Efficient Biosignal Processing with Brain-inspired High-dimensional Computing:

A Universal ExG Classifier

Abbas Rahimi, Pentti Kanerva, Luca Benini, Jan M. Rabaey

ETH Zurich and UC Berkeley

Brain-inspired High-dimensional Computing

[P. Kanerva, An Introduction to Computing in Distributed Representation with High-Dimensional Random Vectors, *Cogn Comput'09*]

- Emulation of cognition by computing with **high-dimensional** vectors as opposed to computing with numbers
- Information distributed in **high-dimensional space**
- Supports **full algebra**

Superb properties:

- **General** and scalable model of computing
- **Well-defined** set of arithmetic operations
- **Fast and one-shot** learning (no need of back-prop)
- Memory-centric with embarrassingly **parallel** operations
- **Extremely robust** against most failure mechanisms and noise
- **Energy efficient**

Brain-inspired High-dimensional Computing

[P. Kanerva, An Introduction to Computing in Distributed Representation with High-Dimensional Random Vectors, *Cogn Comput'09*]

- Emulation of cognition by computing with **high-dimensional** vectors as opposed to computing with numbers
- Information distributed in **high-dimensional space**
- Supports **full algebra**

Superb properties:

- **General** and scalable model of computing
- **Well-defined** set of arithmetic operations
- **Fast and one-shot** learning (no need of back-prop)
- Memory-centric with embarrassingly **parallel** operations
- **Extremely robust** against most failure mechanisms and noise
- **Energy efficient**

What Are HD Vectors?

It is all about data representation

1^{st} 2^{nd} 3^{rd} 4^{th} 10000^{th}

[−1 +1 −1 −1 .... +1]

What Are HD Vectors?

It is all about data representation

1^{st} 2^{nd} 3^{rd} 4^{th} 10000^{th}

[−1 +1 −1 −1 .... +1]

High-dimensional

What Are HD Vectors?

It is all about data representation

$$\begin{array}{ccccccccccccc} 1^{st} & 2^{nd} & 3^{rd} & 4^{th} & & & & & & & & & 10000^{th} \\ [-1 & +1 & -1 & -1 & . & . & . & . & . & . & . & . & +1] \end{array}$$

High-dimensional

Holographic

Distributed

What Are HD Vectors?

It is all about data representation

$$\begin{array}{cccccc} 1^{st} & 2^{nd} & 3^{rd} & 4^{th} & & 10000^{th} \\ [-1 & +1 & -1 & -1 & \ldots\ldots\ldots\ldots\ldots\ldots & +1] \end{array}$$

High-dimensional

Holographic

Distributed

Pseudorandom with i.i.d. components

What Are HD Vectors?

It is all about data representation

1^{st} 2^{nd} 3^{rd} 4^{th} 10000^{th}

[−1 +1 −1 −1 .... +1]

High-dimensional

Holographic

Distributed

Pseudorandom with i.i.d. components

Combine and Compare

What Are HD Vectors?

It is all about data representation

$$\begin{array}{cccccc} 1^{st} & 2^{nd} & 3^{rd} & 4^{th} & & 10000^{th} \\ [-1 & +1 & -1 & -1 & \ldots\ldots\ldots\ldots\ldots & +1] \end{array}$$

High-dimensional | Holographic | Distributed | Pseudorandom with i.i.d. components

Combine and Compare

bipolar
binary
dense
sparse

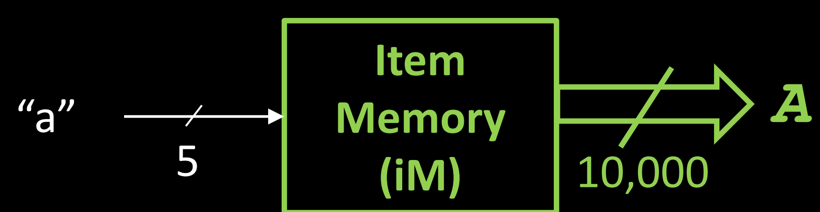
Approximate computation with fixed-size long random patterns that provides a novel look at data representations, associated operations, circuits, and architectures.

Mapping to HD Vectors

- Each letter (symbol) is represented by an HD vector chosen at *random* with 10,000–d:

```
A = [−1 +1 −1 −1 −1 +1 −1 −1 ...]
B = [+1 −1 +1 +1 +1 −1 +1 −1 ...]
C = [−1 −1 −1 +1 +1 −1 +1 −1 ...]
D = [−1 −1 −1 +1 +1 −1 +1 −1 ...]
...
Z = [−1 −1 +1 −1 +1 +1 +1 −1 ...]
```

- Every letter HD vector is dissimilar to others: $\langle A, B \rangle = 0$
- This assignment is fixed throughout computation



HD Arithmetic

- Componentwise addition (+) is good for representing sets, since sum vector is similar to its constituent vectors:

  $\langle A+B, A\rangle = 0.5$

- Componentwise multiplication (∗) is good for binding, since product vector is dissimilar to its constituent vectors:

  $\langle A*B, A\rangle = 0$

- Permutation (ρ) makes a dissimilar vector by rotating, it good for representing sequences:

  $\langle A, \rho A\rangle = 0$

- ∗ and ρ are invertible and preserve distance

Example: Computing Language Profile

How to encode "`Ich bin`"?

Example: Computing Language Profile

How to encode "$\boxed{\texttt{I}}\boxed{\texttt{c}}\boxed{\texttt{h}}$ `bin`"?

$\boxed{\boldsymbol{I} \;=\; +1\; -1\; -1\; +1\; -1\; -1\; \ldots\; +1\; +1\; -1\; -1}$

$\boxed{\boldsymbol{C} \;=\; +1\; -1\; +1\; +1\; +1\; +1\; \ldots\; +1\; -1\; +1\; -1}$

$\boxed{\boldsymbol{H} \;=\; +1\; +1\; +1\; -1\; -1\; +1\; \ldots\; +1\; -1\; +1\; +1}$

Example: Computing Language Profile

How to encode "Ich bin"?

```
Trigram encoding: "Ich" = ρρI * ρC * H
   I  =  +1 -1 -1 +1 -1 -1 ... +1 +1 -1 -1 +1 -1
          /  /  /  /  /  /      /  /  /  /  /  /
   C  =  +1 -1 +1 +1 +1 +1 ... +1 -1 +1 -1 +1
          /  /  /  /  /  /      /  /  /  /  /
   H  =  +1 +1 +1 -1 -1 +1 ... +1 -1 +1 +1
----------------------------------------------
"Ich"=  +1 +1 -1 +1  .... ... +1 +1 -1 -1
```

Example: Computing Language Profile

How to encode "$\boxed{\texttt{Ich bin}}$"?

```
Trigram encoding: "Ich" = ρρI * ρC * H
   I  =  +1 -1 -1 +1 -1 -1 ... +1 +1 -1 -1 +1 -1
          /  /  /  /  /  /      /  /  /  /  /  /
   C  =  +1 -1 +1 +1 +1 +1 ... +1 -1 +1 -1 +1
          /  /  /  /  /  /      /  /  /  /  /
   H  =  +1 +1 +1 -1 -1 +1 ... +1 -1 +1 +1
-----------------------------------------------
"Ich"=  +1 +1 -1 +1  .... ... +1 +1 -1 -1
```

