields, including in-memory computing, edge AI hardware, and brain-inspired circuit design. Prior studies have largely focused on memristor-based implementations of LIF or ALIF neurons using threshold adaptation or leakage control, while dendritic processing has predominantly been explored at the algorithmic or simulation level, with limited hardware realization. Moreover, intrinsic homeostatic regulation implemented directly at the circuit level remains rare, particularly in physical device-based systems. In this context, the present work advances the state of the art by: (1) co-integrating dendritic delay circuits and neurons within a unified hardware framework; (2) combining experimentally validated device- and circuit-level measurements with hardware-calibrated system-level simulations, consistent with best practices in neuromorphic hardware research; and (3) benchmarking against existing CMOS-, RRAM-, and SNN-based hardware implementations while appropriately positioning the contribution as a scalable hardware foundation rather than a fully realized large-scale chip. The conclusions drawn are well supported by the presented evidence.

In response to earlier reviewer concerns, the authors have added: (1) multiple independent training runs with reporting of mean values and standard deviations; (2) direct comparisons with both LIF and ALIF neuron baselines; (3) long-term continuous inference experiments demonstrating the distinctive stability advantage of the proposed homeostatic neurons; and (4) comprehensive analyses of device variability, endurance, retention, and noise tolerance. The claims have also been appropriately moderated, particularly with respect to scalability and system integration. No additional experiments appear necessary, and no major technical flaws are evident. The methodology is sound and consistent with state-of-the-art practices in neuromorphic hardware research.

Any remaining issues are minor and largely editorial in nature (e.g., phrasing consistency and careful claim formulation) and do not preclude publication. At most, they may warrant minor editorial refinement rather than substantive revision.

Overall, this is a carefully revised, technically robust, and conceptually meaningful contribution to neuromorphic hardware research. The authors have thoroughly addressed prior reviewer critiques, strengthened the experimental and statistical foundations of the work, and clarified the scope of their claims. The manuscript meets the standards of *Nature Communications* and should be suitable for publication.

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)

I thank the authors for their thorough responses. The addition of comparisons with SOTA platforms on the example tasks used in this paper, as well as comparison of physical footprints and energy measurements of their proposed circuits address the previous concerns I had with this manuscript. I recommend for publishing.

(Remarks on code availability)

Reviewer #3

(Remarks to the Author)

From a hardware point of view, the proposed system consists of only a single VO₂ memristor device and a small conventional 1T1R array interfaced via an FPGA. Neither the neuron circuits nor the dendritic branches have been fabricated or experimentally tested. As clarified in the response to the reviewers, the reported results are primarily based on circuit-level simulations, since in the board-level prototype the leakage currents of the transistors already reach the microampere (μ A) range, which is comparable to the μ A-level currents modulated in the proposed neuron circuit. Given this limitation, the scientific and technological relevance of the FPGA-based demonstration remains unclear. The manuscript should clarify the added value of the FPGA demonstration under these constraints.

A comparison with the state of the art in terms of area and energy efficiency is provided in Supplementary Tables 1 and 3. While useful, this information should be discussed more clearly in the main text, particularly in relation to other dendritic computing approaches. In particular, comparisons with hardware dendritic implementations based on RRAM are missing from the Introduction and Discussion sections. I would expect a more explicit discussion of the advantages and limitations of different dendritic approaches, for example: architectures where multiple synapses share a common dendritic branch and time constant, versus approaches in which each synapse is assigned a distinct delay or time constant. Regarding the comparison with digital neuromorphic platforms, the manuscript currently mentions Intel's Loihi architecture. However, Loihi's dendritic processing is implemented digitally: state variables corresponding to dendritic compartments are stored in SRAM and updated in discrete time steps. This is fundamentally different from the continuous-time analog dynamics and nonlinear dendritic behavior proposed in this work. It is therefore not clear that Loihi represents an appropriate or fair comparison.

The authors indicate that the achievable dendritic time constant can be extended from the microsecond regime into the millisecond range (15 μ s–5 ms). I would like to see a more detailed analysis of the impact of the dendritic time constant on system-level performance, including simulations with even longer time constants (beyond 5 ms).

Finally, the use of differential synapses to represent positive and negative weights is a well-established design choice in RRAM-based systems. However, the addition of a circuit implementing a ReLU function prior to the neuron increases circuit complexity and likely impacts both area and power consumption. This design choice should be clearly justified in the main text, and its implications for scalability, area, and energy efficiency should be explicitly discussed and, if possible, quantified.

(Remarks on code availability)

Version 2:

Reviewer comments:

Reviewer #3

(Remarks to the Author)

The authors have satisfactorily addressed all my questions and concerns.

(Remarks on code availability)

Open Access This Peer Review File is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

In cases where reviewers are anonymous, credit should be given to 'Anonymous Referee' and the source.

The images or other third party material in this Peer Review File are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

To view a copy of this license, visit <https://creativecommons.org/licenses/by/4.0/>

Title: Homeostatic Dendritic Neuron Based on Co-Integrated Volatile and Non-Volatile Memristors for Neuromorphic Processing

Response to the editor and the reviewers

We would like to express our gratitude to the editor for the kind consideration of our manuscript, and to the reviewers for their thorough assessment of our work and constructive feedback on it, which are extremely valuable in improving our manuscript. We have considered all the reviewers' comments in detail and have made corresponding revisions. Additional experiments, simulations and benchmarking (Supplementary Fig. 4, 19-20, 24-26, 28-30, 33-36, Supplementary Tables 1-4) have been performed. Besides, following the reviewer's suggestion, we conducted repeated simulations of the network and refreshed the reported accuracy results. We also noticed several errors in the original manuscript, and these have been corrected in the revised version. We hope that the editor and reviewers will find the revised manuscript suitable for publication in Nature Communications. The point-to-point responses and changes made are listed below. For clarity, the revised sections quoted from the manuscript files are shown in blue and the exact changes made relevant to the comment are *in italics*.

Comments from Reviewer #1

The manuscript introduces, for the first time, a hardware implementation that co-integrates spiking neurons with intrinsic homeostatic regulation and dendritic delay pathways using memristive devices. The use of volatile VO₂ for neuron circuits and non-volatile RRAM for programmable delays represents a novel device-level approach to emulating complex neuronal behaviors. The authors present a functioning prototype: a 32×18 RRAM array integrated with VO₂ neuron circuits on a custom PCB. VO₂ devices show stable switching over 500 cycles, and RRAM cells reliably support multi-level conductance states. The

neuron circuit achieves a compact footprint ($\sim 76.8 \mu\text{m}^2$) and low energy consumption ($\sim 19.3 \text{ pJ/spike}$), indicating strong engineering optimization. The system is evaluated on two distinct datasets. On the CWRU fault diagnosis task, the architecture achieves $\sim 92.9\%$ accuracy—a 5.84% improvement over baseline. On the GSC keyword spotting task, dendritic circuits alone yield a $\sim 1.56\%$ gain, while combining dendrites with homeostatic neurons results in $\sim 86.6\%$ accuracy ($+3.5\%$). The inclusion of an ablation study across four network variants strengthens the attribution of performance gains. The architecture supports sparse, event-driven processing, with real-time inference managed via FPGA coordination. This hybrid pipeline demonstrates potential for edge computing applications. The manuscript is well-organized, with clear figures and thorough explanations. Mathematical modeling and comparisons to existing neuromorphic designs enhance the technical depth. Supplementary materials provide additional device data and experimental details.

While the new neuromorphic architecture presented in this paper shows promise, there are several key concerns that need to be addressed:

Our response: We would like to express our deepest appreciation to the reviewer for the careful and thorough evaluation of our manuscript, and for the thoughtful summary that accurately captures the essence of our work. We are truly grateful for your positive recognition of the novelty and significance of our contributions. Moreover, we sincerely appreciate your valuable and constructive feedback, which has been highly beneficial in further improving the quality and clarity of our manuscript. In the revised manuscript, we have carefully addressed all the points raised and conducted additional experiments and simulations to further strengthen the underlying rationale. In addition, we have substantially revised the manuscript, refining or clarifying terms that might have caused confusion, in order to improve its overall clarity and readability. Our detailed point-by-point responses to the reviewer's comments and the corresponding revisions are provided below.

1. The architecture is conceptually scalable, but the physical demonstration is limited in scale. The largest network tested in hardware (for fault diagnosis) has 32 neurons in the hidden layer and 10 output classes, and the RRAM array used (576 devices) could store at most a few hundred weights or delay elements. The speech recognition task with a 120–200–35 network was, as I understand, largely evaluated in simulation with the hardware parameters, then partially mapped to the hardware for inference. The authors do use the phrase “system-level simulations” when reporting the SNN accuracy, which suggests that not all layers or neurons were physically instantiated at full scale. The reliance on an FPGA and PC for data preprocessing and coordination, while practical, means this is not a fully self-contained neuromorphic chip. As a result, claims about “unprecedented scalability” and bridging to real-world edge computing need to be taken cautiously. The current hardware is an impressive proof-of-concept, but additional evidence would be needed to show that this approach can be scaled up to large networks (with thousands of neurons/synapses) on a single chip without relying on external control logic. I would encourage the authors to clarify the distinction between what was simulated vs. physically implemented, and perhaps temper statements about scalability to match the demonstrated network sizes.

Our response: Thank you for the reviewer’s valuable comment. We address the reviewer’s question in two parts as follows.

We sincerely thank the reviewer for the detailed and insightful comments regarding the scalability and hardware implementation of our system. We fully acknowledge that the current hardware demonstration is limited in scale, and the system should be regarded as a prototype for validation purposes.

In this work, our primary goal was to experimentally validate the functionality and feasibility of the proposed neuromorphic architecture across different hierarchical levels. At the device level, we characterized the threshold-switching behavior of the fabricated VO₂ devices (Fig. 2b, c, Supplementary Fig. 2-4), as well as the resistive switching characteristics of the RRAM devices (Fig. 4b-d, Supplementary Fig. 19, 20).

At the circuit level, we implemented both current-input and voltage-input neuron circuits and verified their spike-generation capabilities (Fig. 2e, h, Supplementary Fig. 5-8, 10). We also realized a dendritic circuit structure and experimentally examined its delay modulation behavior, thereby obtaining the characteristic response of each fundamental circuit unit (Fig. 4h, j, Supplementary Fig. 21-24). We also mapped a subset of the network weights onto the RRAM array to perform matrix–vector multiplication, as shown in Supplementary Fig. 26.

For the network-level validation, we adopted a system-level simulation approach (Fig. 3c, d, Fig. 5g-j, Fig. 6). Importantly, this simulation was not purely software-based, but rather hardware-informed, utilizing experimentally measured parameters such as the VO₂ threshold and holding voltages, conductance characteristics, and capacitor charge–discharge dynamics. In addition, to further validate that the constructed model accurately reflects the behavior of the designed neuron, we compared the simulated voltage variations at different nodes during the spiking process with the experimentally measured results (as shown in Supplementary Fig. 10). The simulated spike frequency outputs across the network were then compared with measured neuronal firing rates, demonstrating close correspondence (as shown in Fig. 2f, i). The strong consistency between simulation and experimental results validates the accuracy of the model and supports its application to large-scale network-level analysis. The reason we used this hybrid approach is that the current experimental setup and device array size are not yet sufficient to map the entire network physically, rather than due to any functional limitation of the proposed architecture. Moreover, we note that many prior studies^{1–4} have adopted a similar methodological framework—validating device and circuit functionalities on a limited hardware scale, while employing hardware-calibrated simulations to demonstrate large-scale system performance. Our approach is therefore consistent with established practices in the neuromorphic hardware research community.

To make our experimental validation and simulation work clearer, we have revised the corresponding sections in the manuscript to avoid any potential misunderstanding.

For the experimental results, we have revised pages 8, 9, 15 and 20. For the simulation results, we have revised pages 12, 18, and 23.

Page 8:

“The circuit was fabricated and tested on a PCB, and all electrical waveforms were experimentally recorded using an oscilloscope, with the measurement setup schematically shown in Supplementary Fig. 5.”

Page 9:

“Considering that voltage control in circuits is generally more complex than current control, we physically designed and tested a current-input neuron to facilitate system integration and enable more sophisticated functionalities, as illustrated in Fig. 2g. We applied gradient current inputs and experimentally measured the corresponding changes in membrane potential and spike frequency, as shown in Fig. 2h.”

Page 12:

“To verify this homeostatic regulation, we conducted transient simulations by applying a pulse with fixed amplitude to the neuron model and monitored the gate voltage, output spikes, and membrane potential. The simulated waveforms, shown in Fig. 3c, clearly illustrate the transient modulation and subsequent recovery of spike frequency, where each generated pulse causes the voltage at node₁ to increase, leading the output spike frequency to initially decrease and then rise.”

Page 15:

“Fig. 4j and Supplementary Fig. 23 show the experimentally measured output spikes in response to the applied pulse inputs under different RRAM conductance states, exhibiting delays ranging from 45 μ s to 63 μ s.”

Page 18:

“Building upon the neuromorphic hardware architecture, we develop a SNN system that integrates optimized temporal data encoding with our hardware-calibrated homeostatic neuron models, to evaluate their effectiveness through simulation in real-world-inspired applications.”

Page 20:

“During the forward pass, the neuronal state is determined by combining feedback signals from previous timestep outputs processed through recurrent modules and current timestep inputs transformed by weight matrices, for which the matrix–vector multiplication of selected weight vectors and input data is partially mapped onto the RRAM array (Supplementary Fig. 26)”

Page 23, 24:

“To further demonstrate the role of our constructed dendritic structure within the SNN, we performed a keyword spotting task involving 35 keyword categories and 105,829 audio samples from the Google Speech Commands (GSC) dataset⁶³. We preprocessed the raw data as described in the methods section, and the processed data was then fed into our SNN network, as illustrated in Fig. 6a. Building on the previously discussed homeostatic neuron and dendritic delay structure, we further developed an SNN model with richer dynamic behaviors. The network was modeled using experimentally extracted device parameters and corresponding circuit parameters from the implemented hardware system. These parameters were integrated into the network simulations to provide a realistic performance analysis aligned with the behavior of the physical hardware.”

We fully agree with the reviewer’s comment that our current implementation is not a fully self-contained neuromorphic chip. Due to the limited hardware scale and design complexity, it is impractical for the present prototype to incorporate all network functionalities on a single chip or board—nor was this the primary focus of our study. In our implementation, the batch preprocessing of input data is performed on a PC. The preprocessed data are then transmitted to an FPGA unit, which handles data quantization (i.e., converting floating-point data into integer form for DAC conversion) and routing. After these steps, the signals are fed into our designed neuromorphic computing module, which performs analog-domain matrix accumulation operations and neuronal spiking behavior. This system-level partitioning follows the architectural hierarchy typically adopted in large-scale computing clusters or system-on-chip (SoC)

architectures, where the FPGA operates as an interconnect backbone that manages data routing and distribution, while the PC functions as the central processing unit responsible for data scheduling and preprocessing.

Regarding the scalability aspect, our intended meaning refers specifically to the architectural scalability of the hardware system. Although the current prototype is of limited scale, both the neuron circuits and the dendritic arrays are designed in a modular fashion, enabling scalability toward larger hardware systems through array tiling or the fabrication of expanded arrays. As reported in our previous study⁵, the array capacity can be extended up to 1 Mb. Moreover, a 12 MB scale array has been demonstrated at the 40-nm technology node, occupying only 53.8 mm² including word-line/bit-line circuitry and digital selection logic⁶. In addition, several works^{7,8} have explored scalable integration through array tiling, where optimized instruction-set architectures and data distribution strategies ensure efficient inter-array communication and high-performance operation. To further demonstrate the scalability of our design, we analyzed the hardware resources and area consumption of the proposed architecture when processing the GSC dataset. In addition, a comparison with other hardware implementations performing the same task highlights the efficiency and hardware advantages of our design (as shown in Supplementary Table 4). The comparative results indicate that our design attains high accuracy on the 35-class keyword spotting task while maintaining exceptionally low power consumption per inference. Furthermore, even when implemented using a relatively mature 180-nm process technology, the overall system footprint remains remarkably compact, highlighting its strong potential for scalable neuromorphic hardware integration.

We fully agree with the reviewer that claims such as “unprecedented scalability” and bridging to real-world edge computing should be taken cautiously. We acknowledge that our previous phrasing might have led to misunderstanding, and we have revised the corresponding sections in the manuscript accordingly to better clarify this point.

To address this point, we have incorporated Supplementary Table 4 and a corresponding discussion in the main text, and refined several relevant descriptions in the manuscript to enhance clarity and accuracy.

Page 6:

“The system demonstrates a modular and scalable architecture by co-locating memory and computation, thereby alleviating the von Neumann bottleneck that constrains conventional processors. By employing a field-programmable gate array (FPGA) to handle data distribution and routing, the system provides a flexible hardware platform for spiking neural computation while maintaining real-time processing capability. Moreover, the event-driven operation ensures energy-efficient utilization of computational resources, as they are activated only in response to neural activity.”

Page 25, 26:

“In addition, we evaluated the hardware-level performance of the proposed neuromorphic system for the 35-class GSC dataset. The energy consumption and silicon area of our system implementation are quantitatively analyzed at 180nm node and compared with other representative neuromorphic and digital hardware systems, as summarized in Supplementary Table 4. Despite being implemented at a larger technology node (180 nm), our system demonstrates up to an order of magnitude reduction in area and notable improvements in energy efficiency, highlighting its strong potential for scalable and energy-efficient neuromorphic hardware integration for temporal signal processing tasks.”

Supplementary Table 4:

“Supplementary Table 4. Comparison of hardware implementations for the GSC dataset task^{s18–s26}.”

	Technology	Model	Num Classes	Accuracy	Frequency (Hz)	Energy for 1s speech (J)	Latency (s)	Area (μm^2)
Zol. et al. ^{s18,19}	14nm CMOS + IBM PCM	CNN	11	89.30%	400×10^6	78×10^{-6}	348×10^{-6}	144×10^6
Giraldo et al. ^{s20}	65nm	LSTM	10	91%	250×10^3	16.2×10^{-6}	-	2.56×10^6
Li et al. ^{s21}	65nm	GRU	2	98%	-	122.3×10^{-6}	211×10^{-6}	4.38×10^6
Kim et al. ^{s22}	65nm	RNN	12	86%	250×10^3	23×10^{-6}	12.4×10^{-3}	2.03×10^6
Shan et al. ^{s23}	28nm	DSCNN	2	95%	40×10^3	0.98×10^{-6}	-	230×10^3
Liu et al. ^{s24}	65nm	FS-CNN + LIF	4	95%	100×10^3	7.7×10^{-6}	18.7×10^{-3}	917×10^3
Wang et al. ^{s25}	65nm	SNN	5	90%	120×10^3	0.57×10^{-6}	100×10^{-3}	2.71×10^6
Shaban et al. ^{s26}	50um RRAM + 0.5um CMOS	SNN	12	91%	1×10^3	696×10^{-6}	1.2	11×10^3 (sim 65nm)
This work	180nm RRAM + 180nm CMOS	SNN	35	86.54%	180×10^3	12.9×10^{-6}	1.66×10^{-3}	102×10^3

2. Gained by the dendritic structures, especially on the speech dataset, are relatively small (+1.5–3.5%). While any improvement in accuracy for a given model complexity is useful, the practical significance could be questioned. For instance, 1–3% gains might fall within the variance of different runs or might be achievable through other means (e.g., additional training epochs or slightly larger networks) in SNNs. The authors should comment on the statistical significance of these improvements – e.g., were the results averaged over multiple trials or a single train/test split? It would also strengthen the paper to compare against an adaptive LIF (ALIF) neuron baseline. The homeostatic neuron is said to “extend the adaptive threshold behavior of classical ALIF neurons”, yet the comparisons in the experiments were only against non-adaptive LIF neurons. An ALIF neuron model can already provide some firing rate adaptation; how much of the 5.8% accuracy gain (on CWRU) might an ALIF have achieved? The authors claim “significant advantages” over existing ALIF models (mentioning faster convergence and accuracy), but no direct empirical comparison was shown in the results. This is a missed opportunity to validate the specific advantage of

homeostasis vs. simpler adaptation. I recommend the authors include or discuss a baseline with ALIF neurons in the revised manuscript to convincingly demonstrate that the VO₂-based homeostatic mechanism offers more than what a standard adaptive threshold would. Otherwise, the novelty of the neuron model might be called into question.

Our response: Thank you for the reviewer's valuable comment. We address the reviewer's question in two parts as follows.

We sincerely thank the reviewer for raising this valuable point regarding the statistical significance of the reported accuracy improvements.

In the initial submission, the results were indeed based on a single training and testing run. To address this concern, we have now conducted ten independent training runs for each model configuration under identical conditions.

For the bearing fault detection task (CWRU dataset), we trained the two networks (based on LIF and HSLIF neurons) ten times independently and evaluated each trained model on the same test set (Supplementary Fig. 28, 29). We calculate the mean and standard deviation of these ten best accuracies. The results show that the model with HSLIF neurons achieved an average accuracy of 92.14% with a standard deviation of 0.99%, while the model with standard LIF neurons achieved 87.31% with a standard deviation of 0.51%. Although minor variations were observed across runs, the overall trend consistently shows that the network employing homeostatic HSLIF neurons achieves higher accuracy than the baseline LIF-based network as shown in Fig. 5j. In addition, we extend the training process to 400 epochs to examine whether longer training could further improve performance. As shown in Supplementary Fig. 29, the network already achieved stable and satisfactory accuracy within the first 200 epochs, and further increasing the number of epochs did not lead to noticeable improvement in accuracy.

As for the reviewer's concern that slightly larger networks may lead to similar accuracy gains, previous study has demonstrated that merely increasing the network size substantially raises hardware resource consumption while yielding only marginal

improvements in accuracy⁹. In our experiments on the CWRU dataset, we expanded the baseline LIF network from 32 to 64 neurons. The best test accuracy increased only slightly from 87.31% to 88.71% (a gain of 1.4%), whereas the hardware cost approximately doubled, including both the crossbar array area and the neuron count. This trade-off clearly indicates that simple network scaling is inefficient: the accuracy improvement is smaller than that achieved by our proposed HSLIF neuron (+4.83%) while incurring significantly higher hardware overhead. Therefore, the observed performance advantage of HSLIF cannot be attributed to network size but to its intrinsic homeostatic mechanism, which provides a better accuracy–efficiency balance.

For the speech recognition task, we also performed repeated training on four network configurations and computed the mean and standard deviation of the highest accuracies from each run (Fig. 6d, Supplementary Fig. 33, Supplementary Table 2). The results show that the network incorporating dendritic structures achieves 84.61% accuracy with a standard deviation of 0.09%, whereas the network with basic LIF neurons reaches 83.03% with a standard deviation of 0.17%.

We have revised the manuscript accordingly and added the averaged results with standard deviations.

Page 2:

“... the system achieves improved classification accuracies of 92.14%± 0.99% for industrial defect detection and 86.53%± 0.18% for speech recognition ...”

Page 4:

“We demonstrate the effectiveness of our approach through two validation studies: integrating homeostatic neurons into SNN for defect detection task achieves 4.83% accuracy improvement over conventional LIF neurons, while incorporating dendritic structures into SNN architecture for speech recognition task yields 1.58% accuracy enhancement compared to baseline.”

Page20, 21:

“We repeated the training procedure ten times using the same method. The evolution of accuracy and loss across epochs for all runs is shown in Fig. 5h and Supplementary Fig. 29. The maximum test accuracy from each run was recorded, yielding an average of 92.14% with a standard deviation of 0.99%.

