System Energy Efficiency Lab

seelab.ucsd.edu

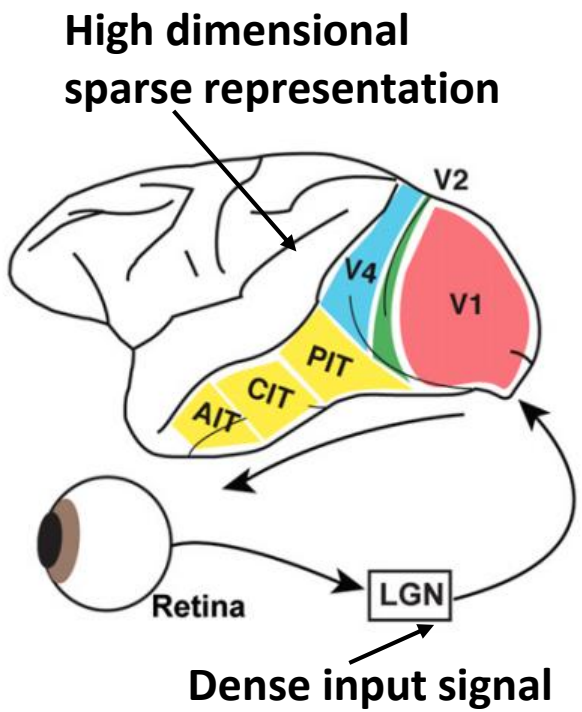
Hyperdimensional Computing & Applications

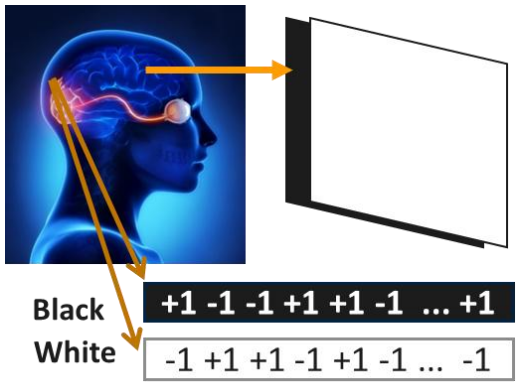
Prof. Tajana Simunic Rosing

& amazing PhD students, postdocs and staff @ UCSD

Brain-inspired Hyperdimensional Computing

Dense sensory input is mapped to
high-dimensional sparse representation
on which brain operates [Babadi, Sompolinsky 2014]





Hyperdimensional (HD) computing:

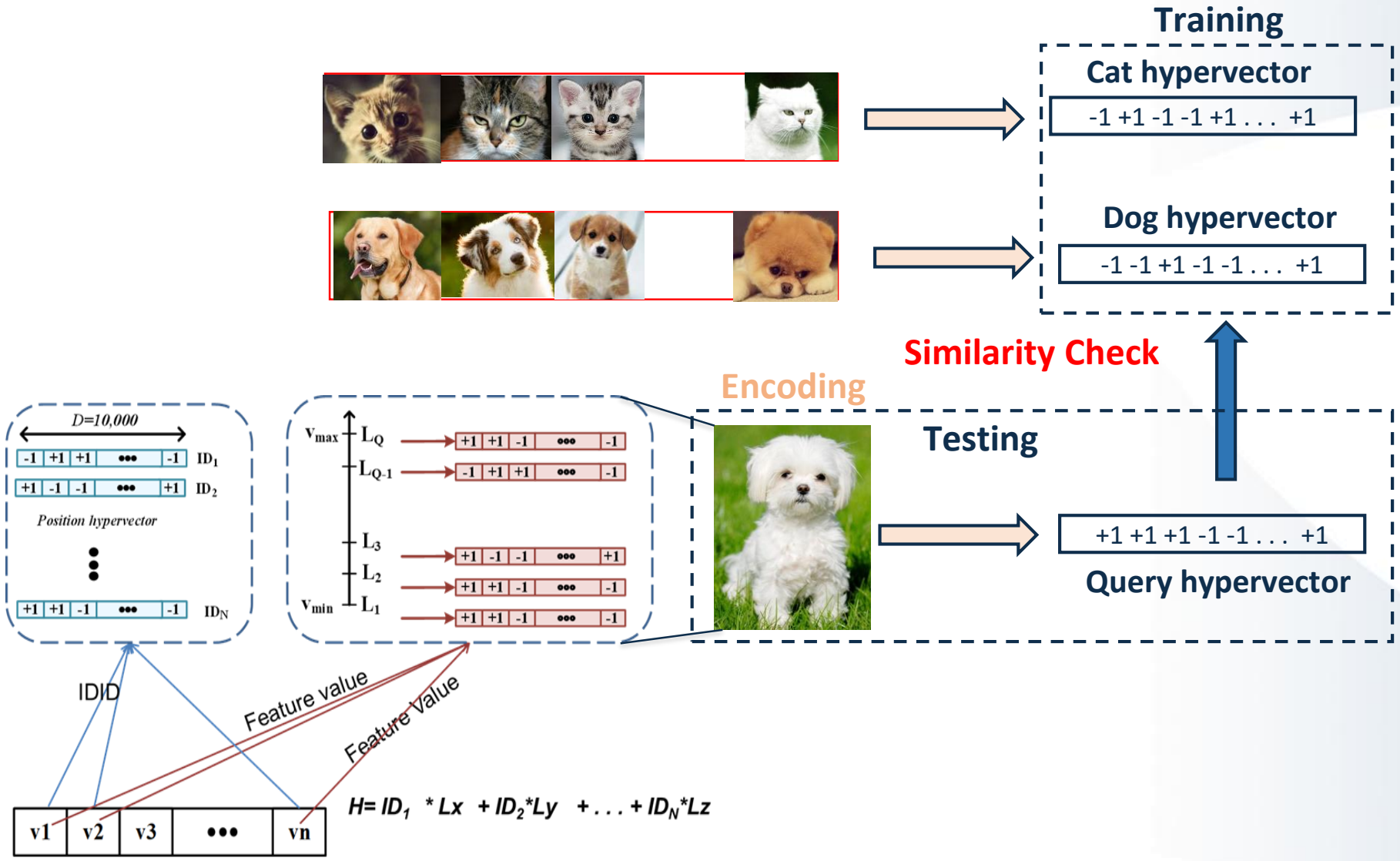
- Encodes data into hypervectors
- Leverages full algebra and works on well-defined set of operations that are easy to parallelize
- Fast single-pass training → online learning
- Supports symbolic reasoning & explainability
- Energy-efficient & robust to noise

Image Classification	ImageNet 150GB
Lidar Segementation	SemanticKitti 80GB
Genomics	Human, COVID Microbiome, ~100TB
Database MassSpec	Proteomics & Metabolomics ~150TB
Virtual Drug Screening	PubChem LIT-PCBA 410k molecules
Clustering	Proteomics ~150TB
Graphs	PubMed 44k edges TUDataset Millions of graphs
Recommendation Systems	Amazon, Yelp ~1TB
Federated & Lifelong Learning	CIFAR100

. . .

Sources: DiCarlo et al "How does the brain solve visual object recognition?." Neuron'12; Kanerva, "HD Computing: An Introduction," Cog.Comp'09

HD Computing Classification: Encoding, Training & Inference



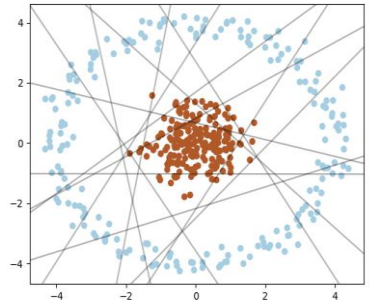
Theoretical Results on Hypervector Dimensionality

Random Projection Encoding:

Encoding dimension depends on desired level of sparsity

To ensure a k-sparse representation which separates classes with probability 1-α

$$d \geq k \frac{\log \alpha}{\log 1-\theta} \text{ where } \theta \approx O\left(1 - \frac{16}{k^2}\right)^n \text{ when } k \text{ is} \geq 10$$



ID-Level Encoding:

Preserves the geometry of data up to additive distortion

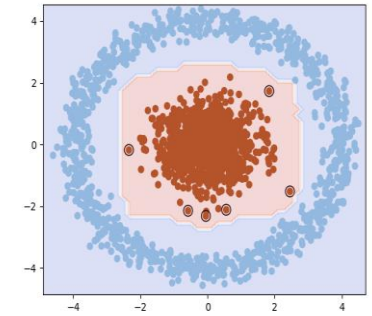
To ensure separability is preserved

$$d > \frac{4n^2}{\delta^2} \log \frac{2(nm)^2}{\alpha}$$

n: dimensionality of the original data

m: quantization levels

δ: distance between classes or centroids in low dimensionality

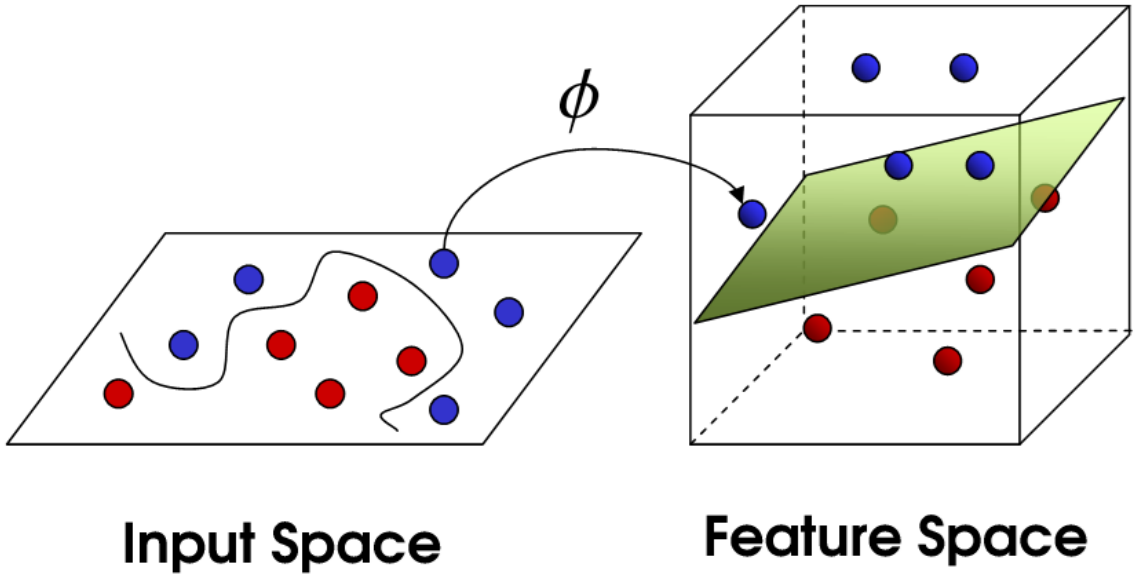


Effects of Noise:

For N symbols drawn from alphabet of size M with noise bound ω: $d_{hv} = O(\omega N \log M)$

A.Thomas, S. Dasgupta, T. Rosing, "A Theoretical Perspective on Hyperdimensional Computing," JAIR'21.

Kernel methods and Non-Linear Encoding for HDC



- **Inner-products in HD space should be reflective of some salient notion of similarity on ambient space.**
- **Idea: Construct HD encoding functions using suitable kernel functions.**

Quanling Zhao, Anthony Thomas, Ari Brin, Xiaofan Yu, Tajana Rosing , "Bridging the Gap between Hyperdimensional Computing and Kernel Methods via the Nyström Method" - AAAI 2025

Random Fourier Features

- Commonly referred as “non-linear” encoding in HDC
- Capable of modeling shift-invariant kernels in HDC (e.g. the Gaussian kernel, polynomial kernel)

Why Nyström method ?

- Random Fourier Features only work with shift-invariant kernels on a Euclidean space, which many useful kernels do not satisfy (e.g kernels on graphs and strings)

Contribution: NysHD

- Directly generate encodings from a desired notion of similarity - inspired by kernel approximation
- Applicable to a wider range of data type
- Still retains efficiency and robustness of HDC

Bridging the Gap between Hyperdimensional Computing and Kernel Methods via the Nyström Method

Experimental results on TUD graph datasets [ICML'20]

Contains millions of graphs with 100s of nodes, 1000s of edges, from bio, social networks & computer vision

	Accuracy %	NCI1	ENZYMES	D&D	BZR	MUTAG	COX2	NCI109	Mutagen
DNN	DGCNN [AAAI'18]	70.2	36.9	74.5	81.5	82.9	78.3	71.1	75.0
DNN	GCN [ICLR'17,Arxiv'19]	79.9	60.7	74.8	84.2	85.5	83.1	80.2	79.9
DNN	GIN [ICLR'2019]	73.4	28.8	67.5	76.4	76.8	78.1	70.8	78.0
DNN	GIUNet [Expert Syst.'24]	72.4	29.9	63.4	78.3	85.0	77.4	68.8	76.5
HDC	GraphHD [DATE'22]	60.0	23.2	67.6	74.9	85.3	81.9	59.9	59.8
HDC	Our NysHD [AAAI'25]	73.8	61.3	76.2	82.0	85.5	74.6	71.8	75.2