Adding trigrams: "Ich bin" =

Example: Computing Language Profile

How to encode "$\boxed{\texttt{Ich bin}}$"?

```
Trigram encoding: "Ich" = ρρI * ρC * H
   I  =  +1 -1 -1 +1 -1 -1 ... +1 +1 -1 -1 +1 -1
          /  /  /  /  /  /      /  /  /  /  /  /
   C  =  +1 -1 +1 +1 +1 +1 ... +1 -1 +1 -1 +1
          /  /  /  /  /  /      /  /  /  /  /
   H  =  +1 +1 +1 -1 -1 +1 ... +1 -1 +1 +1
-----------------------------------------------
"Ich"=  +1 +1 -1 +1  .... ... +1 +1 -1 -1
```

```
Adding trigrams: "Ich bin" =
"Ich" =  +1 +1 -1 +1 -1 +1  ....
"ch " =  -1 -1 +1 +1 -1 +1  ....
"h b" =  -1 -1 +1 +1 +1 -1  ....
" bi" =  +1 -1 +1 -1 -1 -1  ....
"bin" =  -1 +1 +1 -1 -1 +1  ....
```

Example: Computing Language Profile

How to encode "Ich bin"?

```
Trigram encoding: "Ich" = ρρI * ρC * H
   I  =  +1 -1 -1 +1 -1 -1 ... +1 +1 -1 -1 +1 -1
         /  /  /  /  /  /      /  /  /  /  /  /
   C  =  +1 -1 +1 +1 +1 +1 ... +1 -1 +1 -1 +1
         /  /  /  /  /  /      /  /  /  /  /
   H  =  +1 +1 +1 -1 -1 +1 ... +1 -1 +1 +1
-----------------------------------------
"Ich"=  +1 +1 -1 +1  .... ... +1 +1 -1 -1
```

```
 Adding trigrams: "Ich bin" =
 "Ich" =  +1 +1 -1 +1 -1 +1  ....
 "ch " =  -1 -1 +1 +1 -1 +1  ....
+"h b" =  -1 -1 +1 +1 +1 -1  ....
+" bi" =  +1 -1 +1 -1 -1 -1  ....
+"bin" =  -1 +1 +1 -1 -1 +1  ....
+----------------------------
       =  -1 -1 +1 +1 -1 +1  ....
```

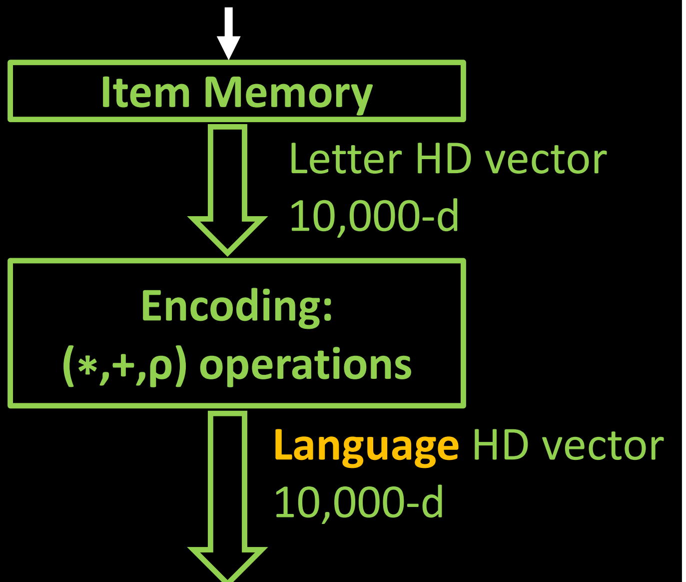
"Ich bin"

EU Language Recognition

Identical hardware for both learning and inference

Train with 100 KB of text from 21 EU languages

Train text: “der emissionserloes soll fuer den weiteren ausbau des qualitativ ...”

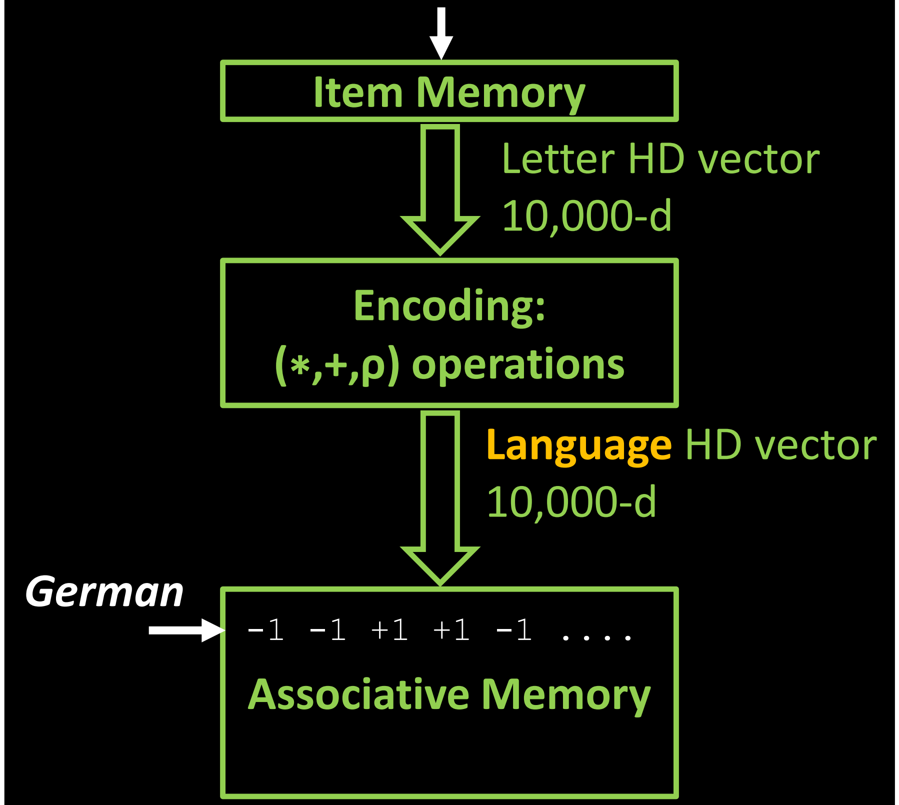


EU Language Recognition

Identical hardware for both learning and inference

Train with 100 KB of text from 21 EU languages

Train text: “der emissionserloes soll fuer den weiteren ausbau des qualitativ ...”

