# Compute-in-Memory: From device to application

Che-Kai Liu[1*†], Mengyuan Li[2‡], Mohsen Imani[3⋆], X. Sharon Hu[2⋆], Xunzhao Yin[1**]
Category: Undergraduate, ACM ID: 4394534, *: {kevinliu,xzyin1}@zju.edu.cn
[1] Zhejiang University, P.R.China; [2] University of Notre Dame, USA; [3] University of California Irvine, USA.

## I. Problem and Motivation

Compute-in-memory (CiM) has emerged as a promising architectural paradigm that overcomes the memory wall issue. By integrating basic processing capabilities, CiM is able to perform parallel operations across the entire memory blocks. With the advances in non-volatile memories (NVMs) such as ferroelectric field effect transistor (FeFET) [1], resistive random access memory (ReRAM) [2], spin-transfer torque RAM (STT-RAM) [3], etc., NVM-based crossbar and content addressable memory (CAM) have been employed to accelerate a number of machine-learning models. Crossbar designs have demonstrated the ability to accelerate vector-matrix multiplication (inner product) [4]. On the other hand, CAM designs have demonstrated great potential as highly energy-efficient associative memories (AMs) for nearest neighbor (NN) search using the Hamming distance metric [5].

However, it can be seen in [6] that the CAM designs implementing Hamming distance-based search achieves energy efficiency at the expense of non-negligible accuracy loss for classification tasks. Sophisticated CiM designs have emerged such as the sigmoid-like-based [6] and Euclidean-based [7] CAM. Yet as a widely used distance metric, cosine similarity-based CiM fabric has not been explored since cosine similarity necessitates division operations and multiplications [8], which are not compatible with cutting-edge CiM designs including CAM [9]. The AM in [10] supports a specific approximated CSS by approximating the denominator of cosine calculation and exploiting the quasi-orthogonal property of the hyperdimensional vector. That said, such approximation still causes slight accuracy loss and is limited only to the HDC application. Therefore, a more general CiM-based CSS that can not only offer energy efficiency and performance improvements but also maintain comparable accuracy to the full precision CSS implemented in software is highly desirable. We propose a novel circuit-level CiM design for cosine similarity detection.

With the proposed in-memory cosine similarity search engine, parallel cosine similarity nearest neighbor can be achieved. However, there are still limitations to existing CAM designs, which hinder them from wide applications. To begin with, in existing CAM designs, one CAM can only support one specific distance metric. CAM designs that implement only a single distance metric are only deployable in limited application scenarios and cannot be reused for many other applications. Still, other different distance metrics are used for various applications, such as the inner product (IP) metric for recommendation systems [11] and natural language processing [12], Euclidean distance ($L_2$) when data is mapped on the same plane [13], Manhattan distance ($L_1$) in graph theory, and Hamming distance for hashing algorithms [14]. we propose a novel approach to designing an NVM-based CAM search engine, C$^2$AM, which can be reconfigured to support multiple distance metrics, such as Hamming, $L_1$, $L_2$, and IP.

Finally, at the interface between the designed hardware and the data from real-world sensors, the mass amount of information acquired from sensors is then converted to digital values and passed to the designed CiM fabrics. In-sensor computing has emerged as a novel paradigm that removes the costly digital-analog conversion for specific applications [15], [16]. We propose an in-sensor paradigm for hyperdimensional computing (HDC) that eliminates the costly DACs and find the optimal ADC precision based on design space exploration.

## II. COSIME: FeFET based associative memory for in-memory cosine similarity search

To address the issues above, we proposed COSIME, a FeFET-based associative memory for in-memory cosine similarity search.

### A. Background

In a number of machine learning models at the edge, e.g. few-shot learning [17], hyperdimensional computing (HDC) [18], etc., an input query is searched across the trained class vectors to find the feature class vector in nearest neighbor (NN) cosine similarity metric. However, frequent cosine similarity-based searches (CSSs) over the class vectors necessitate a large number of multiplications, Euclidean normalizations, division operations, and data movements, provoking heavy hardware energy and latency overheads. Compute-in-memory (CiM) is a promising architectural paradigm as stated. For example, CAM designs have demonstrated great potential as highly energy-efficient associative memories (AMs) for nearest neighbor (NN) search using the Hamming distance metric [5], [6]. However, it can be seen in [6] that the CAM designs supporting Hamming distance-based search achieves energy efficiency at the expense of non-negligible accuracy loss for classification tasks. Moreover, cosine similarity necessitates division operations and multiplications [8], which are incompatible with cutting-edge CiM designs including CAM [9].

### B. Problem formulation

$$\cos\langle\vec{a},\vec{b}\rangle = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\times\|\vec{b}\|} \tag{1}$$

Without loss of generality, we assume that $\vec{a}$ is the binary input vector and $\vec{b}$ is the class binary vector stored in the memory block. The numerator of Eq. 1, i.e. the dot product, can be easily realized by NVMs including Resistive Random Access Memory (RRAM) and ferroelectric FET (FeFET) crossbar array [19]. COSIME aims to obtain the closest vector (i.e., NN) to the input query in terms of cosine similarity. From Eq. (1), cosine similarity can be equivalently expressed in a more circuit-friendly variant without affecting the search output, as shown in (2):

† The author is now with School of ECE, Georgia Institute of Technology, GA, USA.
‡ Collaborator. ⋆ Advisors.

$$\cos^2\langle\vec{a},\vec{b}\rangle = \frac{(\vec{a}\cdot\vec{b})^2}{(\|\vec{a}\|\times\|\vec{b}\|)^2} \quad (2)$$

Note that in (2), the denominator consists of the squared norm of the stored vector $\vec{b}$ which is the number of '1's within $\vec{b}$, and squared norm of input query vector $\vec{a}$ which is shared by all the cosine similarity metrics, and thus can be removed during the CSS. In this sense, the cosine similarity metric can be equivalently expressed as the $X^2/Y$ operator, where $X$ denotes the dot product $(\vec{a}\cdot\vec{b})^2$ and $Y$ denotes $\|\vec{b}\|^2$, i.e., the number of '1's within $\vec{b}$ Based on the above formulation, we illustrate the circuits implementing the computation of $X$, $Y$ and $X^2/Y$.

