

Late Breaking Results: *SP-HD*: Stochastic Projection-Based HyperDimensional Architecture for Near-Sensor Image Classification

Ahmed Mamdouh*, Sabrina Hassan Moon*, Abu Kaisar Mohammad Masum†, Emilien J. Meyer†, Sercan Aygun†, and Dayane Reis*

*Bellini College of AI, Cybersecurity and Computing, University of South Florida, Tampa, FL, USA

†School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, LA, USA

{ahmed749, ms38, dayane3}@usf.edu, {c00591145, emilien.meyer1, sercan.aygun}@louisiana.edu

Abstract—This paper presents *SP-HD*, a near-sensor image classification architecture that combines stochastic computing (SC) and hyperdimensional computing (HDC) to enable energy-efficient and compact embedded intelligence. The proposed approach introduces a *stochastic projection mechanism* that converts input features into bitstreams, enabling bipolar multiplications to be performed with simple logic and in-memory accumulation, thereby eliminating costly multipliers and level hypervectors. A mixed-signal ReRAM-based implementation further reduces data movement by performing projection and accumulation directly within the memory fabric, while binary-weight classification minimizes circuit complexity. *SP-HD* achieves competitive accuracy across multiple image datasets and delivers $3\mu J$ energy per inference with a compact $2.56mm^2$ hardware footprint, significantly outperforming prior ReRAM compute-in-memory accelerators in both energy and area efficiency.

I. INTRODUCTION

The rapid development of Internet-of-Things (IoT) devices has created a growing demand for energy-efficient, always-on, and near-sensor intelligence. Deep Neural Network (DNN) approaches, while accurate, are often resource-intensive and difficult to deploy efficiently on such edge platforms. This has motivated the exploration of alternative computing paradigms that trade precision for robustness, parallelism, and hardware simplicity [1], [2].

Hyperdimensional Computing (HDC) has emerged as a promising alternative due to its brain-inspired representation and information processing in high-dimensional spaces [3]. In HDC, data is encoded as long, distributed hypervectors (\mathcal{HVs}), typically with thousands of dimensions. The key operations, *binding*, *bundling*, and *similarity measurement*, are realized through simple arithmetic and logical functions that are highly parallelizable. These properties allow HDC systems to maintain high classification accuracy even under noise, quantization, or bit errors, making them particularly well suited for hardware implementations [4]–[6].

Although HDC provides a robust framework for representing data in high-dimensional spaces, the projection of raw feature values into \mathcal{HVs} can lead to a loss of fine-grained numerical information. Once features are mapped into discrete bipolar components, operations rely on coarse *sign*-based interactions rather than the original feature magnitudes. Stochastic Computing (SC) offers a complementary computational model that enables arithmetic operations such as bipolar multiplication to be performed directly in the bitstream domain [7]. In SC, numerical values are encoded as *stochastic bitstreams* with probabilities, and operations are implemented using simple logic gates. This approach keeps the probabilistic structure of the input while providing a hardware-efficient computation [8].

By combining HDC with SC, we exploit the synergy between the two paradigms: HDC provides robustness and noise tolerance, while SC offers lightweight arithmetic for high-dimensional operations. Recent efforts have begun exploring the combination of HDC and SC to reduce hardware complexity and energy cost in high-dimensional operations. For instance, reference [8] demonstrated that stochastic bitstreams can replace multipliers in binarized neural networks (BNNs), hinting at the potential of bitstream-based computation for HDC-style vector operations. While BNNs are efficient, their layered

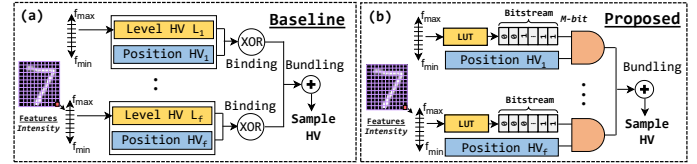


Fig. 1. (a) Baseline HDC and (b) Proposed projection-based bitstream HDC.

structure is noise-sensitive [9], whereas HDC’s distributed representations offer inherent robustness. Reference [10] has explored approximate or quantized HDC encoders to simplify hardware, yet the design relies on digital multipliers and requires large memory to store dense projection matrices with significant accuracy loss in the case of aggressive quantization.