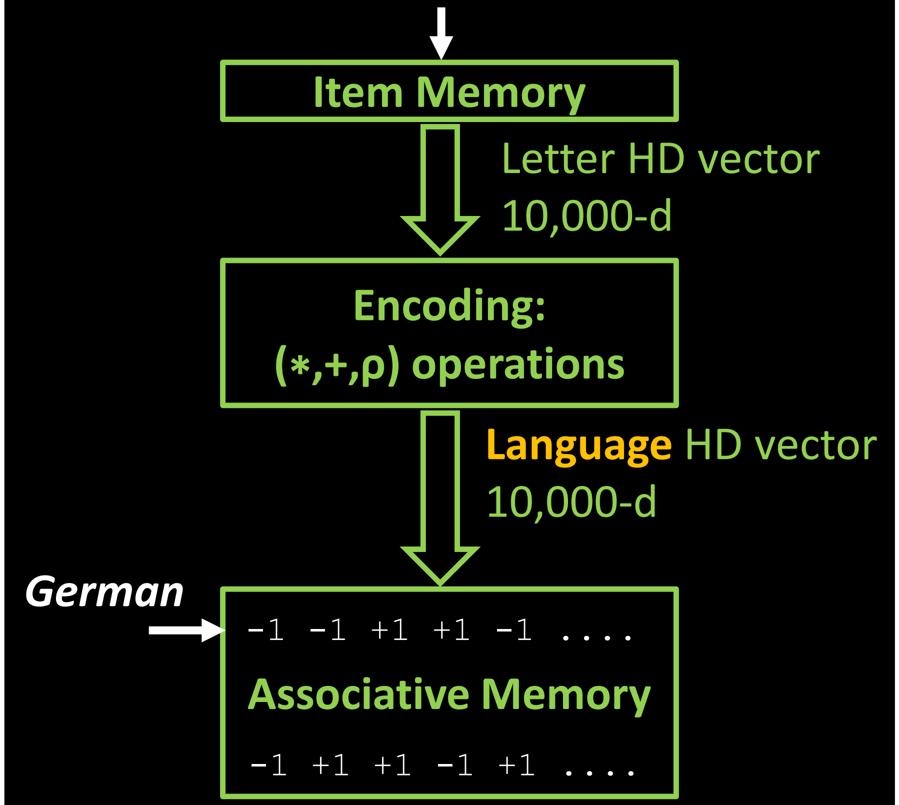


EU Language Recognition

Identical hardware for both learning and inference

Train with 100 KB of text from 21 EU languages

Train text: “der emissionserloes soll fuer den weiteren ausbau des qualitativ ...”