### C. Analog circuit designs

**FeFET Memory Arrays** In this work, the 1FeFET1R structure is adopted and proposed to realize a compact AND gate by storing one operand as the FeFET $V_{TH}$ state and applying the other operand as the gate voltage. The proposed utilization of 1FeFET1R cell is based on the observation in [20] that by connecting a series resistor with proper resistance value on the FeFET source/drain terminal, the ON state current will be only limited by the series resistance. Specifically, with [21] experimentally demonstrated a back-end-of-line 1FeFET1R structure and a $M\Omega$ resistor with less than 8% variability, we propose the possibility to adjust the resistor in the 1FeFET1R to satisfy the required input current range for the translinear circuit following the 1FeFET1R memory.

**Translinear circuits** To implement the key operation $X^2/Y$ for CSS, we propose to employ the translinear circuit from [22] and feed the output currents of FeFET memory arrays $I_x$ and $I_y$ into the analog translinear circuit. Fig. Fig.1(b) shows the schematic of the translinear circuit implementing efficient current-mode squaring and division. This translinear circuit mainly consists of a translinear loop (indicated by the blue arrow) including clockwise (CW) transistors M1, M4 and counterclockwise (CCW) transistors M2, M5. The transistors along the loop are operating in the subthreshold (weak inversion) region, and their drain-source currents can be characterized by the following expression [23]:

$$I_{DS} \approx I_0 \frac{W}{L} e^{\frac{V_{GS}}{\eta V_T}} \quad (3)$$

where $I_0$ denotes the drain current $I_D$ when $V_{GS} = V_T$, $V_T$ denotes the thermal voltage, and $\eta$, the subthreshold slope factor.

The relation between the $V_{GS}$'s of the transistors along the translinear loop follows Kirchoff's Law, i.e., $\sum_{CW} V_{GS} = \sum_{CCW} V_{GS}$ , and from Eq. (3), we obtain:

$$V_{GS} = V_T \eta ln(\frac{I_{DS}}{I_0}\frac{L}{W}) \quad (4)$$

By substituting (4) into the above Kirchoff's Law while keeping the loop transistors in the subthreshold region, the translinear circuit generates the analog output current $I_z$ as below:

$$I_z = \frac{I_x^2}{I_y} \quad (5)$$

**Winner-take-all circuit** Nearest neighbor (NN) search, or, the conventional maximum current selection implementation is a current comparator-based tree structure that requires a huge number of transistors and increases the latency as the number of stored class vectors increases [24]. Here, we propose to employ a current mode winner-take-all (WTA) circuit in [25] shown in Fig.1(c), which can offer efficient and scalable maximum current detection operation. without loss of generality, we assume a small change in $I_{z1}$, so a 2-rail-input WTA's $V_1$ is a linear function of $I_{z1}$ with a slope of $\frac{V_A}{2I_{z1}}$ given in [26]. We then revisit the scalability of a M-rail-input WTA circuit and conclude the winner dynamics to be $\frac{dV_1}{dI_{z1}} = \frac{M-1}{M}\frac{V_A}{I_{z1}}$, while for losers $j \in [2, M]$, to be $\frac{dV_j}{dI_{z1}} = \frac{-1}{M}\frac{V_A}{I_{z1}}$. As the input rail number increases, the behavior only differs by a constant number. We also experimentally justify in Cadence Spectre that the impact of the number of input rails on the winner's output is negligible.

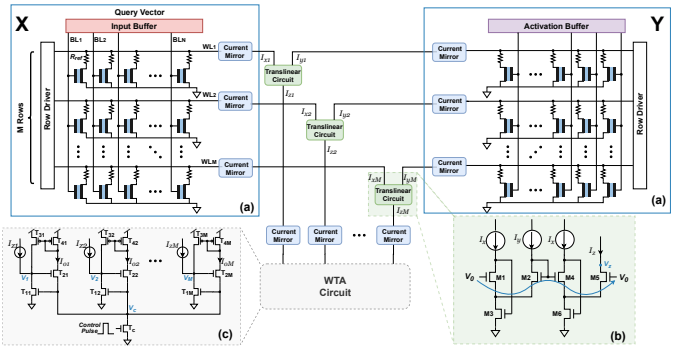


Fig. 1: COSIME overview. (a) 1FeFET1R memory array. (b) Translinear circuit. (c) Winner-Take-All (WTA) circuit.