This work addresses these limitations by proposing a stochastic projection-based HDC architecture that maintains accuracy while drastically reducing hardware overhead. Fig. 1 illustrates two encoding paths: (a) the baseline HDC pipeline, where pixel intensities are bound with positional HVs to form a sample HV, and (b) our proposed pipeline, which replaces intensity binding with a structured projection using an M -bit bitstream to generate raw features into bipolar HVs using a traditional HDC approach, we convert input values into stochastic bitstreams and perform bipolar multiplications using simple logic and analog in-memory accumulation. This mitigates the precision loss associated with HDC’s HV projection and eliminates the need for digital multipliers required by projection-based encoding. Furthermore, we explore a ReRAM crossbar-based in-memory computing architecture, enabling energy-efficient computation across a range of deployment scenarios.

The key contributions of this work are summarized as follows: ① A stochastic projection-based HDC encoding framework that replaces conventional multipliers with bitstream-based bipolar multiplication. ② ReRAM-based crossbar architecture performs stochastic projection and accumulation directly in memory, achieving massively parallel, energy-efficient classification.

II. PROPOSED *SP-HD* ARCHITECTURE

A. Stochastic Projection-Based Encoding

Projection-based encoding provides higher classification accuracy than traditional record-based techniques in HDC systems [10], but at the cost of significant hardware complexity. Implementing projection demands many multipliers in digital designs or DACs in ReRAM-based implementations, both of which incur substantial area, energy, and design overhead. These requirements make projection-based HDC impractical for highly resource-constrained IoT platforms.

To address these limitations, we introduce SC into the projection stage. As shown in Fig. 1, input features are first converted into stochastic bitstreams through Look-Up Tables (LUTs). Each bitstream then acts as a gating signal that selectively enables the addition of its corresponding position hypervector to the final sample HV. This gating occurs serially over M

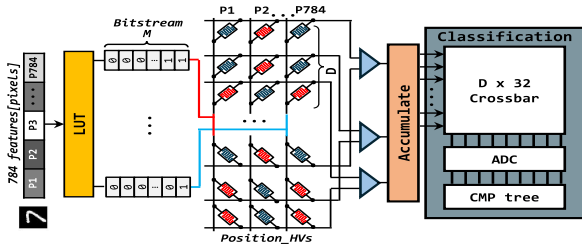


Fig. 2. Proposed *SP-HD* hardware.

TABLE I
ACCURACY COMPARISON W/ LEHDC [11]

Dataset	Method	$D=4K$	$D=1K$	Dataset	Method	$D=4K$	$D=1K$
MNIST [13]	LeHDC	0.90	0.84	F.MNIST [14]	LeHDC	0.82	0.77
MNIST [13]	<i>SP-HD</i>	0.93	0.89	F.MNIST [14]	<i>SP-HD</i>	0.84	0.79
P.MNIST [15]	LeHDC	0.62	0.62	B.MNIST [16]	LeHDC	0.73	0.73
P.MNIST [15]	<i>SP-HD</i>	0.84	0.84	B.MNIST [16]	<i>SP-HD</i>	0.83	0.78

cycles, where M is the bitstream length (much shorter than in regular SC streams and here representing the precision of the data). In effect, this process implements the multiplication step of projection-based encoding using only an AND gate and the standard bundling operation. This substitution significantly reduces hardware cost while preserving the benefits of projection-based encoding. Additionally, unlike baseline HDC systems, the proposed *SP-HD* approach removes the need for level hypervectors, thereby reducing the overall memory footprint.

B. Training Approach

For model training, we adopt the BNN-similar learning approach introduced in LeHDC [11], which optimizes HDC parameters using backpropagation through a binarized representation. Unlike LeHDC, which maintains bipolar weights for inference, we binarize the learned weights after training to further simplify the on-edge inference engine. This reduces storage requirements and eliminates the need for sign-handling logic, enabling a more compact and energy-efficient hardware implementation.

C. ReRAM Crossbar-Based Architecture

We propose a ReRAM crossbar-based architecture for stochastic projection encoding, shown in Fig. 2. The design leverages the analog accumulation of ReRAM arrays to execute projection and summation in-memory, lowering data movement and computational cost. Input features are converted into stochastic bitstreams of length M via LUTs and applied as row activations. Two ReRAM crossbars store the $+1$ and -1 components of the positional hypervectors in a differential format [12]. Each row represents an input feature and each column a hypervector dimension D . During operation, only rows receiving a ‘1’ activation participate in the computation.