The best accuracy

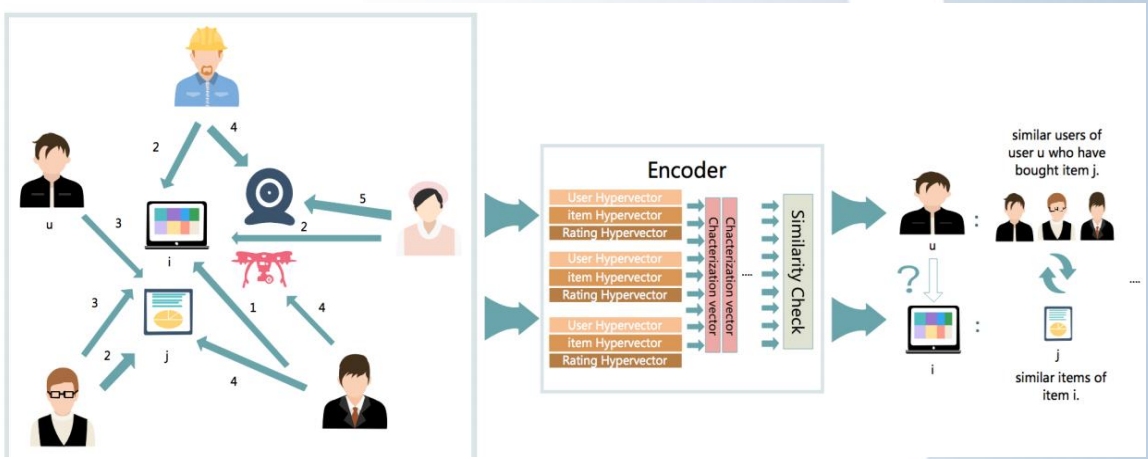
Second best accuracy

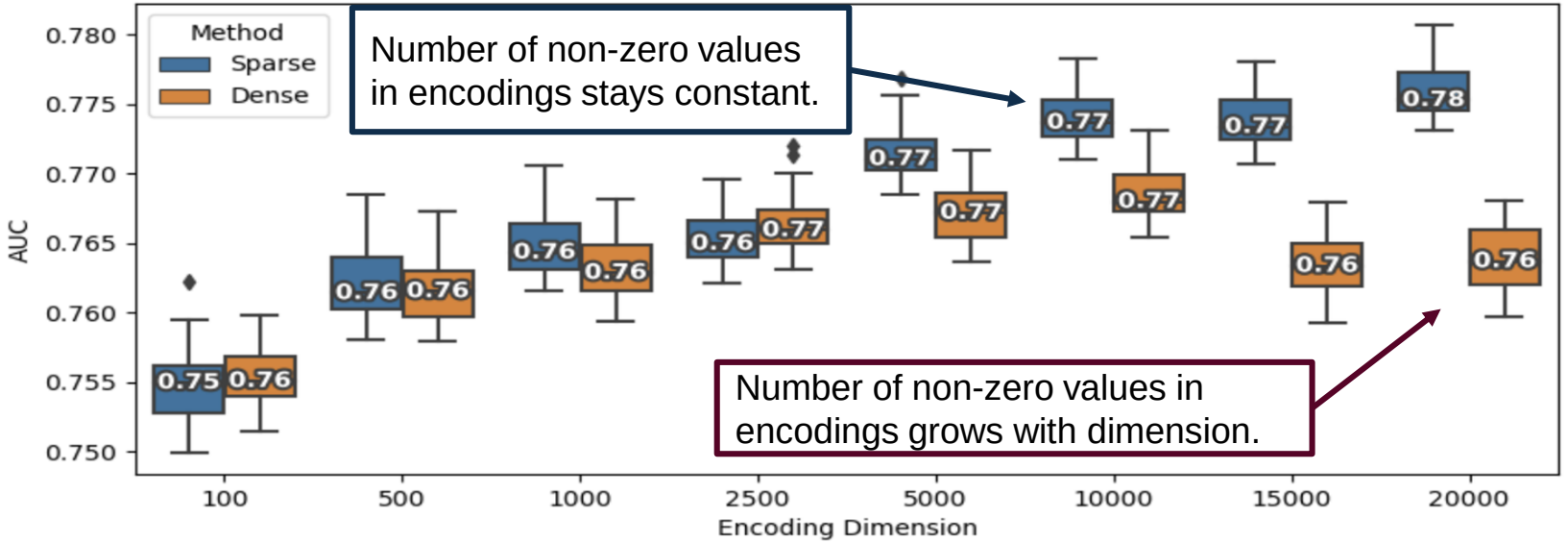
On average 11% better accuracy than previous HDC methods & 52% faster than SOTA GCN

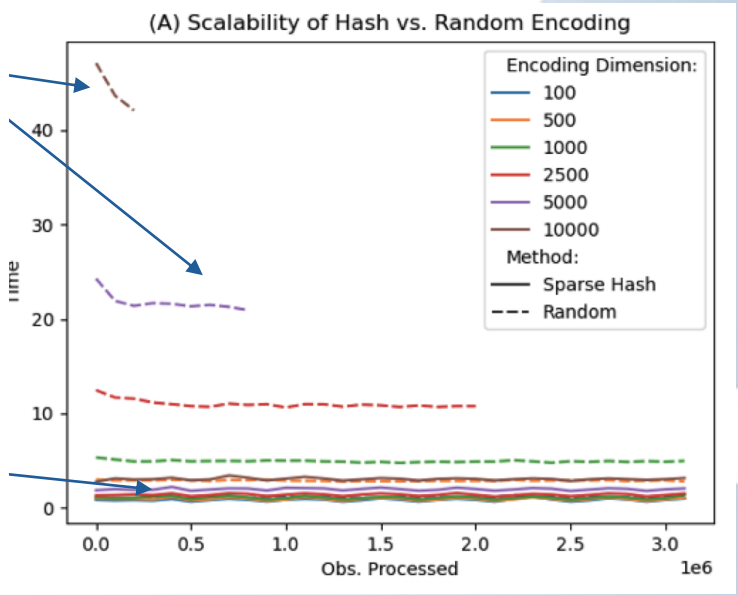
Quanling Zhao, Anthony Thomas, Ari Brin, Xiaofan Yu, Tajana Rosing , "Bridging the Gap between Hyperdimensional Computing and Kernel Methods via the Nyström Method" - AAAI Conference on Artificial Intelligence (AAAI), 2025

HyperRec: HD Computing Recommendation Systems at Scale

- HD Recommendation systems identify similar users & items by using their HD characterization vectors
  - Problem: generate HD encoding for tens of millions of symbols
  - Solution: Instead of storing codewords, we construct them “on-the-fly” by evaluating a handful of hash-functions
- Successfully tested on 1TB of data from Amazon & Yelp
  - Dataset has categorical features defined over a very large alphabet with hundreds of millions of symbols

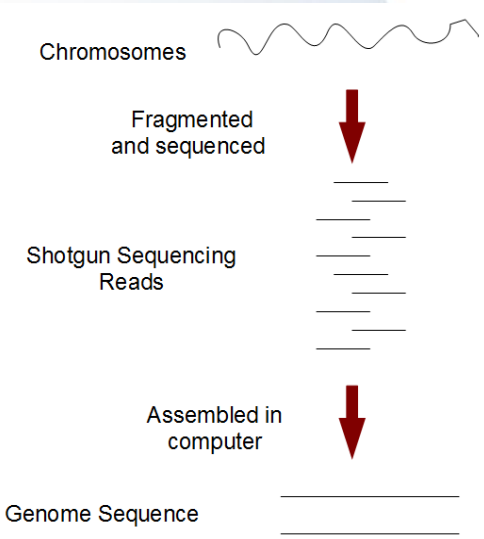




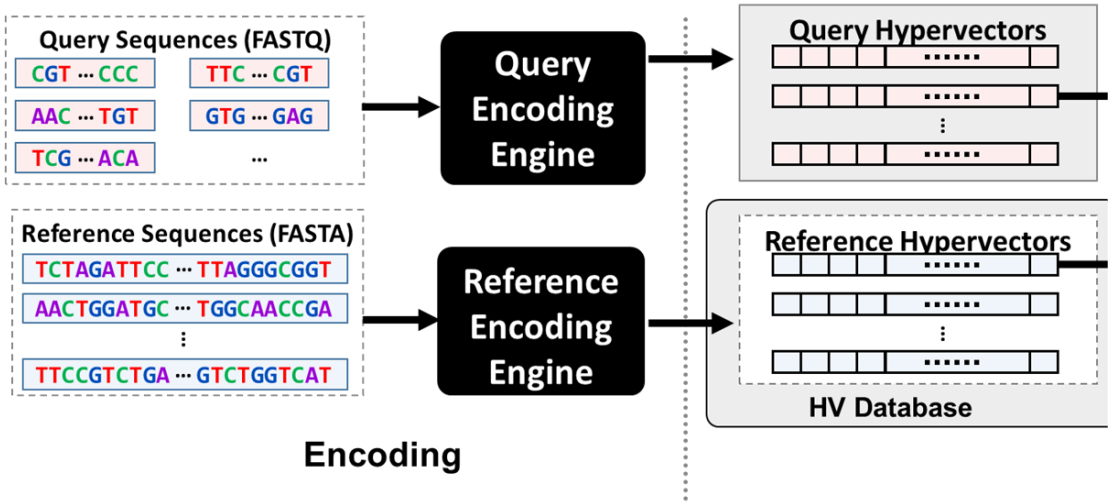


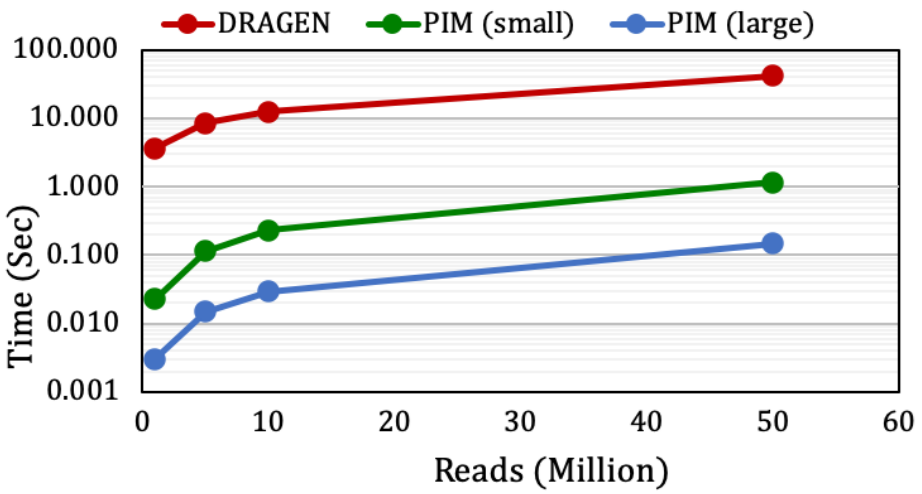
Y. Guo, S. Gupta, M. Imani, Y. Kim, J. Morris, T. Rosing, "HyperRec: Efficient Recommender Systems with Hyperdimensional Computing," ASPDAC 2021 & updated results on ArXiv

GenieHD & RAPID Alignment



- Genome Identity Extractor using HyperDimensional Computing
  - Encode DNA into hypervectors
  - Combine ~1,000 segments of the reference DNA into a hypervector
  - Find the existence of DNA patterns using similarity computation
- RAPID – short sequence alignment accelerated in memory
  - Large PIM can fit human genome and is **1,900x faster** vs. Minimap on CPU, 253x faster vs. DRAGEN FPGA





Y. Kim, M. Imani, N. Moshiri, T. Rosing, “GenieHD: Efficient DNA Pattern Matching Accelerator Using HD Computing”, Best Paper at DATE’20

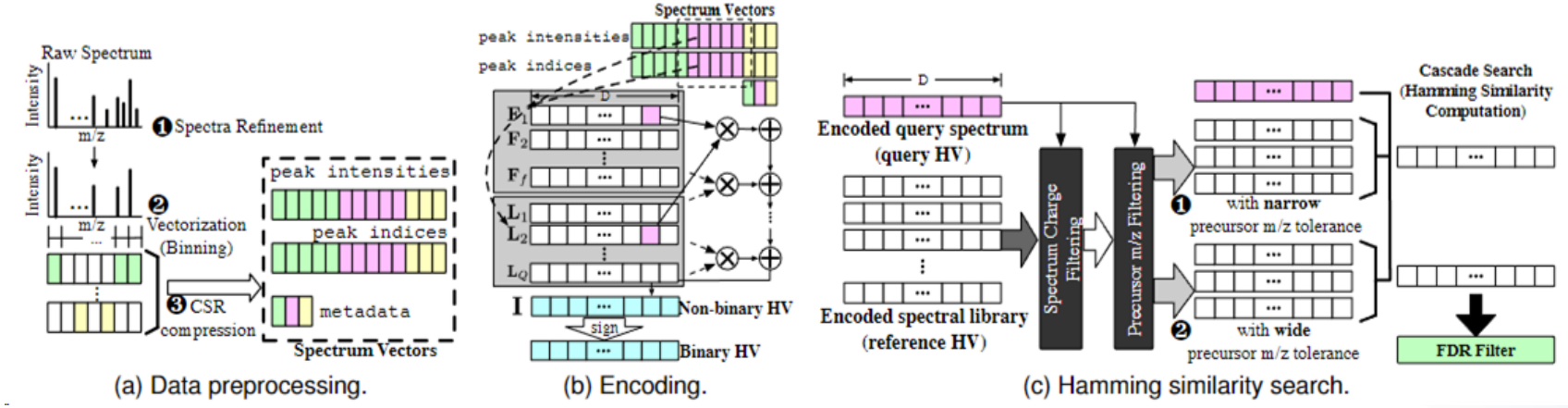
Gupta, S, T. Rosing, et al. "RAPID: A ReRAM processing in-memory architecture for DNA sequence alignment." ISLPED’19.