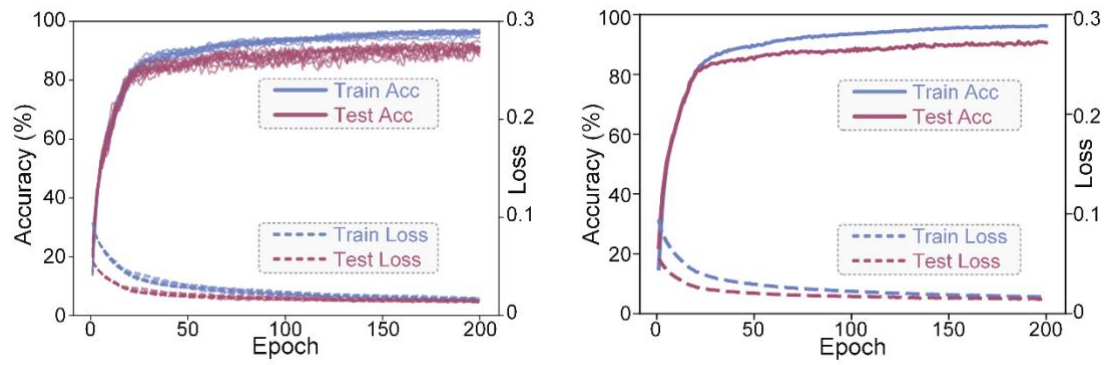
To further evaluate the advantages of the homeostatic neuron over classical LIF and adaptive LIF (ALIF) neurons, we constructed SNNs of identical scale using these neuron types and trained them repeatedly on the same task. The LIF-based and ALIF-based networks achieved accuracies of $87.31\% \pm 0.51\%$ and $90.04\% \pm 0.52\%$, respectively. A comparison of all configurations (Fig. 5j) indicates that the network incorporating our homeostatic neurons demonstrates an improvement in classification performance.”

Page24, 25:

“An SNN with a network scale of $120 \times 200 \times 35$ was constructed and trained for 80 epochs. The training process was repeated ten times, and the best accuracy from each trial was collected. The average of these best accuracies reached 86.53%, with a standard deviation of 0.18% ... Each model was trained for 80 epochs with ten independent runs. The accuracy and loss curves of all runs are shown in Supplementary Fig. 33, and their epoch-wise averages are presented in Fig. 6d. The results indicate that introducing dendritic delays enhances the network’s spatiotemporal processing capability.”

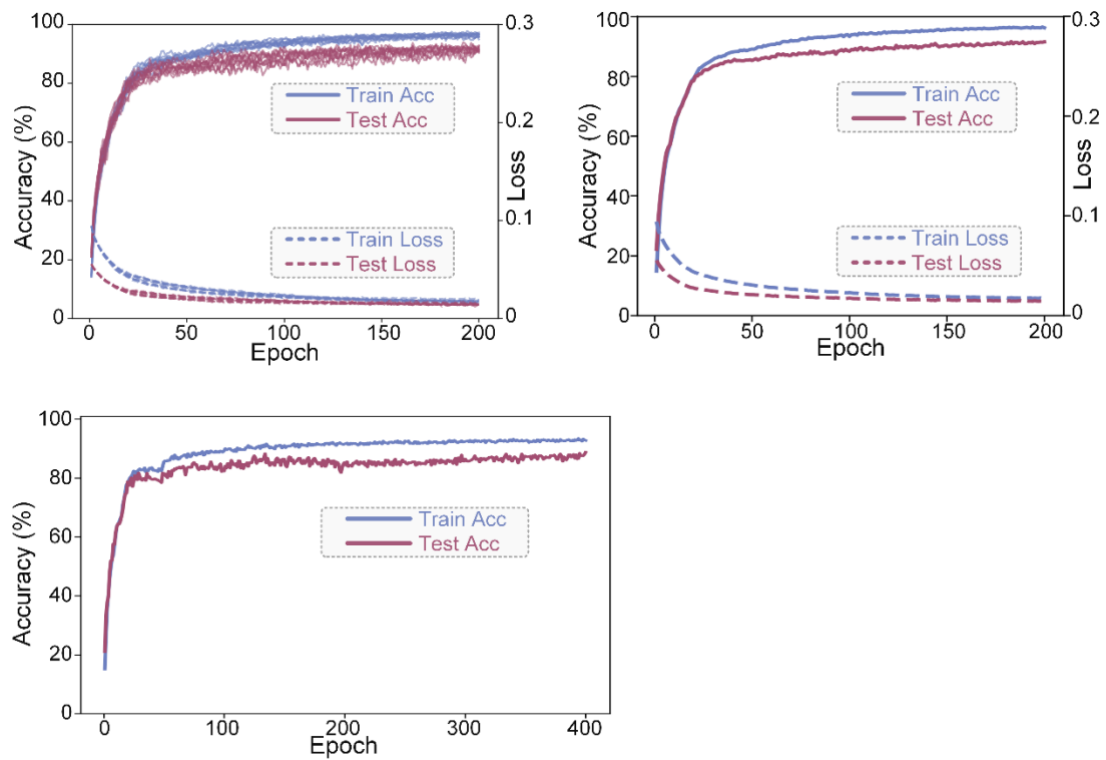
Supplementary Fig. 28:

“Supplementary Fig. 28. Accuracy and loss curves during the training of the SNN with HSLIF. The left picture shows the accuracy and loss curves over 200 epochs for ten independent training runs of the SNN constructed with HSLIF neurons, where each curve corresponds to one training trial. The right picture shows the averaged training and testing accuracy and loss across the ten runs, where for each epoch, the accuracy and loss values were averaged to illustrate the overall training trend.”



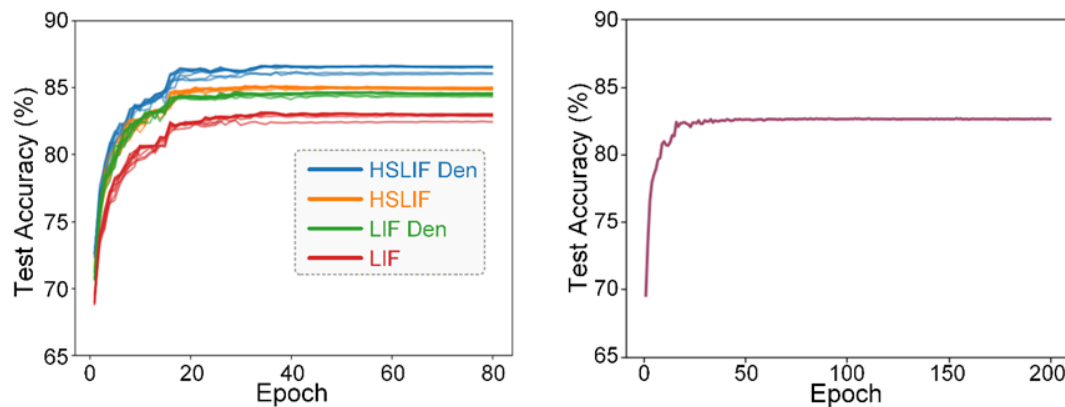
Supplementary Fig. 29:

“Supplementary Fig. 29. Accuracy and loss curves during the training of the SNN with LIF. The left picture shows the accuracy and loss curves over 200 epochs for ten independent training runs of the SNN constructed with basic LIF neurons, where each curve corresponds to one training trial. The right picture shows the averaged training and testing accuracy and loss across the ten runs, where for each epoch, the accuracy and loss values were averaged to illustrate the overall training trend. The bottom figure illustrates the evolution of training accuracy for the same-sized SNN employing LIF neurons, as the number of training epochs was increased from 200 to 400. The results indicate that the network had already converged within 200 epochs, and extending the training to 400 epochs did not yield any further improvement in accuracy.”



Supplementary Fig. 33:

“Supplementary Fig. 33. Accuracy of SNNs with different neuron structures. The left picture shows the accuracy evolution of SNNs constructed with four different neuron configurations from ten independent training runs: (1) regular LIF neurons only, (2) homeostatic neurons only, (3) dendritic structures with regular LIF neurons, and (4) a combination of homeostatic neurons and dendritic structures. Each model was trained independently ten times for 80 epochs. Each curve corresponds to one training run, showing the accuracy variation during the training process. The right picture shows the testing accuracy of the same-sized SNN using regular LIF neurons when the number of training epochs is extended from 80 to 200. The results demonstrate that the network had already converged within 80 epochs, and further increasing the number of epochs did not result in a significant improvement in accuracy.”



Supplementary Table 2:

“Supplementary Table 2. Comparison of Four Network Configurations for the GSC Task.”

Dataset	Structure	Accuracy	Area (μm^2)	Basic circuit unit
GSC (35 classes)	LIF (VO ₂ based)	83.03%	78×10^3	1C 1R 1VO ₂
GSC (35 classes)	HSLIF (VO ₂ based)	85.09%	93×10^3	6T 2C 1R 1VO ₂
GSC (35 classes)	LIF + Dendrite (RRAM & VO ₂)	84.61%	86×10^3	2T 3C 7R 2AMP 1DIO 1RRAM 2VO ₂
GSC (35 classes)	HSLIF + Dendrite (RRAM & VO ₂)	86.53%	102×10^3	8T 4C 7R 2AMP 1DIO 1RRAM 2VO ₂

We sincerely thank the reviewer for this valuable and constructive suggestion regarding the ALIF neuron. Following your advice, we conducted additional experiments using an adaptive LIF (ALIF) neuron baseline with the same network configuration. This comparison enables a more rigorous validation of the specific benefits introduced by the homeostatic regulation in our proposed HSLIF neurons, which go beyond the classical adaptive threshold mechanism of ALIF.

Specifically, we trained the ALIF-based network, and compared the best accuracies with those obtained from the LIF and HSLIF based networks. The ALIF network achieved an average best accuracy of 90.04%, showing a modest improvement over the LIF network (87.31%). However, our proposed HSLIF network reached 92.14%, demonstrating superior performance and confirming that the homeostatic mechanism provides more effective adaptation than the conventional threshold adjustment of ALIF neurons.

As for the term “convergence”, we’d like to clarify that it refers to the stabilization of the neuronal firing rate, rather than the convergence of accuracy during overall network training. Specifically, when the firing frequency of an HSLIF neuron suddenly decreases, it can rapidly return to its original steady-state rate. In contrast, an ALIF neuron typically remains in a prolonged low-frequency firing state after such a decrease, until the neuron is manually reset. To further highlight the unique advantage of our HSLIF neurons in maintaining firing-rate stability, we conducted an additional experiment using concatenated testing data, where some randomly selected segments were combined into a single continuous input sequence. The previously trained LIF- and ALIF-based networks were then used to perform continuous detection on these sequences. As shown in the Supplementary Fig. 30, the HSLIF-based network maintained stable accuracy over long-duration continuous testing, owing to its rapid firing-rate recovery. In contrast, the ALIF-based network exhibited a gradual decline in accuracy over time, as its neurons progressively entered a sustained low firing-rate state. These results clearly demonstrate that our HSLIF neurons possess superior long-term stability and robustness under continuous operation.

To clearly demonstrate the advantages of our proposed HSLIF neurons, we have added the corresponding content in the manuscript and replaced the term “faster convergence” with “enhanced long-term stability”.

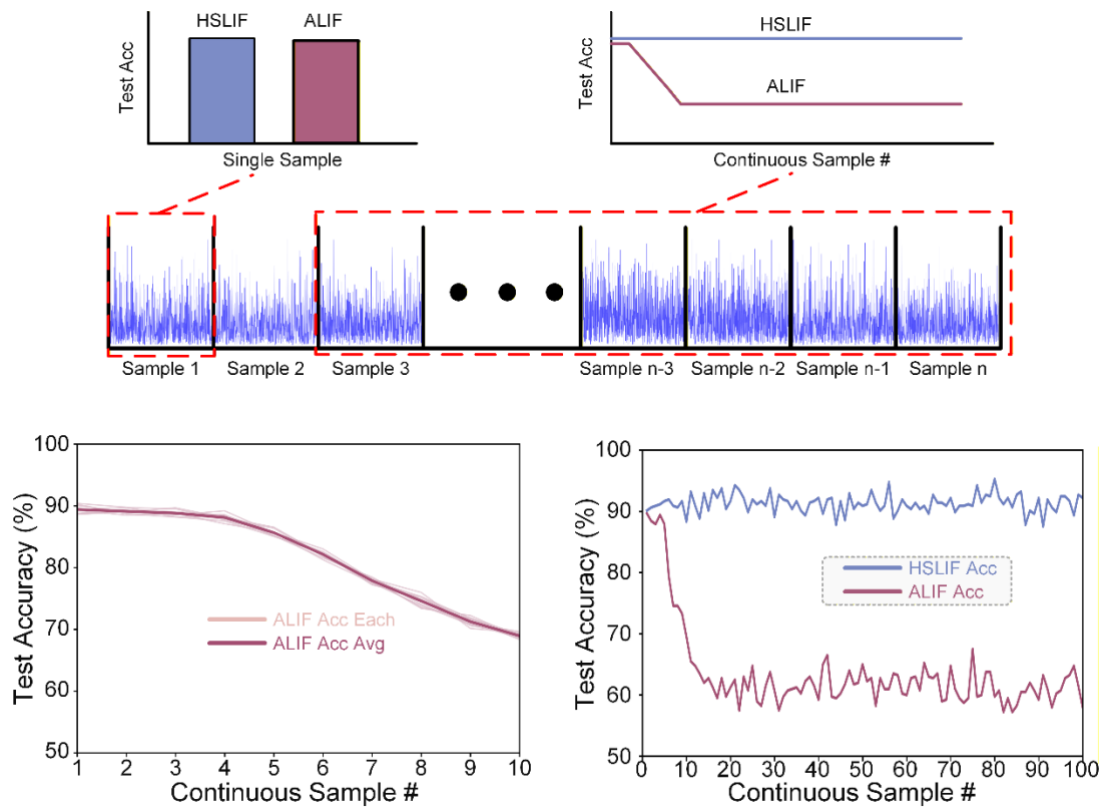
Page 21:

“To further evaluate the advantages of the homeostatic neuron over classical LIF and adaptive LIF (ALIF) neurons, we constructed SNNs of identical scale using these neuron types and trained them repeatedly on the same task. The LIF-based and ALIF-based networks achieved accuracies of $87.31\% \pm 0.51\%$ and $90.04\% \pm 0.52\%$, respectively. A comparison of all configurations (Fig. 5j) indicates that the network incorporating our homeostatic neurons demonstrates an improvement in classification performance. Beyond the improvements in accuracy, our homeostatic neurons exhibit a significant advantage in continuous, long-term information processing compared to ALIF neurons. To assess this, we concatenated multiple test segments into longer sequences (Supplementary Fig. 30) and evaluated the performance of the trained HSLIF (homeostatic LIF neuron) and ALIF-based networks. As shown in Supplementary Fig. 30, the HSLIF-based network maintained stable performance across continuous samples, whereas the ALIF-based network showed a gradual decline in classification accuracy over time. This difference arises from the neurons’ intrinsic dynamics, as HSLIF neurons quickly recover after a sudden firing-rate drop, whereas ALIF neurons remain in a prolonged low-frequency state until manually reset.

Supplementary Fig. 30:

“Supplementary Fig. 30 Performance of HSLIF and ALIF neurons in long-term classification tasks. The top figure illustrates the schematic of our testing methodology. We first trained and evaluated the network model using independent samples. Under this setting, the performance achieved by the HSLIF neurons shows a certain improvement compared with the ALIF neurons, although the enhancement is not very pronounced. We then conducted a continuous testing experiment by concatenating multiple segments to form a long-duration input sequence. The corresponding test results are shown in the two figures at the bottom. The bottom-left picture shows that ten randomly selected data segments were concatenated to form a new continuous

sequence, and the previously trained ALIF-based network was used for testing. The results indicate that the classification accuracy gradually decreases as inference progresses. The bottom-right picture shows that one hundred randomly selected data segments were concatenated to form a longer continuous sequence, and testing was performed using both the previously trained ALIF-based and HSLIF-based networks. The results show that the classification accuracy of the ALIF-based network decreases significantly over time, whereas the HSLIF-based network maintains a consistently high accuracy throughout long-term inference.”



Page 27:

“Compared with existing ALIF models, our proposed homeostatic neuron demonstrates significant advantages, including improved accuracy and enhanced long-term stability.”

3. The dendritic circuit design, while innovative, introduces additional complexity into the neuron model. The paper itself notes that without careful design, the use of RRAM-based delay lines can lead to undesirable effects such as multiple spurious spikes and disrupted timing in deeper network layers. The authors had to engineer a pulse peak-holding circuit and additional transistors to ensure each dendritic input yields only one spike at the neuron. This mitigation is sensible, but it also highlights that the system required non-trivial tuning to function correctly. My concern is how robust such a design is under process variations or device non-idealities. Memristive devices (both RRAM and VO₂) are known to suffer from cycle-to-cycle and device-to-device variability. The manuscript would benefit from a discussion (or data, if available) on how sensitive the network accuracy is to variations in the RRAM conductance states or VO₂ threshold voltages. The authors do show a distribution of conductance levels for the RRAM array, implying 8-level programming is robust across many cells. However, during operation, noise or drift could affect spike timing – was any error-correcting or calibration mechanism needed to maintain consistent delays? Additionally, the endurance of the memristors might be relevant if on-line learning or frequent reprogramming of weights is expected (though in these experiments, weights were presumably written once after training). While these issues do not detract from the concept, they could impact reproducibility and long-term reliability, and should be acknowledged.

Our response: We sincerely thank the reviewer for the valuable comments regarding the robustness of our proposed network.

In the revised manuscript, we have provided detailed information about the cycle-to-cycle (C2C) and device-to-device (D2D) variations of the VO₂ devices. As shown in Fig. 2(b, c), the coefficient of C2C variation (C_v) for the switching threshold voltage (V_{th}) is 1.01%, and for the holding voltage (V_{hold}) is 1.47%. Additionally, we have included the D2D variation results in Supplementary Fig. 4, where the C_v of V_{th} and V_{hold} are 4.93% and 5.70%, respectively.

For the RRAM devices, we employ a write–verify programming scheme to ensure each resistance state is accurately set before use. Nevertheless, as the reviewer correctly pointed out, resistance drift or noise may still occur during operation. To evaluate this, we conducted continuous retention tests across eight distinct conductance states (results in Supplementary Fig. 19). The observed drift errors were 1.5%, 0.01%, 1.18%, 1.03%, 0.6%, 1.2%, 0.15%, and 0.14%, respectively, demonstrating excellent stability across different conductance states.

To further examine the sensitivity of our network to device variations, we analyzed the influence of RRAM conductance fluctuations on network accuracy. During inference, we introduced random noise into the RRAM conductance values and evaluated the classification accuracy under different noise levels. As shown in Supplementary Fig. 35, the network maintained high accuracy (1.4% degradation) with conductance noise up to 10%, demonstrating strong robustness. Similarly, we introduced random variations to the V_{th} and V_{hold} parameters of the VO_2 model, and the results (Supplementary Fig. 36) show only minor degradation (0.57%) in accuracy within a 5% variation range.

These results collectively confirm that our network exhibits strong tolerance to variations in the fabricated devices. Moreover, previous studies¹⁰ have demonstrated that the effect of memristor noise and drift can be further mitigated using optimized denoising pulse schemes, which can be readily adopted in our system to further enhance long-term stability.

Finally, we fully agree with the reviewer’s concern regarding device endurance, which becomes crucial for future online learning implementations. To address this, we have performed endurance measurements on the memristor device by repeatedly applying set/reset voltage pulses for 100k cycles, as shown in Supplementary Fig. 19, demonstrating reliable switching behavior throughout the test.

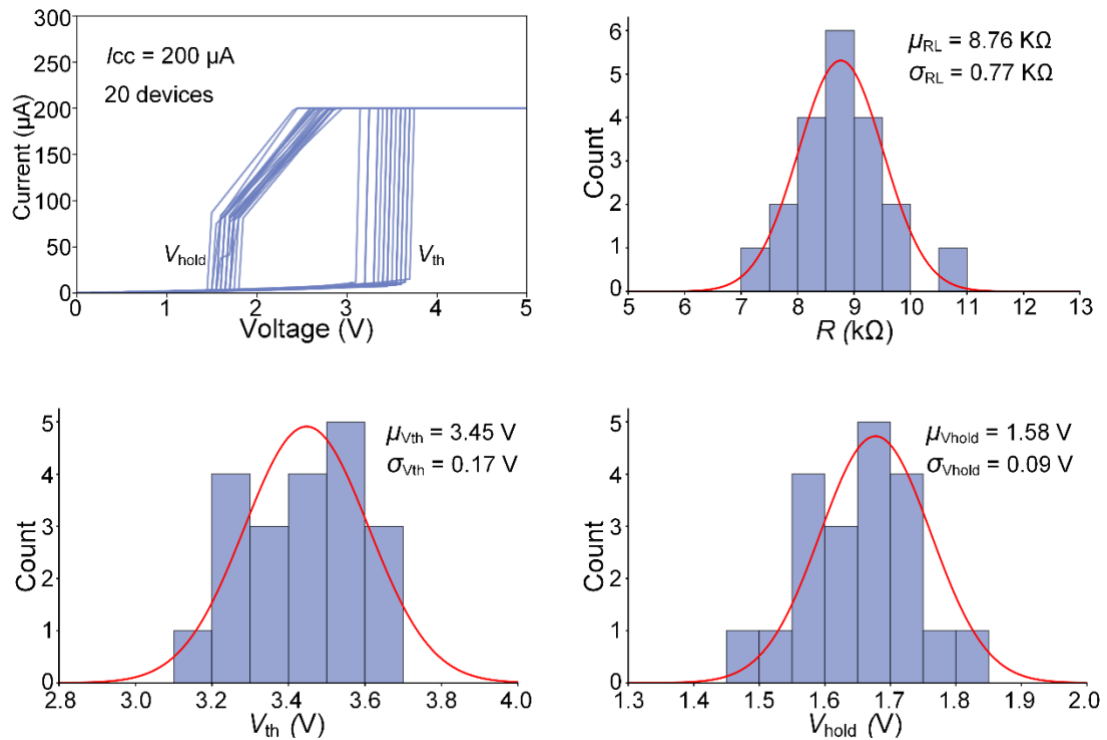
To address this important point, we have added Supplementary Fig. 4, 19, 35, 36, and the following brief statement in the main text.

Page 8:

“In addition, we characterized the device-to-device (D2D) variation of 20 VO₂ memristors and extracted the C_v of both V_{th} and V_{hold} to be 4.93% and 5.70%, respectively (Supplementary Fig. 4).”

Supplementary Fig. 4:

“Supplementary Fig. 4. Device-to-device variation of V_{th} , V_{hold} and the low resistance state (R_{VO2}) of the VO₂ memristors, measured across 20 devices. (a) I-V sweeps of 20 devices superposed to illustrate the decent uniformity. (b) Histogram of R_{VO2} . The C_v is 8.79%. (c) Histogram of V_{th} . The C_v is 4.93%. (d) Histogram of V_{hold} . The C_v is 5.70%.”



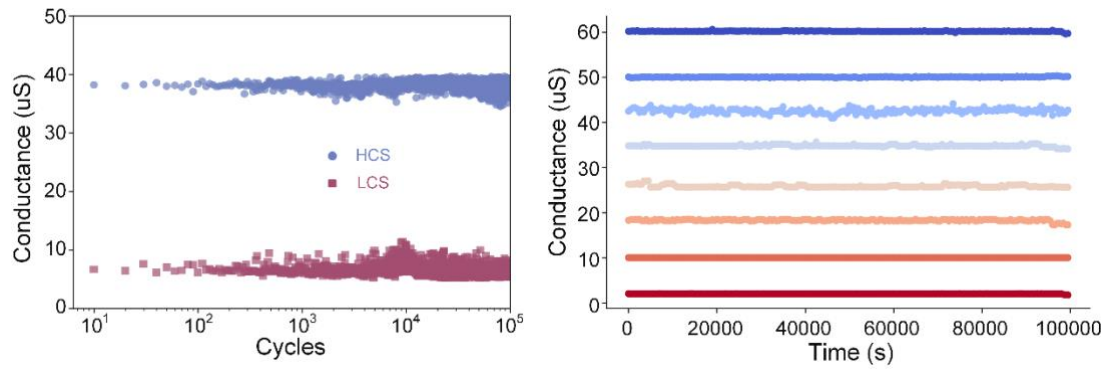
Page 14:

“In addition, we performed endurance tests by repeatedly applying set and reset operations to the device, and the results are shown in Supplementary Fig. 19 ... Fig. 4d and Supplementary Fig. 19 display the cumulative distribution functions (CDFs) across 8 independent conductance states and the excellent retention characteristics of our non-volatile memristor. These characteristics confirm that the array is suitable for

both analog in-memory computing and the implementation of dendritic structures with tunable delays, thereby improving computational performance.”