TABLE I: Comparison of Existing AMs with Different Distance Metrics

| Memory | Technology | Metric | Search Energy per bit$(fJ)$ | Latency$(ns)$ | Area$^*(mm^2)$ | Process (nm) |
|---|---|---|---|---|---|---|
| **A-HAM** [24] | RRAM | Hamming | 0.20(×0.7) | 8.92(×2.9) | 0.524(×26.5) | 45 |
| **FeFET TCAM** [5] | FeFET | Hamming | 0.40(×1.4) | 0.36(×0.12) | 0.010§(×0.51) | 45 |
| **E$^2$-MCAM$^\star$ (1.5 V)** [27] | Flash | Euclidean$^2$ | 0.56(×1.95) | 5.85(×1.95) | 0.192(×9.7) | 55 |
| **Approx. Cosine** [10] | RRAM | Approx. Cosine | 25.9(×90.5) | 1000(×333) | 0.026¶(×1.31) | 90/65† |
| **COSIME (this work)** | FeFET | **Cosine** | **0.286**(×1) | **3**(×1) | **0.0198**(×1) | **45** |

*: Assuming $256 \times 256$ array size. §: Area associated with sensing is not included. ¶: Area is estimated via Neurosim [28] and scaled to 45nm technology for fair comparison. ⋆: E$^2$-MCAM stores 3 bits per cell for search, and the sensing circuitry energy is not included. †: NVM is based on 90nm CMOS while digital peripherals are based on 65m.

### D. Results

We first evaluate the energy and latency of the proposed COSIME with Cadence Spectre at the array level. We then investigate the scalability and robustness of COSIME upon device variations, extracted from [20], [21], and we assume 10% of size, $V_{TH}$, and supply voltage variations. We finally benchmark COSIME for binary HDC inference and compare it with a GPU implementation.

The functionality, scalability, and robustness of COSIME have been validated under a harsh condition, where two non-identical stored vectors are closest to each other, i.e., they only differ by 1 bit at the denominator, and the resulted squared cosine similarities are $\cos^2\theta = 1/4$ and $1/5$. Array level simulations suggest that the maximum error rate is $\approx 10\%$, which would have minimal impact on the application-level accuracy for many machine learning and neuromorphic applications, such as HDC [10], [13], [24]. The energy and latency results of COSIME at the array level indicate 90.5× and 333× improvements over the state-of-the-art approximated CSS design, respectively. HDC application benchmarking suggests that COSIME achieves 47.1× speedup and 98.5× energy efficiency improvement over a GPU implementation.

Table IV demonstrates the area overheads of different AMs. Both the A-HAM in [24] and the E2-MCAM in [27] consume high area overhead since a tree-based loser-take-all (LTA) circuitry and sufficiently large flash cells supporting the 3-bit storage are used, respectively. The approximated CSS design in [10] consumes 1.31× area overhead than COSIME since it adopts ADC for its RRAM readout. The significant improvements of COSIME over the counterpart approximated CSS design mainly benefit from the following aspects: (1) the advantages of FeFET in read/write energy [5]; (2) the 1FeFET1R structure limiting the conducting current within COSIME, which improves the energy efficiency and mitigates the variations; and (3) relatively simple analog circuits in COSIME compared with the capacitor and analog-to-digital converter (ADC) in [10].

Hardware acceleration for CSS is important for edge intelligence and AI models. In this work, we propose for the first time, COSIME, a FeFET-based AM that performs CSS in-memory. Note that the proposed COSIME design is not limited to FeFET technology, but is rather general and can be applied to other NVMs with access transistors. This is because the peripheral circuitry of COSIME is largely independent of the NVM array as long as the array output currents are within the sensing range. Therefore, COSIME paves a promising way toward efficient CiM designs for CSS in data-intensive applications.

## III. C$^2$AM: Content-Addressable Memory with Reconfigurable Distance Metrics

### A. Background

Current associative memory (AM) designs fail to offer reconfigurability in terms of different distance metrics. In this work, we propose a novel approach for designing a FeFET-based configurable CAM search engine (C$^2$AM) which can be reconfigured to support multiple distance metrics, such as Hamming, $L_1$, $L_2$, and Inner Product (IP). Specifically, we propose an adaptive programming (ADP) method to determine the thresholdvoltagesettingsofFeFETs.These $V_T$ settings govern the distance metric implemented in a FeFET-based CAM. ADP systematically finds the VT settings that approximate different distance metrics, based on an optimization problem formulated to minimize the error between the CAM distance and the ideal distance. We further include inherent NVM device variations in our optimization problem and investigate the trade-off between the accuracy and latency of the C$^2$AM.

### B. Reconfigurable CAM designs

*1) Overview - adaptive programming:* ADP design aims to address the reconfigurability of other distance functions such as $L_1$ and $L_2$ in a single two-FeFET CAM cell. For one bit, these distances are the same as Hamming distance, so no additional design is required. However, for multi-bit scenarios, with different query values, the final distance varies in these distances, that is, different discharging currents are required in each CAM cell. By evenly distributing the threshold and search voltages, previous work found that the cell currents or conductances1 of different states can approximate the $L_2$ distance. We observe that by changing the threshold and query voltages, the current distribution can be changed. ***The problem is that is it possible to find the optimal VT and VQ voltages that can approximate any distance metrics with the specified underlying device?***

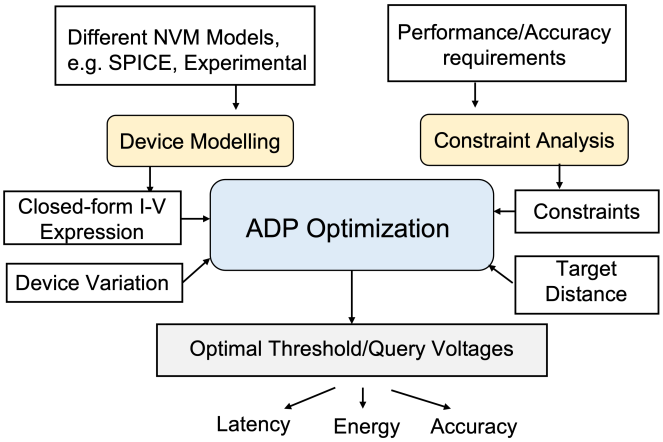


Fig. 2: Overview of the C$^2$AM methodology to find the optimal threshold and query voltages for the target distance on given NVM models.