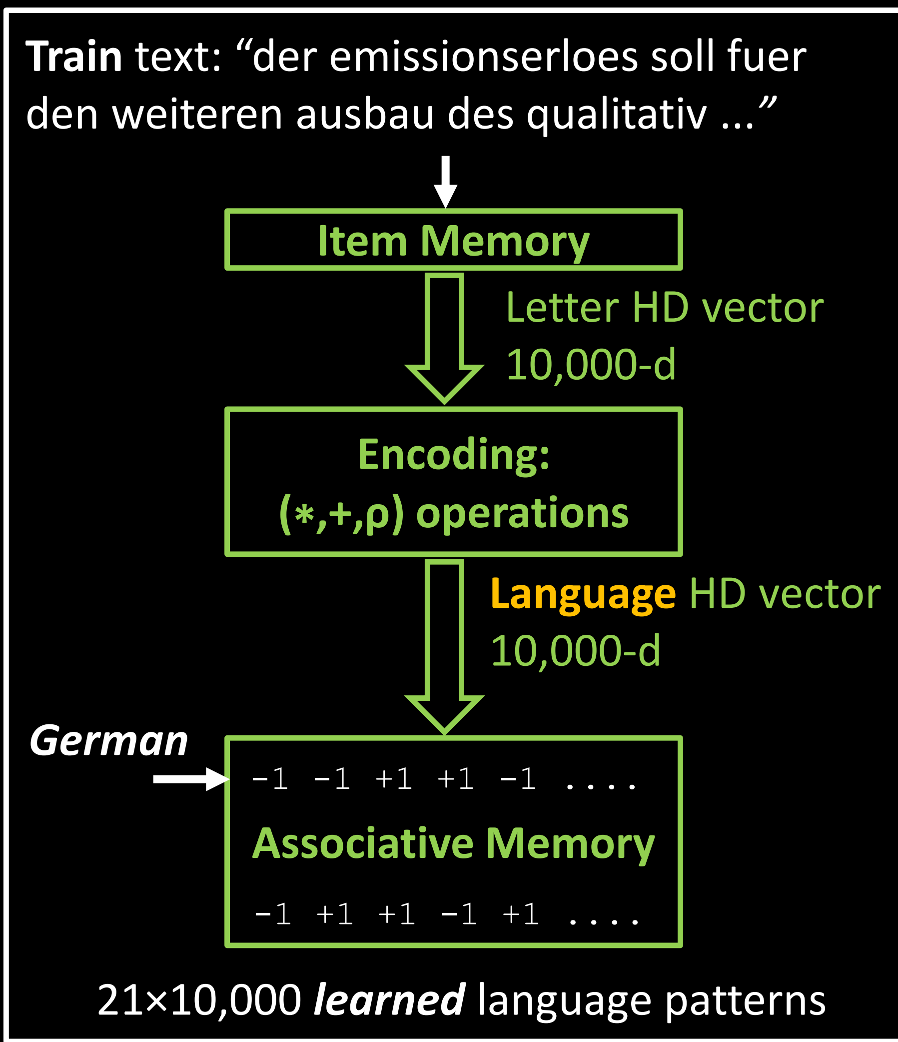


21×10,000 ***learned*** language patterns

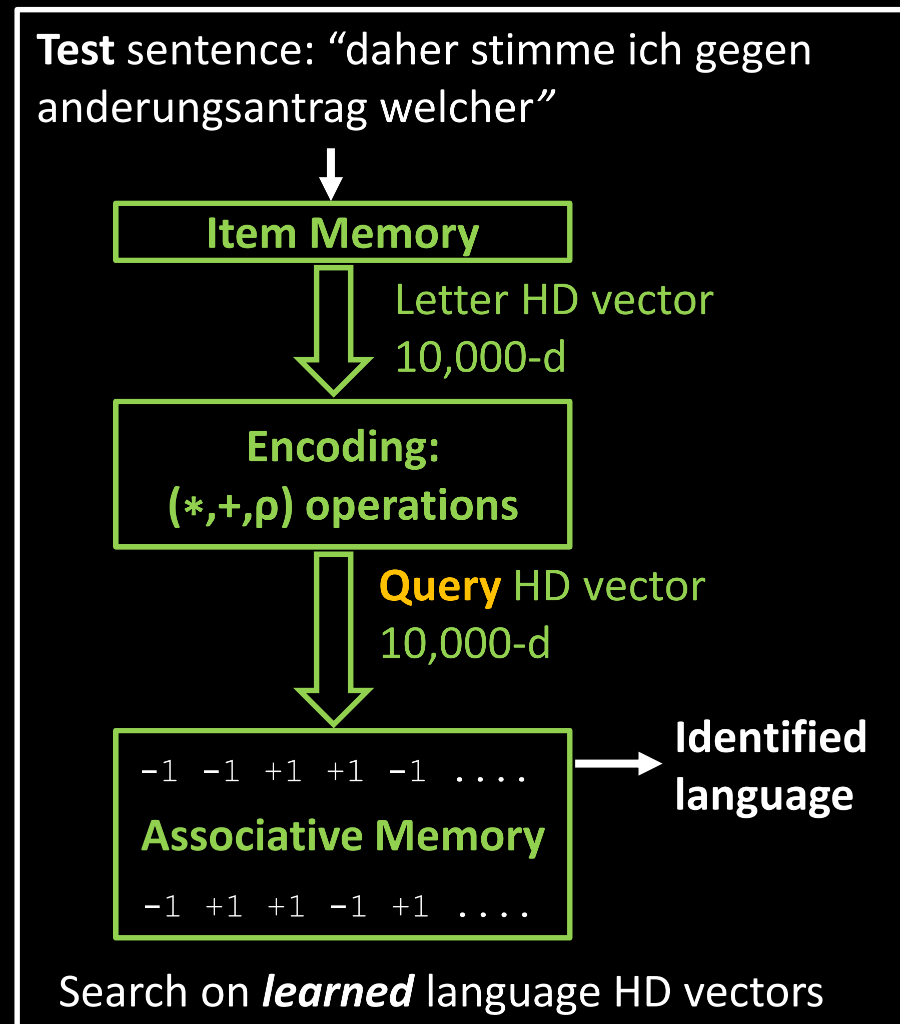
EU Language Recognition

Identical hardware for both learning and inference

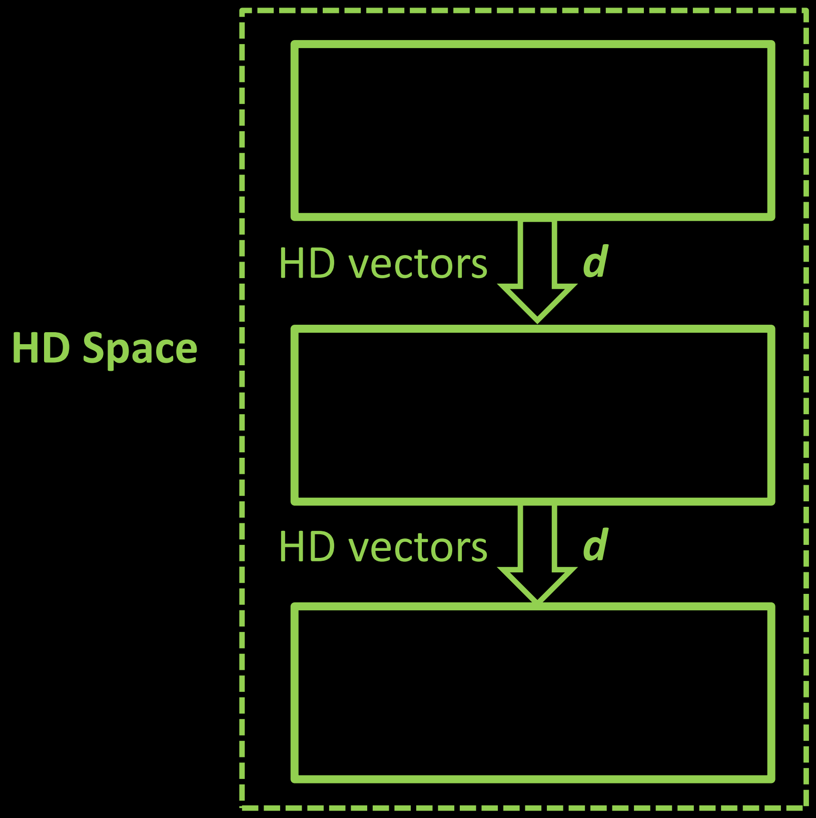
Train with 100 KB of text from 21 EU languages



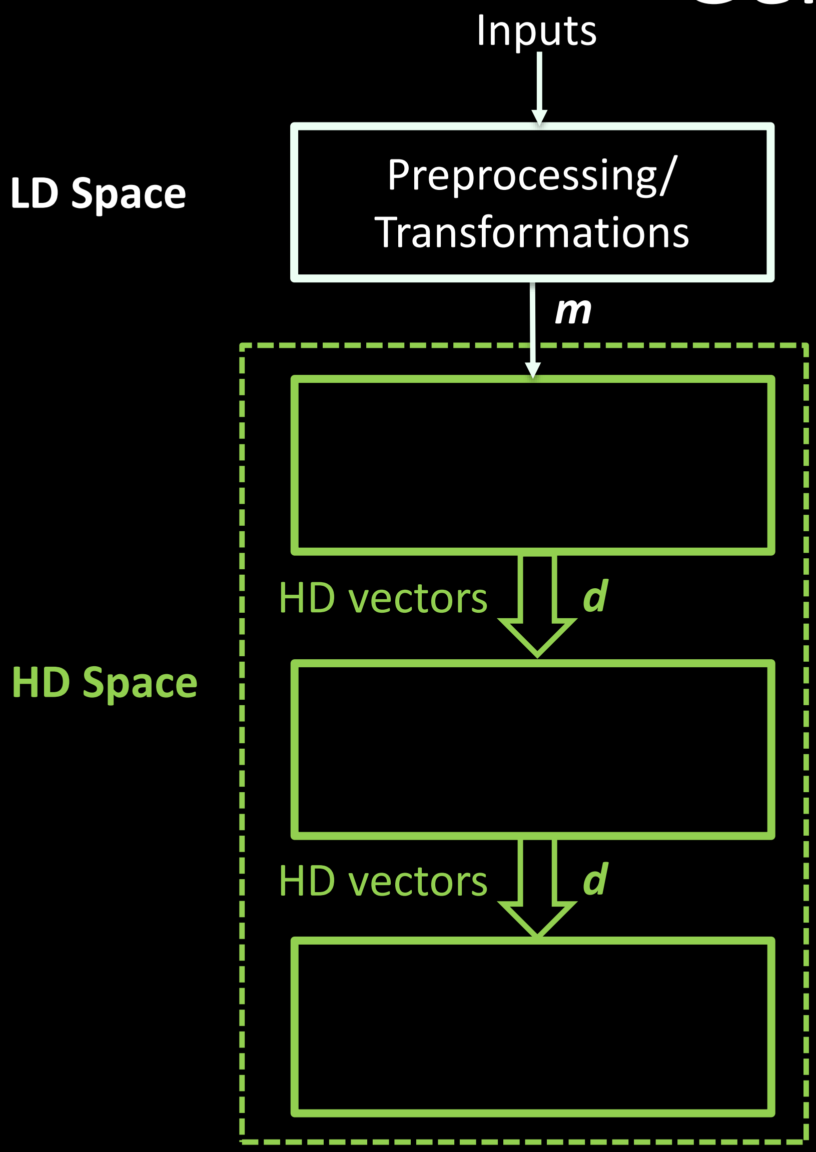
Test with 1,000 sentences for each language