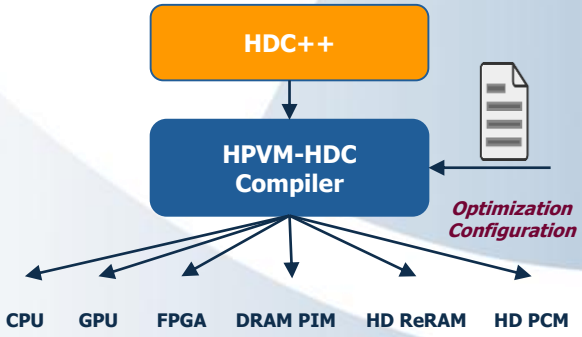
Xu W, Gupta S, Moshiri N, Rosing T. “RAPIDx: High-performance ReRAM Processing in-Memory Accelerator for Sequence Alignment” tbd TCAD’23

Accelerating Mass Spec Database Search in Memory & Storage

PRISM PIs: Tajana Rosing, Vikram Adve, Jason Cong, Sang-Woo Jun, Eric Pop, Mingu Kang, Philip Wong, Shimeng Yu, Suman Datta
Other PIs: Niema Moshiri@UCSD, Pieter Dorrestein Pharmacy@UCSD, Rob Knight, Medicine@UCSD, Sourav Dutta@UTD, Wout Bittremieux@UAntwerp
JUMP 2.0 Centers: Asif Khan, SUPREME; Shimeng Yu, Suman Datta, CHIMES
Industry: Micron, Samsung, IBM, TSMC

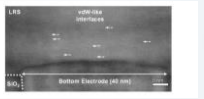
- **Motivation:**
  - Proteomics and metabolomics relay on mass spectrometry as a key tool in design of precision drugs
  - CPU/GPU state of the art tool, AnnSoLo [Journal Proteome Research'19] is very slow due to memory boundedness → gating development of future drugs
    - Novel memory and storage devices provide a capability for in/near memory computation, but have higher bit error rates & do not have compiler support
- **Goal:** understand & quantify what procesisng in/near novel Theme 3 & SUPREME memories can offer to big data analysis on real applications
- **Technical approach:**
  - Leverage resiliency of hyperdimensional computing to benefit from higher capacity along with high parallelism in and near memory/storage compute
  - Automate compiling of HDC-based workloads onto multiple types of accelerators: CPU, GPU, FPGA, and PIM
    - HPVM produces 15x less lines of code that runs >3x faster due to optimizations
- **Grand challenge application**: Open modification search for mass spectrometry based on HDC [BioOxford'23] with comparable accuracy to AnnSoLo [JPR'19]
  - **Smaller dataset**:
    - Reference: Yeast and Human HCD spectral library (1,188,168 spectra; 1GB in HDC representation); Query: iPRG2012 (17,993 spectra, each query is 8,192 bits)
  - **Larger dataset**: 154TB from MaSSiVE database at UCSD
    - Reference: 1,000,000,000 spectra (8TB in HDC), Query: 15,000 spectra (typical for a single instrument run)

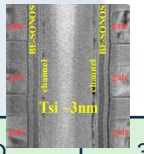


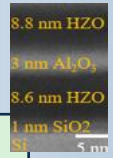


Accelerating Mass Spec Database Search in Memory & Storage

PRISM PIs: Tajana Rosing, Vikram Adve, Jason Cong, Sang-Woo Jun, Eric Pop, Mingu Kang, Philip Wong, Shimeng Yu, Suman Datta
Other PIs: Niema Moshiri@UCSD, Pieter Dorrestein Pharmacy@UCSD, Rob Knight, Medicine@UCSD, Sourav Dutta@UTD, Wout Bittremieux@UAntwerp
JUMP 2.0 Centers: Asif Khan, SUPREME; Shimeng Yu, Suman Datta, CHIMES
Industry: Micron, Samsung, IBM, TSMC





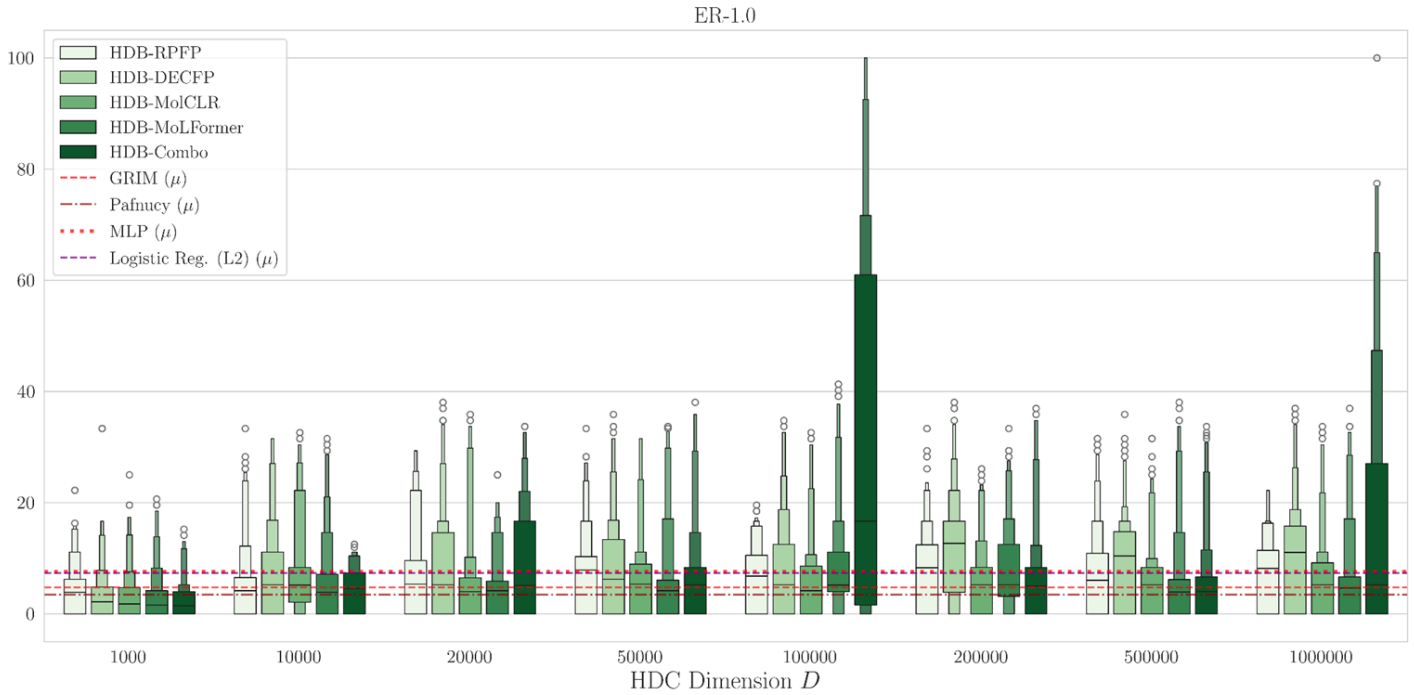


Large Data Comparison	GPU (not HDC) [JPR'19]	GPU [Bio'23]	FPGA [BioSys24]	DRAM [TCAD'24]	MLC ReRAM [DAC'24]	MLC ReRAM [DAC'24]	MLC PCM [JXCDC'24]	3D NAND [MEMSYS'23]	3D FeNAND [submitted]
Algorithm	ANN-SoLo	HOMS-TC	RapidOMS	HyperOMS	HyperOMS	HyperOMS	HyperOMS	HyperOMS	HyperOMS
Technology Node(nm)	H100 w 8TB SSD (5nm)	H100 (5nm)	VP1902 with 1TB SSD (7nm)	22nm DDR4 28nm Compute	130nm RRAM	40nm RRAM	40nm PCM	NAND: 14nm ASIC: 7nm FinFET	FeNAND:14nm ASIC: 7nm FinFET
Speed of search	1x	24x	97x	149x	259x	823x	5,303x	175x	3,779x
Energy Efficiency	1x	34x	272x	955x	48,462x	726,941x	817,808x	65,424x	261,698x
	Measured	Demo'23	Demo'23	Poster'23		Demo'23	Demo'24	Poster'23	Poster'24 Measured & simulated

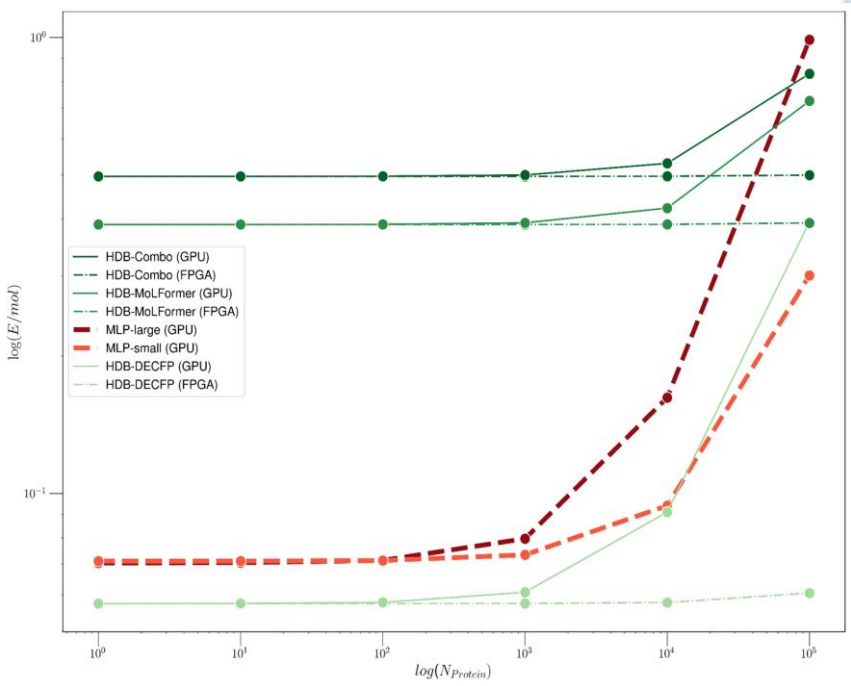
- Key assumptions:
  - Compiler-driven mapping
  - All designs have the reference database in NVM (SSD or PIM). All accelerators have the same capacity and area at the same technology node.
  - We assumed maximum 1.67 Tb per package, with 64 bit interface per package. Query is broadcast to PIM on 200 MHz bus with sequential result readout for index & score.
  - Accelerators are characterized at the device level via measurement, architecture & circuits are simulated; both periphery and memory are accounted for. Results match SOTA accuracy.
- Key takeaways:
  - Processing in/near memory/storage results are up to 221x faster vs GPUs with big data due to decrease in the need for data movement
    - 800 GPUs would be needed to be able to fully fit the reference dataset in GPU memory for SOTA [JPR'19], and 100 GPUs for HDC version [Bio'23]
  - MLC PCM & ReRAM again have the largest improvement in energy efficiency; challenge is scaling and accuracy for more error sensitive workloads;
    - ReRAM used decimals with 64 parallel banks, while PCM used binary with data packing (3x faster) and 128 banks (2x faster) → speed and efficiency are comparable
  - 3D FeNAND offers impressive scalability and significantly higher speed and energy efficiency due to packing more strings in parallel, and not having as long strings as 3D NAND
    - Both suffer from having to read data out sequentially as compared to PCM/ReRAM & would not do as well on applications with many writes

Accelerated Drug Library Screening with HDBind [JSR'24]

- Modern drug screening pipelines combine physical simulation, domain expertise, and machine learning
- HDBind presents molecular encoding methods that can be combined with hardware efficient HDC inference
  - Previous work only considers SMILES string representation, HDBind considers instead the molecular graph information and its combination with SOA LLM-based features
- HDBind is 200x faster on FPGA than SOTA on GPU, has better scaling & excellent accuracy when running on PubChem BioAssay LIT-PCBA dataset which has 410k characterized molecules



HDBind models compare favorably to SOA physics-based scoring (GRIM), deep learning (Pafnucy), and traditional ML methods (MLP & Logistic Regression) using the ROC-enrichment factor metric on the LIT-PCBA dataset of 15 protein targets with experimentally verified activities

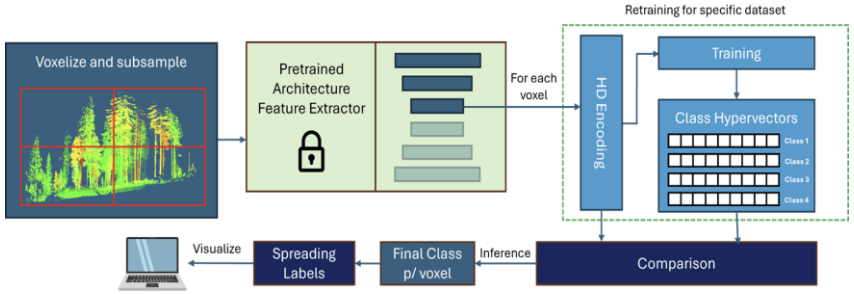


FPGA-based inference enables efficient scaling to repeated screening across the range of HDBind encoding methods

Jones, D., Rosing, T., et. al. (2024). HDBind: encoding of molecular structure with Hyperdimensional binary representations. *Scientific Reports 2024*

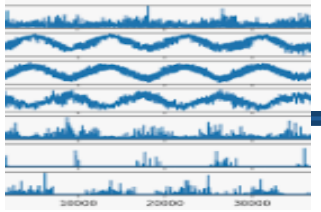
HDC, LLMs & Multimodal Systems

Collaboration with Intel, IBM, TSMC, GlobalFoundries & JUMP 2.0 Centers



LiDAR features are extracted using pretrained, frozen feature extractor and trained with HDC classifier with the goal of implementing on sensor training.

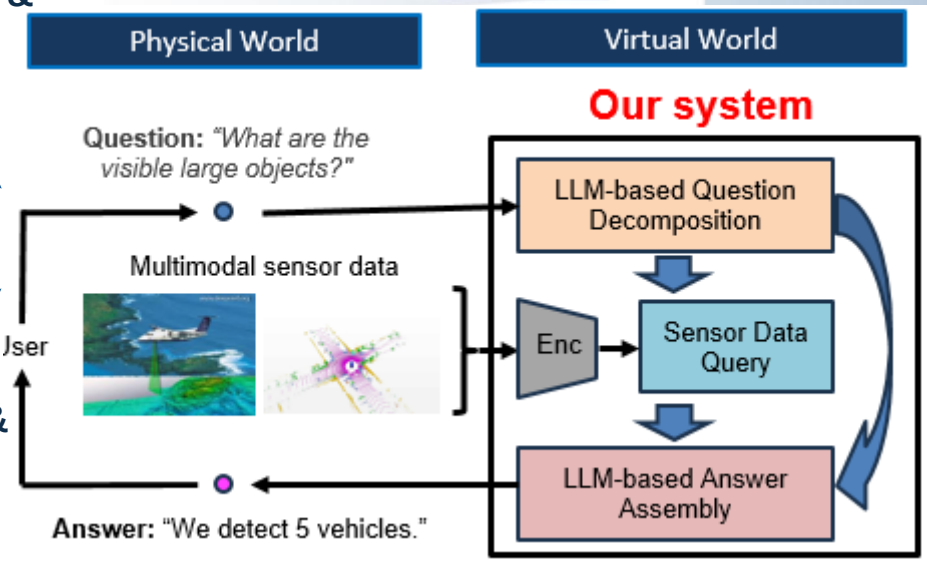
SensorChat combines LLMs with multimodal learning systems to enable natural language interface to sensors at the ***edge***. It obtains higher accuracy vs. SOTA QA systems, with an low end-to-end latency vs. SOTA multimodal LLMs.