For each active row, both crossbars accumulate bitline currents, and the positive and negative outputs are combined into a signed analog value per dimension. A threshold comparator binarizes each column current, and these bits are integrated over M cycles to produce the final sign bit for each hypervector dimension. Classification uses a $D \times 32$ ReRAM crossbar storing the class weight vectors, followed by an ADC and comparator tree. The sample HV sign bits drive the crossbar rows, the digitized currents are then compared, and the class with the strongest response is selected.

III. RESULTS & DISCUSSION

A. Model Accuracy

Table I compares the classification accuracy of the proposed *SP-HD* encoding with LeHDC across four datasets and two

TABLE II
SP-HD BREAKDOWN ANALYSIS

Component	Area (mm ²)	Power mW	Component	Area (mm ²)	Power mW
① LUT	0.08	8.3	② PosHVs Crossb.	1.57	1.88E+04
③ Accumulate	0.18	22.7	④ Classification	0.12	112
Total + (30% PNR): Area = 2.56 mm², Power = 1.89E+04 mW					

hypervector dimensionalities $D = 4096$ and $D = 1024$. Importantly, both approaches use binary weights during inference, allowing a fair comparison of their robustness under highly quantized model parameters. *SP-HD* consistently outperforms LeHDC across all datasets; the most significant gains occur on PneumoniaMNIST [15] and BreastMNIST [16] datasets. Even on simpler datasets such as MNIST [13] and FashionMNIST [14], *SP-HD* provides noticeable improvements (e.g., 0.93 vs. 0.90 on MNIST at $D=4K$). Reducing the dimensionality from $4K$ to $1K$ lowers accuracy for both methods, but *SP-HD* continues to maintain its advantage. Overall, the proposed *SP-HD* architecture achieves higher accuracy, better tolerance to reduced dimensionality, and significantly improved robustness to binary weights compared to LeHDC.

B. Hardware Efficiency

In hardware design, we used the Stanford ReRAM Verilog-A model [17] to design our crossbar. We estimated the area assuming a standard $4F^2$ 1R cell structure, and obtained power directly from SPICE simulations. Digital modules, LUT, accumulate, CMP tree, were synthesized in Cadence Genus using NanGate 45-nm open-cell library [18]. We incorporated current comparator proposed in [19], and the crossbar peripheral units as detailed in Neurosim [20]. The *SP-HD* architecture is designed for $1K$ input features, a hypervector size of $D=4K$, and a bitstream length of only $M=15$ (this is much less than the regular SC architectures [21]). To support real-time near-sensor processing, the system is optimized for 100 MHz operation.

Table II shows the area power distribution across breakdown system components. The total power consumption is approximately $18.9W$, with nearly all of it attributed to the ReRAM crossbar, while the LUTs, accumulator, and classification logic each consume only a few milliwatts. With 16 cycles per inference (15 for stochastic projection and 1 for classification), *SP-HD* achieves a latency of $160ns$ and an energy cost of about $3\mu J$ per inference.

Compared to prior ReRAM accelerators, the efficiency gains are substantial. The design in [22] reports $76.2W$, more than $4\times$ higher than *SP-HD*, and its energy per inference is similarly over four times larger. *SP-HD* also achieves a significant reduction in silicon footprint, occupying $2.56mm^2$ versus the $41mm^2$ reported in [22], an almost $8\times$ area improvement. These benefits were thanks to stochastic bitstream projection and a simplified digital accumulation-comparison setup.

IV. CONCLUSION

This work introduced *SP-HD*, a stochastic projection-based HDC architecture for near-sensor inference. By replacing multipliers with stochastic bitstreams and using ReRAM crossbars for in-memory accumulation, *SP-HD* enables efficient high-dimensional encoding and binary-weight classification with minimal hardware overhead. The method offers improved accuracy and remains resilient to reduced dimensionality and weight binarization. Hardware evaluation shows that *SP-HD* achieves $3\mu J$ per inference and a $2.56mm^2$ area, improving over prior ReRAM accelerators.