To answer the question, we propose the ADP method, which is able to find the optimal programming scheme and approximate the MCAM distance to different distance metrics. We first formulate the problem of finding the optimal thresholds and search voltages for different distance metrics in Section III-B1. The goal of the optimization is to minimize the error between the CAM distance and the ideal distance. Specifically, given two data points $d_1$ and $d_2$, our goal is to find a programming method for the array such that the CAM distance $d_{CAM}$ between $d_1$ and $d_2$ is as close to the ideal distance $d_{ideal}$ as possible.

*2) Optimization problem formulation:* Given a multi-bit CAM model, we can define the problem of achieving different distance metrics using CAM as follows. For a CAM with $N$ states, we have $N$ distinct query voltages $Q = \{V_{Q1}, V_{Q2}, ..., V_{Q2^b}\}$ and $N$ threshold voltages $T = \{V_{T1}, V_{T2}, ..., V_{T2^b}\}$. We model the CAM distance, which is represented by the current between the $i$-th query voltage and the $j$-th threshold voltage, as $I(V_{Qi}, V_{Tj})$. We use $D = \{d_{11}, d_{12}, \ldots, d_{ij}, d_{NN}\}$ to denote the expected distances, which is fixed given the expected distance function. For example, $d_{ij} = |i-j|$ for $L_1$ distance and $d_{ij} = (i-j)^2$ for $L_2$ distance. Given the definition of CAM distance $I(V_{Qi}, V_{Tj})$ and the expected distance $D$, the problem is to find the optimal query voltages $Q$ and threshold voltages $T$ for each state such that the error between each CAM distance $I(V_{Qi}, V_{Tj})$ and the expected distance $d_{ij}$ is minimized. Thus, we formulate the **C$^2$AM optimization problem** as follows:

$$\begin{aligned} \underset{Q,T,\alpha,\gamma}{\text{minimize}} \quad & max(I(V_{Qi}, V_{Tj}) - \alpha \cdot d_{ij} + \gamma) \\ \text{subject to} \quad & V_{Qi} \geq 0, V_{Tj} \geq 0, \forall i, j \end{aligned} \tag{6}$$

where $\alpha$ and $\gamma$ are two additional variables we introduce to tradeoff the performance of different results.

*3) Device modeling:* Figure 2 illustrates the general methodology of C$^2$AM. First, C$^2$AM models the device by the closed-form I-V expression and analyzes the constraints on variables given different NVM models. For example, FeFET follows field-effect transistor characteristics, and FeFET CAM operated in the saturation regime can be characterized by the

following equation:

$$I = K\mu C\frac{W}{L}(V_{GS} - V_T)^2(1 + \lambda V_{DS}) \tag{7}$$

The mobility $\mu$, gate equivalent capacitance $C$, and the width over length $W/L$ are generally combined into a constant $\beta$. Here we add a scaling factor $K$ to model the current given different stored $V_T$ better. However, given different models/experimental data, several constants may be dependent on each other, which can be seen in the later paragraph.

For a case study, we exploit the Preisach model [29] in SPICE for FeFET modeling. We discover that $K \times \beta$ doesn't maintain constant under different $V_T$. Instead, with the increment of $V_T$, $K \times \beta$ increases non-linearly. This can be attributed to the mobility degradation of the transistor [30]:

$$\mu_n = \frac{c_1}{1 + (\frac{V_{GS}+V_T}{c_2})^{c_3}} \tag{8}$$

This is caused by the fact that a high voltage at the gate of the transistor attracts the carriers to the edge of the channel, causing collisions with the oxide interface that slow the carriers. Since this work targets current-based CAM, $(1 + \lambda V_{DS})$ remains constant. We model the device by using minimum square error between Preisach FeFET model and Equation 7 as follows:

$$I = (\frac{0.038}{1.176 - V_T} + 0.257)(V_{GS} - V_T)^2 \tag{9}$$

Notice that $V_T$s are obtained via linear extrapolation in the saturation regime [31] and then $K \times \beta$ is transformed into the function of $V_T$. Thus, we can model the device as a function of only threshold voltages and query voltages. That is, $I(V_{Qi}, V_{Tj}) = (0.038/(1.176 - V_{Tj} + 0.257)(V_{Qi} - V_{Tj})^2$

*4) Constraint analysis:* In the C$^2$AM optimization problem, there are several constraints added to the variables.

Q and T Constraints: First, given the supply voltage Vdd and write voltage limitation. $V_{Tj}$ and $V_{Qi}$ are limited in certain range, e.g.,$V_{Qi} \in [0, Vdd]$, and $V_{Tj} \in [-0.5, 1.2]$ for the FeFET Preisach model. Second, we include the constraint that for each state, $V_{Qi} \geq V_{Ti}$ to avoid the case that $I = 0$ when $V_{Qi} < V_{Ti}$,

$\alpha$ and $\gamma$ Constraints: We consider the impact of non-ideal factors and the performance requirements to determine the constraints for both $\alpha$ and $\gamma$. Non-idealities of NVMs play a critical role in their circuit and architecture. Device-to-device [32] and cycle-to-cycle [33] variations are detrimental to the reliability of the circuit and architecture. Also, the margin of the sensing circuit impacts the actual CAM distance distribution. The nonideal factors have a greater impact on the set with lower magnitude, making it difficult to differentiate each state. Thus, two additional variables, $\alpha$ and $\gamma$, are introduced in the objectives to adjust the ideal distance $D$ to $\alpha * d_{ij} + \gamma$, which controls the trends of the CAM distances. This allows us to find solutions that balance performance and accuracy requirements by adding constraints on the variables $\alpha$ and $\gamma$.