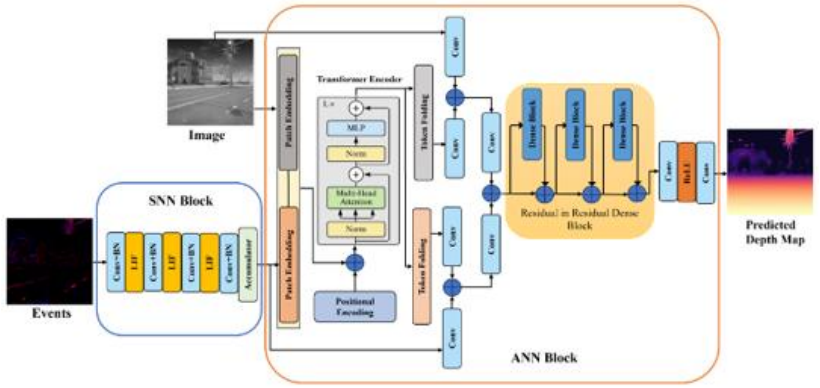


Time series sensor data

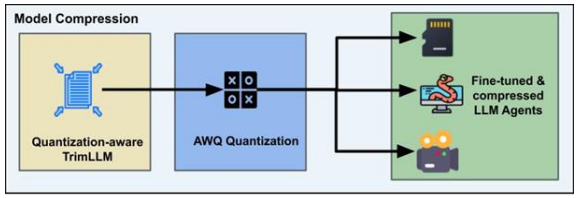
LiDAR & Radar

RGB & DVS





SNN-HDC, HDnn, SNN-ANN Hybrid Networks performed using TransPIM in memory accelerator to enable fast recognition



TinyAgent for edge LLMs uses layer dropping and post-training quantization in an activation-aware way resulting in smaller memory footprint and faster execution [ICML"24]

HDnn: Large Image Classification
Jointly with TSMC, IBM and Intel

- Combine HD with a feature extractor derived from the CNN
  - Prune and cut many of CNN layers; add HD as the last layer

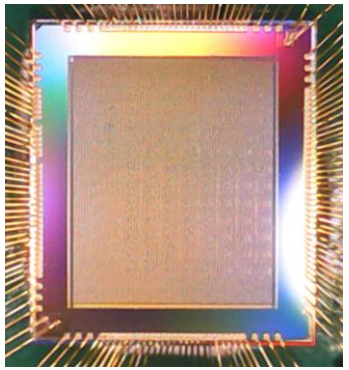
Accuracy comparison of HD, CNN, and HDnn.

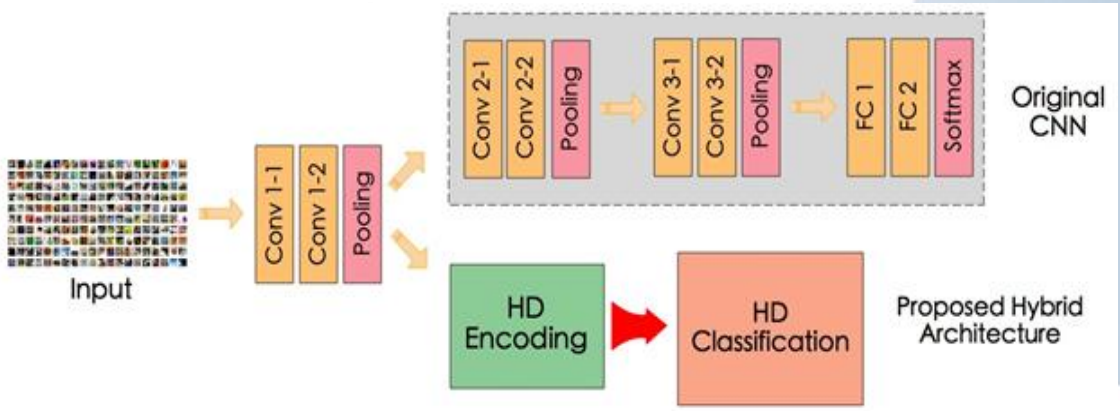
Model↓ Dataset→	MNIST	CIFAR-10	CIFAR-100	Flowers
HD (RP) [9]	94%	26.9%	9%	19.6%
HD (non-linear) [10]	97%	45.5%	27.7%	31.5%
StocHD [11]	98%	N/A	N/A	N/A

Dataset	Baseline VGG16	binary HDnn
22 super categories of ImageNet	72.88%	75.52%
22 vehicle subcategories of ImageNet	79.91%	79.91%

HDnn runs ImageNet & CIFAR100 as accurately as CNNs but more efficiently

- Three chips with TSMC
  - HDnn 40nm ASIC [ESSRC'24]
    - Few-shot learning
  - HD MLC ReRAM [DAC'24]
    - Collaboration w PRISM Philip Wong
  - HDnn 40nm ReRAM [VLSI'25]
    - Key innovation: analog HDC-PIM; testing in progress





[9] M. Imani, J. Morris, *et al.*, "Bric: Locality-based encoding for energy-efficient brain-inspired hyperdimensional computing," in *56th Annual Design Automation Conference*, pp. 1–6, 2019.

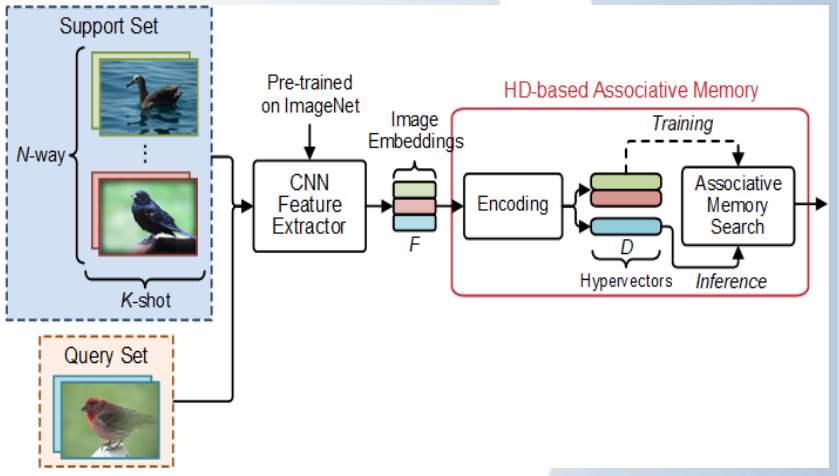
[10] Z. Zou, Y. Kim, M. H. Najafi, and M. Imani, "Manihd: Efficient hyper-dimensional learning using manifold trainable encoder," in *2021 Design, Automation Test in Europe Conference Exhibition (DATE)*, pp. 850–855, 2021.

[11] P. Poduval, Z. Zou, H. Najafi, H. Homayoun, and M. Imani, "Stochd: Stochastic hyperdimensional system for efficient and robust learning from raw data," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*, pp. 1195–1200, 2021.

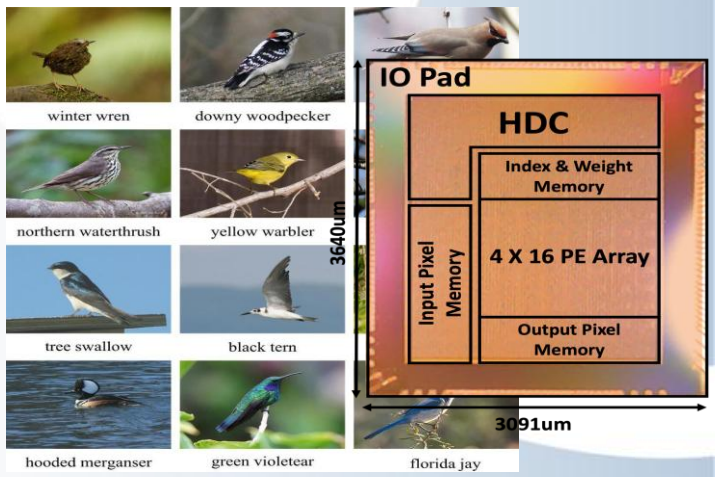
[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

HDC Few Shot Learning [DATE'23, ESSERC'24]
Jointly with TSMC, IBM & Intel

- **Few-shot learning (FSL):** classifying new data with only a few training samples
  - **N**-way, **K**-shot: **N** new classes & **K** training samples for each class
  - CNN feature extractor with HDC-based classifier
- **Experimental Setup**:
  - FSL setting: 10-way, 5-shot
  - Front-end CNN model: ResNet-18 with 512 features
  - Back-end HD classifier: D=2,048; binary
- **Datasets**: CIFAR-100 and Caltech-UCSD Birds 200
  - CIFAR-100 w 100 classes, Caltech-UCSD Birds with 200 classes
- **Results**: **FSL-HD is 20x faster** with comparable accuracy to SOTA; 4.9% higher accuracy compared to kNN-based design for FSL tasks [T-ED'21]



BER	0	2%	6.5%	11.0%	15%
CIFAR-100	61.4%	61.2%	61.0%	59.2%	54.5%
CU-Birds 200	85.1%	84.8%	84.3%	82.6%	77.9%



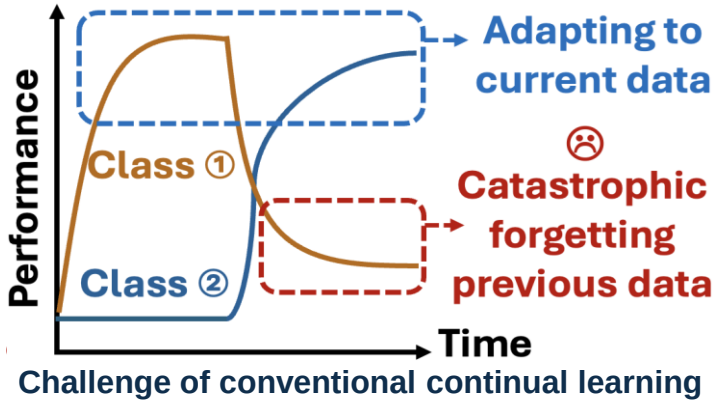
Amazing resilience of HD -> accuracy stable even in very high BER regimes!

Weihong Xu, Jaeyoung Kang, and Tajana Rosing, "FSL-HD: Accelerating Few-Shot Learning using Hyperdimensional Computing." DATE 2023.

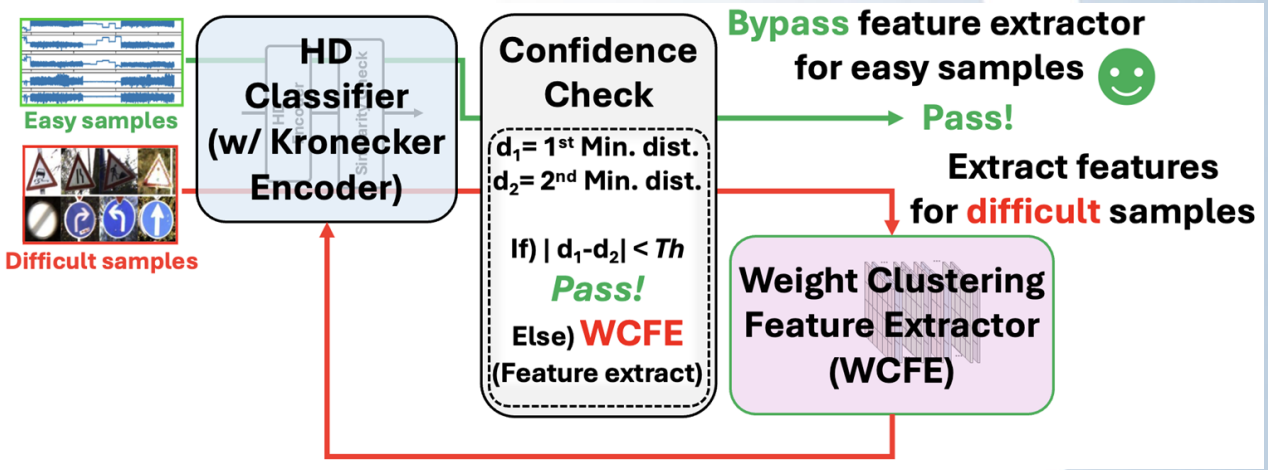
H. Yang*, C. E. Song*, W. Xu, B. Khaleghi, U. Mallappa, M. Shah, K. Fan, M. Kang, and T. Rosing ***"FSL-HDnn: A 5.7 TOPS/W End-to-end Few-shot Learning Classifier Accelerator with Feature Extraction and Hyperdimensional Computing"***, IEEE European Solid-State Electronics Research Conference (**ESSERC**), 2024

Clo-HDnn 40nm ASIC: Continual On Device Learning

Jointly with TSMC [VLSI'25]



Challenge of conventional continual learning



Clo-HDnn Processing Flow with Dual Mode

Challenges of conventional continual on device learning

- Forgetting the previous data due to limited memory on the edge device
- Requires less energy consumption for continual learning

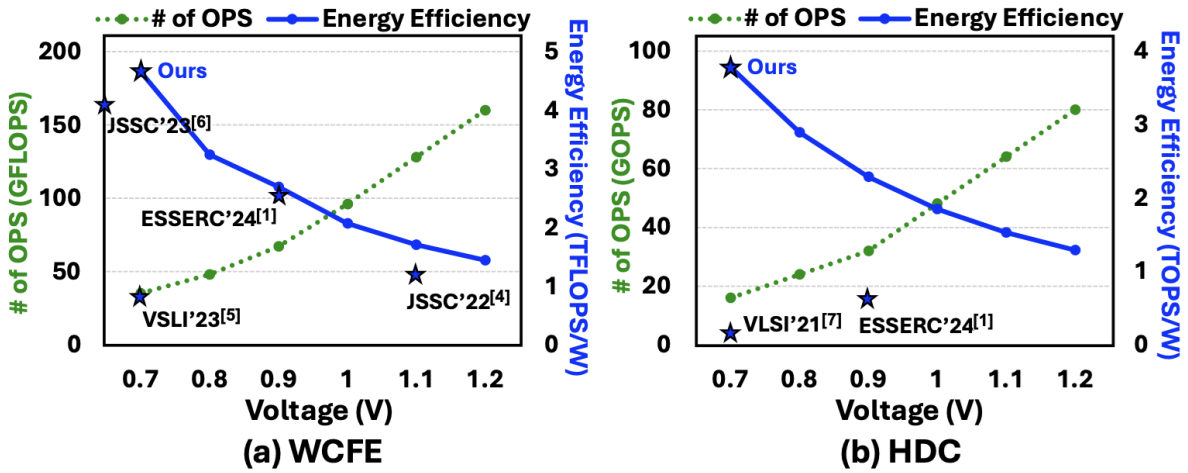
Proposed Clo-HDnn: **C**ontinua**l** **o**n device learning with HDnn without catastrophic forgetting