Generic HD Processing Unit

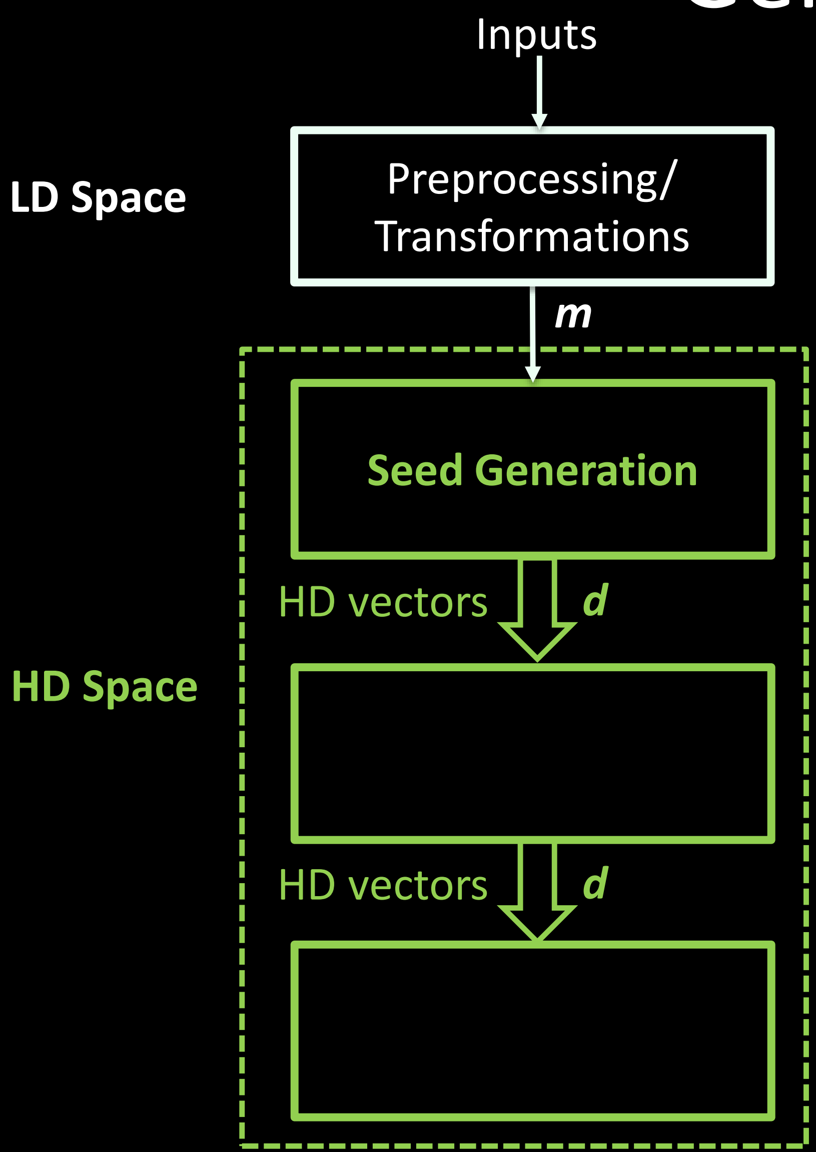


Generic HD Processing Unit



Part of preprocessing can be eliminated

Generic HD Processing Unit

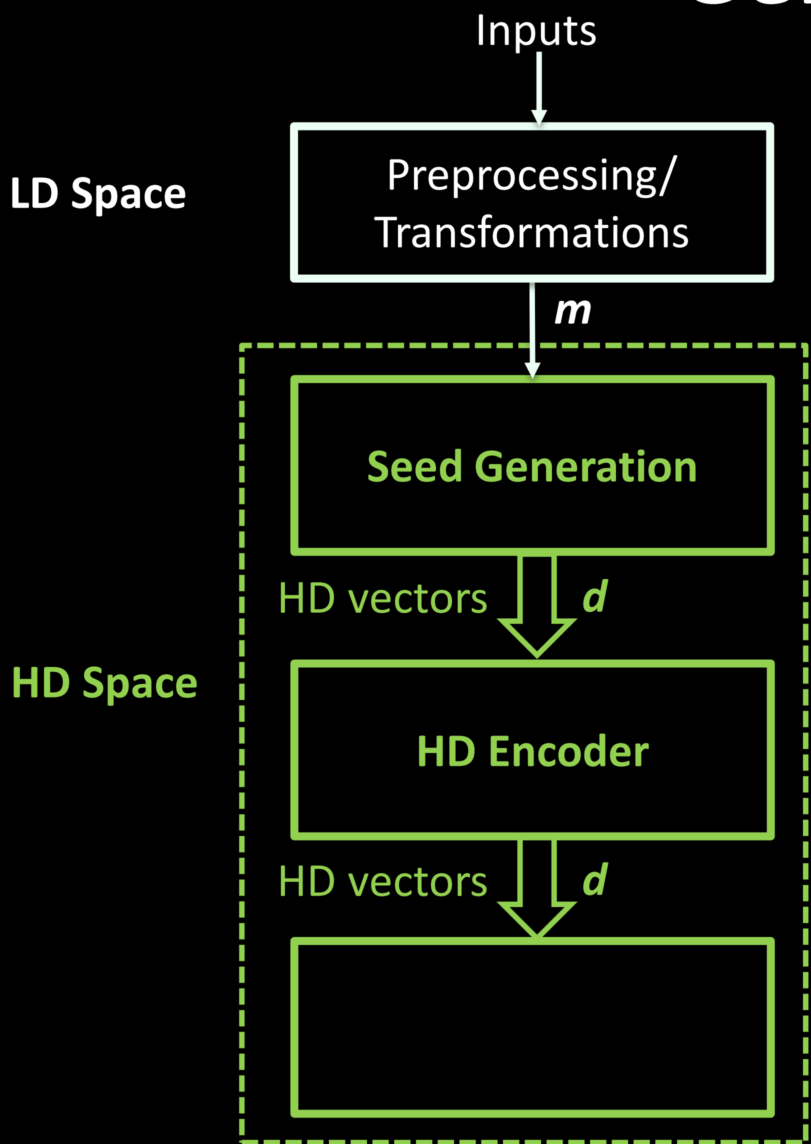


Part of preprocessing can be eliminated

Maps input vectors into d-dimensional (**d≈10,000**) pseudo-orthogonal random vectors

Nanodevice opportunity to **exploit process randomness and utilize variability**

Generic HD Processing Unit



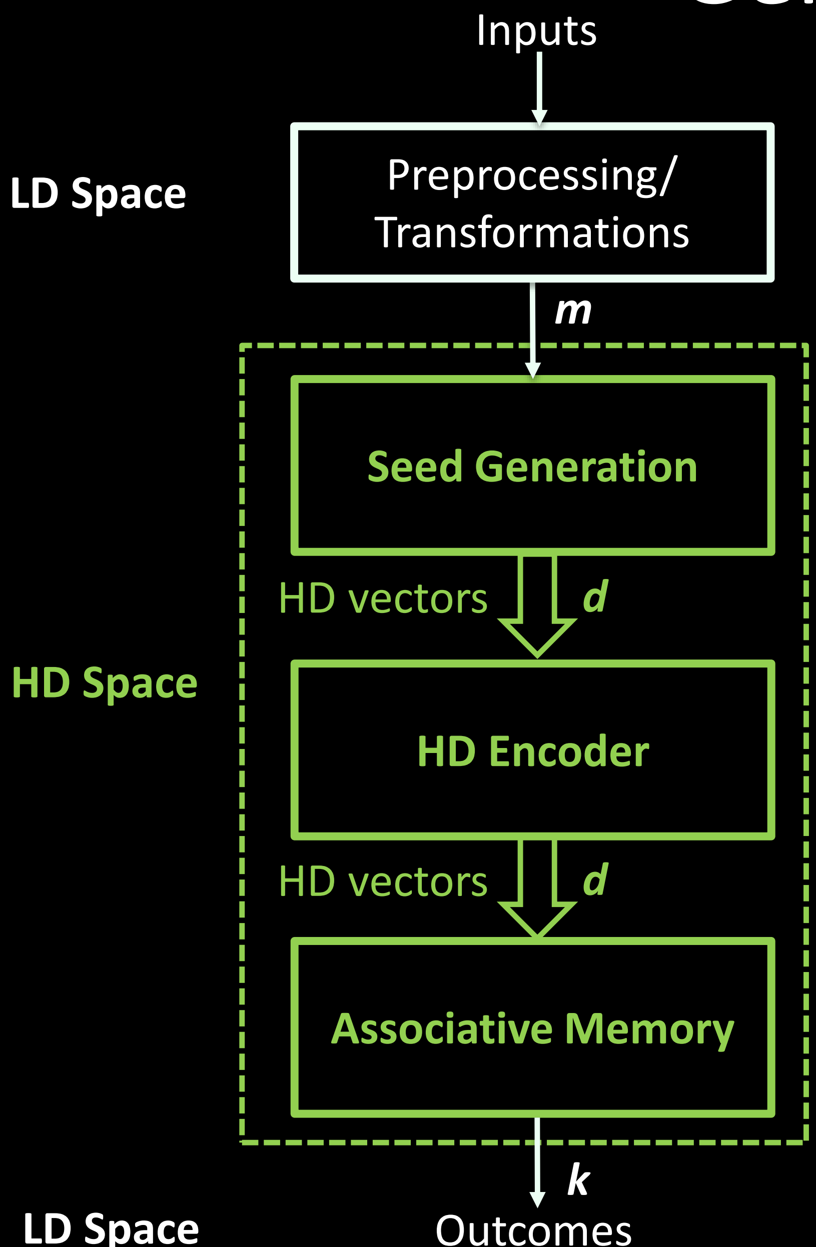
Part of preprocessing can be eliminated

Maps input vectors into d-dimensional (**d≈10,000**) pseudo-orthogonal random vectors