Supplementary Fig. 19:

“Supplementary Fig. 19 The endurance and retention performance of the non-volatile memristor. (a) Endurance of 10^5 cycles. (b) 8 conductance states can be retained for 10^5 seconds. From the lowest to the highest conductance state, the observed drift errors were 1.5%, 0.01%, 1.18%, 1.03%, 0.6%, 1.2%, 0.15%, and 0.14%.”

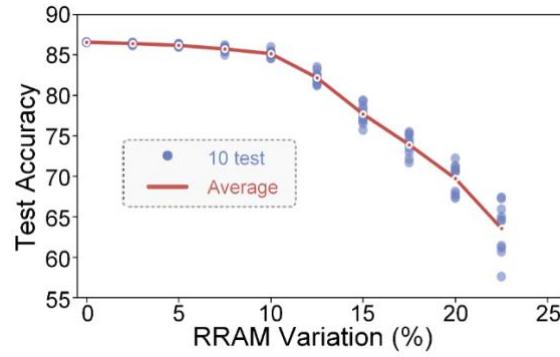


Page 25:

“Since the RRAM and VO_2 devices used in our system may suffer from inherent variations, we conducted simulation analyses to evaluate the impact of such device variations on network performance. As shown in Supplementary Figs. 35 and 36, the classification accuracy decreased by only 1.42% when the RRAM conductance variation reached 10%, and by only 0.57% when the VO_2 threshold and holding voltages varied by 5%, demonstrating that the proposed network exhibits strong robustness against device variations.

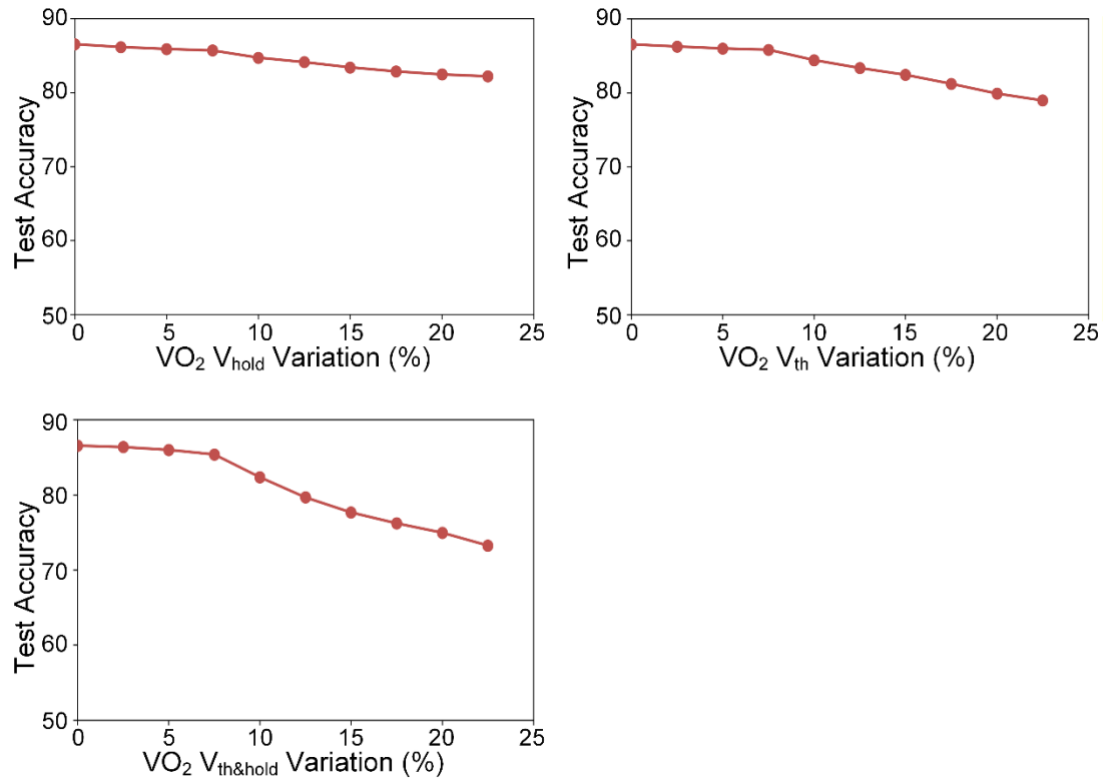
Supplementary Fig. 35:

“Supplementary Fig. 35. Accuracy of the proposed SNN under RRAM conductance variations.”



Supplementary Fig. 36:

“Supplementary Fig. 36. SNN Accuracy under VO_2 Device Variations. (a). Impact of V_{hold} variation on SNN accuracy. (b). Impact of V_{th} variation on SNN accuracy. (c). Combined impact of V_{th} and V_{hold} variations on SNN accuracy.”



4. The network training was conducted in software (with custom neuron models), and then the parameters were transferred to hardware for inference. One concern is the potential mismatch between the software model and hardware behavior. The authors derived equations (1)–(3) to model the VO₂ neuron dynamics and used those in simulation. It's important that these models accurately capture the analog circuit behavior; otherwise, a network trained on the model might perform sub-optimally on the real hardware. Did the authors verify the hardware outputs against the simulation (e.g., by measuring the confusion matrix or accuracy directly from the hardware runs)? The workflow in Fig. 5e implies a full forward and backward pass simulation. I suspect the final accuracies (92.88%, 86.64%) were obtained from the simulation model rather than purely measured from hardware, due to the logistical challenge of measuring thousands of classifications on the custom board. Clarifying this in the paper would be helpful. If the hardware was only used to evaluate a subset (like the neuron and dendrite response, energy per spike, etc.), the authors should make clear which results are simulated vs. measured. This is crucial for readers who might attempt to reproduce or build on this work – they need to know how much can be expected to work in physical memristors versus what was idealized.

Our response: We sincerely appreciate the reviewer's insightful comments regarding the potential mismatch between the software model and hardware implementation. In our study, the modeling and validation are carefully performed across multiple hierarchical levels to ensure consistency between simulation and physical behavior.

At the device and circuit levels, we first model the devices based on experimentally measured I–V characteristics (as shown in Fig. 2b and Supplementary Figs. 2, 3). The derived device models are then integrated into the circuit simulations to construct the neuron circuit model. The simulated voltage waveforms and firing behaviors are directly compared with the experimentally measured results (Fig. 2f, i, and Supplementary Fig. 10), showing excellent agreement. This confirms that the developed model accurately captures the spiking dynamics of the VO₂-based neuron.

At the network level, we acknowledge that device-to-device (D2D) variations may cause slight behavioral differences among neurons. To account for this, random variations in the VO_2 threshold and holding voltages are introduced into the neuron models during simulation, as shown in Supplementary Fig. 36. This analysis demonstrates that the network maintains robust performance even under 10% VO_2 variability. Similarly, random variations in the RRAM conductance were introduced during simulation, and the results indicate that the network exhibits only a 1.4% accuracy degradation under conductance noise of up to 10%.

Regarding the accuracy results, these accuracies are obtained from system-level simulations that incorporate experimentally calibrated device and circuit parameters—including VO_2 threshold and holding voltages, RRAM conductance values, and capacitor charge-discharge dynamics. The neuron and dendrite circuits, as well as the RRAM-based matrix-vector multiplication, are experimentally implemented and characterized (Fig. 2e, h; Fig. 4h, j; Supplementary Figs. 6–8, 10, 21, 23, 26). The hardware measurements serve to validate each functional block, while large-scale network inference is performed through hardware-informed simulations due to the current array size limitations.

This hierarchical validation approach—combining experimental verification at the device and circuit levels with hardware-calibrated network simulations—is consistent with widely adopted methodologies in neuromorphic hardware research.

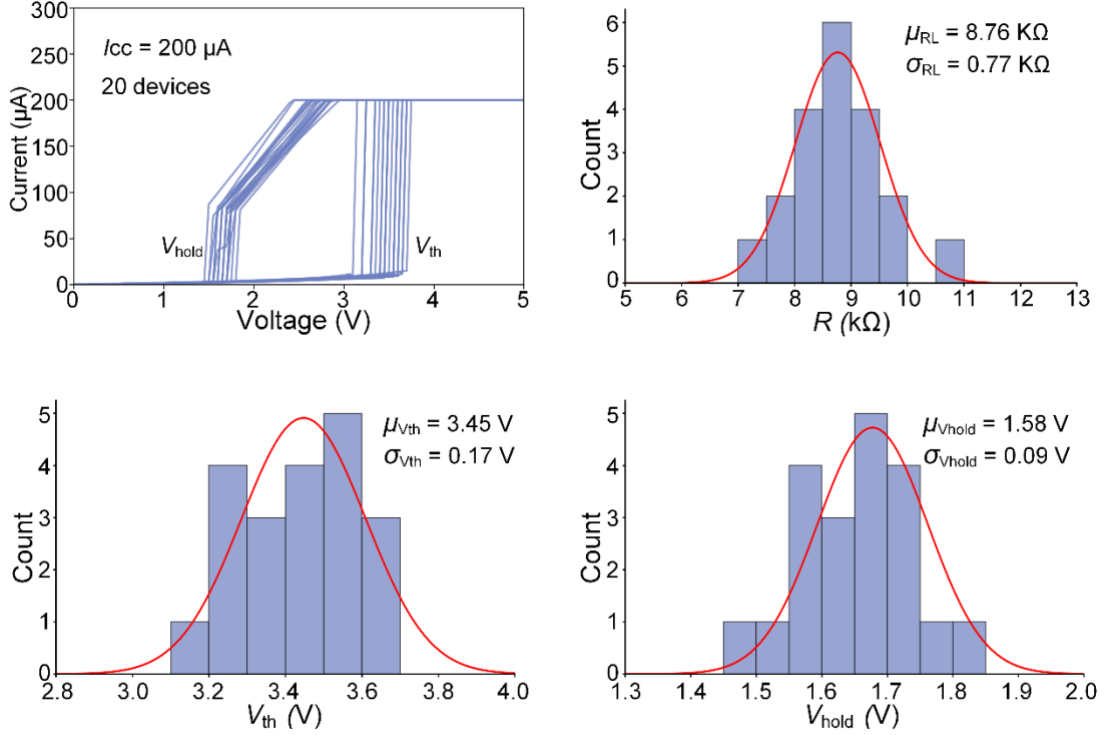
To address this important point, we have added Supplementary Fig 4, and the following brief statement in the main text. To clarify that the network accuracy results were obtained through system-level simulations, we have revised the content on pages 18 and 24. Additional explanations regarding simulations and experimental results have also been provided in our response to the reviewer’s first comment.

Page 8:

“In addition, we characterized the device-to-device (D2D) variation of 20 VO_2 memristors and extracted the C_v of both V_{th} and V_{hold} to be 4.93% and 5.70%, respectively (Supplementary Fig. 4).”

Supplementary Fig. 4:

“Supplementary Fig. 4. Device-to-device variation of V_{th} , V_{hold} and the low resistance state (R_{VO2}) of the VO_2 memristors, measured across 20 devices. (a) I-V sweeps of 20 devices superposed to illustrate the decent uniformity. (b) Histogram of R_{VO2} . The C_v is 8.79%. (c) Histogram of V_{th} . The C_v is 4.93%. (d) Histogram of V_{hold} . The C_v is 5.70%.”

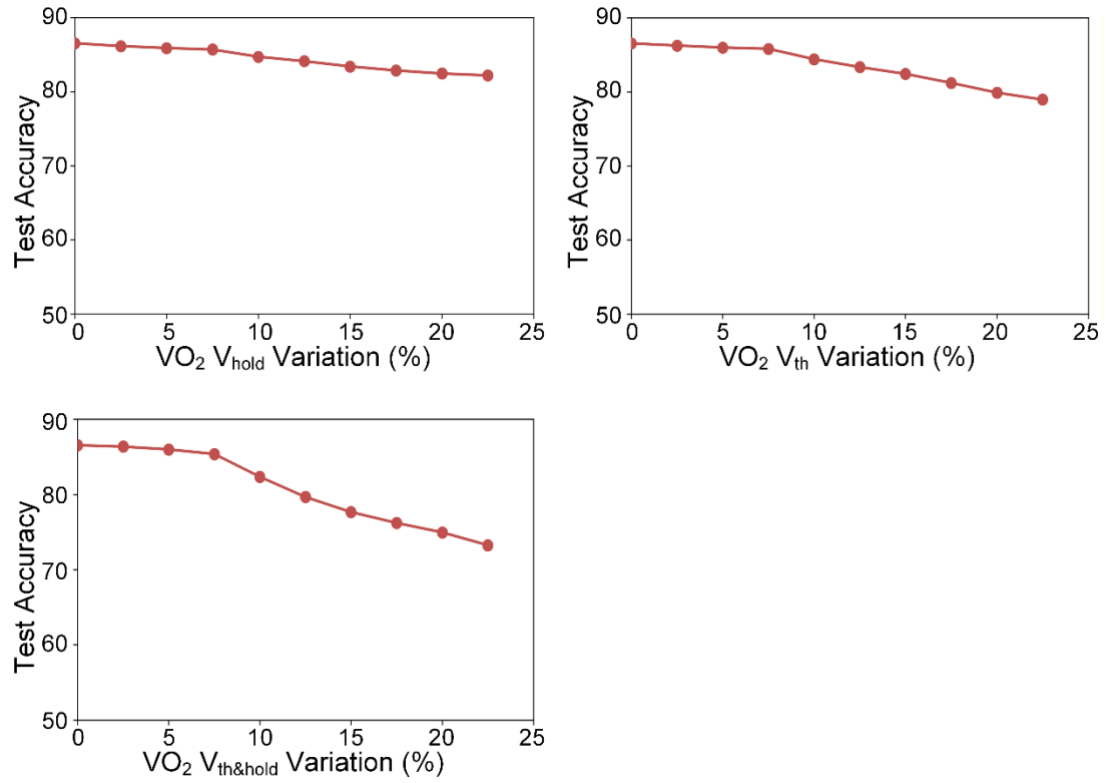


Page 25:

“Since the RRAM and VO_2 devices used in our system may suffer from inherent variations, we conducted simulation analyses to evaluate the impact of such device variations on network performance. As shown in Supplementary Figs. 35 and 36, the classification accuracy decreased by only 1.42% when the RRAM conductance variation reached 10%, and by only 0.57% when the VO_2 threshold and holding voltages varied by 5%, demonstrating that the proposed network exhibits strong robustness against device variations.”

Supplementary Fig. 36:

“Supplementary Fig. 36 SNN Accuracy under VO₂ Device Variations. (a). Impact of V_{th} variation on SNN accuracy. (b). Impact of V_{hold} variation on SNN accuracy. (c). Combined impact of V_{th} and V_{hold} variations on SNN accuracy.”



Page 18:

“Building upon the neuromorphic hardware architecture, we develop a SNN system that integrates optimized temporal data encoding with our hardware-calibrated homeostatic neuron models, to evaluate their effectiveness through simulation in real-world-inspired applications.”

Page 24:

“The network was modeled using experimentally extracted device parameters and corresponding circuit parameters from the implemented hardware system. These parameters were integrated into the network simulations to provide a realistic performance analysis aligned with the behavior of the physical hardware.”

5. While the paper is mostly well-written, a few points of clarity could be improved. The description of the dendritic circuit and its modification (Section on Fig. 4h–4j) is quite dense and could confuse readers not intimately familiar with analog circuit design. The authors might consider simplifying that explanation or moving some details to supplementary material. Moreover, there are minor editorial issues to fix: for example, on page 15 (Fig. 4 description) it reads “During reaout, the current charges the membrance potential...”, which contains typos (“reaout” for readout, “membrance” for membrane). These do not affect the science but polishing them will improve professionalism. Another minor issue is inconsistent notation (e.g., in equations the use of variables could be better defined in the text). Ensuring all acronyms (like STFT, GSC) are defined when first used would also aid readers. None of these are major problems, but they should be addressed in revision for clarity and precision.

Our response: We sincerely thank the reviewer for these constructive suggestions to improve the clarity and readability of our manuscript. (1) We have revised the description of the dendritic circuit (Section related to Fig. 4h–4j) to make the explanation more accessible to readers who are less familiar with analog circuit design. (2) The typographical errors noted by the reviewer (e.g., “reaout” → “readout” and “membrance” → “membrane”) have been corrected. We also performed a careful proofreading throughout the manuscript to eliminate other minor editorial issues. (3) We have also standardized the variable notations throughout the text and ensured that all acronyms (such as STFT and GSC) are clearly defined upon their first appearance. We greatly appreciate the reviewer’s valuable feedback, which has helped us enhance the manuscript’s clarity, precision, and professionalism.

We have revised and refined the following content from the original text to improve the clarity and readability of our manuscript. We have also corrected typographical errors, standardized the variable notations, and ensured that all acronyms are clearly defined upon their first appearance.

“Although the dendritic circuit in Fig. 4g can introduce a delay between the input and output, its behavior is limited by the width of the input pulse and often results in multiple output spikes from a single pulse input ... To address this, we designed a modified dendritic circuit incorporating a pulse peak-holding mechanism, which preserves the membrane voltage after the read pulse is removed. This modification stabilizes the neuron’s input and prevents multiple spike firing. The detailed schematic and performance evaluation are provided in Supplementary Fig. 22. We further optimized the circuit design by employing the operational amplifier of the TIA circuit as the first stage of the pulse-hold circuit, instead of directly widening the input pulse, as illustrated in Fig. 4i. Fig. 4j and Supplementary Fig. 23 show the experimentally measured output spikes in response to the applied pulse inputs under different RRAM conductance states, exhibiting delays ranging from 45 μ s to 63 μ s. When the RRAM device is programmed to a low-conductance state, the pulse input results in a larger delay in the output. Notably, by varying the capacitance parameters, the temporal delay can be flexibly tuned across a broad range from 15 μ s up to 5 ms, as shown in Supplementary Fig. 24.”

Comments from Reviewer #2

This manuscript presents a novel hardware implementations of neuromorphic neuron models leveraging VO₂ devices. First the authors describe a basic LIF neuron implementation that leverages the inherent switching behavior of a VO₂ memristor to emulate spiking behavior. The authors then incorporate a dynamic feedback loop that allows the neuron to self-regulate its firing behavior, referred to as the homeostatic neuron. Finally, the authors use a RRAM array to mimic the temporal delays of biological dendrites (between synapse and soma). The authors argue that the additional temporal complexity introduced by the homeostatic and dendritic mechanisms offer superior temporal signal processing, using a temporal classification task on an industrial defect detection dataset (CWRU, the Case Western Reserve University bearing fault dataset) and a speech recognition task (using GSC, the Google Speech Commands dataset).

Our response: We sincerely appreciate the reviewer's thoughtful and comprehensive summary of our work. The reviewer has precisely captured the key contributions presented in our manuscript, and we are grateful for the recognition of its overall structure and underlying motivation.

Neuromorphic implementations of spiking neural networks have been experiencing a resurgence and this paper focuses on processing of temporally-varying signals, and area where I think neuromorphic processing has a lot to offer. I wasn't entirely convinced by the authors' argument that these new neuron models have concrete advantages to offer, however. The authors mention a 5.84% accuracy improvement in defect detection for homeostatic neurons and a 1.56% accuracy improvement in the speech recognition for neurons with dendritic structures (relative to baseline models with LIF neurons). While these are promising results, the authors do not document what accuracies are produced by

conventional or state-of-the-art approaches. Accuracy rates exceeding 98-99% have been reported in the literature for both datasets. While these results do not leverage neuromorphic approaches, it seems like the authors might need a more compelling application — the reported results (92.88% for CWRU and <90% for GSC) don't really seem viable by themselves. Maybe if the authors could provide some energy-efficiency comparison?

Our response: We sincerely appreciate and feel encouraged by the reviewer's recognition of the potential of neuromorphic processing for temporally varying signals.

We also fully understand the reviewer's concern regarding the reported accuracy of our proposed models. As correctly pointed out, our original submission did not include comparisons with conventional or state-of-the-art (SOTA) approaches. To address this issue and make our evaluation more comprehensive, we have now added detailed comparison tables in the revised Supplementary Information (Supplementary Table 1, 3) for both the CWRU (defect detection) and GSC (speech recognition) tasks. These tables summarize the accuracy, parameter count, and FLOPs of representative networks as well as our proposed models.

As shown in the updated results, while our neuromorphic networks achieve slightly lower accuracies than the SOTA deep learning models (92.14% vs. 99.92% on CWRU, and 86.53% vs. 98.7% on GSC), they are an order of magnitude smaller in both model parameters and computational operations. Such reductions in network size are highly meaningful in neuromorphic and edge scenarios where energy efficiency and latency are key design constraints. Therefore, even though our networks are not primarily optimized for peak accuracy, they deliver a superior balance between performance and efficiency, which is essential for practical neuromorphic deployment.

Furthermore, following the reviewer's valuable suggestion, we have added an evaluation of the hardware-level performance of our neuromorphic implementation. Specifically, we analyzed the energy consumption and silicon area of our system when executing the GSC task, and compared the results with other representative neuromorphic and digital implementations reported in the literature. As summarized in

Supplementary Table 4, our system achieves an energy consumption of 12.9 uJ per inference and a compact core area of 0.102 mm² (at 180nm technology node). In comparison, previously reported neuromorphic or digital hardware systems—implemented in more advanced technology nodes (e.g., 65 nm or 14 nm)—typically consume 7.7 - 122.3 uJ per second of speech, with chip areas ranging from 0.9 mm² to 144 mm², and classification accuracies between 86% and 95%. These comparisons highlight that, even under a larger technology node (180 nm), our system achieves a comparable accuracy with up to an order of magnitude reduction in silicon area and notable energy savings relative to existing designs.

In summary, we have revised the manuscript and Supplementary Information to include these new comparisons and analyses. We believe the additional results strengthen the overall contribution of our work by clearly demonstrating that the proposed neuromorphic neurons, while not targeting the peak accuracy, offer a compelling efficiency advantage and represent a promising step toward energy-efficient temporal signal processing hardware.

Page 21:

“In addition, we have compared the performance of our constructed network with other models reported in the literature on the CWRU dataset. As presented in Supplementary Table 1, our proposed approach achieves slightly lower classification accuracy compared to some state-of-the-art networks; however, it demonstrates a remarkable reduction in both parameter count and computational cost. Consequently, under the same testing platform, our network is expected to deliver lower inference power consumption and latency, highlighting its efficiency and suitability for resource-constrained neuromorphic applications.”

Supplementary Table 1:

“Supplementary Table 1. Comparative benchmark of different networks on the CWRU dataset.^{s1 – s7}”

	Model	Accuracy	Params	Flops
Chen et al. ^{s1}	ResNet50	99.92%	47.2M	718M
Yoo et al. ^{s2}	CNN0	96.64%	901K	1.8M
Rosa et al. ^{s3}	WDCNN	89.70%	40.7K	-
Wang et al. ^{s4}	CWT-2DCNN	83.10%	1.34M	840M
Wang et al. ^{s5}	MS-ER	98.80%	3.95M	2.8G
Lim et al. ^{s6}	SpikeCAN	99.86%	1.45M	-
Fang et al. ^{s7}	Spiking-ResNet	87.46%	3.85M	-
This work	SNN-HSLIF	92.14%	22K	140K

Page 25:

“In addition, we compared the constructed network with other models reported in the literature on the GSC dataset. As summarized in Supplementary Table 3, our network demonstrates significant advantages in terms of parameter count and computational cost while maintaining competitive classification accuracy.”