- Dual mode, Normal & Bypass:
  - Normal mode → feature extractor + HDC (for difficult samples)
  - Bypass mode → HDC only (for easy samples)
- Efficient HD Encoder (Kronecker encoder) – simple computation instead of a large projection matrix
- Progressive Search - early decision with minimal hypervector size → minimizes associative cache memory

C. E. Song*, W. Xu*, K. Fan, S. Jain, G. Hota, H. Yang, **L. Liu, K. Akarvardar, M. F. Chang, C. H. Diaz**, G. Cauwenberghs, T. Rosing, and M. Kang "**Clo-HDnn: A 4.66 TFLOPS/W and 3.78 TOPS/W Continual On-Device Learning Accelerator with Energy-efficient HD Computing via Progressive Search**" IEEE Symposium on VLSI Technology and Circuits 2025

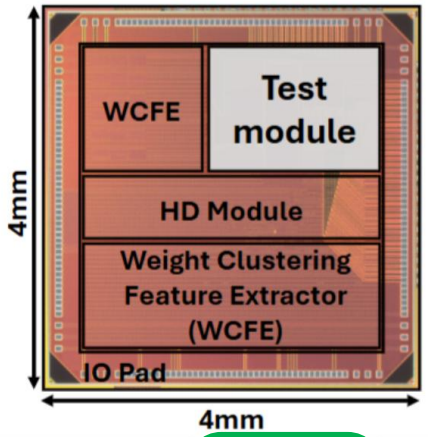
Clo-HDnn ASIC: Results

Jointly with TSMC



Energy efficiency and peak throughput of (a) Feature extractor, (b) HDC running CIFAR-100

- **7.8x** and **4.9x** higher energy efficiency for the weight clustering feature extractor (WCFE) and classifier (HDC) vs. SOTA
- **88%** and **94%** of latency and energy reduction due to bypassing



Chip Summary Table	
Technology	40nm CMOS
Die Size	14.4 mm²
Capacity (kB)	SRAM: 168 (WCFE), 32 (HDC)
Supply Voltage	0.7V–1.2V
Frequency	50MHz – 250MHz
Model	CNN (WCFE) + HDC
Precision	BF16 (CNN) INT1-8 (HDC training) INT8 (HDC inference)
Feature Dimension (*F*)	8-1024
HDC Dimension (*D*)	1024-8192
Max # of Class	128
Peak Energy Efficiency	CNN (WCFE): 1.44-4.66 TFLOPS/W HDC: 1.29-3.78 TOPS/W

	Our work	ESSERC'24 [4]	VLSI'23 [8]	JSSC'23 [9]	JSSC'22 [3]	VLSI'21 [10]
Technology	40nm	40nm	28nm	28nm	40nm	40nm
Learning Mode	CL HDC	FSL HDC	LET	Sparse BP	Low-rank BP	OSL
Design	Digital	Digital	Digital + CIM	Digital	Digital + CIM	ReRAM CIM
Encoder Type	Kronecker	cRP-based	-	-	-	-
Precision	BF16/INT1-8	BF16/INT16	BF16	FP8/16	INT8	FP32
On-chip Mem. (kB)	SRAM: 200	SRAM: 424	SRAM: 329	SRAM: 1280	ReRAM: 204 SRAM:512	ReRAM: 8
Area (mm²)	14.4	11.3	5.8	16.4	29.2	0.2
Frequency (MHz)	50-250	100-250	20-450	75-340	200	200
Supply voltage (V)	0.7-1.2	0.9-1.2	0.56-1.05	0.6-1.1	1.1	-
Scaled EE (TFLOPS/W) (CNN)	4.66 @ResNet18	2.69 @VGG16	0.6-0.87	4.1@ ResNet20	1.1* @ResNet18 (2.2 TOPS/W)	-
Scaled EE (TOPS/W) (Classifier)	3.78 (HDC)	0.78	-	-	-	0.12

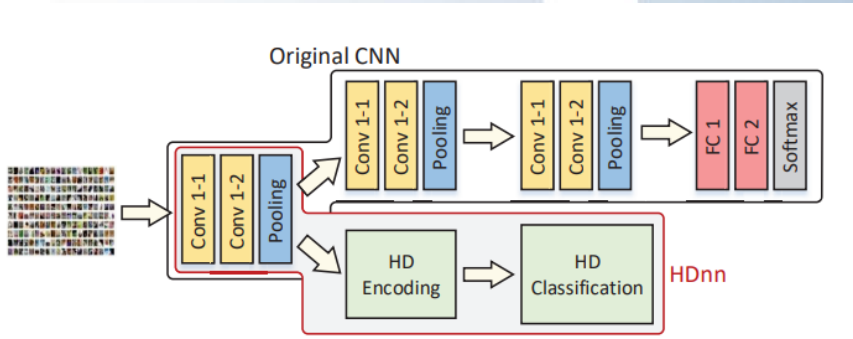
All the energy efficiency (EE) is scaled to 40nm technology. * Scaled INT8 (TOPS/W) to BF16 (TFLOPS/W)

C. E. Song*, W. Xu*, K. Fan, S. Jain, G. Hota, H. Yang, L. Liu, K. Akarvardar, M. F. Chang, C. H. Diaz, G. Cauwenberghs, T. Rosing, and M. Kang "**Clo-HDnn: A 4.66 TFLOPS/W and 3.78 TOPS/W Continual On-Device Learning Accelerator with Energy-efficient HD Computing via Progressive Search"** IEEE Symposium on VLSI Technology and Circuits 2025

H. Yang*, C. E. Song*, W. Xu, B. Khaleghi, U. Mallappa, M. Shah, K. Fan, M. Kang, and T. Rosing ***"FSL-HDnn: A 5.7 TOPS/W End-to-end Few-shot Learning Classifier Accelerator with Feature Extraction and Hyperdimensional Computing"***, IEEE European Solid-State Electronics Research Conference (**ESSERC**), 2024

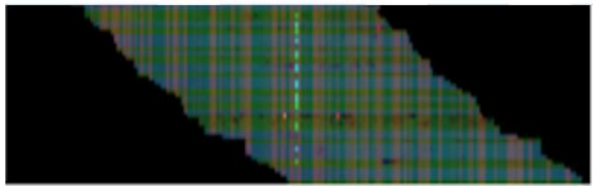
DeepVariant & HDnn Acceleration

- Variant calling finds changes in genomes
  - e.g. mutations in cancer
- DeepVariant[Nature'18] uses CNNs
  - Converts aligned sequences into images & then detects variants
  - HDnn accelerates image classification in memory

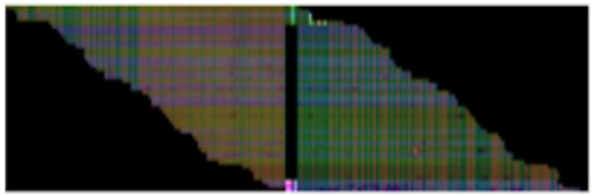


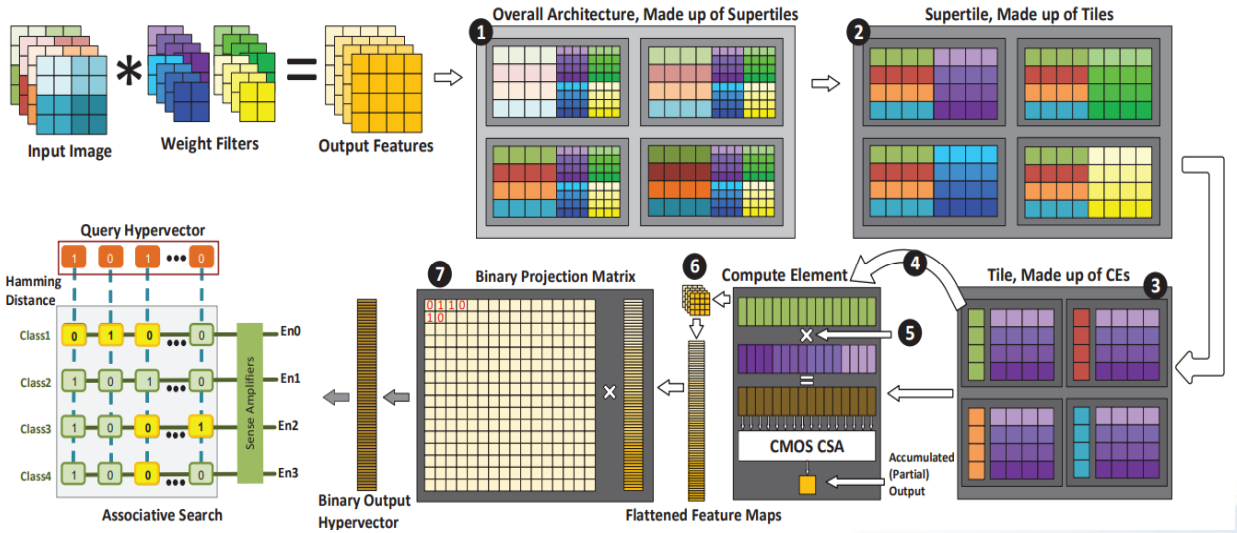
Examples of DeepVariant Images

No variants



Variants in two chromosomes





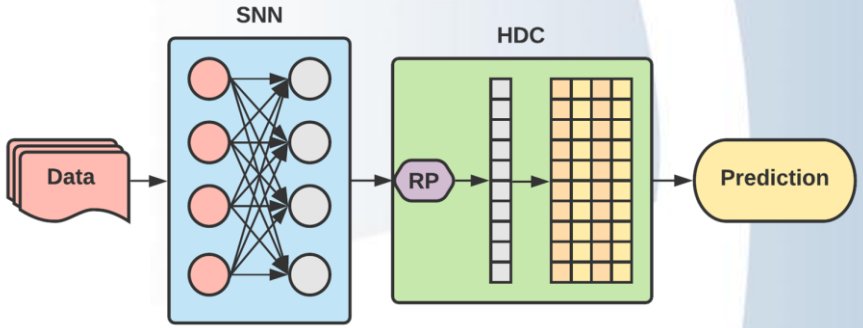
HDnn has at least **233x** higher throughput vs. SOTA

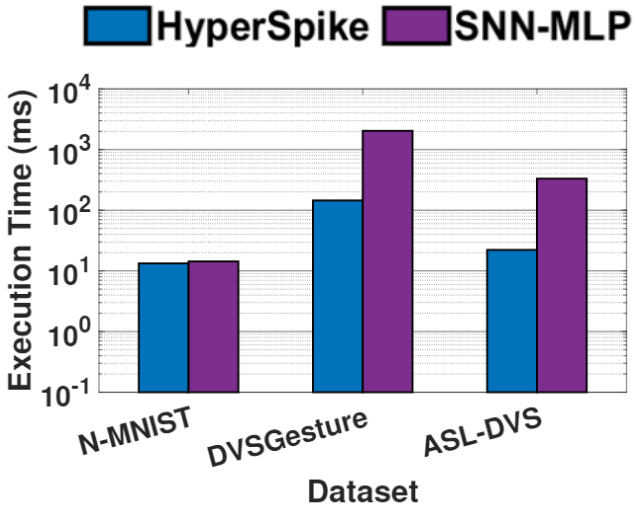
A. Dutta, Gupta, S., Rosing, T et al. "HDnn-PIM: Efficient in Memory Design of Hyperdimensional Computing with Feature Extraction," GLVLSI'22

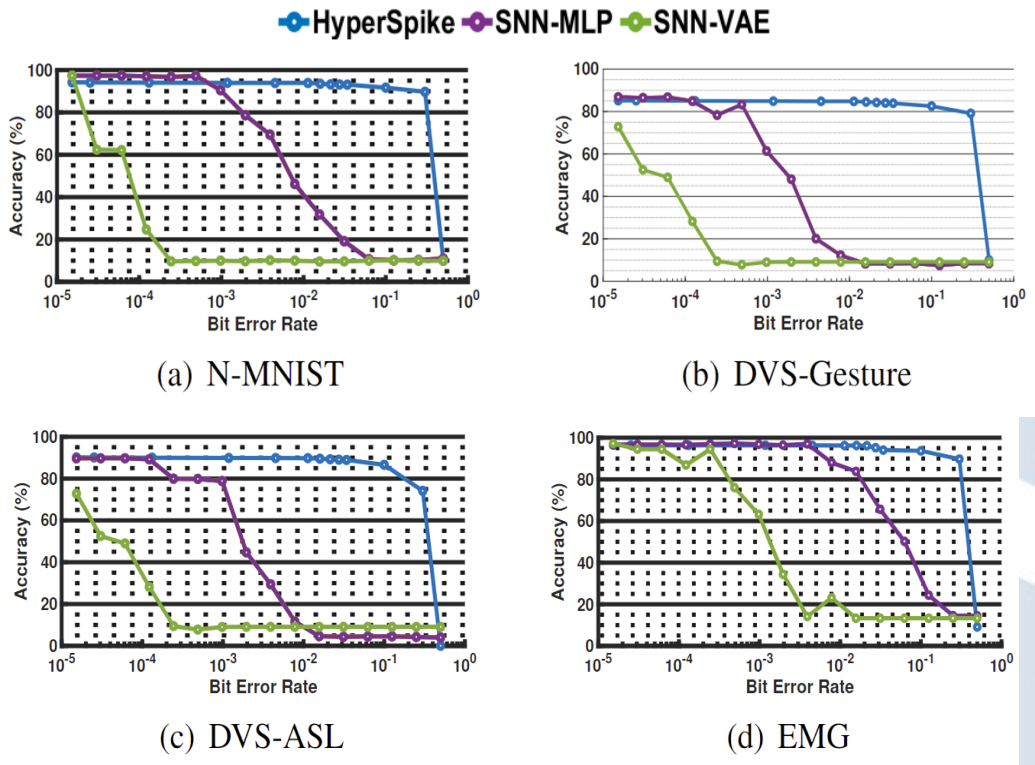
Poplin, Ryan, et al. "A universal SNP and small-indel variant caller using deep neural networks." Nature biotechnology'18 – source of variant images