Nanodevice opportunity to **exploit process randomness and utilize variability**

Encodes all input information into single HD vector using simple local operators （$*$, $+$, ρ）

Generic HD Processing Unit



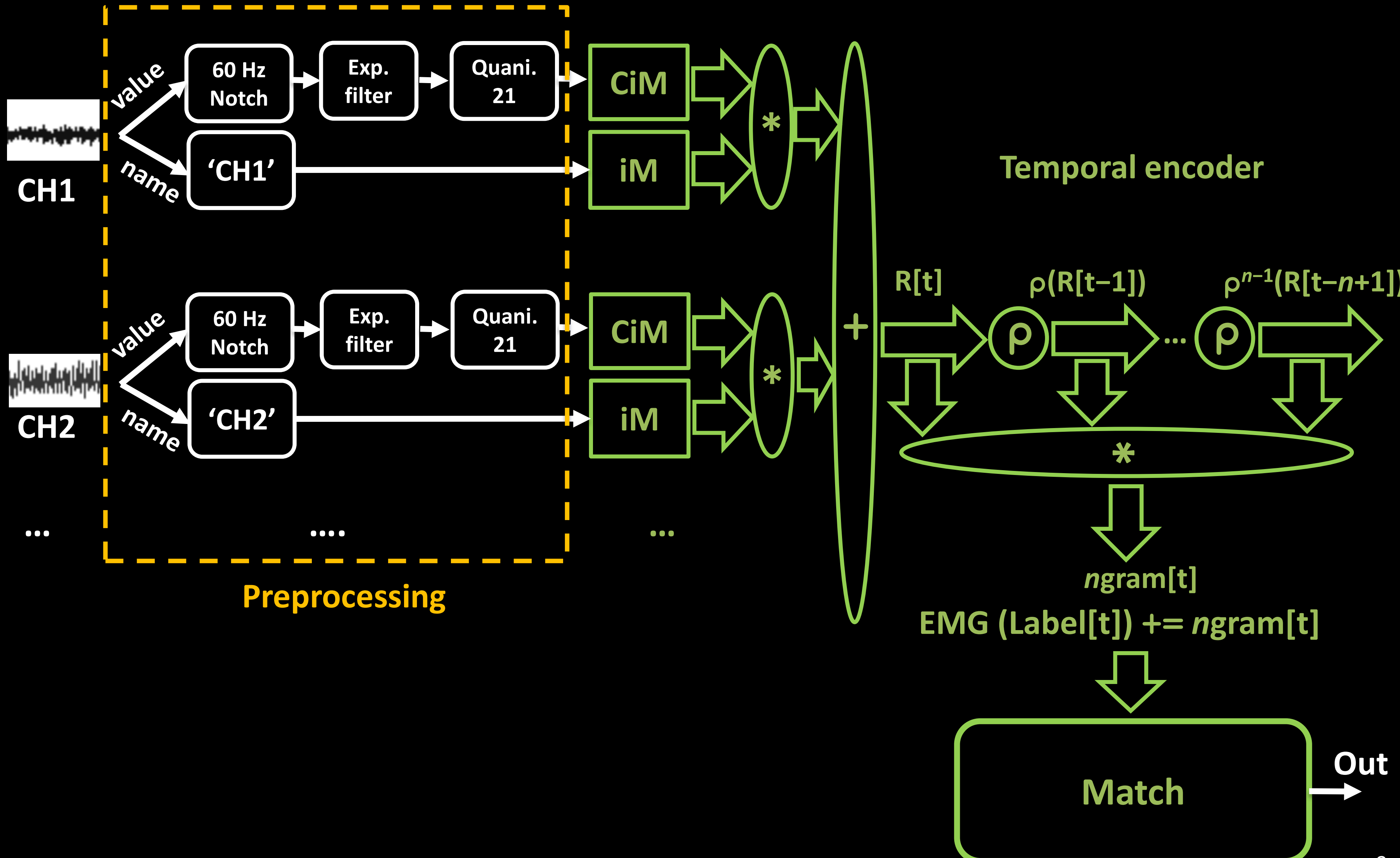
Part of preprocessing can be eliminated

Maps input vectors into d-dimensional (**d≈10,000**) pseudo-orthogonal random vectors
Nanodevice opportunity to **exploit process randomness and utilize variability**

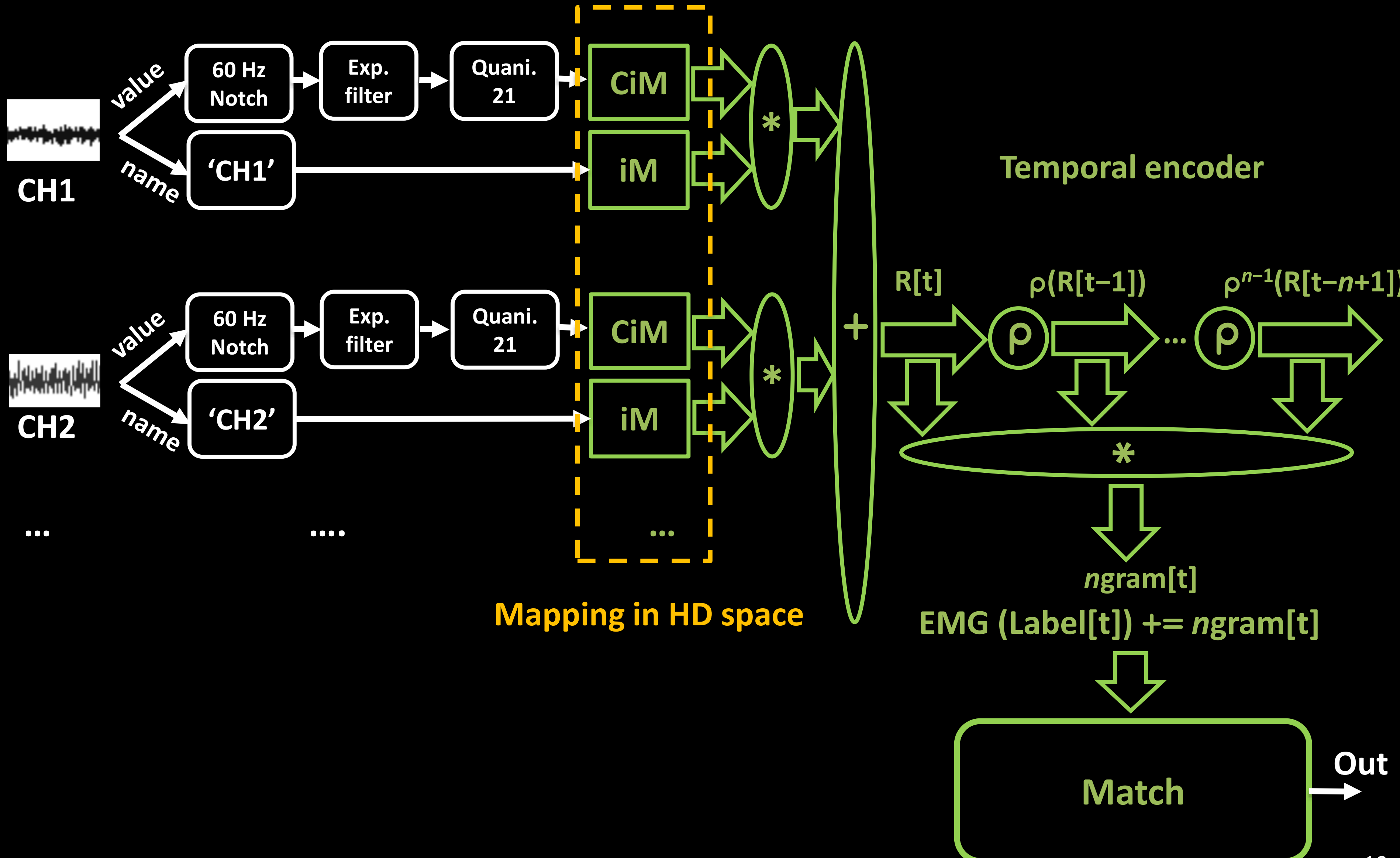
Encodes all input information into single HD vector using simple local operators （**∗**，+，**ρ**）