*5) Problem solution:* As such, the C$^2$AM problem transforms into a multivariate non-linear problem given a device model and expected distance metric. Here we adopt the Preisach FeFET model and $L_1$ distance as an example.

$$h_{i,j}(V_{Qi}, V_{Tj}, \alpha, \gamma) = (\frac{0.038}{1.176 - t_j} + 0.257)(V_{Qi} - V_{Tj})^2 + \gamma - \alpha|i - j| \tag{10}$$

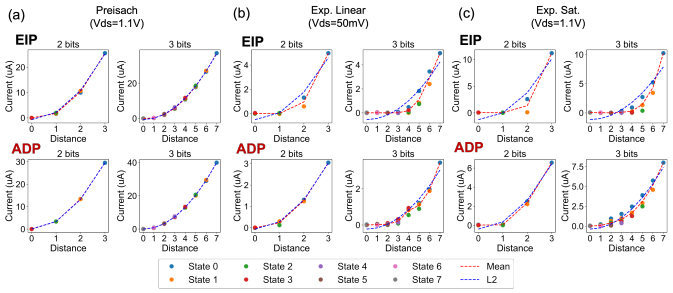


Fig. 3: (a) 2-bit / 3-bit $L_2$ distance found using previous EIP method [6] and the proposed ADP method found in the FeFET preisach model. (b) 2-bit/ 3-bit $L_2$ distance verified with FeFET experimental data using EIP and ADP methods when the device is in linear region ($V_{DS} = 50mV$). (c) 2-bit/ 3-bit $L_2$ distance verified with FeFET experimental data using EIP and ADP methods when the device is in saturation region ($V_{DS} = 1.1V$).

where the inputs are $V_{T1}, V_{T2}, ..., V_{T2^b}$, $V_{Q1}, V_{Q2}, ..., V_{Q2^b}$, $\alpha$, $\gamma$; $|i - j|$ is the expected $L_1$ distance, and $\alpha$ as well as $\gamma$ are variables for performance tradeoff. The error between the CAM distance and ideal distance can be calculated as the maximum of $h_{i,j} \forall i, j$, and the optimization problem can be solved by minimizing the error.

To solve the multivariate nonlinear problem, we use an iterative method Levenberg-Marquardt algorithm [34] which combines the steepest descent and the Gauss-Newton methods. This algorithm is based on the Taylor series for the multivariate optimization problem. In each step $t$, the method optimize $X^t$ as follows:

$$X^{t+1} = X^t - (J_f^T J_f + \lambda I)^{-1} J_f^T f(X^t) \tag{11}$$

where $J_f$ is the Jacobian matrix of function $f$, and $\lambda$ is a parameter. A damping factor, $\lambda$, is used to control the step size of each iteration and is adjusted dynamically to ensure convergence. The algorithm starts with a large $\lambda$ value, which is gradually decreased until the desired accuracy is achieved. At the end, the optimal threshold and query voltages can be found through the optimization process.

### C. Results

Figure 3 visualizes the $L_2$ distance in a single FeFET cell based on prior art and the proposed ADP method, both leveraging the Preisach model for a fair comparison. Moreover, we illustrate the $L_1$ distance metric in the linear and saturation region with the proposed ADP methodology. For the first time, we demonstrate the $L_1$ distance metric under the FeFET Preisach model, experimental FeFET in the linear region, and experimental FeFET in the saturation region. It can be seen from Table II that the fabricated device that works in the linear region has a lower MSE.

## IV. HDSense

Figure 4 shows the proposed HDSense. A FeFET-based crossbar array in Figure 4 is used for the encoding phase of HDC. MUXes are needed as ADCs are shared at the crossbar output, and the switch matrix that consists of multiple transmission gates is used for row-by-row updates of the crossbar. After encoding, the data is then input to the CAM array to identify the nearest neighbor Hamming distance w.r.t the stored class hypervectors in the CAM array. Directly removing

| Designs | MSE (Preisach) | Delay (Preisach) | Energy (Preisach) | MSE (Experiments) linear/saturation |
|---|---|---|---|---|
| 2 bits L2 | | | | |
| EIP [6] | 0.257 | 1.46ns | 1.16fJ | 0.26/2.22 |
| ADP | 0.011 | 1.06ns | 1.23fJ | 0.001/0.09 |
| 3 bits L2 | | | | |
| EIP [6] | 0.125 | 1.34ns | 1.13fJ | 0.30/1.99 |
| ADP | 0.002 | 1.20ns | 1.16fJ | 0.059/0.15 |
| 2 bits L1 | | | | |
| ADP | 1.14 | 449ps | 1.21fJ | 0.016/0.18 |
| 3 bits L1 | | | | |
| ADP | 3.68 | 598ps | 1.33fJ | 0.018/0.389 |

TABLE II: MSE, Latency, Energy comparison using the proposed ADP method on the Preisach model for 2/3 bit L1 and L2 distance, and MSE using the proposed ADP method on the fabricated FeFETs.