Event-based Cameras & Sensors [TCAD'23]
HyperSpike: HD Computing with Spiking Neural Networks

- Single layer **untrained SNN** extracts features
- HDC does reasoning
  - Random projection, binary quantization, Hamming distance
- Implemented using Intel Loihi & TinyHD
  - **15x faster** and 4.6x more energy efficient
  - **58x more robust** at 3.4% BER



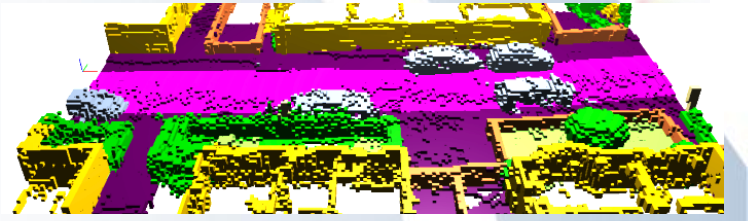




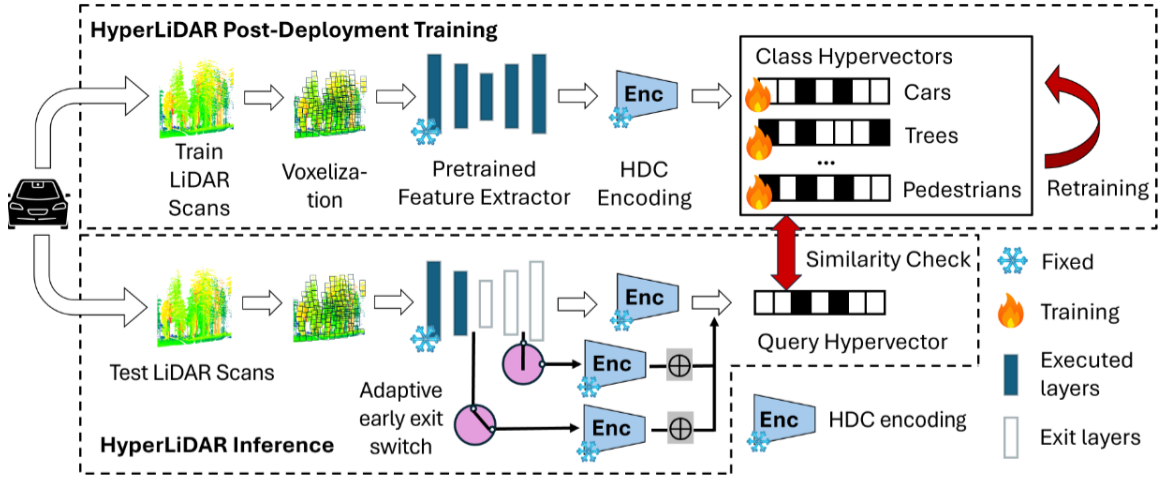
T. Zhang, J Morris, HW Lui, K Stewart, B Khaleghi, A Thomas, T Marback, B Aksanli, E Neftci, T Rosing, "HyperSpikeASIC: HD Computing for More Efficient and Robust Spiking Neural Networks, TCAD'23

LiDAR Segmentation with HD Computing on >80GB data

PI Tajana Rosing, Yi Yao, Flavio Ponzina, Xiaofan Yu, Ivannia Gomez Moreno @UCSD
PI Hun Seok Kim (CogniSense), Mingyu Yang (CogniSense)



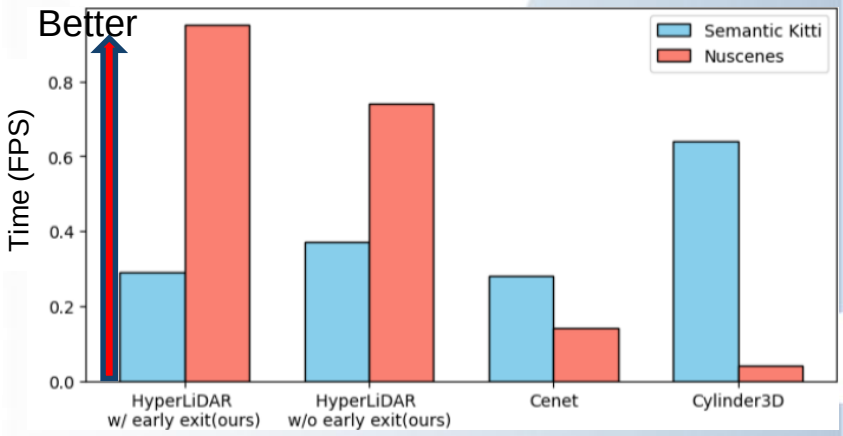
- **Problem & Opportunity**:
  - SOTA LiDAR segmentation algorithms cannot be trained online due to multiple layers of encoder-decoder
- **Technical Approach**:
  - Random points are sampled from the point cloud, passed through a frozen feature extractor, features are passed to HD classifier for training to the specific dataset → online learning possible via HDC
- **HyperLiDAR is at least 2x faster** in training than CENET [ICME '22] & Cylinder3D [CVPR '21], and faster than both for inference
  - HyperLiDAR is the only method capable of online learning/training
- **HyperLiDAR achieves comparable accuracy** for nuScenes (1k sequences; 30k points per scan) & SemanticKITTI (360º view, 43k scans, 25 classes; ~60k points per scan)



SEGMENTATION ACCURACY COMPARISON ON NUSCENES-MINI AND SEMANTICKITTI DATASETS.

Dataset	Model	mIoU	Car	Truck	Bicycle	Person	Road	Sidewalk	Building	Vegetation
nuScenes-mini	CENet [10]	19.5	44.6	0.0	0.0	16.6	79.6	17.4	-	47.2
	Cylinder3D [51]	30.4	78.9	0.0	0.0	51.9	87.2	22.3	-	68.0
	HyperLiDAR w/o early-exit (Ours)	**76.4**	**85.4**	**86.2**	**27.0**	**66.7**	**92.1**	**64.3**	**-**	**85.1**
	HyperLiDAR w/ early-exit (Ours)	67.3	72.6	76.4	26.5	64.1	89.4	48.3	-	85.08
SemanticKITTI	CENet [10]	63.5	96.4	87.7	50.8	66.8	95.3	82.5	86.0	86.0
	Cylinder3D [51]	56.6	94.5	58.3	33.5	59.9	93.0	78.3	88.0	89.0
	HyperLiDAR w/o early-exit (Ours)	57.0	85.5	63.1	36.2	38.8	84.2	66.5	84.0	82.3
	HyperLiDAR w/ early-exit (Ours)	54.9	82.8	67.8	39.2	40.9	70.7	55.6	81.6	82.9

Training time in Frames/sec



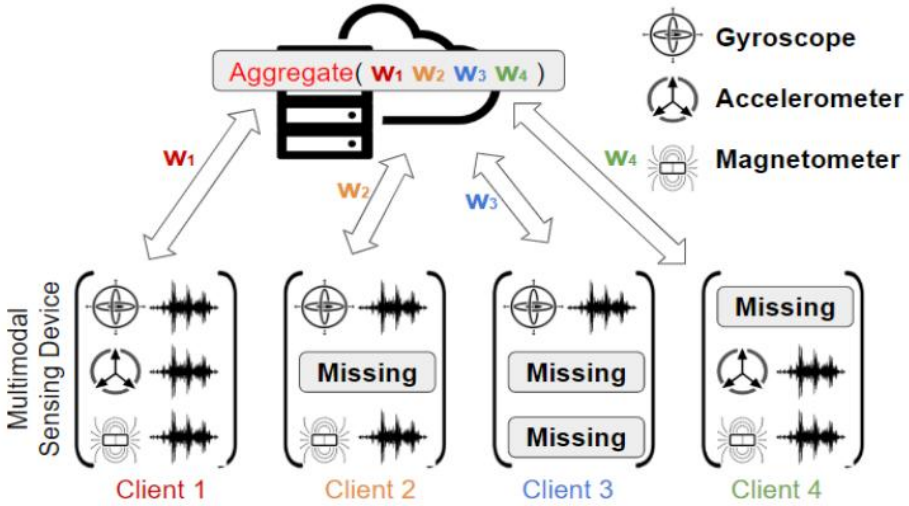
Baselines: https://github.com/DarthIV02/LidarSegHD, Model: https://github.com/DarthIV02/3DLabelProp

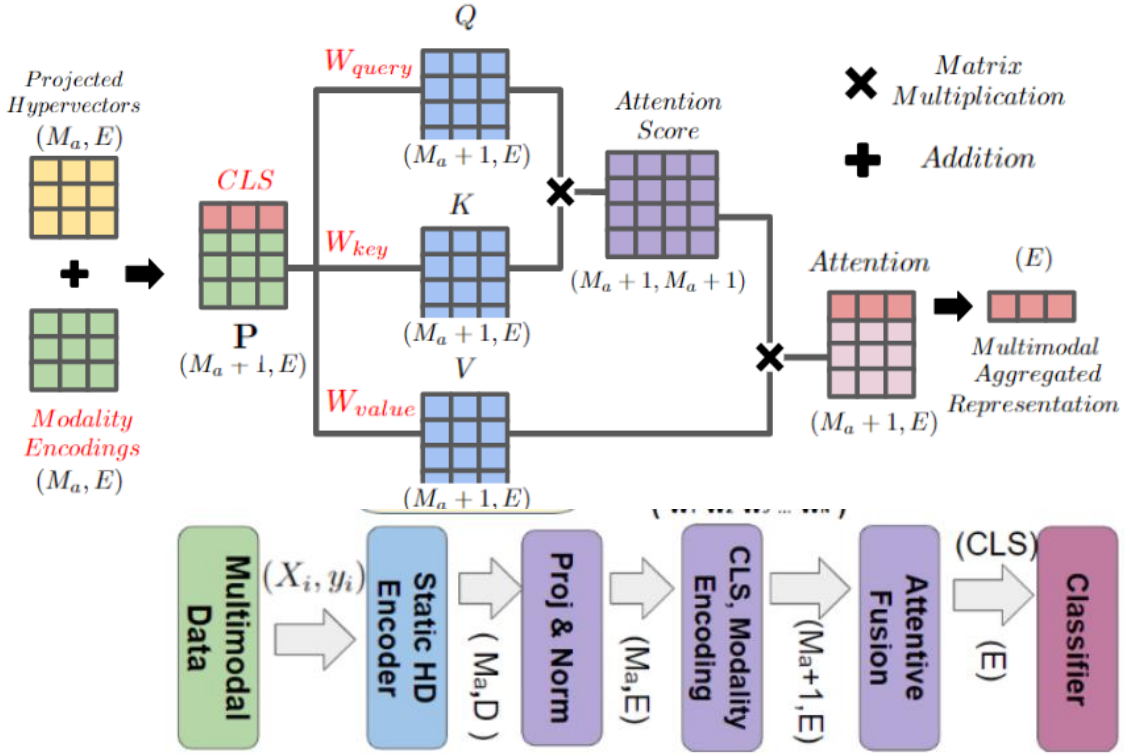
Training Latency experiments on NVIDIA A10 GPU

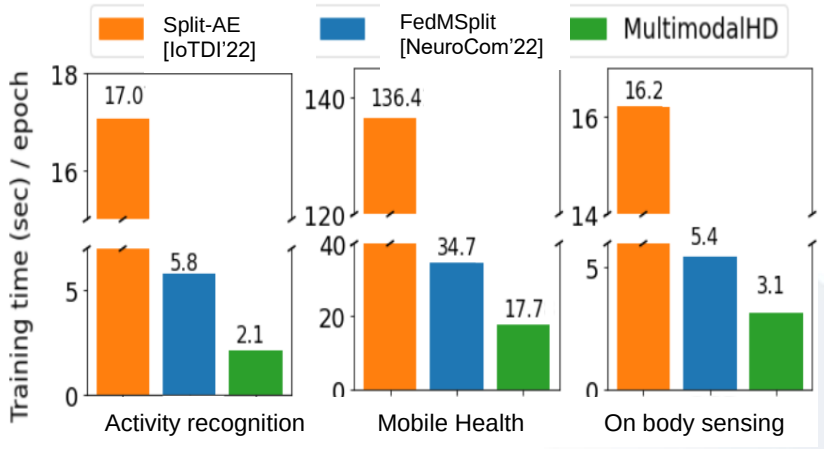
Y. Yi, I.Gomez Moreno, X. Yu, M. Sullivan, T. Rosing, "HyperLiDAR: Label- and Energy-Efficient LiDAR Segmentation with HD Computing" submitted.