REFERENCES

- [1] J. Chen and X. Ran, "Deep learning with edge computing: A review," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1655–1674, 2019.
- [2] W. Lin, A. Adetomi, and T. Arslan, "Low-power ultra-small edge ai accelerators for image recognition with convolution neural networks: Analysis and future directions," *Electronics*, vol. 10, no. 17, p. 2048, 2021.
- [3] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive computation*, vol. 1, pp. 139–159, 2009.
- [4] S. Zhang, K. Juretus, and X. Jiao, "Exploring hyperdimensional computing robustness against hardware errors," *IEEE Transactions on Computers*, vol. 74, no. 6, pp. 1963–1977, 2025.
- [5] S. Aygun, M. S. Moghadam, M. H. Najafi, and M. Imani, "Learning from hypervectors: A survey on hypervector encoding," 2023.
- [6] S. Aygun, M. S. Moghadam, and M. H. Najafi, "uhd: Unary processing for lightweight and dynamic hyperdimensional computing," in *2024 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2024, pp. 1–6.
- [7] S. Liu, J. L. Rosselló, S. Liu, X. Tang, J. Font-Rosselló, C. F. Frasser, W. Qian, J. Han, P. Reviriego, and F. Lombardi, "From multipliers to integrators: A survey of stochastic computing primitives," *IEEE Transactions on Nanotechnology*, vol. 23, pp. 238–249, 2024.
- [8] S. Aygun, E. O. Gunes, and C. De Vleeschouwer, "Efficient and robust bitstream processing in binarised neural networks," *Electronics Letters*, vol. 57, no. 5, pp. 219–222, 2021.
- [9] S. Buschjäger, J.-J. Chen, K.-H. Chen, M. Günzel, C. Hakert, K. Morik, R. Novkin, L. Pfahler, and M. Yayla, "Towards explainable bit error tolerance of resistive ram-based binarized neural networks," *arXiv preprint arXiv:2002.00909*, 2020.
- [10] F. Ponzina and T. Rosing, "Microhd: An accuracy-driven optimization of hyperdimensional computing algorithms for tinyml systems," *arXiv preprint arXiv:2404.00039*, 2024.
- [11] S. Duan, Y. Liu, S. Ren, and X. Xu, "Lehdc: Learning-based hyperdimensional computing classifier," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022, pp. 1111–1116.
- [12] P. Chi, S. Li, C. Xu, T. Zhang, J. Zhao, Y. Liu, Y. Wang, and Y. Xie, "Prime: A novel processing-in-memory architecture for neural network computation in rram-based main memory," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 3, pp. 27–39, 2016.
- [13] Y. Lecun, "Mnist," <https://api.openml.org/d/554>, accessed: 2024-11-22.
- [14] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.
- [15] J. Yang, R. Shi, and B. Ni, "Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis," in *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 2021, pp. 191–195.
- [16] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, "Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification," *Scientific Data*, vol. 10, no. 1, p. 41, 2023.
- [17] Z. Jiang, Y. Wu, S. Yu, L. Yang, K. Song, Z. Karim, and H.-S. P. Wong, "A compact model for metal-oxide resistive random access memory with experiment verification," *IEEE Transactions on Electron Devices*, vol. 63, no. 5, pp. 1884–1892, 2016.
- [18] J. Knudsen, "Nangate 45nm open cell library," *CDNLive, EMEA*, 2008.
- [19] X. Tang and K.-P. Pun, "High-performance cmos current comparator," *Electronics Letters*, vol. 45, no. 20, pp. 1007–1009, 2009.
- [20] P. Chen, X. Peng, and S. Yu, "NeuroSim: A circuit-level macro model for benchmarking neuro-inspired architectures in online learning," *IEEE TCAD*, vol. 37, 2018.
- [21] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Trans. Embed. Comput. Syst.*, vol. 12, no. 2s, May 2013. [Online]. Available: <https://doi.org/10.1145/2465787.2465794>
- [22] C. Liu, K. Wu, H. Liu, H. Jin, X. Liao, Z. Duan, J. Xu, H. Li, Y. Zhang, and J. Yang, "A ReRAM-Based Processing-In-Memory Architecture for Hyperdimensional Computing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2024.