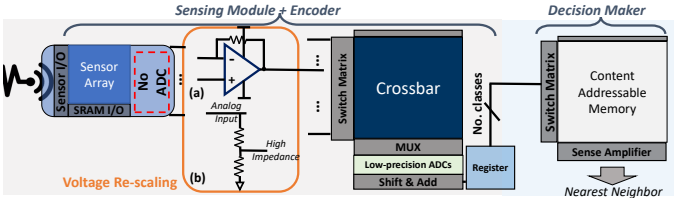


Fig. 4: HDSense, the proposed in-sensor CiM architecture for HDC.

the ADC and DAC blocks can be problematic. The analog voltage sensed from each antenna or pixel should pass through an interface that is responsible for re-scaling the voltage for the NVM input. For example, our SPICE simulation for 22nm FeFET shows the read voltages at the gate terminal should be no more than 1.3 volts in order to maintain the stored characteristics [35]. Figure 4 illustrates possible ways of integrating a basic voltage divider. If choosing ReRAM as the targeted NVM, the key challenge is the low on-state resistance in ReRAM. The WL load of the ReRAM crossbar consists of parallel-connected resistors, and therefore, a voltage buffer (typically an operational amplifier) with low output-impedance is required to provide stable WL voltage for inference [36], and at the same time, re-scale the input analog voltages into the desired input voltage range of ReRAM. On the other hand, the advantage of FeFET is that the WL connects to the transistor's gate, which is a capacitive load. Implementing resistive voltage dividers (Figure 4(b)) for the input analog voltages becomes feasible and thus introduces negligible hardware overhead.

### A. Evaluation

we developed a calibrated non-linear quantized HDC model for the HDC. The framework is implemented by Python language and Pytorch packages and supports both full-precision and quantized HDC encoding and classification. It can be seen that a full-precision query with a 4-bit crossbar array reaches a maximum accuracy of $D \approx 3k$ dimensions of hypervectors. Due to the approximated computation of HDC, the proposed quantized HDC paradigm can easily exploit low-precision ADCs to reach the same accuracy. For example, using comparators, or 1-bit ADCs, HDC can reach a maximum accuracy with $D \approx 5k$ dimensions of hypervectors. By design

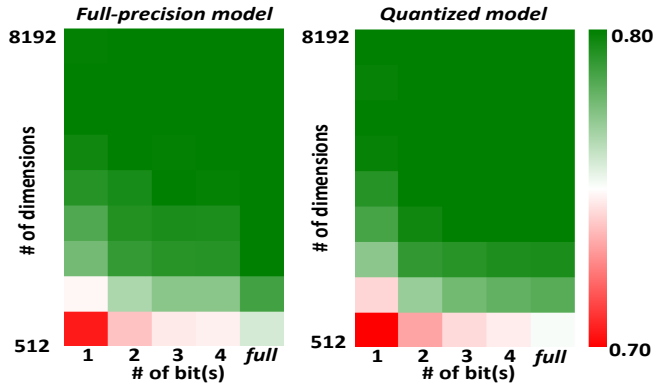


Fig. 5: In-sensor HDC by varying dimensions of the array and ADC bit-precisions.

TABLE III: Varying ADC bit-precision

| ADC bit-precision | $FP32$ | 4 bits | 3 bits | 2 bits | 1 bit |
|---|---|---|---|---|---|
| **Full precision Effective** $D$ | 2048 | 4096 | 4096 | 5120 | 6144 |
| **CiM Effective** $D$ | 3072 | 3072 | 3072 | 4096 | 5120 |
| **CiM Energy** $(nJ)$ | 38977 | 590 | 412 | 469 | 509 |
| **CiM Latency** $(\mu s)$ | 150 | 13.5 | 13.4 | 18.5 | 23.2 |
| **CiM Area** $(mm^2)$ | 278 | 13 | 9.5 | 9 | 8.6 |

TABLE IV: Cross-platform comparisons.

| | CiM | ASIC [37] | Jetson Orin | ZCU104 (DPU) |
|---|---|---|---|---|
| **Latency** $(ms)$ | 0.019 | 0.16 | 0.7 | 0.72 |
| **Energy** $(\mu J)$ | 0.3 | 0.3 | $4.2 \times 10^3$ | $3.2 \times 10^3$ |

space exploration based on Neurosim [] and SPICE for crossbar/peripherals and CAM respectively, the optimal hardware designs can be found via Table III. Finally, a cross-platform comparison is performed in Table IV, which illustrates the low-latency and low-power merit of CiM-based HDSense.

## V. Conclusions

Evaluation results for COSIME at the array level suggest that the proposed COSIME design achieves 333X and 90.5X latency and energy improvements, respectively, and realizes better classification accuracy when compared with an AM design implementing approximated CSS. C$^2$AM illustrate the effectiveness of ADP algorithm, offering on average 43X (Preisach), 132X (experimental linear region FeFET), and 19X (experimental saturation region FeFET) improvement over the $L_2$ distance in terms of mean square error (MSE) obtained via the existing programming scheme. In addition, for the first time, the $L_1$ distance along with fabricated FeFET is demonstrated in this work. Finally, an in-sensor HDC paradigm, called HDSense, is proposed for ultra-efficient edge inference by significantly reducing the analog-digital overhead. The CiM fabric shows 37X/38X latency and $1.4 \times 10^4$/$1.0 \times 10^4$ energy improvement over the eSoC and FPGA implementations respectively, and on average the same energy consumption with 8.4X speedup compared to state-of-the-art HDC ASIC accelerator.

***Research Impacts***

My undergraduate research in compute-in-memory co-design between device/circuit and application has led to publications at ICCAD'22 [38] and DATE'23 [39] (to appear), and several submitted papers at EDA / AI / Algorithm top venues, workshops, and journals including ISLPED'23 [40], MLSys-SNAP'23 [41], TC'23 [42], ICCAD'23 [43], in which I served as the leading author in ICCAD'22 [38], ISLPED'23 [40], and TC'23 [42] (co-first author), and main contributor in MLSys-SNAP'23 [41] and ICCAD'23 [43].