Finds closest match in trained data
Can be continuously updated

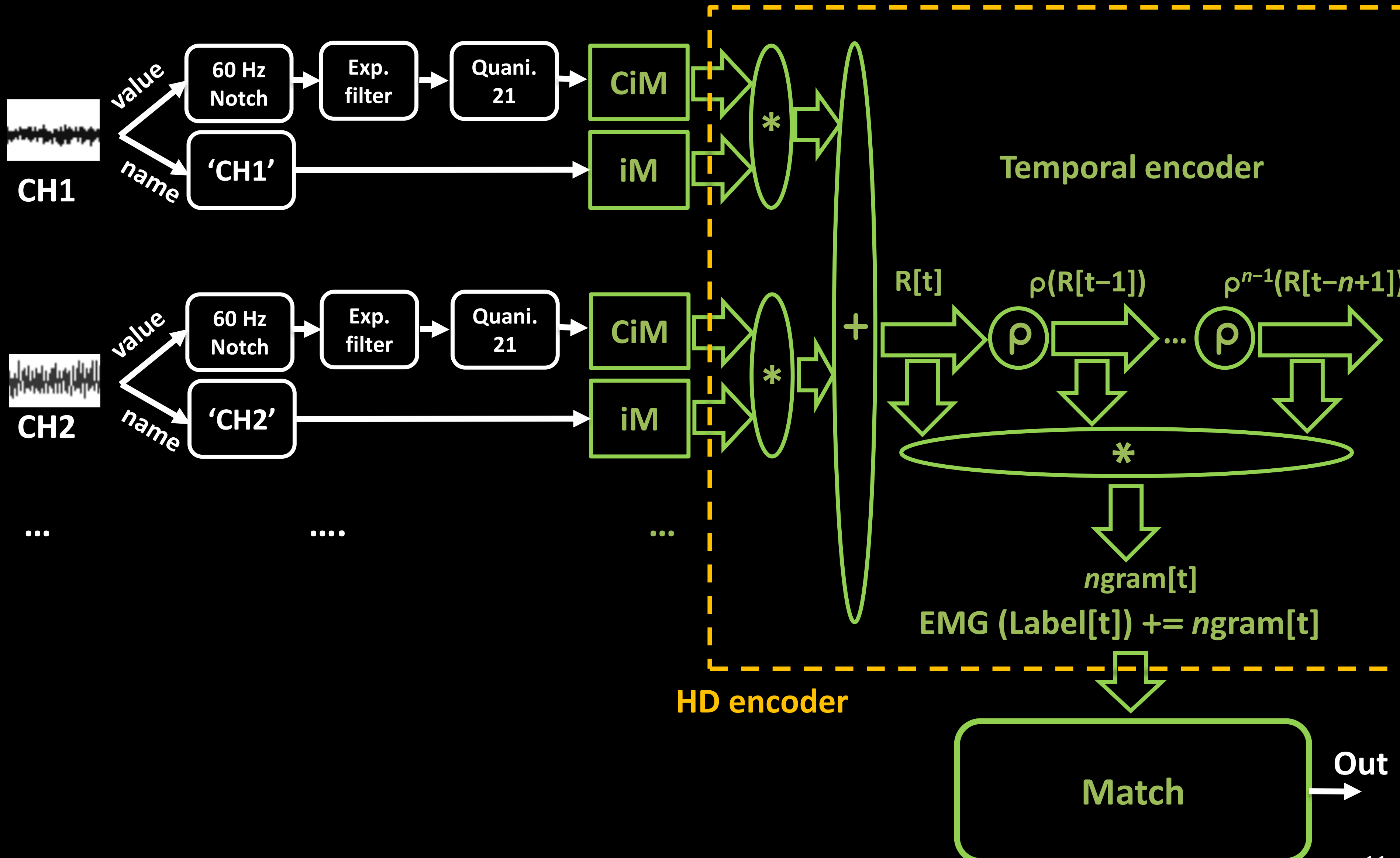
HD Processing for Gesture Recognition



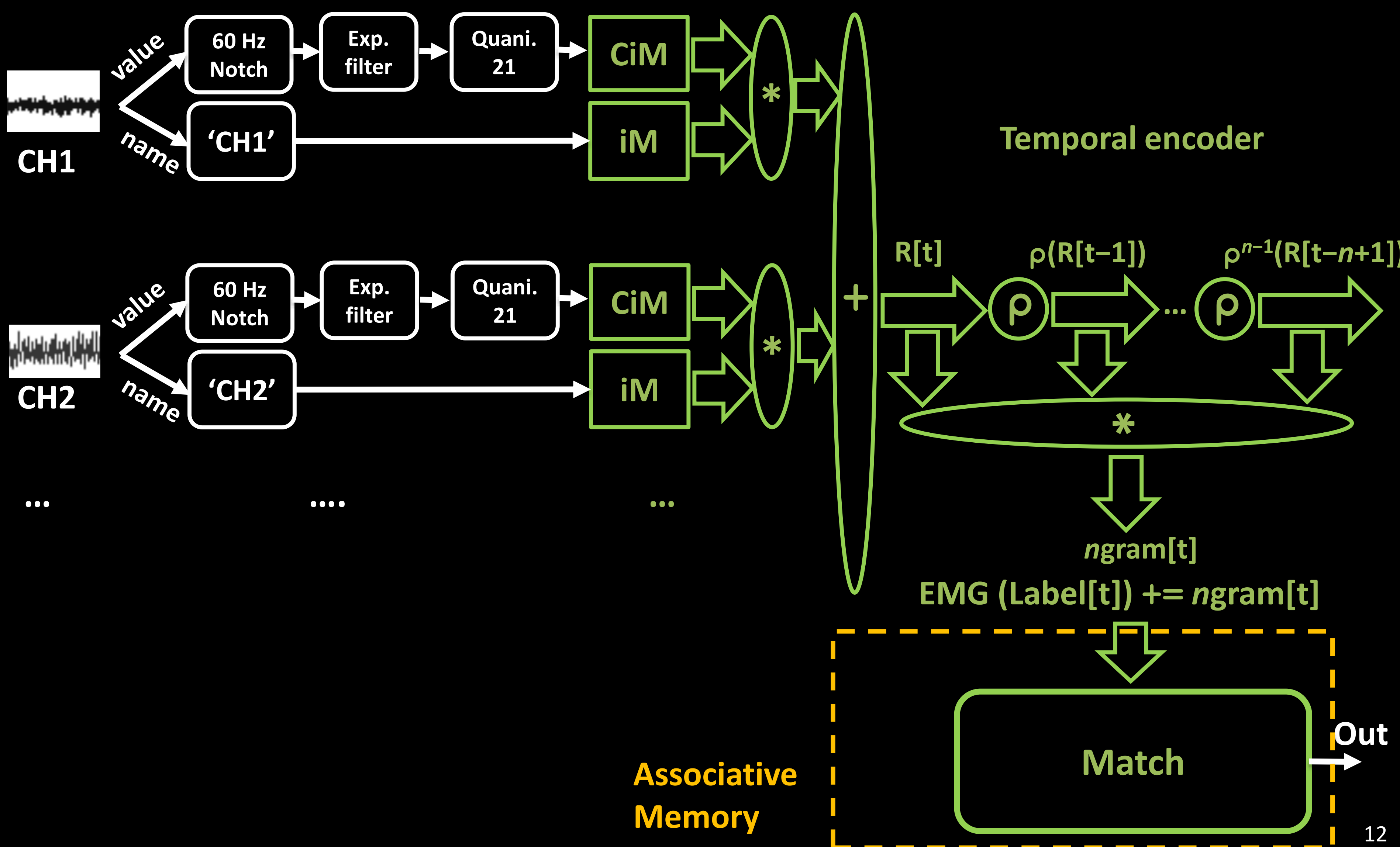
HD Processing for Gesture Recognition



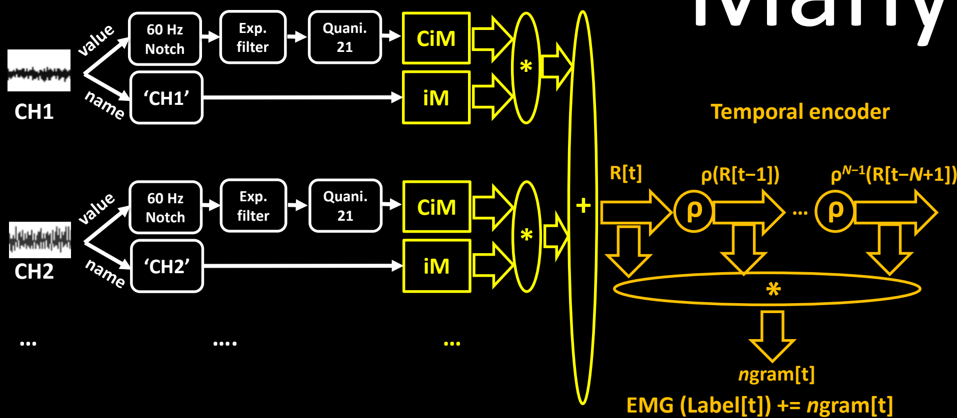