Supplementary Table 3:

“Supplementary Table 3. Comparative benchmark of different networks on the GSC dataset.^{s8 – s17}”

	Model	Num Classes	Accuracy	Params	Flops
Niizumi et al. ^{s8}	M2D	35	98.50%	86M	-
Chen et al. ^{s9}	EAT	35	98.30%	88M	-
Andrade et al. ^{s10}	Attention-RNN	20	94.50%	202K	44.6M
Kim et al. ^{s11}	BC-ResNet-8	12	98.70%	321K	171.8M
Majumdar et al. ^{s12}	Matchbox-Net	35	97.40%	140K	34.2M
Bittar et al. ^{s13}	SNN-RadLIF	35	94.51%	1.2M	-
Hammouamri et al. ^{s14}	SNN-delays	35	95.35%	2.5M	-
Deckers et al. ^{s15}	SNN-d-cAdLIF	35	95.69%	610K	-
Sun et al. ^{s16}	SNN-delays	20(SHD)	92.45%	109K	-
D'agostino et al. ^{s17}	SNN-delays(fixed)	20(SHD)	90.88%	224K	-
Deckers et al. ^{s15}	SNN-LIF	35	86.21%	300k	-
This work	SNN-HSLIF-delays	35	86.53%	33K	3.13M

Page 25, 26:

“In addition, we evaluated the hardware-level performance of the proposed neuromorphic system for the 35-class GSC dataset. The energy consumption and silicon area of our system implementation are quantitatively analyzed at 180nm node and compared with other representative neuromorphic and digital hardware systems, as summarized in Supplementary Table 4. Despite being implemented at a larger technology node (180 nm), our system demonstrates up to an order of magnitude reduction in area and notable improvements in energy efficiency, highlighting its strong potential for scalable and energy-efficient neuromorphic hardware integration for temporal signal processing tasks.”

Supplementary Table 4:

“Supplementary Table 4. Comparison of hardware implementations for the GSC dataset task.^{s18 – s26}”

	Technology	Model	Num Classes	Accuracy	Frequency (Hz)	Energy for 1s speech (J)	Latency (s)	Area (um ²)
Zol. et al. ^{s18,19}	14nm CMOS + IBM PCM	CNN	11	89.30%	400x10 ⁶	78x10 ⁻⁶	348x10 ⁻⁶	144x10 ⁶
Giraldo et al. ^{s20}	65nm	LSTM	10	91%	250x10 ³	16.2x10 ⁻⁶	-	2.56x10 ⁶
Li et al. ^{s21}	65nm	GRU	2	98%	-	122.3x10 ⁻⁶	211x10 ⁻⁶	4.38x10 ⁶
Kim et al. ^{s22}	65nm	RNN	12	86%	250x10 ³	23x10 ⁻⁶	12.4x10 ⁻³	2.03x10 ⁶
Shan et al. ^{s23}	28nm	DSCNN	2	95%	40x10 ³	0.98x10 ⁻⁶	-	230x10 ³
Liu et al. ^{s24}	65nm	FS-CNN + LIF	4	95%	100x10 ³	7.7x10 ⁻⁶	18.7x10 ⁻³	917x10 ³
Wang et al. ^{s25}	65nm	SNN	5	90%	120x10 ³	0.57x10 ⁻⁶	100x10 ⁻³	2.71x10 ⁶
Shaban et al. ^{s26}	50um RRAM + 0.5um CMOS	SNN	12	91%	1x10 ³	696x10 ⁻⁶	1.2	11x10 ³ (sim 65nm)
This work	180nmRRAM + 180nm CMOS	SNN	35	86.54%	180x10 ³	12.9x10 ⁻⁶	1.66x10 ⁻³	102x10 ³

Page 27:

“At the system level, we evaluated the hardware performance of the proposed neuromorphic network for the GSC dataset. The implementation achieves a compact area of 0.102 mm² and energy consumption of 12.9 × 10⁻⁶ J per second of speech while maintaining the classification accuracy. Compared with other representative works, our design demonstrates significant improvements in area and energy efficiency (Supplementary Table 4). Together, the single-neuron optimizations and system-level design enable a highly efficient and scalable architecture capable of performing real-

time time-series processing tasks with low power consumption and compact silicon footprint.”

Also, could the authors comment on the size/physical footprint for these circuits? It seems like the additional temporal complexity will require a bigger physical footprint. While the authors show an accuracy improvement over a baseline LIF-based network, understanding the trade-off space would improve the broad interest of this manuscript.

Our response: We sincerely appreciate the reviewer’s insightful comment regarding the trade-off between temporal complexity and physical footprint. We fully agree that introducing additional temporal dynamics—such as homeostatic and dendritic mechanisms—inevitably increases circuit complexity to some extent. To address this point, we have conducted a detailed quantitative analysis of the silicon area of each proposed neuron configuration when implementing the 35-class GSC dataset, as summarized in Supplementary Table 2.

As summarized in the table, the total silicon area increases from $78 \times 10^3 \mu\text{m}^2$ for the baseline LIF circuit to $102 \times 10^3 \mu\text{m}^2$ for the HSLIF + Dendrite configuration, corresponding to an approximately 30% increase in physical footprint. However, as shown in Supplementary Table 4, the overall area of our system remains an order of magnitude smaller than that of conventional digital or neuromorphic implementations, which typically exceed 1 mm^2 . This observation confirms that the additional homeostatic and dendritic modules introduce only a minor area overhead.

Taken together, these results demonstrate a favorable trade-off between area and accuracy. The proposed circuits achieve a balanced integration of functional complexity and resource efficiency: a modest increase in area yields meaningful gains in accuracy. We believe this analysis provides valuable insights into the co-design of scalable and area-efficient neuromorphic circuits capable of implementing advanced temporal learning mechanisms.

To address these important issues, we have added the Supplementary Table 2, along with a brief statement in the main text.

Page 25:

“To assess the hardware implications of these designs, we further evaluated and compared the silicon area of the four different model configurations for this task, as summarized in Supplementary Table 2. The results indicate that the additional homeostatic and dendritic modules introduce only a minor increase in area, while providing a 3.5% improvement in classification accuracy.”

Supplementary Table 2:

“Supplementary Table 2. Comparison of Four Network Configurations for the GSC Task.”

Dataset	Structure	Accuracy	Area (μm^2)	Basic circuit unit
GSC (35 classes)	LIF (VO ₂ based)	83.03%	78x10 ³	1C 1R 1VO ₂
GSC (35 classes)	HSLIF (VO ₂ based)	85.09%	93x10 ³	6T 2C 1R 1VO ₂
GSC (35 classes)	LIF + Dendrite (RRAM & VO ₂)	84.61%	86x10 ³	2T 3C 7R 2AMP 1DIO 1RRAM 2VO ₂
GSC (35 classes)	HSLIF + Dendrite (RRAM & VO ₂)	86.53%	102x10 ³	8T 4C 7R 2AMP 1DIO 1RRAM 2VO ₂

Minor:

I could have used a table that collected all the benchmark results so that performance of different models (LIF, homeostatic, dendritic, homeostatic + dendritic) could be compared with more conventional approaches on the different tasks/datasets.

page 15: “readout” is misspelled

page 15: I think “result” should be “results”.

Our response: We thank the reviewer for the valuable suggestion regarding comprehensive benchmark tables. In response, we have included detailed comparison tables in the Supplementary Materials. These tables summarize the performance of all four network configurations (LIF, homeostatic, dendritic, and homeostatic + dendritic) as well as other approaches on both the GSC and CWRU tasks. Specifically:

Supplementary Tables 1 and 3 report parameter counts and classification accuracies for different networks on the CWRU and GSC datasets, respectively.

Supplementary Table 2 compares the performance of the same network scale under different configurations (LIF, homeostatic, dendritic, and homeostatic + dendritic), highlighting the trade-offs between area and accuracy.

Supplementary Table 4 presents hardware-level evaluation results, including energy consumption and silicon area, and compares our system with other reported hardware implementations.

These additions provide a comprehensive view of model efficiency and performance across datasets, network configurations, and hardware implementations.

We also sincerely thank the reviewer for pointing out typographical and grammatical errors. We have carefully checked the entire manuscript to ensure that similar mistakes are corrected.

Supplementary Table 1:

“Supplementary Table 1. Comparative benchmark of different networks on the CWRU dataset.^{s1 – s7}”

	Model	Accuracy	Params	Flops
Chen et al. ^{s1}	ResNet50	99.92%	47.2M	718M
Yoo et al. ^{s2}	CNN0	96.64%	901K	1.8M
Rosa et al. ^{s3}	WDCNN	89.70%	40.7K	-
Wang et al. ^{s4}	CWT-2DCNN	83.10%	1.34M	840M
Wang et al. ^{s5}	MS-ER	98.80%	3.95M	2.8G
Lim et al. ^{s6}	SpikeCAN	99.86%	1.45M	-
Fang et al. ^{s7}	Spiking-ResNet	87.46%	3.85M	-
This work	SNN-HSLIF	92.14%	22K	140K

Supplementary Table 2:

“Supplementary Table 2. Comparison of Four Network Configurations for the GSC Task.”

Dataset	Structure	Accuracy	Area (μm^2)	Basic circuit unit
GSC (35 classes)	LIF (VO ₂ based)	83.03%	78x10 ³	1C 1R 1VO ₂
GSC (35 classes)	HSLIF (VO ₂ based)	85.09%	93x10 ³	6T 2C 1R 1VO ₂
GSC (35 classes)	LIF + Dendrite (RRAM & VO ₂)	84.61%	86x10 ³	2T 3C 7R 2AMP 1DIO 1RRAM 2VO ₂
GSC (35 classes)	HSLIF + Dendrite (RRAM & VO ₂)	86.53%	102x10 ³	8T 4C 7R 2AMP 1DIO 1RRAM 2VO ₂

Supplementary Table 3:

“Supplementary Table 3. Comparative benchmark of different networks on the GSC dataset.^{s8 – s17}”

	Model	Num Classes	Accuracy	Params	Flops
Niizumi et al. ^{s8}	M2D	35	98.50%	86M	-
Chen et al. ^{s9}	EAT	35	98.30%	88M	-
Andrade et al. ^{s10}	Attention-RNN	20	94.50%	202K	44.6M
Kim et al. ^{s11}	BC-ResNet-8	12	98.70%	321K	171.8M
Majumdar et al. ^{s12}	Matchbox-Net	35	97.40%	140K	34.2M
Bittar et al. ^{s13}	SNN-RadLIF	35	94.51%	1.2M	-
Hammouamri et al. ^{s14}	SNN-delays	35	95.35%	2.5M	-
Deckers et al. ^{s15}	SNN-d-cAdLIF	35	95.69%	610K	-
Sun et al. ^{s16}	SNN-delays	20(SHD)	92.45%	109K	-
D'agostino et al. ^{s17}	SNN-delays(fixed)	20(SHD)	90.88%	224K	-
Deckers et al. ^{s15}	SNN-LIF	35	86.21%	300k	-
This work	SNN-HSLIF-delays	35	86.53%	33K	3.13M

Supplementary Table 4:

“Supplementary Table 4. Comparison of hardware implementations for the GSC dataset task.^{s18 – s26}”

	Technology	Model	Num Classes	Accuracy	Frequency (Hz)	Energy for 1s speech (J)	Latency (s)	Area (um ²)
Zol. et al. ^{s18,19}	14nm CMOS + IBM PCM	CNN	11	89.30%	400x10 ⁶	78x10 ⁻⁶	348x10 ⁻⁶	144x10 ⁶
Giraldo et al. ^{s20}	65nm	LSTM	10	91%	250x10 ³	16.2x10 ⁻⁶	-	2.56x10 ⁶
Li et al. ^{s21}	65nm	GRU	2	98%	-	122.3x10 ⁻⁶	211x10 ⁻⁶	4.38x10 ⁶
Kim et al. ^{s22}	65nm	RNN	12	86%	250x10 ³	23x10 ⁻⁶	12.4x10 ⁻³	2.03x10 ⁶
Shan et al. ^{s23}	28nm	DSCNN	2	95%	40x10 ³	0.98x10 ⁻⁶	-	230x10 ³
Liu et al. ^{s24}	65nm	FS-CNN + LIF	4	95%	100x10 ³	7.7x10 ⁻⁶	18.7x10 ⁻³	917x10 ³
Wang et al. ^{s25}	65nm	SNN	5	90%	120x10 ³	0.57x10 ⁻⁶	100x10 ⁻³	2.71x10 ⁶
Shaban et al. ^{s26}	50um RRAM + 0.5um CMOS	SNN	12	91%	1x10 ³	696x10 ⁻⁶	1.2	11x10 ³ (sim 65nm)
This work	180nmRRAM + 180nm CMOS	SNN	35	86.54%	180x10 ³	12.9x10 ⁻⁶	1.66x10 ⁻³	102x10 ³

Comments from Reviewer #3

The manuscript reports on the utilization of VO₂-based devices to implement homeostatic neurons, and on a combination of RRAM and VO₂ devices to implement dendritic branches. The authors build a system consisting of a single VO₂ memristor device and a small 1T1R conventional array connected through an FPGA. System-level simulations of the proposed SNN, incorporating homeostatic neurons and dendritic structures, are tested on the CWRU dataset and for speech recognition tasks. Results are promising; however, there is a lack of comparison with state-of-the-art approaches, making it difficult to assess the benefits of the proposed architecture. While I find the presented method and conclusions valuable, I believe the authors need to more clearly articulate the novelty of their work and substantiate it with more detailed experimental results. Additionally, certain aspects of the manuscript require supplementary data and explanations to aid reader comprehension.

Our response: We sincerely appreciate the reviewer's positive evaluation of our work and the constructive suggestions.

Regarding the comparison with state-of-the-art approaches: We fully agree that a direct comparison is essential to highlight the advantages of the proposed VO₂ and RRAM based neuromorphic system. Accordingly, we have added comprehensive benchmark tables in the revised manuscript and supplementary materials (see Supplementary Table 1-5). These tables summarize classification accuracy, network complexity (number of parameters and FLOPs), and energy efficiency compared to hardware implementations on speech recognition datasets. The results demonstrate that our system achieves comparable performance while maintaining significantly lower computational cost and physical footprint. In response to the reviewer's valuable comments regarding the articulation of novelty and the need for supplementary data, we have substantially revised and improved both the main manuscript and the supplementary materials accordingly.

1. The idea of using a single Threshold Switching (TS) device with a capacitor — the TS/C structure — to implement a spiking neuron is not novel, and no comparison with previously published works is provided. The only apparent novelty in Fig. 2 is that the input is a current rather than a voltage; however, the advantages of this choice are not explained.

Our response: We thank the reviewer for this valuable comment. We would like to clarify that the main novelty of our work does not lie in using the TS/C structure itself to implement a spiking neuron, which has indeed been explored in prior studies, but rather in leveraging this structure as a fundamental building block to construct higher-level functional units—including homeostatic neurons and dendritic delay circuits. In our architecture, the TS/C unit serves as the physical neuron core that exhibits self-sustained threshold switching and recovery dynamics. When embedded into a feedback circuit, it realizes homeostatic firing regulation, where the firing frequency can be adaptively modulated under constant input amplitude, mimicking biological self-regulation mechanisms. Furthermore, by coupling TS/C structure with RRAM-based dendritic branch, we demonstrate tunable dendritic delays, which collectively enable richer temporal processing at the circuit level. This hierarchical integration of TS-based neurons and RRAM-based dendritic elements represents a novel system-level realization that has not been reported in previous TS/C neuron studies.

The adoption of a current-driven configuration in Fig. 2 is a deliberate design choice that enables both homeostatic functionality and system-level compatibility within the proposed neuromorphic framework. In contrast to voltage-driven implementations, where the input amplitude directly determines the firing behavior and the voltage applied to the TS/C structure is difficult to regulate, our current-driven neuron enables spike-frequency modulation through a feedback loop under a constant input amplitude. This mechanism, utilized in the homeostatic neuron design shown in Fig. 3, provides a direct circuit-level means to achieve homeostasis of neural activity, which is essential for processing temporally continuous signals in neuromorphic systems. Moreover, since RRAM-based synapses naturally generate current outputs,

the use of a current-input neuron removes the need for additional current-voltage conversion interfaces. This approach greatly simplifies the interconnect circuitry, reduces both power and area overhead, and facilitates efficient hierarchical integration of dendritic computation, synaptic weighting, and neuronal firing.

In response to the reviewer's concern, we have revised the corresponding sections in the manuscript as detailed below.

Page 9:

“The voltage-driven neuron designed for spike encoding generates spike signals primarily through voltage division between the VO₂ device and a fixed-value resistor, along with the threshold switching behavior of the VO₂ device. While this structure effectively emulates spiking behavior, realizing more advanced functions such as homeostatic regulation would require structural enhancements to the neuron, including the incorporation of feedback mechanisms that dynamically adjust the input. Considering that voltage control in circuits is generally more complex than current control, we physically designed and tested a current-input neuron to facilitate system integration and enable more sophisticated functionalities, as illustrated in Fig. 2g. We applied gradient current inputs and experimentally measured the corresponding changes in membrane potential and spike frequency, as shown in Fig. 2h.”

2. From my understanding, the neuron parameters in the proposed design are fixed, determined by the values of the resistances and capacitors. In contrast, biological neural networks typically contain neurons with heterogeneous parameters that can adapt through homeostatic plasticity rules. What the authors refer to as “homeostatic plasticity” in Fig. 2g–i appears to be simply a change in output frequency when the input signal (current, in this case) is varied. In neuroscience, neuron homeostatic plasticity refers to a set of mechanisms that maintain stable activity levels over time, despite changes in input or network dynamics. It acts as a self-regulating process to prevent neural circuits from becoming either overactive or underactive.

Our response: We thank the reviewer for this insightful comment and the opportunity to clarify our design.

As correctly pointed out, the neuron parameters in Fig. 2 are fixed, determined by the resistance and capacitance values, and the firing frequency is modulated by the input current amplitude. However, the neuron shown in Fig. 2 represents the basic neuron structure, rather than the homeostatic neuron itself. Its purpose is to analyze the fundamental behavior of the basic neuron, which serves as the building block for the homeostatic neuron design presented later in Fig. 3. The homeostatic neuron in Fig. 3 implements the functionality pointed out by the reviewer, acting as a self-regulating mechanism to prevent neural circuits from becoming either overactive or underactive.

In the homeostatic neuron (Fig. 3), an additional feedback loop is incorporated to automatically regulate the output spiking activity, thereby realizing homeostatic functionality at the circuit level. The results in Fig. 2g – i therefore do not represent homeostatic plasticity by itself, but rather illustrate how the basic neuron responds to different input currents, providing the foundation for the homeostatic mechanism introduced subsequently.

To avoid potential misunderstanding by readers, we have revised the following parts of the manuscript accordingly.

Page 9:

“The voltage-driven neuron designed for spike encoding generates spike signals primarily through voltage division between the VO₂ device and a fixed-value resistor, along with the threshold switching behavior of the VO₂ device. While this structure effectively emulates spiking behavior, realizing more advanced functions such as homeostatic regulation would require structural enhancements to the neuron, including the incorporation of feedback mechanisms that dynamically adjust the input. Considering that voltage control in circuits is generally more complex than current control, we physically designed and tested a current-input neuron to facilitate system integration and enable more sophisticated functionalities, as illustrated in Fig. 2g. We applied gradient current inputs and experimentally measured the corresponding changes in membrane potential and spike frequency, as shown in Fig. 2h.”

3. The circuit proposed in Fig. 3 implements excitatory/inhibitory behavior; however, this is different from homeostatic plasticity. Excitatory/inhibitory behavior describes the instantaneous effect of an input on a neuron, whereas homeostatic plasticity is a long-term adaptive process that adjusts the excitatory/inhibitory balance to keep firing rates within a healthy range. Implementing true homeostatic plasticity would require some form of non-volatile memory to store parameter changes over the long term.

Our response: Thank you very much for raising this important point and giving us the opportunity to clarify our intention.

First, we agree with the reviewer that excitatory or inhibitory behavior is fundamentally different from neuronal homeostatic regulation, and that such excitatory or inhibitory responses typically refer to a neuron's reaction to inputs. However, we would like to clarify that what we demonstrate in this work is homeostatic regulation, rather than a simple excitatory or inhibitory response.

According to Davis' work¹¹, "a homeostatic system can be defined as having a constant output" and such systems typically satisfy the following requirements: maintaining a stable output, having a feedback mechanism, and being able to readjust activity levels following perturbations. This reference does not specify the exact time scale of homeostatic regulation in neurons. From other studies^{12–15}, we found that neuronal homeostatic regulation can occur over long time scales, such as several hours to days, but there are also reports showing that some neurons can achieve rapid homeostatic adjustment over very short time scales, e.g., a few minutes^{16–18}.

Furthermore, some studies have shown that homeostatic regulation in neurons is primarily achieved through synaptic scaling, membrane excitability, and intrinsic excitability^{11,19}. Regarding the reviewer's comment on "long-term adaptation," this mainly refers to one mechanism of homeostatic regulation—synaptic scaling or synaptic plasticity—which involves local changes in synaptic activation to produce local adaptations, as well as network-wide changes in activity to maintain the overall balance between excitation and inhibition²⁰. Such regulation typically involves synaptic

receptors, related proteins, and the activation of enzymes, which takes a relatively long timescale to complete. Therefore, in hardware implementations, as the reviewer suggested, achieving this form of long-term plasticity may indeed require some form of non-volatile memory to store the long-term regulatory state.

However, the homeostatic neuron proposed in our manuscript refers specifically to the functional behavior of homeostasis, i.e., the ability of the neuron to restore its firing rate to a stable state after a perturbation. We do not claim to reproduce the molecular mechanisms underlying biological synaptic plasticity. Instead, our goal is to emulate the behavioral manifestation of firing-rate homeostasis at the circuit level. What's more, we note that similar neuromorphic hardware implementations also realize homeostatic regulation on short timescales. For example, in the work by Ning Qiao et al.²¹, a CMOS homeostatic neuron exhibits firing-rate stabilization on the order of seconds by modulating capacitor charging/discharging dynamics. Likewise, the DYNAP-SE2 multi-core neuromorphic computing chip by Ole Richter et al.²² incorporates homeostatic mechanisms that operate on similar time scales rather than long-term biological timescales.

In our design, the neuron output spikes are routed through a feedback loop that charges a capacitor near the input node. The increasing voltage modulates the conduction of two transistors, thereby regulating the VO₂-based spike frequency. Although our circuit does not store information over long durations as non-volatile devices do, the capacitor-based feedback enables short-timescale activity regulation.

To avoid potential misunderstandings and to more clearly convey the contributions of our work, we have revised the corresponding passage in the manuscript as follows.

Page 11:

“Although basic LIF neurons are easy to implement, they tend to generate a fixed spike frequency under constant input and lack biologically inspired mechanisms like homeostatic regulation to modulate their firing behavior. To address these limitations, we propose a homeostatic neuron, as illustrated in the middle of Fig. 3a, which incorporates a feedback loop to regulate its firing behavior. This feedback mechanism

enables the neuron to partially restore its firing rate after perturbations, thereby improving its ability to encode temporal information.”

4. It also appears that the results shown in Fig. 3 are purely simulation-based. If this is the case, it should be explicitly stated in the manuscript.

Our response: We sincerely thank the reviewer for highlighting this point. The results shown in Fig. 3 are indeed obtained from circuit-level simulations. We would also like to explain the practical reasons why these results are currently limited to simulations. In our board-level prototype, the leakage currents of the transistors already reach the microampere (μA) range, which is comparable to the μA -level currents we intend to modulate in the proposed neuron circuit. As a result, the desired behavior is extremely difficult to measure reliably at the board level. However, precise control of μA -level currents can be achieved at the chip (wafer) level. Implementing this at the chip level represents the next step of our work, and we are actively pursuing this goal in the upcoming year. At the current stage, we therefore rely on circuit simulations to demonstrate the phenomenon shown in Fig. 3. We hope this explanation addresses the reviewer’s concern, and we sincerely appreciate their careful and constructive feedback. To ensure clarity for readers, we have now explicitly stated this in both the main text and the figure caption.