## REFERENCES


[1] T. Böscke, J. Müller, D. Bräuhaus, U. Schröder, and U. Böttger, "Ferroelectricity in hafnium oxide: Cmos compatible ferroelectric field effect transistors," in *2011 International electron devices meeting*. IEEE, 2011, pp. 24–5.

[2] H.-S. P. Wong, H.-Y. Lee, S. Yu, Y.-S. Chen, Y. Wu, P.-S. Chen, B. Lee, F. T. Chen, and M.-J. Tsai, "Metal–oxide rram," *Proceedings of the IEEE*, vol. 100, pp. 1951–1970, 2012.

[3] C. Zhuo, Z. Yang, K. Ni, M. Imani, Y. Luo, S. Wang, D. Zhang, and X. Yin, "Design of ultra-compact content addressable memory exploiting 1t-1mtj cell," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2022.

[4] M. Jerry, P.-Y. Chen, J. Zhang, P. Sharma, K. Ni, S. Yu, and S. Datta, "Ferroelectric fet analog synapse for acceleration of deep neural network training," in *2017 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2017, pp. 6–2.

[5] K. Ni, X. Yin, A. F. Laguna, S. Joshi, S. Dünkel, M. Trentzsch, J. Müller, S. Beyer, M. Niemier, X. S. Hu *et al.*, "Ferroelectric ternary content-addressable memory for one-shot learning," *Nature Electronics*, vol. 2, pp. 521–529, 2019.

[6] A. Kazemi, M. M. Sharifi, A. F. Laguna, F. Müller, X. Yin, T. Kämpfe, M. Niemier, and X. S. Hu, "Fefet multi-bit content-addressable memories for in-memory nearest neighbor search," *IEEE Transactions on Computers*, vol. 71, pp. 2565–2576, 2021.

[7] A. Kazemi, F. Müller, M. M. Sharifi, H. Errahmouni, G. Gerlach, T. Kämpfe, M. Imani, X. S. Hu, and M. Niemier, "Achieving software-equivalent accuracy for hyperdimensional computing with ferroelectric-based in-memory computing," *Scientific reports*, vol. 12, p. 19201, 2022.

[8] A. F. Laguna, M. Niemier, and X. S. Hu, "Design of hardware-friendly memory enhanced neural networks," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 1583–1586.

[9] X. S. Hu, M. Niemier, A. Kazemi, A. F. Laguna, K. Ni, R. Rajaei, M. M. Sharifi, and X. Yin, "In-memory computing with associative memories: a cross-layer perspective," in *2021 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2021, pp. 25–2.

[10] G. Karunaratne, M. Schmuck, M. Le Gallo, G. Cherubini, L. Benini, A. Sebastian, and A. Rahimi, "Robust high-dimensional memory-augmented neural networks," *Nature communications*, vol. 12, p. 2468, 2021.

[11] M. Li, A. F. Laguna, D. Reis, X. Yin, M. Niemier, and X. S. Hu, "imars: an in-memory-computing architecture for recommendation systems," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022, pp. 463–468.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[13] G. Karunaratne, M. Le Gallo, G. Cherubini, L. Benini, A. Rahimi, and A. Sebastian, "In-memory hyperdimensional computing," *Nature Electronics*, vol. 3, pp. 327–337, 2020.

[14] L. Paulevé, H. Jégou, and L. Amsaleg, "Locality sensitive hashing: A comparison of hash function types and querying mechanisms," *Pattern recognition letters*, vol. 31, pp. 1348–1358, 2010.

[15] G. Datta, S. Kundu, Z. Yin, R. T. Lakkireddy, J. Mathai, A. P. Jacob, P. A. Beerel, and A. R. Jaiswal, "A processing-in-pixel-in-memory paradigm for resource-constrained tinyml applications," *Scientific Reports*, vol. 12, p. 14396, 2022.

[16] S. Angizi *et al.*, "Pisa: A non-volatile processing-in-sensor accelerator for edge image processing," *IEEE Transactions on emerging topics in computing*, 2022.

[17] M. Hersche, G. Karunaratne, G. Cherubini, L. Benini, A. Sebastian, and A. Rahimi, "Constrained few-shot class-incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9057–9067.

[18] A. Hernandez-Cane, N. Matsumoto, E. Ping, and M. Imani, "Onlinehd: Robust, efficient, and single-pass online learning using hyperdimensional system," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2021, pp. 56–61.

[19] A. Shafiee, A. Nag, N. Muralimanohar, R. Balasubramonian, J. P. Strachan, M. Hu, R. S. Williams, and V. Srikumar, "Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," *ACM SIGARCH Computer Architecture News*, vol. 44, pp. 14–26, 2016.

[20] T. Soliman, F. Müller, T. Kirchner, T. Hoffmann, H. Ganem, E. Karimov, T. Ali, M. Lederer, C. Sudarshan, T. Kämpfe *et al.*, "Ultra-low power flexible precision fefet based analog in-memory computing," in *2020 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2020, pp. 29–2.

[21] D. Saito, T. Kobayashi, H. Koga, N. Ronchi, K. Banerjee, Y. Shuto, J. Okuno, K. Konishi, L. Di Piazza, A. Mallik *et al.*, "Analog in-memory computing in fefet-based 1t1r array for edge ai applications," in *2021 Symposium on VLSI Technology*. IEEE, 2021, pp. 1–2.