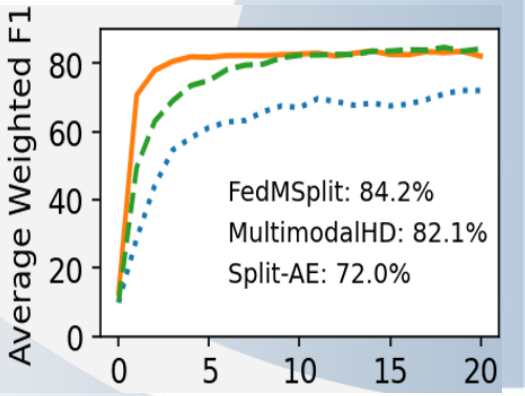
MultimodalHD: Federated Learning Over Heterogeneous Sensor Modalities using Hyperdimensional Computing [DATE'24]

- Multimodal learning challenges on time-series sensor data
  - DNN: relies on slow Recurrent NN models
    - Split-AE [IoTDI'22] & FedMSplit [NeuroComp'22]
  - HDC: fast, but not accurate when capturing dynamics of multimodal data
- Attentive Multimodal Attention on HDC representation
  - Tokenizes modalites and uses attention mechanism for multimodal fusion
  - Combines the efficiency of HDC and the strengths of attention representations









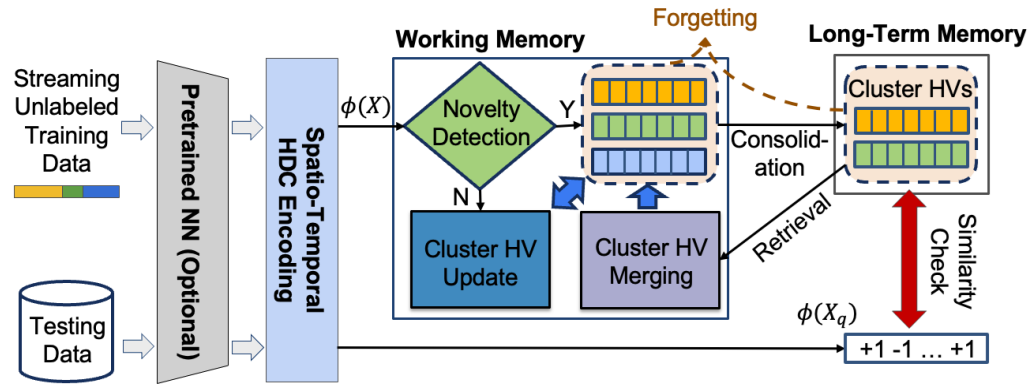
Up to 8x faster training at SOTA accuracy

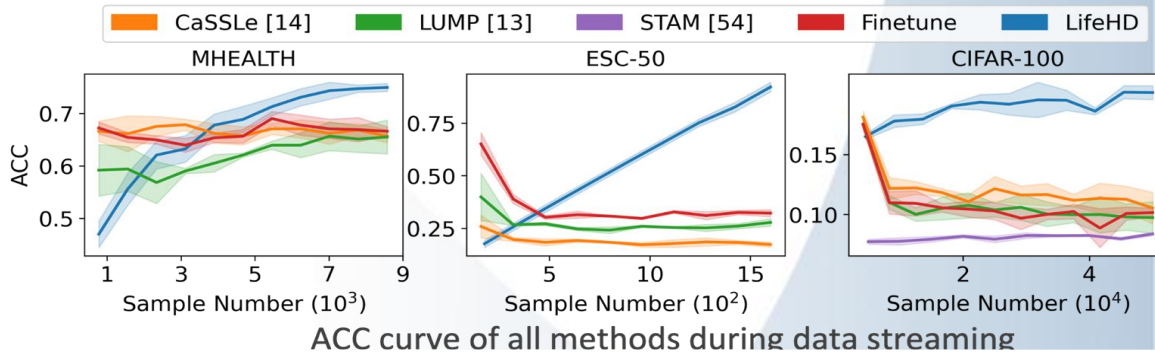
Quanling Zhao, Xiaofan Yu, Shengfan Hu, Tajana Rosing, "MultimodalHD: Federated Learning Over Heterogeneous Sensor Modalities using HD Computing" DATE'24

LifeHD: Lifelong Intelligence Beyond the Edge using HDC [IPSN'24]

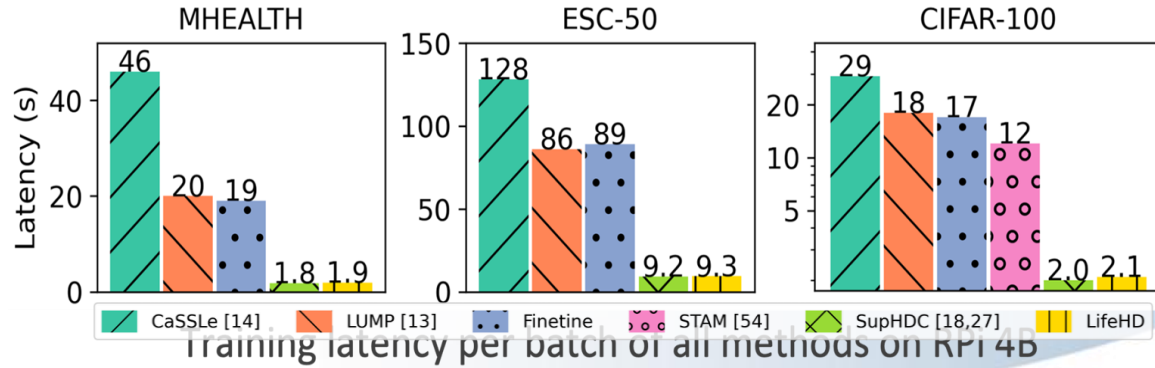
Jointly with Intel

- Goal: Learn and adapt to changing environment after deployment, without supervision or prior data
- LifeHD has three key components:
  - Novelty detection, online cluster update & merging
- SOTA comparison algorithms:
  - Unsupervised lifelong learning based on DNNs
    - CaSSLe [CVPR'22]: past knowledge distillation
    - LUMP [ICLR'22]: memory replay
  - Neurally-inspired lightweight algorithms; fully supervised
    - FlyModel [Shen'21], SDMLP [ICLR'23]: sparse coding and associative memory
    - STAM [IJCAI'21]: progressive memory architecture
- Results for LifeHD vs. SOTA
  - 74.8% better unsupervised clustering accuracy
  - Up to 34.3x better energy efficiency
  - Faster training time





ACC curve of all methods during data streaming



Training latency per batch of all methods on RPi 4B

CoCoSys
CENTER FOR THE CO-DESIGN OF COGNITIVE SYSTEMS

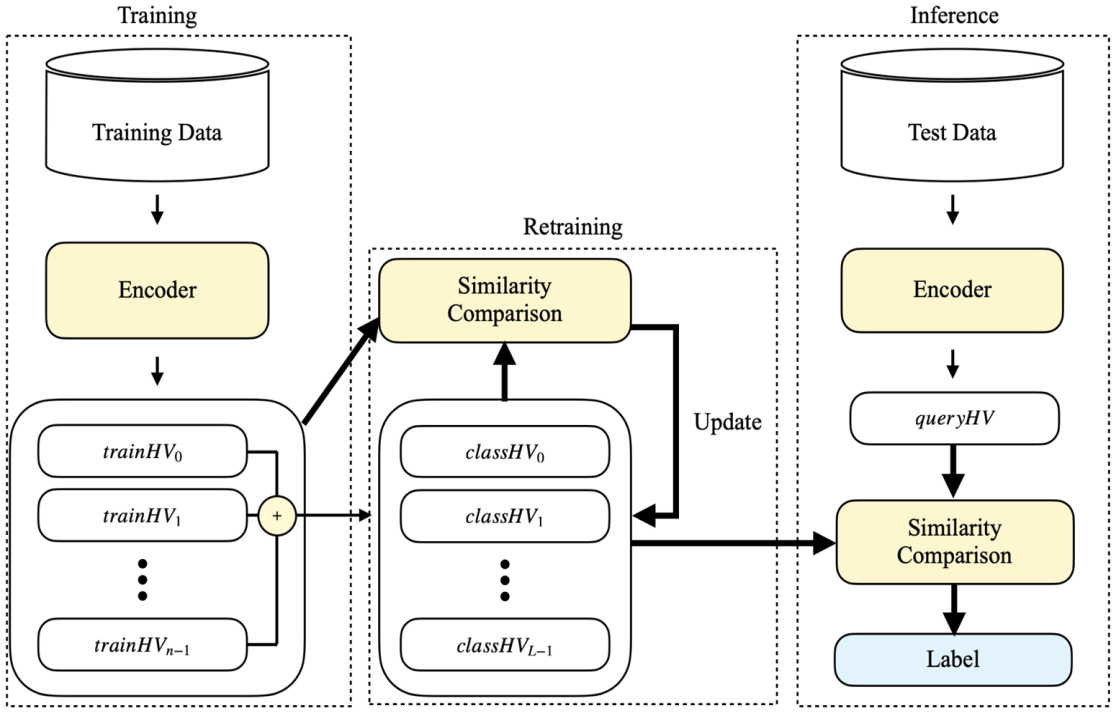
X. Yu, A. Thomas, I. Gomez Moreno, L. Gutierrez, T. Rosing, "Lifelong Intelligence beyond the Edge using Hyperdimensional Computing", IPSN'24

FHE-HD: End-to-End FHE-based ML with HD Computing

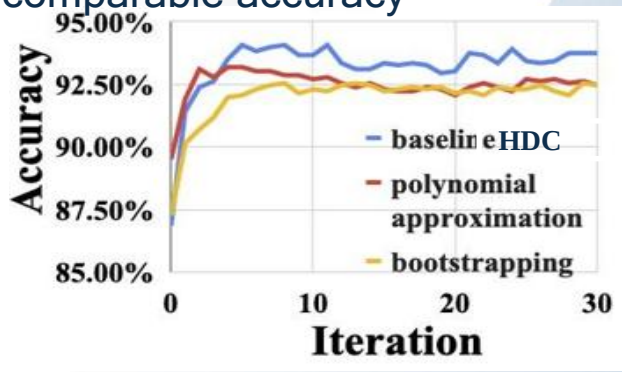
- Problem:
  - FHE is currently only done for inference [PMLR'22, Asia CCS'22] & simple tasks [AAAI'19, NeurIPS'20]
- Approach: use HD computing for FHE:
  - Encoding: vector-matrix multiplication + nonlinear activation
  - Training: addition of hypervectors (HVs)
  - Inference: similarity check between a query and class HV
  - Retraining: subtract/add vectors based on the inference results

FHE-HD is the first end-to-end learning system that performs ALL operations in the FHE domain

Work	#Iter	Latency		
		1-epoch	Total training	Inference
MLP 1 [1]	5	14 days	69.9 days	-
MLP 2 [2]	5	17.4 days	86.8 days	-
FHE-HD	5	2.9 days	15 days	2.58 s
RNN [3]	-	-	-	49 min



- >1,000x faster inference vs. FHE-based RNN [Asia CCS'22]
- Up to 5.8x faster training vs. FHE-based MLP [NeurIPS'22, CVPR'19] with comparable accuracy

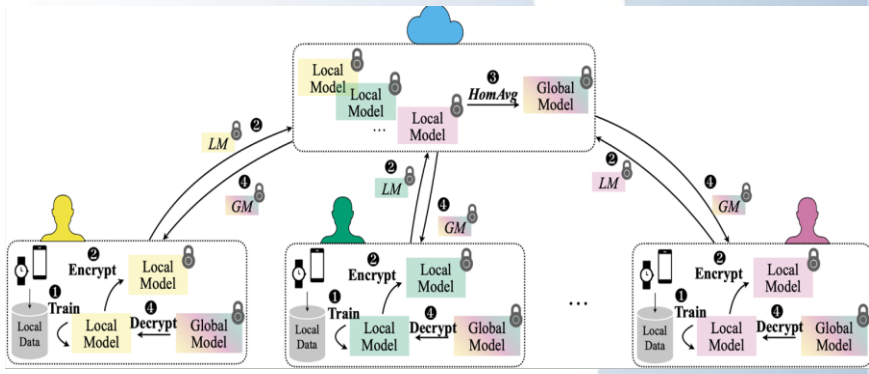


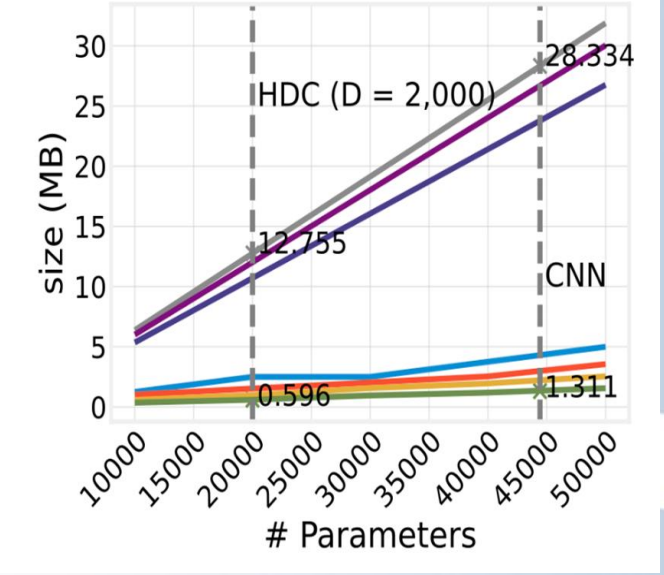
Y. Nam, M. Zhou, S. Gupta, G. De Micheli, R. Cammarota, C. Wilkerson, D. Micciancio, T. Rosing, "Efficient Machine Learning on Encrypted Data using HD Computing," ISLPED"23.

Rhychee-FL: Robust and Efficient Hyperdimensional Federated Learning with FHE

PI Tajana Rosing, PI Priyadarshini Panda (Yale, CoCoSys), Yujin Nam (PhD, UCSD), Xiaofan Yu (PhD, UCSD), Xuan Wang (PhD, UCSD), Minxuan Zhou (PI IIT), Yeshwanth Venkatesha (PhD, Yale, CoCoSys), Abhishek Moitra (PhD, Yale, CoCoSys), Gabrielle De Micheli (PostDoc, UCSD), Augusto Vega (IBM)

- **Problem & Opportunity**:
  - Federated Learning (FL) faces privacy concerns from sharing locally trained models that may reveal sensitive data.
- **Technical Approach**:
  - Rhychee-FL integrates Fully Homomorphic Encryption (FHE) with Hyperdimensional Computing (HDC) to create a lightweight, noise-resilient FL framework that reduces communication and computation overhead while preserving accuracy and privacy.
  - Reduced communication size by up to 21.4×, improved aggregation latency by up to 20.5×, achieved high accuracy 6× faster and 2.2× more communication-efficient compared to CNN-based FL, through optimized parameter tuning and FHE simulation.
- **Key results and metrics vs. SoTA**:
  - 2.2× less data sent with 6× faster convergence at comparable accuracy vs. SOTA CNN [ICDE'22]