In response to the reviewer’s comment, we have revised the corresponding section of the manuscript to explicitly indicate that these results are obtained from circuit-level simulations.

Page 11, 12:

“To illustrate how these transistors modulate the spike frequency through current supply and leakage, we simulated the output spike frequency under a fixed 40 μA current while varying the voltage at node1. The simulated results in the left of Fig. 3a demonstrate that as the voltage at node1 increases, the output spike frequency shows a non-monotonic variation, first decreasing and then rising. Further details regarding

the specific contributions of each NMOS transistor can be found in Supplementary Figs. 11, 12. We simulated the effect of applying a pulse with a period of 10 μ s and a pulse width of 500 ns as the spike output of the neuron to the feedback module. The simulation results shown in the right of Fig. 3a, demonstrate that as the number of applied pulses to the feedback module increases, the voltage at node1 gradually rises under different gate bias voltages of M_6 . The charging rate of node1 can be controlled by adjusting the gate bias voltage of M_6 . These simulations confirm that the feedback module meets the requirements for the frequency self-tuning capability of our neuron.

Page 12:

“To verify this homeostatic regulation, we conducted transient simulations by applying a pulse with fixed amplitude to the neuron model and monitored the gate voltage, output spikes, and membrane potential. The simulated waveforms, shown in Fig. 3c, clearly illustrate the transient modulation and subsequent recovery of spike frequency, where each generated pulse causes the voltage at node1 to increase, leading the output spike frequency to initially decrease and then rise.”

5. Additionally, the proposed neuron (both in Fig. 2 and Fig. 3) can only integrate positive currents. While the target application described here does not require integration of negative currents, high-precision spiking neural networks generally require handling both positive and negative weights. This means neurons should be able to integrate both excitatory and inhibitory currents. Without this capability, the proposed design may face limitations in broader SNN implementations.

Our response: We sincerely appreciate the reviewer’s insightful comment. We fully agree that supporting both excitatory and inhibitory currents would broaden the applicability of the proposed neuron to more general SNN architectures.

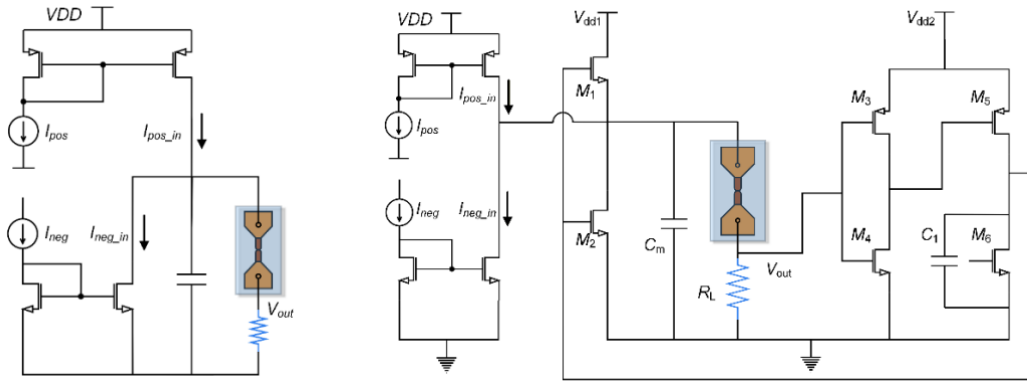
Regarding the reviewer’s comment on the “integration of negative currents”, our understanding is that this refers to the weighted summation of input signals after they

have been multiplied by either positive or negative synaptic weights. In this context, the accumulation of both positive and negative weighted currents is typically performed at the synaptic or crossbar array level prior to entering the neuron. In our implementation, this weighted summation is supported through a differential-array structure, as illustrated in the Supplementary Fig. 25. The positive and negative weight contributions are represented by two complementary current paths, whose difference naturally yields a net input current that already incorporates both excitatory and inhibitory effects. This net current is then fed into our neuron circuit. Therefore, our design enables the integration of negative currents through differential-weight encoding and current-mode accumulation.

Using the differential-weight implementation described above, the input signals are multiplied by their corresponding positive and negative weights, and the resulting currents are accumulated. In most cases, the net accumulated current is positive; however, in certain situations the summed result may indeed become negative. If this is the type of “integration of negative currents” the reviewer refers to, we would like to clarify that in our network simulation, the accumulated current is passed through a ReLU-type nonlinearity before being delivered to the neuron. This ensures that the neuron always receives a non-negative input current, which is compatible with the circuit structure we propose. This treatment is consistent with many existing SNN implementations that use memristor-based synaptic arrays. In tasks such as speech recognition^{23–25}, image classification^{26–28}, and physiological signal processing^{29,30}, prior works have also adopted ReLU or similar rectification at the synapse–neuron interface and have reported competitive performance. These results indicate that rectifying the accumulated synaptic current does not significantly degrade model accuracy and supports the feasibility of our design choice.

While the above mechanism is sufficient for the requirements of our current system, we agree that enabling the neuron circuit itself to directly process both positive and negative input currents would further generalize the design. To accommodate this capability, an extended version of our neuron structure can be implemented by adding a dedicated negative-current conduction path near the membrane capacitor, as

illustrated in the following figures. In this extended implementation, two complementary current mirrors are used: the top current mirror injects positive currents into the membrane node, while the bottom current mirror injects negative currents. Consequently, the membrane capacitor is charged when a positive current is applied and discharged when a negative current is applied. This symmetric current-handling structure allows the neuron to explicitly integrate both excitatory and inhibitory inputs at the circuit level.



The proposed circuits enable hardware support for both positive and negative input currents by incorporating two complementary current-mirror branches. The upper current mirror generates a positive charging current I_{pos_in} , which increases the membrane capacitor voltage in response to positive inputs. Conversely, the lower current mirror produces a discharging current I_{neg_in} , allowing negative inputs to decrease the capacitor voltage. This bidirectional current-integration mechanism effectively extends the neuron’s capability to handle signed inputs. The VO₂ device and the load resistor connected at the output node (V_{out}) provide the subsequent threshold-switching behavior used in the neuron model.

To address this comment, we have added Supplementary Fig. 25 and Supplementary Note 4 and the following brief discussions in the main text:

Page 17:

“Each dendritic branch within the circuit array incorporates multiple synaptic processing elements connected through programmable delay units, mimicking the temporal dynamics inherent in biological dendritic computation. Synaptic weights are realized using RRAM devices that provide non-volatile and continuously tunable

conductance states for flexible weight modulation. By adopting a paired differential RRAM configuration, the system reliably encodes both positive and negative weights, providing hardware-level support for signed computational parameters. The details of the signed-weight mechanism are provided in the Supplementary Fig. 25 and Supplementary Note. 4.”

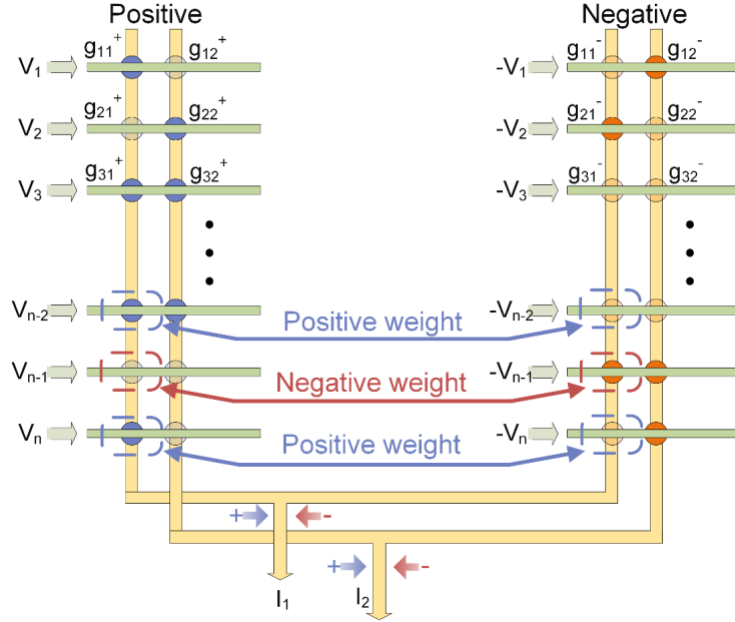
Supplementary Note. 4:

“To support both positive and negative synaptic weights in the proposed hardware neural network, we implemented a differential-pair conductance mapping scheme based on a dual-column RRAM crossbar structure (Supplementary Fig. 25). For each input voltage V_i , two parallel RRAM devices (the positive branch g_{i1}^+ and the negative branch g_{i1}^-) are allocated symmetrically in the crossbar. The left devices store the positive component of the weight, whereas the right devices store the negative component. A scalar synaptic weight w_i is decomposed into two non-negative conductance values: $w_i = G_i^+ - G_i^-$, where G_i^+ is programmed into the positive RRAM device g_{i1}^+ , G_i^- is programmed into the negative RRAM device g_{i1}^- . To realize weight subtraction physically, the two halves of the array are biased using opposite input polarities. The source lines on both sides are routed and merged at the output summing node using a pair of differential current-collecting paths. This differential configuration naturally enables both positive and negative weight representations without relying on devices that provide negative conductance. In addition, device variations and non-idealities are alleviated through differential cancellation, as the two branches experience nearly identical voltage conditions, wire resistances, and temporal fluctuations.”

Supplementary Fig. 25:

“Supplementary Fig. 25 Schematic illustration of the differential-pair conductance mapping used to realize positive and negative weights in an RRAM crossbar array. Each input V_i is connected to two parallel devices storing the positive and negative conductance components, respectively. The left devices receive the original input voltages ($+V_i$), while the right devices receive their inverted counterparts ($-V_i$). The resulting branch currents are summed through separate routing lines and combined at

the output to produce a signed weighted current via differential subtraction. This structure enables the hardware representation of signed weights using only positive-conductance memory devices.”



6. A comparison with purely CMOS-based solutions would also be important to better illustrate the advantages of the proposed approach.

Our response: We thank the reviewer for the suggestion to provide a comparison with purely CMOS-based solutions to better illustrate the advantages of our approach. To address this point, we performed a two-level comparison: at the single-neuron level and at the system/application level.

We compared our VO₂-based neuron with a range of published neuron designs, including purely CMOS implementations and other hybrid approaches. Key metrics include area, energy per spike, firing frequency, and circuit composition. As summarized in the Supplementary Table 5, our neuron supports both frequency increase and decrease with return to a stable state—a feature not fully supported in most CMOS-only neurons—while maintaining competitive energy efficiency (19.29 pJ/spike), firing frequency (180 kHz) and a compact transistor area (76.8μm²). In contrast, purely

CMOS neurons generally require a larger number of transistors and auxiliary components to achieve similar functionalities, leading to higher energy consumption and area.

We further compared the performance of our SNN system on the GSC dataset with other hardware implementations of neural networks, including SNNs, CNNs, and LSTM based systems. The metrics considered include number of classes, classification accuracy, operating frequency, energy, latency, and area. As shown in the following table, our system achieves a high operating frequency (180 kHz) with low energy consumption (12.9 μ J for 1-second speech) and compact area ($102 \times 10^3 \mu\text{m}^2$), while maintaining competitive accuracy (86.54%). Compared with purely CMOS implementations (e.g., 14 nm CNN, 65 nm LSTM/GRU, 65nm SNN), our approach demonstrates superior energy efficiency and area efficiency, highlighting the advantages of the proposed system.

In summary, these comparisons clearly illustrate that our VO₂-based neuron and system provide favorable trade-offs between energy, area, and functional flexibility compared with purely CMOS-based solutions.

To address this, we have included Supplementary Tables 4 and 5, including the discussion above as well as more detailed descriptions of the simulations on Page 25.

Page 25, 26:

“In addition, we evaluated the hardware-level performance of the proposed neuromorphic system for the 35-class GSC dataset. The energy consumption and silicon area of our system implementation are quantitatively analyzed at 180nm node and compared with other representative neuromorphic and digital hardware systems, as summarized in Supplementary Table 4. Despite being implemented at a larger technology node (180 nm), our system demonstrates up to an order of magnitude reduction in area and notable improvements in energy efficiency, highlighting its strong potential for scalable and energy-efficient neuromorphic hardware integration for temporal signal processing tasks.”

Supplementary Table 4:

“Supplementary Table 4. Comparison of hardware implementations for the GSC dataset task.^{s18 – s26}”

	Technology	Model	Num Classes	Accuracy	Frequency (Hz)	Energy for 1s speech (J)	Latency (s)	Area (μm^2)
Zol. et al. ^{s18,19}	14nm CMOS + IBM PCM	CNN	11	89.30%	400×10^6	78×10^{-6}	348×10^{-6}	144×10^6
Giraldo et al. ^{s20}	65nm	LSTM	10	91%	250×10^3	16.2×10^{-6}	-	2.56×10^6
Li et al. ^{s21}	65nm	GRU	2	98%	-	122.3×10^{-6}	211×10^{-6}	4.38×10^6
Kim et al. ^{s22}	65nm	RNN	12	86%	250×10^3	23×10^{-6}	12.4×10^{-3}	2.03×10^6
Shan et al. ^{s23}	28nm	DSCNN	2	95%	40×10^3	0.98×10^{-6}	-	230×10^3
Liu et al. ^{s24}	65nm	FS-CNN + LIF	4	95%	100×10^3	7.7×10^{-6}	18.7×10^{-3}	917×10^3
Wang et al. ^{s25}	65nm	SNN	5	90%	120×10^3	0.57×10^{-6}	100×10^{-3}	2.71×10^6
Shaban et al. ^{s26}	50um RRAM + 0.5um CMOS	SNN	12	91%	1×10^3	696×10^{-6}	1.2	11×10^3 (sim 65nm)
This work	180nmRRAM + 180nm CMOS	SNN	35	86.54%	180×10^3	12.9×10^{-6}	1.66×10^{-3}	102×10^3

Page 27:

“We conduct a comparative analysis between the constructed neuron circuit and representative CMOS or memristor based implementations. The results indicate that the proposed memristor-based system for time-series signal processing offers improved area and energy efficiency. The homeostatic neuron, optimized at both the device and circuit levels, attains a compact area of $76.8 \mu\text{m}^2$ and ultra-low energy consumption of 19.29 pJ per spike (Supplementary Table 5) ... At the system level, we evaluated the hardware performance of the proposed neuromorphic network for the GSC dataset. The implementation achieves a compact area of 0.102 mm^2 and energy consumption of $12.9 \times 10^{-6} \text{ J}$ per second of speech while maintaining the classification accuracy. Compared with other representative works, our design demonstrates significant improvements in area and energy efficiency (Supplementary Table 4). Together, the single-neuron optimizations and system-level design enable a highly efficient and scalable architecture capable of performing real-time time-series processing tasks with low power consumption and compact silicon footprint.”

Supplementary Table 5:

“Supplementary Table 5. Detailed comparison between neuron circuits.^{s26 – s32}”

	Neuron type	Area (um ²)	Energy (J/spike)	Frequency (Hz)	Circuit
Shaban et al. ^{s26}	ALIF (frequency increase)	11435.7 (all CMOS 65nm)	158x10 ⁻¹²	1x10 ³	6T + 1OxRAM
Ahma. et al. ^{s27}	ALIF (frequency increase)	863 (CMOS 180nm)	140x10 ⁻¹²	~1x10 ³	24T + 3C + RRAM
Moradi et al. ^{s28}	ALIF (frequency decrease)	8572 (CMOS 180nm)	260x10 ⁻¹²	~100x10 ³	30T + 3C
Chak. et al. ^{s29}	ALIF (frequency decrease)	2177 (CMOS 65nm)	~28.6x10 ⁻⁹	~6.7x10 ³	T and C
Yang et al. ^{s30}	HH	100 (5um)	1.06x10 ⁻¹²	320X10 ³	2R + 2C + 2NbO ₂ + 2 voltage source
Xiao et al. ^{s31}	LIF (frequency decrease)	no area info	~600x10 ⁻¹²	160x10 ³	2R + 2C + 2NbO ₂ + 2ECRAM
Qiao et al. ^{s32}	LIF (frequency increase or decrease and return to stable)	1848 (CMOS 180nm)	~72x10 ⁻¹²	< 1x10 ³	Purely CMOS
This work	LIF (frequency increase or decrease and return to stable)	~30197(main cost by cap) / ~76.8 (only transistors)	19.29x10 ⁻¹²	180x10 ³	6T + 2C + 1R + 1VO ₂

7. Regarding dendritic behavior, my understanding is that it requires the ability to modulate both the synaptic weight and the delay for each synapse [1]. If I have understood correctly, the proposed circuit uses non-volatile RRAM to modulate delays — but how are the weights modulated? Furthermore, what is the range of time constants achievable with your design? In many realistic models, time constants on the order of milliseconds or longer are required.

Reference:

[1] D’Agostino, S., Moro, F., Torchet, T. et al. DenRAM: neuromorphic dendritic architecture with RRAM for efficient temporal processing with delays. Nat Commun 15, 3446 (2024).

Our response: We thank the reviewer for the insightful comments regarding dendritic behavior and the modulation of both synaptic weight and delay.

First, we carefully reviewed the reference suggested by the reviewer. In that work, dendritic behavior is implemented by assigning each synapse both a programmable RRAM for weight modulation and an additional RRAM device for delay. We acknowledge that this represents a valid hardware-level solution. However, such an implementation is not the only architectural choice. We also surveyed other studies in which the dendritic processing pathway and the weighting element are treated as separate functional modules. These works demonstrate that delay mechanisms or dendritic processing can be independently modeled or realized without binding them to the synaptic weighting device itself, thereby offering alternative circuit-level or architectural design options.

For example, Zheng et al.³¹ exploited dendritic temporal computing capabilities to construct an SNN architecture in which multiple synapses reside on the same dendritic branch and share a common dendritic time constant. In their approach, the input–weight multiplication and accumulation (synapses’ function) are performed first, and only then passed into the dendritic module for further spatiotemporal processing. Patino-Saucedo et al.³¹ adopted strategy in which multiple synapses share the same delay value, rather than assigning a distinct delay to each synaptic input. Similarly, in Intel’s neuromorphic processor Loihi³², the architectural diagrams clearly indicate that synaptic units and dendritic units are implemented as separate functional blocks. The large-scale neuromorphic architecture explored by Boyle et al.³³ also adopts a physically separated “synapse–dendrite” design.

Consistent with these earlier designs, our design also separates the synaptic computation from dendritic temporal processing, in which multiple synaptic weights can share a dendritic delay, thereby greatly reducing the number of required delay circuits as well as the area cost. In our implementation, each neuron is equipped with multiple dendritic branches, and different branches are assigned different delay values. Synapses located on the same dendritic branch share an identical delay, whereas synapses on different branches experience distinct delays. Specifically, the synaptic weights are implemented using a 1T1R RRAM array, which performs the weighted summation of inputs. The resulting output is then fed into the dendritic circuit, which

is responsible for generating the desired temporal delay. Thus, the circuit shown in Fig. 4 primarily implements the delay mechanism.

Regarding the reviewer's question on time constants, our design provides tunability through two parameters: (1) the conductance state of the RRAM device, and (2) the capacitance connected in parallel with the VO₂ device. As shown in Fig. 4i and Supplementary Fig. 23, when the capacitance is fixed at 800 pF, varying the RRAM conductance results in delays ranging from approximately 45 μs to 63 μs. Furthermore, when the RRAM device is programmed to a conductance of 20 μS and the capacitance is increased (see Supplementary Fig. 24), the achievable time constant can be extended from the microsecond range into the millisecond regime (15 μs-5ms). This demonstrates that our design can cover the biologically relevant time scales required by many dendritic computation models.

To clearly present our work and avoid potential misunderstandings, we have revised the original text in page 2, 15, 17 and added Supplementary Fig. 24, 25, Supplementary Note. 4.

Page 2:

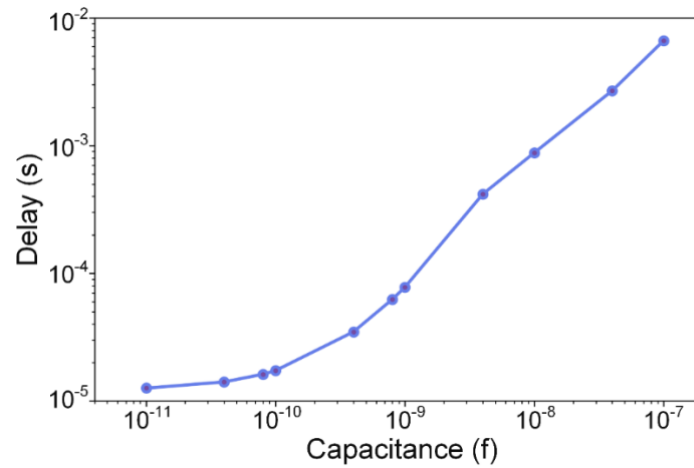
“The dendritic module, realized as a CMOS-RRAM and VO₂ device co-designed and integrated at the board level, includes programmable parameters for adjusting spike delays, enabling temporal feature extraction across multiple timescales.”

Page 15, 16:

“Fig. 4j and Supplementary Fig. 23 show the experimentally measured output spikes in response to the applied pulse inputs under different RRAM conductance states, exhibiting delays ranging from 45 μs to 63 μs. When the RRAM device is programmed to a low-conductance state, the pulse input results in a larger delay in the output. Notably, by varying the capacitance parameters, the temporal delay can be flexibly tuned across a broad range from 15 μs up to 5 ms, as shown in Supplementary Fig. 24.”

Supplementary Fig. 24:

“Supplementary Fig. 24 The RRAM device was programmed to a conductance state of $20\ \mu\text{S}$. By adjusting the capacitance connected in parallel with the VO_2 device, the output pulse delay can be effectively tuned. Experimental results indicate that as the parallel capacitance increases, the delay in pulse generation becomes longer.”



Page 17:

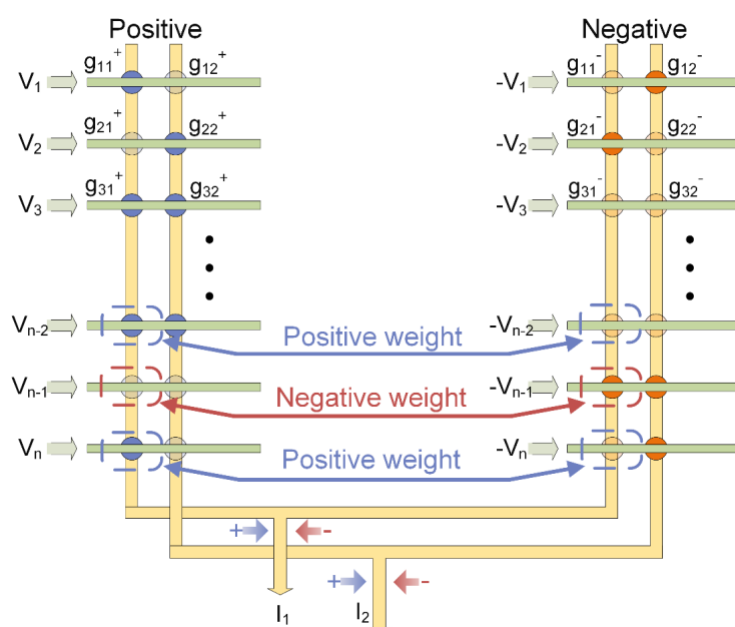
“Each dendritic branch within the circuit array incorporates multiple synaptic processing elements connected through programmable delay units, mimicking the temporal dynamics inherent in biological dendritic computation. Synaptic weights are realized using RRAM devices that provide non-volatile and continuously tunable conductance states for flexible weight modulation. By adopting a paired differential RRAM configuration, the system reliably encodes both positive and negative weights, providing hardware-level support for signed computational parameters. The details of the signed-weight mechanism are provided in the Supplementary Fig. 25 and Supplementary Note. 4.”