[22] B. A. Minch, "Translinear circuits," Tech. Rep., 2009. [Online]. Available: http://madvlsi.olin.edu/bminch/talks/090402_atact.pdf

[23] A. G. Andreou and K. A. Boahen, "Translinear circuits in subthreshold mos," *Analog Integrated Circuits and Signal Processing*, vol. 9, pp. 141–166, 1996.

[24] M. Imani, A. Rahimi, D. Kong, T. Rosing, and J. M. Rabaey, "Exploring hyperdimensional associative memory," in *2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 2017, pp. 445–456.

[25] J. A. Starzyk and X. Fang, "Cmos current mode winner-take-all circuit with both excitatory and inhibitory feedback," *ELECTRONICS LETTERS*, 1993.

[26] J. Lazzaro, S. Ryckebush, M. Mahowald, and C. A. Mead, "Winner-take-all networks of o(n) complexity," Cal-tech, Tech. Rep., 1988. [Online]. Available: https://john-lazzaro.github.io/biblio/wta-tech.pdf

[27] A. Kazemi, S. Sahay, A. Saxena, M. M. Sharifi, M. Niemier, and X. S. Hu, "A flash-based multi-bit content-addressable memory with euclidean squared distance," in *2021 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*. IEEE, 2021, pp. 1–6.

[28] P.-Y. Chen, X. Peng, and S. Yu, "Neurosim+: An integrated device-to-algorithm framework for benchmarking synaptic devices and array architectures," in *2017 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2017, pp. 6–1.

[29] K. Ni, M. Jerry, J. A. Smith, and S. Datta, "A circuit compatible accurate compact model for ferroelectric-fets," in *2018 IEEE symposium on VLSI technology*. IEEE, 2018, pp. 131–132.

[30] N. H. Weste and D. Harris, *CMOS VLSI design: a circuits and systems perspective*. Pearson Education India, 2015.

[31] A. Ortiz-Conde, F. G. Sánchez, J. J. Liou, A. Cerdeira, M. Estrada, and Y. Yue, "A review of recent mosfet threshold voltage extraction methods," *Microelectronics reliability*, vol. 42, pp. 583–596, 2002.

[32] K. Ni, A. Gupta, O. Prakash, S. Thomann, X. S. Hu, and H. Amrouch, "Impact of extrinsic variation sources on the device-to-device variation in ferroelectric fet," in *2020 IEEE International Reliability Physics Symposium (IRPS)*. IEEE, 2020, pp. 1–5.

[33] S. Yu, X. Guan, and H.-S. P. Wong, "On the switching parameter variation of metal oxide rram—part ii: Model corroboration and device design strategy," *IEEE Transactions on Electron Devices*, vol. 59, pp. 1183–1188, 2012.

[34] A. Ranganathan, "The levenberg-marquardt algorithm," *Tutoral on LM algorithm*, vol. 11, pp. 101–110, 2004.

[35] H. Mulaosmanovic, S. Dünkel, J. Müller, M. Trentzsch, S. Beyer, E. T. Breyer, T. Mikolajick, and S. Slesazeck, "Impact of read operation on the performance of hfo 2-based ferroelectric fets," *IEEE Electron Device Letters*, vol. 41, pp. 1420–1423, 2020.

[36] Y. Long, T. Na, P. Rastogi, K. Rao, A. I. Khan, S. Yalamanchili, and S. Mukhopadhyay, "A ferroelectric fet based power-efficient architecture for data-intensive computing," in *Proceedings of the International Conference on Computer-Aided Design*, 2018, pp. 1–8.

[37] B. Khaleghi, J. Kang, H. Xu, J. Morris, and T. Rosing, "Generic: highly efficient learning engine on edge using hyperdimensional computing," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022, pp. 1117–1122.

[38] **Che-Kai Liu**, H. Chen, M. Imani, K. Ni, A. Kazemi, A. F. Laguna, M. Niemier, X. S. Hu, L. Zhao, C. Zhuo, and X. Yin, "Cosime: Fefet based associative memory for in-memory cosine similarity search," in *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 2022, pp. 1–9.

[39] H. E. Barkam, S. Yun, P. R. Genssler, Z. Zou, **Che-Kai Liu**, H. Amrouch, and M. Imani, "Hdgim: Hyperdimensional genome sequence matching on unreliable highly-scaled fefet," in *Proceedings of the IEEE/ACM Design Automation and Test in Europe.* IEEE/ACM, 2023.

[40] **Che-Kai Liu**, H. Barkam, Z. Zou, H. Chen, S. Yun, X. Yin, and M. Imani, "Hdsense:," in *Submission to Proceedings of the IEEE/ACM International Symposium on Low Power Electronic Design*. IEEE/ACM, 2023.

[41] Z. Wan, **Che-Kai Liu**, H. Yang, C. Li, H. You, Y. Fu, C. Wan, T. Krishna, Y. C. Lin, and A. Raychowdhury, "Towards cognitive ai system: A survey and prospective on neuro-symbolic ai," in *Workshop on Systems for Next-Gen AI Paradigms, Sixth Conference on Machine Learning and Systems*, 2023.

[42] M. Li*, **Che-Kai Liu***, Z. Jiang, K. Ni, and X. S. Hu, "Fusion of expression values and protein interaction information using multi-objective optimization for improving gene clustering," **** equal contributions with alphabetical ordering.*** *Submission to Transactions on Computers*, 2023.

[43] S. Shou, **Che-Kai Liu** *et al.*, "An ultra-dense 2fefet-1t multi-bit content addressable memory for brain-inspired associative search," in *Submission to Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*. IEEE/ACM, 2023.