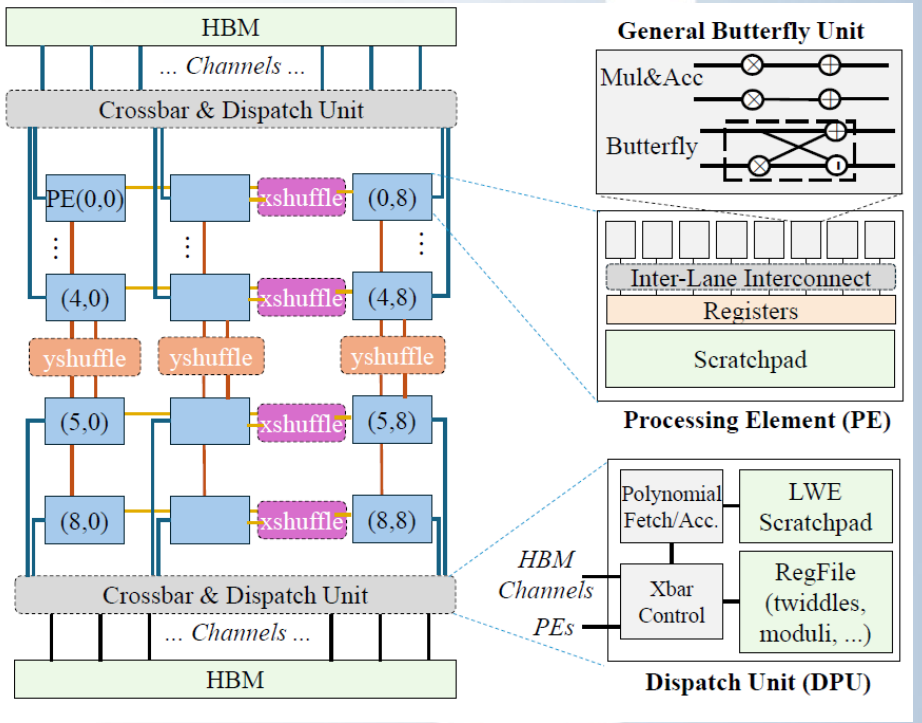


Yujin Nam, Xiaofan Yu, Xuan Wang, Minxuan Zhou, Yeshwanth Venkatesha, Abhishek Moitra, Gabrielle De Micheli, Augusto Vega, Priyadarshini Panda and Tajana Rosing "Rhychee-FL: Robust and Efficient Hyperdimensional Federated Learning with Homomorphic Encryption", DATE, 2025

Unified Accelerator for Fully Homomorphic Encryption

Tajana Rosing (PI@UCSD), Chris Wilkerson (Intel), Rosario Cammarota (Intel), Sanu Mathew (Intel), Raghavan Kumar (Intel), Sachin Tajene (Intel), Minxuan Zhou (Postdoc, UCSD), Yujin Nam (PhD, UCSD), Xuan Wang (PhD, UCSD), Youhak Lee (PhD, UCSD)

- **Problem & Opportunity**:
  - Existing FHE accelerators do not support hybrid-scheme FHE, resulting in limited support for efficient FHE computations
- **Technical Approach**:
  - Unified architecture that consists of low-level primitive function units with efficient interconnect for high-throughput processing of hybrid-scheme FHE applications
  - Novel software-hardware co-design to fully utilize the proposed hardware without introducing costly chip overhead
- **Key results and metrics vs. SOTA**:
  - 6.0× faster and 1.6× better energy-delay-area efficiency vs. FHE accelerators [ISCA'23, MICRO'23].
  - Tape out at Intel planned for '25, earlier version taped out in '24

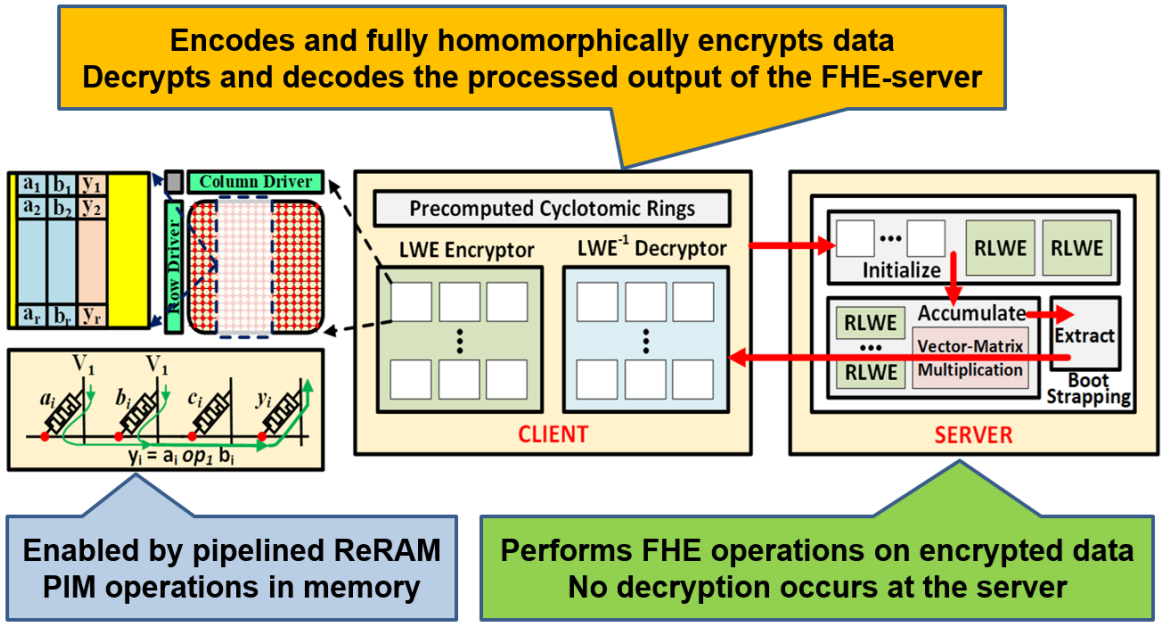


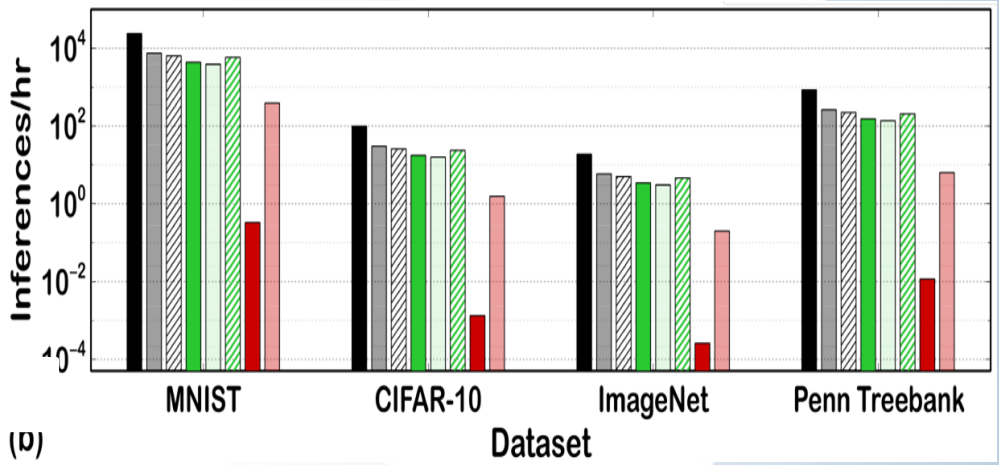
Architecture of the unified accelerator for FHE

Minxuan Zhou, Yujin Nam, Xuan Wang, Youhak Lee, Chris Wilkerson, Rahavan Kumar, Sachin Taneja, Sanu Mathew, Rosario Cammarota, and Tajana Rosing, "UFC: A Unified Accelerator for Fully Homomorphic Encryption", MICRO'24, SRC#383127

FHE computing with HDC in PIM [TECS'24, TETC'25]

- Fully homomorphic encryption removes the need for decryption => all processing in encrypted domain
  - Problem: Explosion of data and operations; e.g. int turns into 20kB, int multiply takes >10M ops
- MemFHE design implements 3rd generation fully homomorphic encryption in 1TB ReRAM PIM
- We compare MemFHE with TDNN-FHE (TDNN-Lvl) [NeurIPS'19] with 163-bit (152-bit) classical security that runs on Intel Xeon E7-4850 CPU. 1TB DRAM





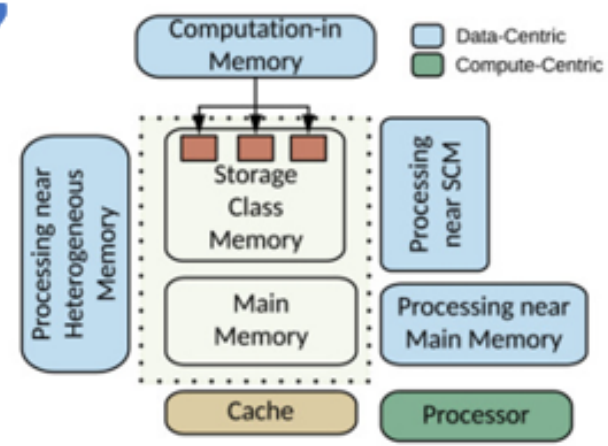
MemFHE DNN vs. TDNN-FHE [NeurIPS'19]
- **Normal mode: 36,007x higher throughput**
- **Quantum-safe mode: 15,000x higher throughput**

MemFHE + HDC provides another 10-30x in speed vs SOTA

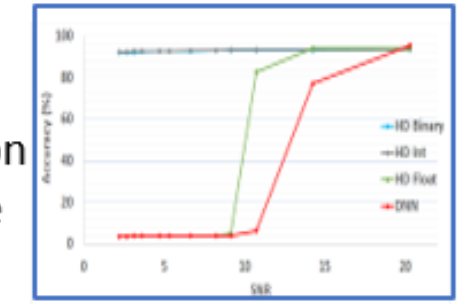
Minxuan Zhou, Yujin Nam, Pranav Gangwar, Weihong Xu, Arpan Dutta, Chris Wilkerson, Rosario Cammarota, Saransh Gupta, Tajana Rosing," FHEmem: A processing in-memory accelerator for fully homomorphic encryption," IEEE TETC'25

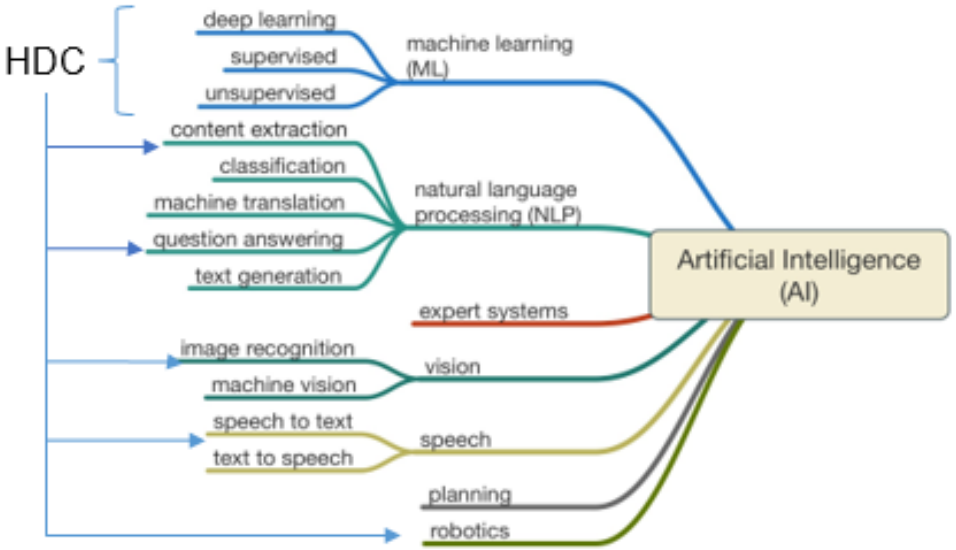
S. Gupta, T. Rosing et al "MemFHE: FHE acceleration in memory," ACM TECS'24

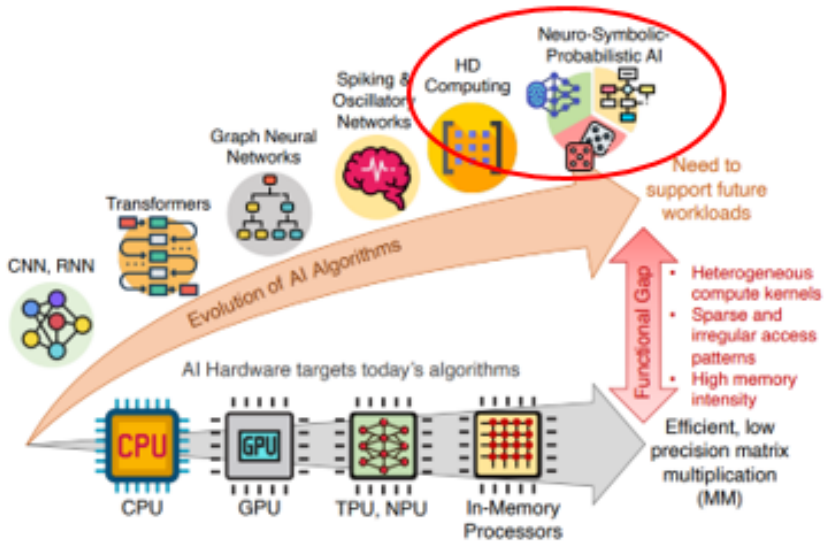
Where to next? Real-Time, Lightweight, Robust & Secure Data Analytics at Scale



- **Vision: Create novel intelligent memory and storage architectures that**
  - Answer when, where and how to store and process which data
  - Seamlessly integrate diversity of memory, storage, compute & software
  - Optimize for best performance, power, area and cost tradeoffs
- **HD Computing (HDC) is a promising solution for future systems**
  - Learns adaptively due to fast training & inference => secure lifelong & federated learning at scale
  - Handles big data => e.g. recommendation systems, mass spectrometry, image processing, ML
  - Inherently robust => excellent for new memory and storage devices & error-prone communication
  - Efficiently combines neuro-symbolic reasoning with probabilistic learning while being explainable







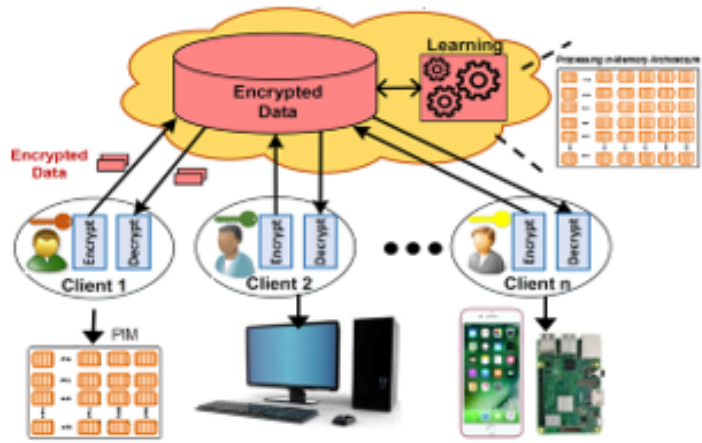


Figure sources: JUMP 2.0 Centers PRISM & CoCoSys


ACADEMIA • INDUSTRY • GOVERNMENT •