Supplementary Note. 4:

“To support both positive and negative synaptic weights in the proposed hardware neural network, we implemented a differential-pair conductance mapping scheme based on a dual-column RRAM crossbar structure (Supplementary Fig. 25). For each input voltage V_i , two parallel RRAM devices (the positive branch g_{i1}^+ and the negative branch g_{i1}^-) are allocated symmetrically in the crossbar. The left devices store the positive component of the weight, whereas the right devices store the negative component. A scalar synaptic weight w_i is decomposed into two non-negative conductance values: $w_i = G_i^+ - G_i^-$, where G_i^+ is programmed into the positive RRAM device g_{i1}^+ , G_i^- is programmed into the negative RRAM device g_{i1}^- . To realize weight subtraction physically, the two halves of the array are biased using opposite input polarities. The source lines on both sides are routed and merged at the output summing node using a pair of differential current-collecting paths. This differential configuration naturally enables both positive and negative weight representations without relying on devices that provide negative conductance. In addition, device variations and non-idealities are alleviated through differential cancellation, as the two branches experience nearly identical voltage conditions, wire resistances, and temporal fluctuations.”

Supplementary Fig. 25:

“Supplementary Fig. 25 Schematic illustration of the differential-pair conductance mapping used to realize positive and negative weights in an RRAM crossbar array. Each input V_i is connected to two parallel devices storing the positive and negative conductance components, respectively. The left devices receive the original input voltages ($+V_i$), while the right devices receive their inverted counterparts ($-V_i$). The resulting branch currents are summed through separate routing lines and combined at the output to produce a signed weighted current via differential subtraction. This structure enables the hardware representation of signed weights using only positive-conductance memory devices.”



8. The proposed circuit requires the integration of VO₂ memristors, conventional non-volatile memristors, and CMOS circuitry. However, the feasibility and cost implications of co-integrating VO₂ memristors with conventional non-volatile memristors are not discussed.

Our response: We appreciate the reviewer’s insightful comment. In the current study, our goal is to demonstrate the functional feasibility of the proposed architecture rather

than to realize monolithic wafer- or chip-level co-integration of VO₂ devices, non-volatile memristors, and CMOS circuits.

At this stage, our implementation relies on a PCB-level system integration approach, where the control circuits, the packaged VO₂ devices, and the RRAM array are mounted on the same board. This method allows us to experimentally validate the system behavior with lower cost and complexity.

We agree that wafer-level co-integration is an important long-term direction. In fact, several existing studies already indicate that a monolithic integration pathway for VO₂ devices, HfO₂/TaO_x RRAMs, and silicon-based MOSFETs is technically feasible. When pursuing such integration, considerations would include the compatibility of VO₂ deposition with the front-end-of-line (FEOL) processes, the interaction between VO₂ and the HfO₂/TaO₂ active layers, and the performance of VO₂ devices when implemented in the back-end-of-line (BEOL) stack. Firstly, VO₂ films can be deposited at conditions below the thermal budget and prior works have successfully demonstrated the integration of VO₂ devices with silicon MOSFETs³⁴, HfO₂-based memristors³⁵, and TaO_x-based memristors³⁶. Besides, the deposition temperature of VO₂ films can be further lowered by incorporating a Cr₂O₃ buffer layer beneath the VO₂ layer³⁷. Secondly, the integration of VO₂, even in direct contact with active HfO₂³⁵ or TaO_x³⁶ layers, does not degrade the performance of the nonvolatile memristor, indicating a high degree of material compatibility. Finally, VO₂ devices deposited on Si or SiO₂ substrates^{34,38}, instead of sapphire, still demonstrate excellent volatile threshold-switching behavior suitable for implementing the proposed neuron. These results collectively indicate that monolithic CMOS-RRAM-VO₂ integration is a realistic option, with a clear technological pathway for ensuring that all device types maintain their intended functionality after integration.

To address these important issues, we have revised the content on pages 2, 5, 17 to avoid any potential misunderstanding.

Page 2:

“The dendritic module, realized as a CMOS-RRAM and VO₂ device co-designed and integrated at the board level, includes programmable parameters for adjusting spike delays, enabling temporal feature extraction across multiple timescales.”

Page 5:

“Furthermore, we implement dendritic circuits by integrating packaged VO₂ devices with RRAM arrays and CMOS circuits on a printed circuit board (PCB), incorporating programmable delays across branches to enable temporal processing.”

Page 17:

“The dendritic computation is achieved through hybrid analog circuits assembled at the PCB level using packaged VO₂ devices, RRAM arrays, and discrete CMOS amplifier components, enabling programmable temporal dynamics and nonlinear processing.”

9. The authors claim: “The dendritic computation is achieved through hybrid analog circuits incorporating VO₂ devices and CMOS amplifiers, facilitating programmable temporal dynamics and nonlinear integration.” Can you confirm whether these hybrid circuits have actually been designed and tested, and whether VO₂ devices have been co-integrated on CMOS circuits? It is not clear what is simulated and what has actually been fabricated and tested.

Our response: We thank the reviewer for raising this important point.

We would like to clarify that our work involves PCB-level system integration rather than chip-level or wafer-level co-integration of VO₂ devices with CMOS circuitry. The statement “hybrid analog circuits incorporating VO₂ devices and CMOS amplifiers” refers specifically to the assembly of discrete CMOS components together with packaged VO₂ devices on a printed circuit board, where the complete neuromorphic circuits were constructed and experimentally characterized. We did not integrate VO₂ devices monolithically into a CMOS process. The CMOS elements used in the

experiments, such as operational amplifiers, are commercial off-the-shelf discrete components. The demonstration of programmable dendritic delays is carried out at the PCB level using these discrete CMOS devices together with packaged VO₂ devices.

To present our work more accurately and prevent any potential misunderstanding, we have revised the relevant descriptions on pages 2, 5, 17 to explicitly state that the demonstrated circuits are implemented through PCB-level integration of discrete CMOS components, RRAM devices, and packaged VO₂ devices.

Page 2:

“The dendritic module, realized as a CMOS-RRAM and VO₂ device co-designed and integrated at the board level, includes programmable parameters for adjusting spike delays, enabling temporal feature extraction across multiple timescales.”

Page 5:

“Furthermore, we implement dendritic circuits by integrating packaged VO₂ devices with RRAM arrays and CMOS circuits on a printed circuit board (PCB), incorporating programmable delays across branches to enable temporal processing.”

Page 17:

“The dendritic computation is achieved through hybrid analog circuits assembled at the PCB level using packaged VO₂ devices, RRAM arrays, and discrete CMOS amplifier components, enabling programmable temporal dynamics and nonlinear processing.”

10. Is the RRAM array capable of performing analog in-memory computing, or is it simply a conventional RRAM array? Can you provide basic characterization of the array, such as multi-level programming capability and variability?

Our response: We thank the reviewer for raising this important question regarding the capability and characterization of our RRAM array. We would like to clarify that the RRAM array used in this work supports analog in-memory computing.

The HfO_x/TaO_x based RRAM array exhibits stable resistive switching and fine-grained conductance modulation, enabling accurate analog weight representation for in-memory computation. The TaO_x layer serves as an oxygen reservoir and facilitates gradual filament modulation, allowing multi-level analog tuning similar to previously reported artificial synapses.³⁹ To illustrate this, we have included the conductance results after programming the devices to the intended values (Supplementary Fig. 20). The measured conductance values follow the intended target values with high fidelity (5–60 μ S range), confirming that the array is suitable for conductance-based analog in-memory computing.

We also mapped a subset of the trained weights from the CWRU task onto the RRAM array to perform a matrix–vector multiplication test. As shown in Supplementary Fig. 26, the outputs obtained from the array exhibit strong agreement with the expected results, demonstrating the capability of our RRAM array to perform analog in-memory computing.

We have further evaluated the multi-level programming capability of the array. Using an incremental write–verify scheme, each device can be reliably programmed into eight distinct conductance states. As shown in Fig. 4d, the cumulative distribution functions of the eight programmed levels exhibit well-separated conductance windows, confirming stable and controllable intermediate states. The tight distribution of each conductance level indicates that the devices can maintain clearly distinguishable analog states despite intrinsic device-to-device variations.

In addition to demonstrating precise programming into eight distinct conductance levels, we also evaluated the retention characteristics of all programmed states to further confirm the suitability of the RRAM array for analog in-memory computing. As shown in Supplementary Fig. 19, devices programmed to the eight conductance targets exhibit excellent long-term stability, maintaining their conductance values over an extended retention period of 10⁵ seconds without significant drift. Furthermore, we assessed the endurance ability of the devices by subjecting them to 10⁵ consecutive set–reset cycles. Supplementary Fig. 19 shows that the devices maintain a clear and stable switching window throughout the entire cycling process.

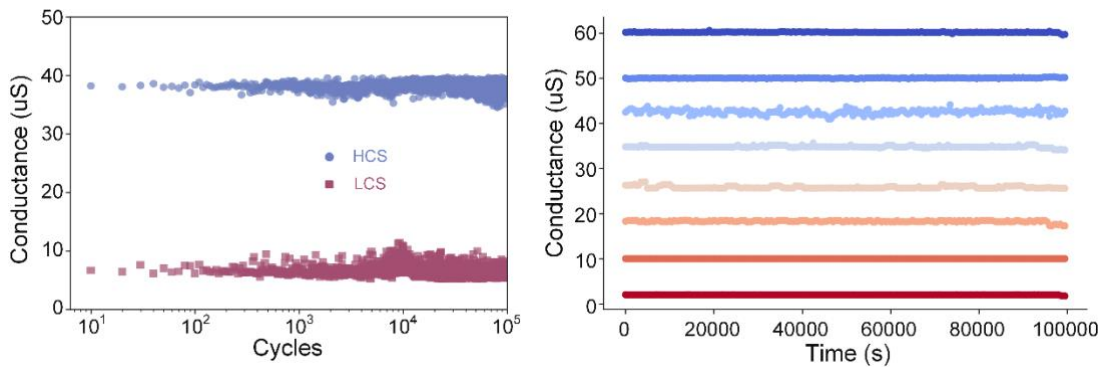
To address this comment, we have added Supplementary Fig. 19, 20, and the following brief discussions in the main text:

Page 14:

“In addition, we performed endurance tests by repeatedly applying set and reset operations to the device, and the results are shown in Supplementary Fig. 19. Our device demonstrates analog resistive switching and multilevel behavior by applying a series of set-reset pulses which gradually increase the pulse amplitude (Fig. 4c). Furthermore, to demonstrate that the array is suitable for analog in-memory computing, we measured the conductance after programming the analog values, and the conductance exhibits only minor deviations from the target values (Supplementary Fig. 20). Fig. 4d and Supplementary Fig. 19 display the cumulative distribution functions (CDFs) across 8 independent conductance states and the excellent retention characteristics of our non-volatile memristor. These characteristics confirm that the array is suitable for both analog in-memory computing and the implementation of dendritic structures with tunable delays, thereby improving computational performance.”

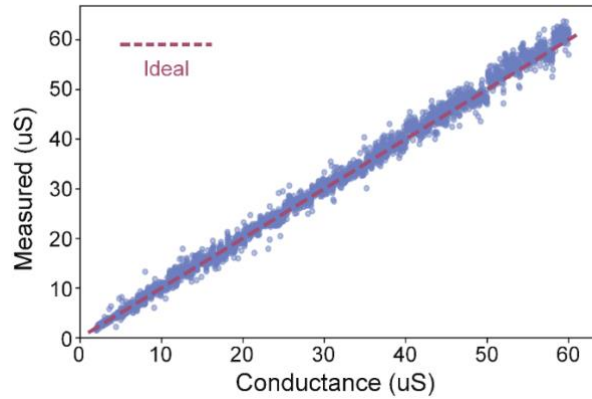
Supplementary Fig. 19:

“Supplementary Fig. 19 The endurance and retention performance of the non-volatile memristor. (a) Endurance of 10^5 cycles. (b) 8 conductance states can be retained for 10^5 seconds. From the lowest to the highest conductance state, the observed drift errors were 1.5%, 0.01%, 1.18%, 1.03%, 0.6%, 1.2%, 0.15%, and 0.14%.”



Supplementary Fig. 20:

“Supplementary Fig. 20 Continuous analog values programmed into the array, and the measured conductance closely follow the intended target values.”

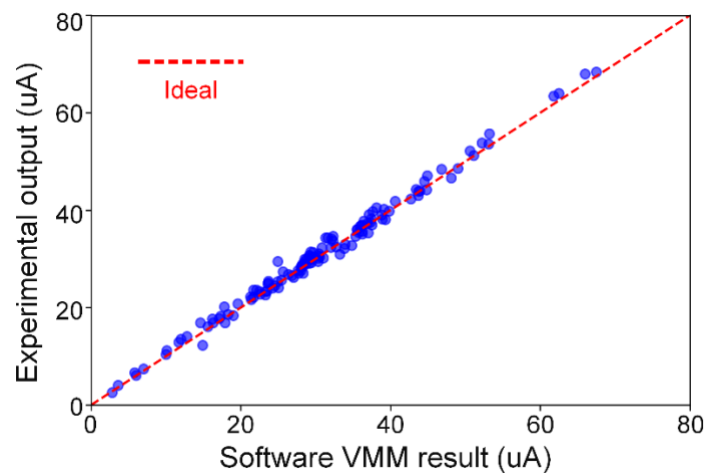


Page 20:

“During the forward pass, the neuronal state is determined by combining feedback signals from previous timestep outputs processed through recurrent modules and current timestep inputs transformed by weight matrices, for which the matrix–vector multiplication of selected weight vectors and input data is partially mapped onto the RRAM array (Supplementary Fig. 26).”

Supplementary Fig. 26:

“Supplementary Fig. 26 Matrix–vector multiplication test on the RRAM array. A subset of the trained weights from the CWRU task was mapped onto the array. The outputs obtained from the array show strong agreement with the expected results, demonstrating the capability of the RRAM array to perform analog in-memory computing.”



11. If I understand correctly, in the system-level SNN simulations you have only one delay per dendrite, and you train this delay. This implies that models with dendrites have more trainable parameters. Can you comment on this point and compare it with approaches using non-trained delays?

Our response: We thank the reviewer for the insightful comment.

As correctly noted, in our system-level SNN simulations each dendritic branch contains one trainable delay parameter. In our configuration, the hidden layer consists of 120 neurons, each receiving 200 synaptic inputs that converge through 8 dendritic delay branches. Although each dendrite introduces a trainable delay, the total number of additional parameters remains small. Specifically, the number of delay parameters scales linearly with the number of neurons (in our case <2K additional parameters). Importantly, achieving an equivalent level of temporal complexity with conventional LIF neurons would require stacking multiple layers, whose parameter count would scale quadratically with the neuron number. Therefore, our dendritic model achieves enhanced temporal modeling capacity at a very modest parameter cost.

To further address the reviewer’s question, we additionally conducted a comparative experiment using fixed, non-trainable delays (83.71%), and compared its performance with that of our proposed SNN with trainable dendritic delays (86.53%), as well as a baseline SNN without any dendritic delays (83.03%). As shown in Supplementary Fig. 34, fixed delays do improve performance compared to the no-delay baseline, indicating that introducing temporal pathways in dendrites is inherently beneficial. However, the performance gain remains lower than that of our trainable-delay model.

To address this comment, we have added Supplementary Fig. 34, and the following brief discussions in the main text:

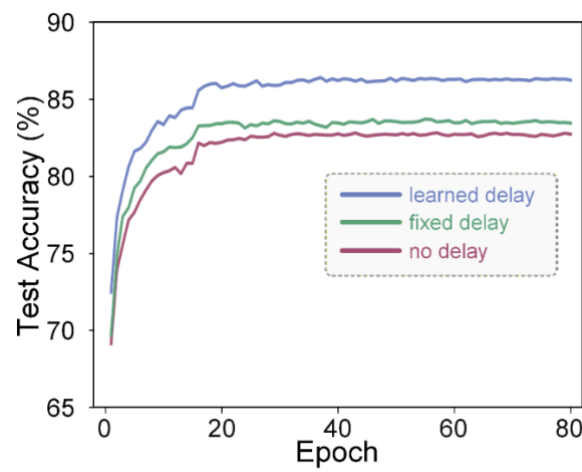
Page. 25:

“To investigate whether the observed performance improvement is due to the learnability of the delays rather than merely the presence of delays, we conducted a

control experiment with fixed delays. As shown in Supplementary Fig. 34, this fixed-delay model yielded a modest improvement over the baseline network without delays, yet consistently underperformed compared to the model with trainable delays. These results indicate that learnable dendritic delays contribute more to performance improvement than randomly assigned delays.”

Supplementary Fig. 34:

“Supplementary Fig. 34. Performance comparison between trainable delays, fixed (non-trainable) delays, and the baseline SNN without delays.”



12. The authors claim a 5.84% improvement over conventional LIF neurons, but to my understanding this refers to an LIF neuron designed by the authors, not to a state-of-the-art LIF neuron.

Our response: We sincerely thank the reviewer for this insightful comment.

We would like to clarify that the “conventional LIF neuron” designed by us (as shown in Fig. 2) strictly corresponds to the classical LIF neuron model widely used in the neuroscience and computational modeling literature. According to the standard formulation^{40,41}, the LIF dynamics are given by $C \frac{dV}{dt} = -gV + I_{in}$, with spike emission when the membrane potential reaches a threshold V_{th} and an instantaneous reset to V_r . In our circuit shown in Fig. 2, the VO₂ device and membrane capacitor

naturally realize this formulation. Denoting the VO₂ resistance as R_{VO_2} , the series load resistance as R_L , and the input current as I_{in} , we obtain $C \frac{dV}{dt} = -\frac{V}{R_{VO_2} + R_L} + I_{in} = -gV + I_{in}$. The intrinsic threshold switching mechanism of the VO₂ device directly provides the spike threshold V_{th} and reset voltage V_r . Thus, our “conventional LIF neuron” fully corresponds to the classical LIF neuron model.

Furthermore, we compared our network with representative SNN and DNN models reported for the same tasks. For CWRU, we benchmarked against multiple previously reported SNN and DNN approaches (see Supplementary Table 1). For GSC, we evaluated our model alongside several recent architectures, including those suggested by the reviewer (see Supplementary Table 3). These comparisons show that our network attains competitive accuracy with substantially lower parameters and computational cost, underscoring the robustness of our evaluation.

Finally, we interpret the reviewer’s reference to “state-of-the-art LIF” as likely referring to more advanced neuron models, for example, the Adaptive LIF (ALIF). To address this, we conducted additional experiments comparing LIF, ALIF, and our proposed HSLIF neurons using the same network configuration on the CWRU dataset. As shown in Fig. 5j, the ALIF network achieved an average best accuracy of 90.04%, slightly higher than the LIF network (87.31%). However, our proposed HSLIF network reached 92.14%, demonstrating higher performance. This suggests that the homeostatic mechanism in HSLIF provides more effective temporal adaptation than the threshold-adaptation mechanism in ALIF. Moreover, ALIF neurons tend to enter a persistently low firing-rate state after adaptation, and their performance gradually degrades during long-term inference unless manually reset. To illustrate this limitation, we conducted an additional experiment using concatenated long-duration sequences. The previously trained ALIF- and HSLIF-based networks were directly evaluated on these sequences. As shown in Supplementary Fig.30, the HSLIF network maintained stable accuracy over extended continuous inference due to its rapid firing-rate recovery, whereas the ALIF network exhibited a progressive accuracy drop as neurons gradually entered a

sustained low-activity state. These observations clearly demonstrate that HSLIF neurons provide superior long-term stability and robustness under continuous operation.

To address this comment, we have added Supplementary Table 1, 3, Supplementary Fig. 30, and the following brief discussions in the main text:

Page 21:

“To further evaluate the advantages of the homeostatic neuron over classical LIF and adaptive LIF (ALIF) neurons, we constructed SNNs of identical scale using these neuron types and trained them repeatedly on the same task. The LIF-based and ALIF-based networks achieved accuracies of $87.31\% \pm 0.51\%$ and $90.04\% \pm 0.52\%$, respectively. A comparison of all configurations (Fig. 5j) indicates that the network incorporating our homeostatic neurons demonstrates an improvement in classification performance. Beyond the improvements in accuracy, our homeostatic neurons exhibit a significant advantage in continuous, long-term information processing compared to ALIF neurons. To assess this, we concatenated multiple test segments into longer sequences (Supplementary Fig. 30) and evaluated the performance of the trained HSLIF (homeostatic LIF neuron) and ALIF-based networks. As shown in Supplementary Fig. 30, the HSLIF-based network maintained stable performance across continuous samples, whereas the ALIF-based network showed a gradual decline in classification accuracy over time. This difference arises from the neurons’ intrinsic dynamics, as HSLIF neurons quickly recover after a sudden firing-rate drop, whereas ALIF neurons remain in a prolonged low-frequency state until manually reset. In addition, we have compared the performance of our constructed network with other models reported in the literature on the CWRU dataset. As presented in Supplementary Table 1, our proposed approach achieves slightly lower classification accuracy compared to some state-of-the-art networks; however, it demonstrates a remarkable reduction in both parameter count and computational cost. Consequently, under the same testing platform, our network is expected to deliver lower inference power consumption and latency, highlighting its efficiency and suitability for resource-constrained neuromorphic applications.”

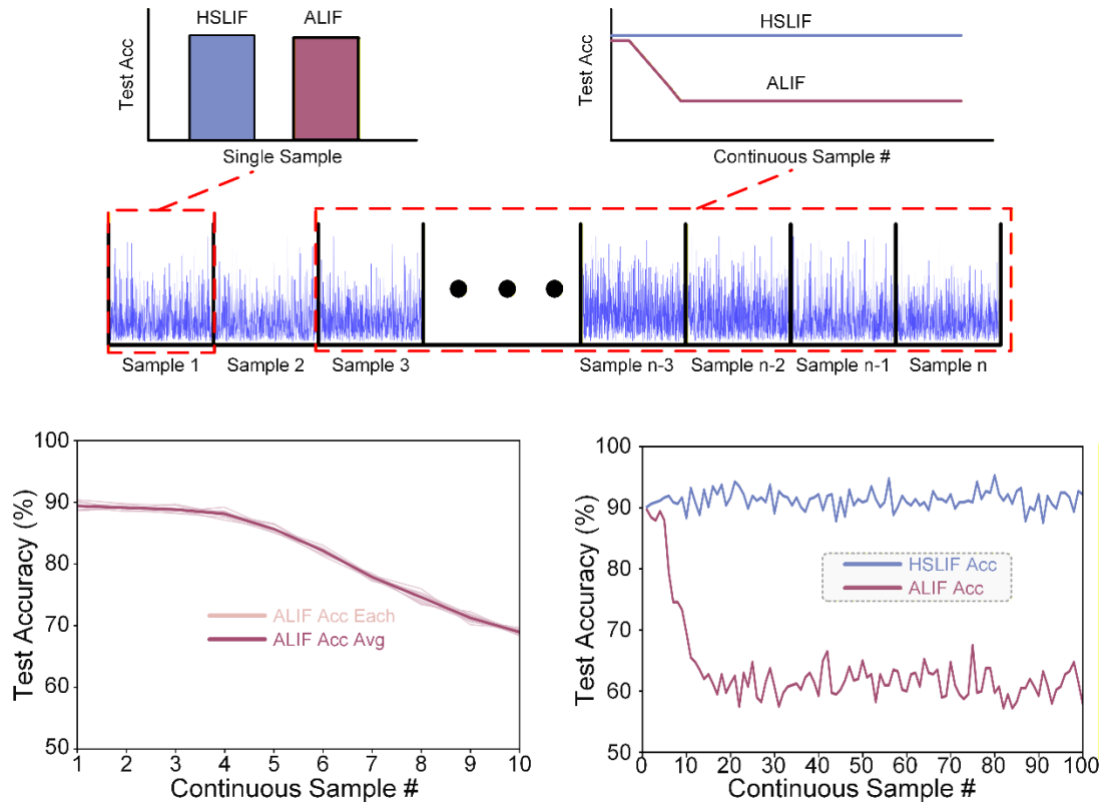
Supplementary Table. 1

“Supplementary Table 1. Comparative benchmark of different networks on the CWRU dataset.^{s1 – s7}”

	Model	Accuracy	Params	Flops
Chen et al. ^{s1}	ResNet50	99.92%	47.2M	718M
Yoo et al. ^{s2}	CNN0	96.64%	901K	1.8M
Rosa et al. ^{s3}	WDCNN	89.70%	40.7K	-
Wang et al. ^{s4}	CWT-2DCNN	83.10%	1.34M	840M
Wang et al. ^{s5}	MS-ER	98.80%	3.95M	2.8G
Lim et al. ^{s6}	SpikeCAN	99.86%	1.45M	-
Fang et al. ^{s7}	Spiking-ResNet	87.46%	3.85M	-
This work	SNN-HSLIF	92.14%	22K	140K

Supplementary Fig. 30:

“Supplementary Fig. 30 Performance of HSLIF and ALIF neurons in long-term classification tasks. The top figure illustrates the schematic of our testing methodology. We first trained and evaluated the network model using independent samples. Under this setting, the performance achieved by the HSLIF neurons shows a certain improvement compared with the ALIF neurons, although the enhancement is not very pronounced. We then conducted a continuous testing experiment by concatenating multiple segments to form a long-duration input sequence. The corresponding test results are shown in the two figures at the bottom. The bottom-left picture shows that ten randomly selected data segments were concatenated to form a new continuous sequence, and the previously trained ALIF-based network was used for testing. The results indicate that the classification accuracy gradually decreases as inference progresses. The bottom-right picture shows that one hundred randomly selected data segments were concatenated to form a longer continuous sequence, and testing was performed using both the previously trained ALIF-based and HSLIF-based networks. The results show that the classification accuracy of the ALIF-based network decreases significantly over time, whereas the HSLIF-based network maintains a consistently high accuracy throughout long-term inference.”



Page 25:

“In addition, we compared the constructed network with other models reported in the literature on the GSC dataset. As summarized in Supplementary Table 3, our network demonstrates significant advantages in terms of parameter count and computational cost while maintaining competitive classification accuracy.”

Supplementary Table 3:

“Supplementary Table 3. Comparative benchmark of different networks on the GSC dataset.^{s8–s17}”

	Model	Num Classes	Accuracy	Params	Flops
Niizumi et al. ^{s8}	M2D	35	98.50%	86M	-
Chen et al. ^{s9}	EAT	35	98.30%	88M	-
Andrade et al. ^{s10}	Attention-RNN	20	94.50%	202K	44.6M
Kim et al. ^{s11}	BC-ResNet-8	12	98.70%	321K	171.8M
Majumdar et al. ^{s12}	Matchbox-Net	35	97.40%	140K	34.2M
Bittar et al. ^{s13}	SNN-RadLIF	35	94.51%	1.2M	-
Hammouamri et al. ^{s14}	SNN-delays	35	95.35%	2.5M	-
Deckers et al. ^{s15}	SNN-d-cAdLIF	35	95.69%	610K	-
Sun et al. ^{s16}	SNN-delays	20(SHD)	92.45%	109K	-
D'agostino et al. ^{s17}	SNN-delays(fixed)	20(SHD)	90.88%	224K	-
Deckers et al. ^{s15}	SNN-LIF	35	86.21%	300k	-
This work	SNN-HSLIF-delays	35	86.53%	33K	3.13M

13. Overall, the comparison with state-of-the-art is missing, and the main advantages of the proposed approach — relative to classical SNNs and other dendrite-based implementations (see [1]–[3])— are not clearly established.

References:

- [1] D'Agostino, S., Moro, F., Torchet, T. et al. DenRAM: neuromorphic dendritic architecture with RRAM for efficient temporal processing with delays. Nat Commun 15, 3446 (2024).
- [2] Hammouamri, I., Khalfaoui-Hassani, I. & Masquelier, T. Learning delays in spiking neural networks using dilated convolutions with learnable spacings. ICLR (2024).
- [3] Sun, P., Eqlimi, E., Chua, Y., Devos, P. & Botteldooren, D. Adaptive axonal delays in feedforward spiking neural networks for accurate spoken word recognition. ICASSP 2023, 1–5 (IEEE, 2023).

Our response: We thank the reviewer for the valuable comment.

We would like to clarify the advantages of our proposed approach. First, our model introduces dendritic delays with only a small number of additional parameters, which grow linearly with the number of neurons. Achieving comparable temporal complexity in classical SNNs typically requires adding additional layers or neurons, resulting in a quadratic increase in model size. Second, the proposed HSLIF neurons maintain stable firing rates under long-duration and continuous inputs, providing richer neuronal dynamics. This stands in contrast to ALIF or other adaptive neuron models whose firing rates gradually decay over time, leading to degraded performance in long-term tasks, as demonstrated in Supplementary Fig. 30. Finally, following the reviewer’s suggestion, we included comparisons with representative SNN and DNN models reported on the same tasks (Supplementary Table 1, 3). These results show that our network achieves competitive accuracy with substantially fewer parameters and lower computational cost, highlighting the robustness and efficiency of our proposed approach. In addition, we note that the hardware system built using our method achieves higher energy efficiency and a more compact silicon area while maintaining high accuracy, as shown in Supplementary Table 4.

To address this comment, we have added Supplementary Table 1, 3, 4, and the following brief discussions in the main text:

Page 27:

“At the system level, we evaluated the hardware performance of the proposed neuromorphic network for the GSC dataset. The implementation achieves a compact area of 0.102 mm^2 and energy consumption of $12.9 \times 10^{-6}\text{ J}$ per second of speech while maintaining the classification accuracy. Compared with other representative works, our design demonstrates significant improvements in area and energy efficiency (Supplementary Table 4). Together, the single-neuron optimizations and system-level design enable a highly efficient and scalable architecture capable of performing real-time time-series processing tasks with low power consumption and compact silicon footprint.”

Supplementary Table. 1

“Supplementary Table 1. Comparative benchmark of different networks on the CWRU dataset.^{s1 – s7}”

	Model	Accuracy	Params	Flops
Chen et al. ^{s1}	ResNet50	99.92%	47.2M	718M
Yoo et al. ^{s2}	CNN0	96.64%	901K	1.8M
Rosa et al. ^{s3}	WDCNN	89.70%	40.7K	-
Wang et al. ^{s4}	CWT-2DCNN	83.10%	1.34M	840M
Wang et al. ^{s5}	MS-ER	98.80%	3.95M	2.8G
Lim et al. ^{s6}	SpikeCAN	99.86%	1.45M	-
Fang et al. ^{s7}	Spiking-ResNet	87.46%	3.85M	-
This work	SNN-HSLIF	92.14%	22K	140K

Supplementary Table 3:

“Supplementary Table 3. Comparative benchmark of different networks on the GSC dataset.^{s8 – s17}”

	Model	Num Classes	Accuracy	Params	Flops
Niizumi et al. ^{s8}	M2D	35	98.50%	86M	-
Chen et al. ^{s9}	EAT	35	98.30%	88M	-
Andrade et al. ^{s10}	Attention-RNN	20	94.50%	202K	44.6M
Kim et al. ^{s11}	BC-ResNet-8	12	98.70%	321K	171.8M
Majumdar et al. ^{s12}	Matchbox-Net	35	97.40%	140K	34.2M
Bittar et al. ^{s13}	SNN-RadLIF	35	94.51%	1.2M	-
Hammouamri et al. ^{s14}	SNN-delays	35	95.35%	2.5M	-
Deckers et al. ^{s15}	SNN-d-cAdLIF	35	95.69%	610K	-
Sun et al. ^{s16}	SNN-delays	20(SHD)	92.45%	109K	-
D'agostino et al. ^{s17}	SNN-delays(fixed)	20(SHD)	90.88%	224K	-
Deckers et al. ^{s15}	SNN-LIF	35	86.21%	300k	-
This work	SNN-HSLIF-delays	35	86.53%	33K	3.13M

Supplementary Table 4:

“Supplementary Table 4. Comparison of hardware implementations for the GSC dataset task ^{s18 – s26}.”

	Technology	Model	Num Classes	Accuracy	Frequency (Hz)	Energy for 1s speech (J)	Latency (s)	Area (um ²)
Zol. et al. ^{s18,19}	14nm CMOS + IBM PCM	CNN	11	89.30%	400x10 ⁶	78x10 ⁻⁶	348x10 ⁻⁶	144x10 ⁶
Giraldo et al. ^{s20}	65nm	LSTM	10	91%	250x10 ³	16.2x10 ⁻⁶	-	2.56x10 ⁶
Li et al. ^{s21}	65nm	GRU	2	98%	-	122.3x10 ⁻⁶	211x10 ⁻⁶	4.38x10 ⁶
Kim et al. ^{s22}	65nm	RNN	12	86%	250x10 ³	23x10 ⁻⁶	12.4x10 ⁻³	2.03x10 ⁶
Shan et al. ^{s23}	28nm	DSCNN	2	95%	40x10 ³	0.98x10 ⁻⁶	-	230x10 ³
Liu et al. ^{s24}	65nm	FS-CNN + LIF	4	95%	100x10 ³	7.7x10 ⁻⁶	18.7x10 ⁻³	917x10 ³
Wang et al. ^{s25}	65nm	SNN	5	90%	120x10 ³	0.57x10 ⁻⁶	100x10 ⁻³	2.71x10 ⁶
Shaban et al. ^{s26}	50um RRAM + 0.5um CMOS	SNN	12	91%	1x10 ³	696x10 ⁻⁶	1.2	11x10 ³ (sim 65nm)
This work	180nmRRAM + 180nm CMOS	SNN	35	86.54%	180x10 ³	12.9x10 ⁻⁶	1.66x10 ⁻³	102x10 ³

References

1. Xiao, Y. *et al.* Bio-plausible reconfigurable spiking neuron for neuromorphic computing. *Science Advances* **11**, eadr6733 (2025).
2. Jeong, H. *et al.* Self-supervised video processing with self-calibration on an analogue computing platform based on a selector-less memristor array. *Nat Electron* **8**, 168–178 (2025).
3. Li, F. *et al.* An artificial visual neuron with multiplexed rate and time-to-first-spike coding. *Nat Commun* **15**, 3689 (2024).
4. Yi, S., Kendall, J. D., Williams, R. S. & Kumar, S. Activity-difference training of deep neural networks using memristor crossbars. *Nat Electron* **6**, 45–51 (2023).
5. Wang, J. *et al.* Design and Implementation of a Hybrid, ADC/DAC-Free, Input-Sparsity-Aware, Precision Reconfigurable RRAM Processing-in-Memory Chip. *IEEE Journal of Solid-State Circuits* **59**, 595–604 (2024).
6. Prabhu, K. *et al.* MINOTAUR: A Posit-Based 0.42–0.50-TOPS/W Edge Transformer Inference and Training Accelerator. *IEEE Journal of Solid-State Circuits* **60**, 1311–1323 (2025).
7. An, Y. *et al.* StreamPIM: Streaming Matrix Computation in Racetrack Memory. in *2024 IEEE International Symposium on High-Performance Computer Architecture (HPCA)* 297–311 (2024). doi:10.1109/HPCA57654.2024.00031.
8. Zhu, Y. *et al.* MeMCISA: Memristor-Enabled Memory-Centric Instruction-Set Architecture for Database Workloads. in *2024 57th IEEE/ACM International Symposium on Microarchitecture (MICRO)* 1678–1692 (2024). doi:10.1109/MICRO61859.2024.00122.
9. Putra, R. V. W. & Shafique, M. SpiKernel: A Kernel Size Exploration Methodology for Improving Accuracy of the Embedded Spiking Neural Network Systems. *IEEE Embedded Systems Letters* **17**, 151–155 (2025).
10. Rao, M. *et al.* Thousands of conductance levels in memristors integrated on CMOS. *Nature* **615**, 823–829 (2023).
11. Davis, G. W. Homeostatic control of neural activity: from phenomenology to molecular design. *Annu Rev Neurosci* **29**, 307–323 (2006).

-
12. Desai, N. S., Rutherford, L. C. & Turrigiano, G. G. Plasticity in the intrinsic excitability of cortical pyramidal neurons. *Nat Neurosci* **2**, 515–520 (1999).
 13. Leslie, K. R., Nelson, S. B. & Turrigiano, G. G. Postsynaptic Depolarization Scales Quantal Amplitude in Cortical Pyramidal Neurons. *J. Neurosci.* **21**, RC170–RC170 (2001).
 14. Kilman, V., van Rossum, M. C. W. & Turrigiano, G. G. Activity deprivation reduces miniature IPSC amplitude by decreasing the number of postsynaptic GABA(A) receptors clustered at neocortical synapses. *J Neurosci* **22**, 1328–1337 (2002).
 15. Aizenman, C. D., Akerman, C. J., Jensen, K. R. & Cline, H. T. Visually Driven Regulation of Intrinsic Neuronal Excitability Improves Stimulus Detection In Vivo. *Neuron* **39**, 831–842 (2003).
 16. Frank, C. A., Kennedy, M. J., Goold, C. P., Marek, K. W. & Davis, G. W. Mechanisms Underlying the Rapid Induction and Sustained Expression of Synaptic Homeostasis. *Neuron* **52**, 663–677 (2006).
 17. Rieggle, K. C. & Meyer, R. L. Rapid Homeostatic Plasticity in the Intact Adult Visual System. *J. Neurosci.* **27**, 10556–10567 (2007).
 18. Sutton, M. A. *et al.* Miniature neurotransmission stabilizes synaptic function via tonic suppression of local dendritic protein synthesis. *Cell* **125**, 785–799 (2006).
 19. Tyssowski, K. M. *et al.* Firing Rate Homeostasis Can Occur in the Absence of Neuronal Activity-Regulated Transcription. *J Neurosci* **39**, 9885–9899 (2019).
 20. Turrigiano, G. Homeostatic Synaptic Plasticity: Local and Global Mechanisms for Stabilizing Neuronal Function. *Cold Spring Harb Perspect Biol* **4**, a005736 (2012).
 21. Qiao, N., Bartolozzi, C. & Indiveri, G. An Ultralow Leakage Synaptic Scaling Homeostatic Plasticity Circuit With Configurable Time Scales up to 100 ks. *IEEE Transactions on Biomedical Circuits and Systems* **11**, 1271–1277 (2017).
 22. Richter, O. *et al.* DYNAP-SE2: a scalable multi-core dynamic neuromorphic asynchronous spiking neural network processor. *Neuromorph. Comput. Eng.* **4**, 014003 (2024).

-
23. Liu, Q. & Zhang, Z. Ultralow Power Always-On Intelligent and Connected SNN-Based System for Multimedia IoT-Enabled Applications. *IEEE Internet Things J.* **9**, 15570–15577 (2022).
 24. Zhang, Z. & Liu, Q. Spike-Event-Driven Deep Spiking Neural Network With Temporal Encoding. *IEEE Signal Process. Lett.* **28**, 484–488 (2021).
 25. Guo, Y., Huang, X. & Ma, Z. Direct learning-based deep spiking neural networks: a review. *Front. Neurosci.* **17**, (2023).
 26. Zhang, M. *et al.* Rectified Linear Postsynaptic Potential Function for Backpropagation in Deep Spiking Neural Networks. Preprint at <https://doi.org/10.48550/arXiv.2003.11837> (2020).
 27. Wei, W. *et al.* Temporal-Coded Spiking Neural Networks with Dynamic Firing Threshold: Learning with Event-Driven Backpropagation. in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)* 10518–10528 (IEEE, Paris, France, 2023). doi:10.1109/ICCV51070.2023.00968.
 28. Fang, W., Xuan, Z., Chen, S. & Kang, Y. An 1.38nJ/Inference Clock-Free Mixed-Signal Neuromorphic Architecture Using ReL-PSP Function and Computing-in-Memory. in *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)* 1–5 (2023). doi:10.1109/BioCAS58349.2023.10388821.
 29. Xing, Y. *et al.* Accurate ECG Classification Based on Spiking Neural Network and Attentional Mechanism for Real-Time Implementation on Personal Portable Devices. *Electronics* **11**, 1889 (2022).
 30. Choi, S. H. Spiking neural networks for biomedical signal analysis. *Biomed Eng Lett* **14**, 955–966 (2024).
 31. Patiño-Saucedo, A. *et al.* Co-optimized training of models with synaptic delays for digital neuromorphic accelerators. in *2024 IEEE International Symposium on Circuits and Systems (ISCAS)* 1–5 (2024). doi:10.1109/ISCAS58744.2024.10558209.
 32. Davies, M. *et al.* Loihi: A Neuromorphic Manycore Processor with On-Chip Learning. *IEEE Micro* **38**, 82–99 (2018).

-
33. Boyle, J. A. *et al.* Exploring Dendrites in Large-Scale Neuromorphic Architectures. in *Proceedings of the International Conference on Neuromorphic Systems* 49–56 (IEEE Press, Seattle, Washington, USA, 2025). doi:10.1109/ICONS69015.2025.00018.
 34. Frougier, J. *et al.* Phase-Transition-FET exhibiting steep switching slope of 8mV/decade and 36% enhanced ON current. in *2016 IEEE Symposium on VLSI Technology* 1–2 (2016). doi:10.1109/VLSIT.2016.7573445.
 35. Shan, L. *et al.* In Materia Neuron Spiking Plasticity for Sequential Event Processing Based on Dual-Mode Memristor. *Advanced Intelligent Systems* **4**, 2100264 (2022).
 36. Wang, Z. *et al.* Self-Activation Neural Network Based on Self-Selective Memory Device With Rectified Multilevel States. *IEEE Transactions on Electron Devices* **67**, 4166–4171 (2020).
 37. Li, Z. *et al.* Crossmodal sensory neurons based on high-performance flexible memristors for human-machine in-sensor computing system. *Nat Commun* **15**, 7275 (2024).
 38. Corti, E. *et al.* Coupled VO₂ Oscillators Circuit as Analog First Layer Filter in Convolutional Neural Networks. *Front. Neurosci.* **15**, (2021).
 39. Dang, B. *et al.* Reconfigurable in-sensor processing based on a multi-phototransistor–one-memristor array. *Nat Electron* **7**, 991–1003 (2024).
 40. Tuckwell, H. C. (Henry C. *Introduction to Theoretical Neurobiology*. (Cambridge : Cambridge University Press, 1988).
 41. Fourcaud-Trocme, N., Hansel, D., Vreeswijk, C. van & Brunel, N. How Spike Generation Mechanisms Determine the Neuronal Response to Fluctuating Inputs. *J. Neurosci.* **23**, 11628–11640 (2003).

Title: Homeostatic Dendritic Neuron Based on Co-Integrated Volatile and Non-Volatile Memristors for Neuromorphic Processing

Response to the editor and the reviewers

We would like to express our gratitude to the editor and the reviewers once again for their time and thoughtful follow-up comments on our revised manuscript. The constructive feedback provided in both rounds of review has greatly improved the quality and clarity of our work. We have carefully addressed the additional comments raised by the reviewers. We have provided further clarifications and updated the manuscript to improve readability (Pages 4, 5, 18-20, 26, 28, Supplementary Fig. 35 and Supplementary Table 6). We hope that the editor and reviewers will find the revised manuscript suitable for publication in Nature Communications. The point-by-point responses and corresponding revisions are provided below. For clarity, the revised sections quoted from the manuscript files are shown in blue and the exact changes made relevant to the comment are *in italics*.

This manuscript presents a co-integrated neuromorphic hardware system that combines VO₂-based volatile memristors with non-volatile RRAM devices to realize, in hardware, two biologically important yet rarely co-implemented functionalities: neuronal homeostatic regulation and dendritic delay-based temporal processing. Specifically, the authors demonstrate: (1) a VO₂-memristor-based homeostatic spiking neuron that autonomously restores its firing rate following excitatory or inhibitory perturbations, extending beyond conventional LIF and ALIF neuron models; (2) a programmable dendritic architecture employing multi-level RRAM conductance states to implement tunable temporal delays; and (3) a hardware-informed spiking neural network (SNN) framework that achieves statistically significant performance improvements on representative temporal tasks, including bearing fault diagnosis (CWRU, 92.14% $\pm$ 0.99%, outperforming both LIF and ALIF baselines) and speech recognition (GSC, 86.53% $\pm$ 0.18%), where dendritic delays consistently enhance accuracy. The system further demonstrates low-energy operation ($\sim$ 19 pJ per spike) and compact circuit footprints, underscoring its relevance for edge and neuromorphic computing applications. The revised manuscript incorporates ablation studies, repeated training runs, ALIF benchmarks, and robustness analyses, substantially strengthening the technical rigor and credibility of the results.

The work is of clear significance to the neuromorphic hardware community and is also relevant to related fields, including in-memory computing, edge AI hardware, and brain-inspired circuit design. Prior studies have largely focused on memristor-based implementations of LIF or ALIF neurons using threshold adaptation or leakage control, while dendritic processing has predominantly been explored at the algorithmic or simulation level, with limited hardware realization. Moreover, intrinsic homeostatic regulation implemented directly at the circuit level remains rare, particularly in physical device-based systems. In this context, the present work advances the state of the art by: (1) co-integrating dendritic delay

circuits and neurons within a unified hardware framework; (2) combining experimentally validated device- and circuit-level measurements with hardware-calibrated system-level simulations, consistent with best practices in neuromorphic hardware research; and (3) benchmarking against existing CMOS-, RRAM-, and SNN-based hardware implementations while appropriately positioning the contribution as a scalable hardware foundation rather than a fully realized large-scale chip. The conclusions drawn are well supported by the presented evidence.

In response to earlier reviewer concerns, the authors have added: (1) multiple independent training runs with reporting of mean values and standard deviations; (2) direct comparisons with both LIF and ALIF neuron baselines; (3) long-term continuous inference experiments demonstrating the distinctive stability advantage of the proposed homeostatic neurons; and (4) comprehensive analyses of device variability, endurance, retention, and noise tolerance. The claims have also been appropriately moderated, particularly with respect to scalability and system integration. No additional experiments appear necessary, and no major technical flaws are evident. The methodology is sound and consistent with state-of-the-art practices in neuromorphic hardware research.

Any remaining issues are minor and largely editorial in nature (e.g., phrasing consistency and careful claim formulation) and do not preclude publication. At most, they may warrant minor editorial refinement rather than substantive revision.

Overall, this is a carefully revised, technically robust, and conceptually meaningful contribution to neuromorphic hardware research. The authors have thoroughly addressed prior reviewer critiques, strengthened the experimental and statistical foundations of the work, and clarified the scope of their claims. The manuscript meets the standards of Nature Communications and should be suitable for publication.

Our response: We sincerely thank the reviewer for the thorough evaluation of our revised manuscript and for the positive assessment of the technical rigor and significance of our work. In accordance with the reviewer's suggestion, we have carefully reviewed the entire manuscript to further improve phrasing consistency and ensure precise claim formulation throughout. Minor editorial refinements have been implemented to enhance clarity and readability, without altering the technical content or conclusions. We greatly appreciate the reviewer's constructive feedback throughout the review process, which has significantly strengthened the quality of this work.

Comments from Reviewer #2

I thank the authors for their thorough responses. The addition of comparisons with SOTA platforms on the example tasks used in this paper, as well as comparison of physical footprints and energy measurements of their proposed circuits address the previous concerns I had with this manuscript. I recommend for publishing.

Our response: We sincerely thank the reviewer for the positive assessment of our revised manuscript. We are pleased that the added SOTA comparisons and the analysis of physical footprint and energy consumption have addressed the previous concerns. We appreciate the reviewer's thoughtful feedback and support.

1. From a hardware point of view, the proposed system consists of only a single VO₂ memristor device and a small conventional 1T1R array interfaced via an FPGA. Neither the neuron circuits nor the dendritic branches have been fabricated or experimentally tested. As clarified in the response to the reviewers, the reported results are primarily based on circuit-level simulations, since in the board-level prototype the leakage currents of the transistors already reach the microampere (μ A) range, which is comparable to the μ A-level currents modulated in the proposed neuron circuit. Given this limitation, the scientific and technological relevance of the FPGA-based demonstration remains unclear. The manuscript should clarify the added value of the FPGA demonstration under these constraints.

Our response: We sincerely thank the reviewer for the insightful and technically rigorous comment.

We acknowledge that the present work does not report a fully integrated chip-level implementation. Instead, we adopted a staged validation approach combining FPGA control and board-level circuitry to experimentally verify key functions of the architecture. For example, the basic LIF neuron in Fig. 2 was implemented to demonstrate spike generation. The dendritic structure in Fig. 4 was realized to validate programmable spike delay. Furthermore, as shown in Supplementary Fig. 26, we have experimentally implemented matrix-vector multiplication (MVM) operations on the board-level platform using the RRAM array and its peripheral circuitry.

Although a fully integrated system-on-chip was not constructed, the board-level implementation serves several critical validation purposes. We would like to respectfully clarify the specific objectives of our board-level demonstration.

Firstly, through board-level implementation, we demonstrated that the basic LIF neuron circuit effectively emulates spiking behavior by leveraging the intrinsic threshold-switching characteristics of VO₂ device. We experimentally verified that the

spiking frequency can be modulated via current control, thereby establishing a solid hardware foundation for the subsequent design of homeostatic neurons.

Furthermore, the dendritic structure was realized using a board-level hybrid integration of volatile and non-volatile memristors, with the delay characteristics programmed by the FPGA.. To emulate the control logic of a complete system-on-chip, we utilized the FPGA to perform closed-loop conductance tuning. This involves a "write-verify" sequence where the FPGA reads the RRAM state and executes further SET/RESET operations to precisely match target values. By employing this methodology, we have experimentally confirmed the capability of the dendritic architecture to generate diverse delay profiles, providing a robust proof-of-concept for the overall design.

Finally, we implemented the matrix–vector multiplication (MVM) operation on the constructed board-level platform and mapped a portion of the network weights onto the RRAM array for experimental computation (as shown in Supplementary Fig. 26). Using the FPGA as a high-precision controller, the RRAM devices within the array were first programmed to the targeted conductance states. Subsequently, the FPGA dispatched synchronized input vector signals to the row drivers on the PCB. The resulting currents, representing the weighted sums, were successfully captured and processed by the peripheral sensing circuitry. In this configuration, the FPGA functions as the central processing and synchronization unit, managing the complex timing required for parallel data conversion and address decoding. This demonstration confirms that our hardware platform can reliably execute the fundamental multiply–accumulate (MAC) operations, providing an experimental foundation for larger-scale SNN inference tasks.

Such a board-level platform provides a vital rapid prototyping bridge between pure simulation and full silicon integration, functioning as a common methodology in the development of emerging neuromorphic hardware.

Regarding the issue of discrete transistor leakage current being on the same order of magnitude as the target uA-level operating current for the homeostatic LIF (HSLIF) neuron, we acknowledge that this constraint hindered a perfect demonstration of the

HSLIF neuron circuit at the board level. However, this limitation arises strictly from the available discrete components, rather than a flaw in our design principles. Our simulation results based on a 180nm technology node confirm that in a fully integrated silicon implementation, subthreshold leakage and parasitic effects can be tightly controlled to achieve the intended functionality. Although a full integrated demonstration of the complete HSLIF neuron and dendritic delay is currently unfeasible, the modular board-level validation (basic neuron behavior and dendritic delay) combined with our simulations strongly supports the feasibility of our design. Therefore, these results indicate that our system architecture is scalable to larger-scale, fully integrated silicon chips.

In summary, the FPGA and board-level demonstrations have realized functional verification, confirming the operational viability of our neuron-dendrite computational framework. By progressing systematically from device characterization and simulation to partial hardware validation, we have established the architectural feasibility of our design and paved the way for full silicon integration. This phased approach is a common practice in neuromorphic hardware development.

We have revised the relevant section in the manuscript to better highlight the functionality and purpose of our FPGA-board synergy.

Page 18:

“The hardware platform developed in this work serves as a partial experimental realization of key functional modules of the proposed SNN architecture, rather than a fully monolithically integrated system. The architecture is designed to emulate essential features of dendritic processing and history-dependent spiking dynamics. Conceptually, the system comprises dendritic delay circuits coupled with neuron structures (Fig. 5a).”

Page 18, 19:

“The experimental setup implements a heterogeneous hardware validation framework that bridges software preprocessing with board-level neuromorphic modules (Fig. 5b). Raw input data undergoes preprocessing on a conventional PC before being transmitted to FPGA controller. The FPGA serves as the primary control and interface unit, managing real-time spike data streaming, conductance

programming of RRAM devices, and temporal control of the test board. The board-level platform hosts the experimentally implemented dendritic circuits and basic LIF neuron modules, while the homeostatic LIF neurons are simulated.”

Page 19:

“In this framework, the FPGA functions analogously to a control unit (or MCU) in a future fully integrated silicon implementation, while the PCB serves as a testbed for validating programmable dendritic delay and neuron–dendrite interaction. This staged validation architecture provides experimental verification of key functional mechanisms and programmable temporal dynamics.”

2. A comparison with the state of the art in terms of area and energy efficiency is provided in Supplementary Tables 1 and 3. While useful, this information should be discussed more clearly in the main text, particularly in relation to other dendritic computing approaches. In particular, comparisons with hardware dendritic implementations based on RRAM are missing from the Introduction and Discussion sections. I would expect a more explicit discussion of the advantages and limitations of different dendritic approaches, for example: architectures where multiple synapses share a common dendritic branch and time constant, versus approaches in which each synapse is assigned a distinct delay or time constant. Regarding the comparison with digital neuromorphic platforms, the manuscript currently mentions Intel’s Loihi architecture. However, Loihi’s dendritic processing is implemented digitally: state variables corresponding to dendritic compartments are stored in SRAM and updated in discrete time steps. This is fundamentally different from the continuous-time analog dynamics and nonlinear dendritic behavior proposed in this work. It is therefore not clear that Loihi represents an appropriate or fair comparison.

Our response: We thank the reviewer for the valuable suggestion. We agree that a clearer positioning of our work with respect to existing dendritic hardware approaches,

as well as a more explicit discussion of different dendritic design philosophies, would strengthen the manuscript.

In the original Supplementary Materials, we compared our work with state-of-the-art approaches from both the algorithmic and hardware implementation perspectives.

Supplementary Tables 1 and 3 primarily focus on algorithm-level comparisons. According to the results reported in these tables, our model requires fewer parameters and reduced computational complexity compared to prior works. Therefore, if implemented on the same hardware platform or fabricated using the same process node for a dedicated accelerator, our approach is expected to achieve superior area efficiency and energy efficiency due to its inherently lower computational demand.

Supplementary Tables 4 and 5 further evaluated hardware-related metrics. In Supplementary Table 4, we report the silicon area and energy consumption of the complete system for the Google Speech Commands (GSC) recognition task and compare it with prior works. Despite being simulated based on a 180 nm process, the proposed system exhibits a compact silicon footprint of 0.102 mm² and an energy consumption of 12.9 μ J per second of speech. The relatively low inference latency (1.66 ms) stems from the intrinsic continuous-time integration mechanism of the neuron circuits, which eliminates the sequential clock cycles required in digital architectures. In Supplementary Table 5, we further compare the area and energy consumption of the proposed HSLIF neuron with representative neuron circuits. These designs span fundamentally different computational paradigms and can be categorized into three groups: (i) purely CMOS adaptive neurons relying on capacitor charging/discharging and bias-controlled feedback (s28, s29, s32); (ii) hybrid CMOS–memristive neurons utilizing RRAM or ECRAM devices to emulate adaptive conductance modulation (s26, s27, s31); and (iii) phase-transition-driven oscillatory neurons exploiting the threshold switching characteristics of NbO₂ or related materials (s30, s31). In our design, spike generation originates from the intrinsic physical threshold switching of the device, without requiring clock-driven integration cycles to compute membrane potential. This mechanism allows the neuron to operate at relatively higher frequencies, making

threshold-switching-based neurons inherently well-suited for fast neuromorphic signal-processing tasks. The transistor footprint in 180 nm technology is $76.8 \mu\text{m}^2$, which is smaller than several CMOS-based adaptive LIF neurons implemented in comparable or more advanced process nodes. Furthermore, the proposed neuron exhibits lower energy per spike than CMOS-based adaptive implementations and achieves performance comparable to other threshold-switching neuron designs.

In the revised manuscript, we have added a dedicated comparison of our dendritic structure with representative hardware dendritic implementations, particularly those based on RRAM devices. The detailed comparison is now provided in Supplementary Table 6 and is discussed explicitly in the Introduction and Discussion sections. As summarized in the table, existing hardware dendritic implementations can be broadly categorized according to their structural organization and temporal processing strategy: (i) One-synapse-per-dendrite architectures (e.g., s17, s37), where each synapse is assigned an independent dendritic branch with its own time constant. (ii) Multi-synapse shared-dendrite architectures (e.g., s33–s36), where multiple synapses converge onto a common dendritic unit and share unified temporal dynamics.

The first category provides granular temporal control, essential for tasks requiring precise sequence-dependent integration and asymmetric temporal summation, such as direction or sequence recognition. These systems typically leverage hybrid RRAM-CMOS architectures. Nevertheless, the requirement for dedicated branches per synapse imposes a heavy circuit overhead. Data from Supplementary Table 6 reveals that these designs—incorporating tens of transistors per branch (e.g., s17 and s37)—result in a significant area penalty and increased static power dissipation due to the complex peripheral circuitry.

In contrast, the second strategy offers a more resource-efficient alternative, reducing hardware overhead by multiplexing multiple synapses onto a shared dendritic branch. Rather than relying on redundant circuitry, it exploits the intrinsic nonlinear dynamics of emerging devices (such as memristors, multi-gate FETs, and anti-ambipolar transistors) alongside analog CMOS blocks to implement complex temporal

filtering and meta plasticity. Consequently, this strategy enables robust dendritic computation with a minimal hardware footprint.

In this work, we adopt a multi-synapse shared-dendrite architecture, departing from previous RRAM-based implementations that primarily utilize a one-synapse-per-dendrite approach. This strategic shift substantially reduces the required number of dendritic structures (scaling from $I_{inputs} \times N_{neurons}$ down to the number of neurons $N_{neurons}$), thereby minimizing the overall hardware footprint. A key distinction of our design lies in the spike generation mechanism: instead of relying on sophisticated CMOS circuitry, we leverage the intrinsic threshold switching characteristics of VO₂ devices. This transition significantly curtails the transistor count, as detailed in Supplementary Table 6. However, the shared-dendrite architecture is not universally superior to the one-synapse-per-dendrite paradigm; instead, they represent a fundamental architectural trade-off. The latter imposes significant circuit overhead and scaling costs, which restricts its application to simpler tasks such as direction discrimination that require fewer parameters. Conversely, the former enhances area and energy efficiency through shared temporal dynamics and reduced structural redundancy, thereby rendering it more effective for complex tasks that necessitate large-scale parameter integration.

We also sincerely appreciate the reviewer's insightful comment regarding the architectural differences between Loihi and our proposed system. We fully acknowledge that Loihi utilizes a digital implementation where variables are stored in SRAM and updated at discrete time steps, which inherently contrasts with the analog-based, continuous-time dendritic dynamics presented in this work. However, we would like to clarify that the inclusion of the Loihi architecture in our discussion was intended to provide a comprehensive overview of the diverse landscape of dendritic computation, rather than to imply a direct performance superiority of one paradigm over the other. Each approach possesses distinct advantages tailored to specific application requirements. To ensure a more rigorous and fair comparison, we have added Supplementary Table 6, which specifically focuses on analog-based dendritic

implementations characterized by continuous-time dynamics. This addition enables a more focused evaluation of our work within the context of similar physical-domain computing systems.

We have updated the manuscript to address the reviewer's feedback by adding a comparison of different dendritic computation methods in the Introduction and Discussion. We have also included Supplementary Table 6 and expanded the associated discussions to ensure a fair and comprehensive evaluation.

Page 4:

“Architecturally, hardware dendrites involve a trade-off between one-synapse-per-dendrite structures^{45,46}, which offer granular temporal control at the expense of hardware redundancy, and multi-synapse shared-dendrite configurations^{47,48}, which maximize area efficiency by multiplexing inputs onto shared temporal dynamics.”

Page 4, 5:

“By adopting a resource-efficient shared-dendrite architecture, our approach scales the number of dendritic units down to the number of neurons, significantly reducing hardware overhead compared to traditional RRAM-based implementations that require per-synapse dendritic structures. Furthermore, instead of relying on sophisticated peripheral circuitry for spike generation, we leverage the intrinsic threshold-switching characteristics of VO₂ devices to directly implement neuronal firing mechanisms.”

Page 26:

“As summarized in Supplementary Table 3, our network demonstrates significant advantages in terms of parameter count and computational cost while maintaining competitive classification accuracy. Since the silicon area is primarily dictated by the parameter storage and the energy consumption scales with the number of operations, our approach is inherently positioned to achieve superior area and energy efficiency when implemented on a unified hardware platform. These reductions in memory footprint and operational intensity translate directly into a more compact hardware

design and minimized power consumption, thereby validating the suitability of our architecture for resource-constrained neuromorphic applications.”

Page 28:

“Specifically, the homeostatic neuron, optimized at both the device and circuit levels, attains a compact area of $76.8 \mu\text{m}^2$ and ultra-low energy consumption of 19.29 pJ per spike (Supplementary Table 5). Beyond its minimal area overhead, the neuron is capable of operating at higher spiking frequencies via capacitor optimization, which offers improved trade-offs across area, speed, and energy consumption. In our design, the dendritic structure adopts a multi-synapse shared-dendrite architecture, which achieves a significant reduction in hardware overhead by multiplexing multiple inputs onto a common dendritic branch. Furthermore, by leveraging the intrinsic threshold-switching characteristics of VO_2 devices, the dendritic structure further minimizes the consumption of hardware resources (Supplementary Table 6). These results underscore the potential of VO_2 based memristive devices as a scalable and high-performance building block for future hardware neural networks.”

Supplementary Table 6:

“Supplementary Table 6. Detailed comparison between dendritic circuits^{s17,s33–s37}.”

	Device	Dendritic Mechanism	Dendritic Type	Energy (J/spike)	Time Constant (s)	Dendritic Circuit
D'Ago. et al. ^{s17}	RRAM + 180nm CMOS	temporal process & sequence selectivity	one synapse per dendrite	58.5×10^{-12}	8×10^{-3} $\sim 58 \times 10^{-3}$	40T + 1C + 2RRAM (analysis-based)
Sche. et al. ^{s33}	65nm CMOS	non-linearity & multi-compartment	multiple synapses share a dendrite	$\sim 174 \times 10^{-12}$	N/A	N/A
Li et al. ^{s34}	memristor (Pt/TaO _x /AlO _x /Al)	filter & integrate	multiple synapses share a dendrite	$\sim 600 \times 10^{-12}$	$\sim 10 \times 10^{-3}$	1 memristor
Baek et al. ^{s35}	multigate silicon nanowire FET	non-linearity & direction selectivity	multiple synapses share a dendrite	$\sim 130 \times 10^{-12}$	250×10^{-3}	1FET + 1 sol-gel layer
Yang et al. ^{s36}	anti-ambipolar transistor	meta-plasticity	multiple synapses share a dendrite	N/A	$\sim 5 \times 10^{-6}$	1AAB transistor
Moro et al. ^{s37}	RRAM + 130nm CMOS	temporal process & coincidence detection	one synapse per dendrite	N/A	$\sim 20 \times 10^{-6}$	20T + 3C + 2RRAM + 1AMP
This work	VO_2 + RRAM + 180nm CMOS	integrate & temporal process	multiple synapses share a dendrite	19.29×10^{-12}	15×10^{-6} $\sim 5 \times 10^{-3}$	2T + 6R + 2RRAM + 1AMP

References:

45. D'Agostino, S. et al. DenRAM: neuromorphic dendritic architecture with RRAM for efficient temporal processing with delays. Nat Commun 15, 3446 (2024).

46. Moro, F. et al. Neuromorphic object localization using resistive memories and ultrasonic transducers. *Nat Commun* 13, 3506 (2022).

47. Schemmel, J., Kriener, L., Müller, P. & Meier, K. An accelerated analog neuromorphic hardware system emulating NMDA- and calcium-based non-linear dendrites. in *2017 International Joint Conference on Neural Networks (IJCNN)* 2217–2226 (2017).

48. Li, X. et al. Power-efficient neural network with artificial dendrites. *Nat. Nanotechnol.* 15, 776–782 (2020).

3. The authors indicate that the achievable dendritic time constant can be extended from the microsecond regime into the millisecond range (15 μ s–5 ms). I would like to see a more detailed analysis of the impact of the dendritic time constant on system-level performance, including simulations with even longer time constants (beyond 5 ms).

Our response: We sincerely thank the reviewer for this insightful comment. In response, we have conducted additional analyses to investigate the impact of the dendritic time constant on system-level performance.

First, we would like to clarify the temporal representation of our data. The original speech samples from the Google Speech Commands (GSC) dataset consist of 1-second recordings sampled at 16 kHz. During preprocessing, log-Mel filterbank features are extracted, resulting in 120 feature channels and 101 temporal frames per utterance (with a 10 ms hop size). After this preprocessing stage, the input to the neural network is a sequence of discrete feature frames rather than the original continuous-time waveform. Therefore, the processing speed of the proposed system is no longer constrained by the physical duration of the original signal. Instead, it is determined by the effective temporal dynamics of the neuron model, particularly the dendritic time constant. A smaller time constant allows the neurons to process input frames at a higher frequency, effectively "compressing" the task into a much shorter physical time scale. This

flexibility allows our hardware to accelerate inference significantly compared to real-time execution.

The above discussion provides a qualitative interpretation of our design. To further address the reviewer's concern, we additionally performed a quantitative analysis of how the dendritic delay affects system-level performance. In our proposed system, the temporal dimension of the preprocessed signals is represented by 101 discrete feature frames, rather than the original physical time scale. To facilitate quantitative analysis, we rescaled the temporal axis by assuming that each frame corresponds to an effective duration of 10 μs in the system-level simulation. Consequently, all temporal values reported below are expressed under this scaling assumption. As shown in Supplementary Fig. 35, when the dendritic delay range increases from 20 μs to 160 μs , the classification accuracy on the speech recognition task improves significantly. This improvement can be attributed to the fact that a longer dendritic delay creates a wider temporal synchronization window, enabling effective spatiotemporal integration of signals across multiple input frames. When the dendritic delay is further increased, the network accuracy remains relatively stable at a high level. This behavior suggests that the critical temporal features of the task are primarily captured within a time scale corresponding to hundreds of microseconds in the rescaled frame domain. However, when the dendritic delay increases beyond approximately 500 μs , we observe a slight degradation in performance. We attribute this decline to excessive temporal tailing and overlap of the input features, which introduces temporal ambiguity and reduces the discriminability of the encoded signals. These additional analyses demonstrate that the dendritic time constant plays an important role in balancing temporal integration and temporal resolution, and that there exists an optimal delay range that maximizes system-level performance.

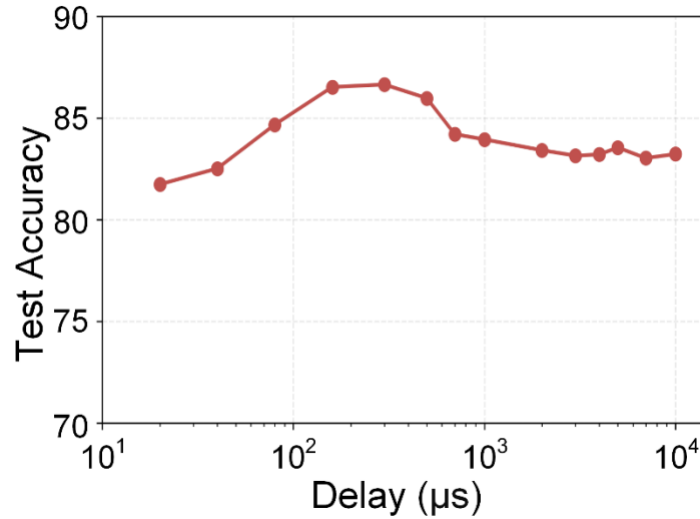
To address the reviewer's concern, we have revised the manuscript accordingly.

Page 26:

"We further analyzed the impact of dendritic delay range on network performance (Supplementary Fig. 35). The results indicate that an optimal delay range exists that maximizes system performance."

Supplementary Fig. 35:

“Supplementary Fig. 35. Impact of dendritic delay on system-level performance. In the analysis, the temporal dimension of the input is represented by 101 discrete feature frames obtained from log-Mel preprocessing. For quantitative evaluation, each frame is rescaled to an effective duration of 10 μ s in the system-level simulation. As the dendritic delay increases from 20 μ s to 160 μ s, the classification accuracy improves due to enhanced spatiotemporal integration across multiple frames. The accuracy remains relatively stable for larger delays but slightly degrades when the delay exceeds $\sim$ 500 μ s, which is attributed to excessive temporal overlap of the input features.”



4. Finally, the use of differential synapses to represent positive and negative weights is a well-established design choice in RRAM-based systems. However, the addition of a circuit implementing a ReLU function prior to the neuron increases circuit complexity and likely impacts both area and power consumption. This design choice should be clearly justified in the main text, and its implications for scalability, area, and energy efficiency should be explicitly discussed and, if possible, quantified.

Our response: We sincerely thank the reviewer for this insightful comment. We agree that the use of differential RRAM structures to represent positive and negative weights is a well-established approach.

Regarding the ReLU functionality, we realize that our previous description of the ReLU function may have caused some misunderstanding. (i.e., “If this is the type of ‘integration of negative currents’ the reviewer refers to, we would like to clarify that in our network simulation, the accumulated current is passed through a ReLU-type nonlinearity before being delivered to the neuron”).

So, we would like to further clarify that no additional hardware module or specific ReLU circuit was added to our design. Instead, the ReLU function is only applied during system-level simulations to ensure that the simulated neuron behavior matches that of our designed neuron circuit. In our hardware design, the neuron circuit receives input currents through a current mirror. Since the current mirror can only source or sink current in one direction, any negative current from the RRAM crossbar naturally results in a zero input to the neuron as the transistors enter the cut-off region. Therefore, the inclusion of the ReLU function in simulation is solely for the purpose of accurately reflecting the behavior of the neuron circuit at the system level.

In the hardware system, utilizing a current mirror inherently restricts the input current to the neuron to be positive, effectively realizing the ReLU function without the need for additional dedicated circuitry. For scalability, this approach simplifies the per-neuron peripheral circuitry, which is beneficial for extending the system to large-scale neuromorphic computing. Regarding area, no additional hardware is required for the activation function, thereby avoiding the overhead typically associated with dedicated circuits. In terms of energy efficiency, only a small amount of static power is consumed to perform the rectification, and the circuit remains effectively idle when the input is negative, resulting in negligible energy overhead.

To clearly present our work and avoid potential misunderstandings, we have revised the manuscript accordingly.

Page 20:

“During network simulations, neuron dynamics are modeled based on their circuit designs using the following membrane equations (1, 2, 3). The input current I is passed through a ReLU function in simulations to reflect the hardware behavior, as the physical neuron circuit receives current through a current mirror that inherently restricts the input to positive values without requiring additional circuitry.”