HD Processing for Gesture Recognition

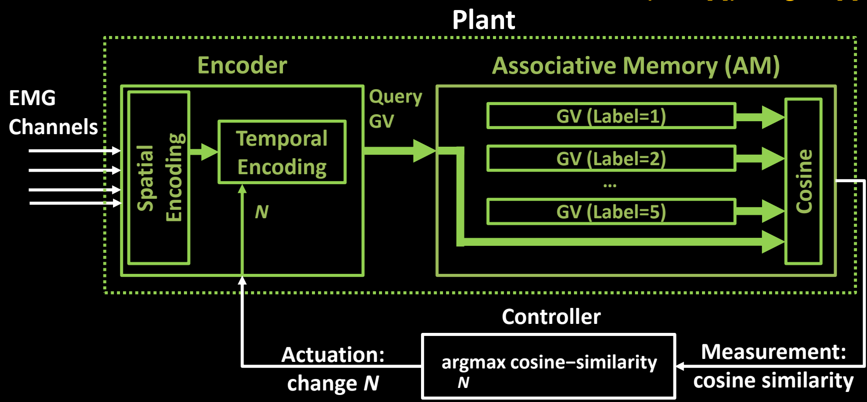


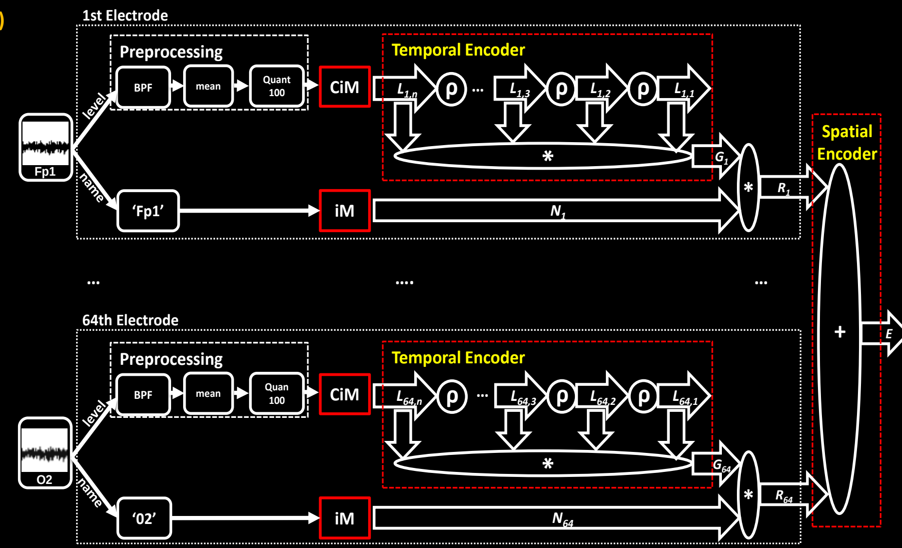
HD Processing for Gesture Recognition



Many variants of same …

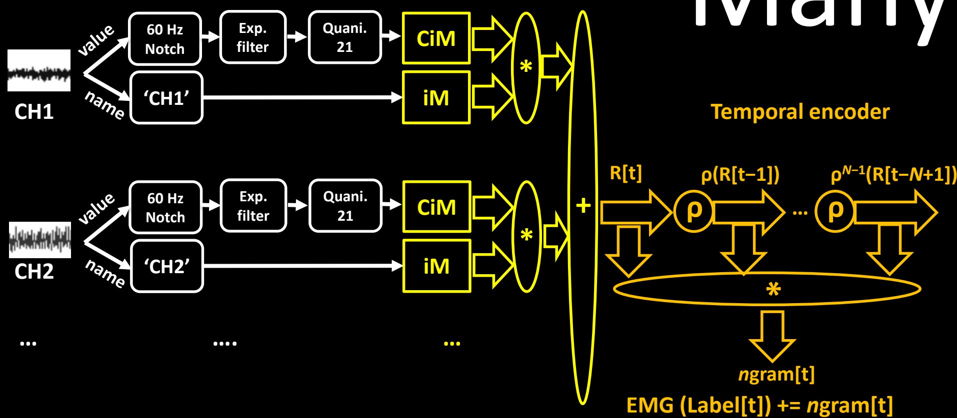


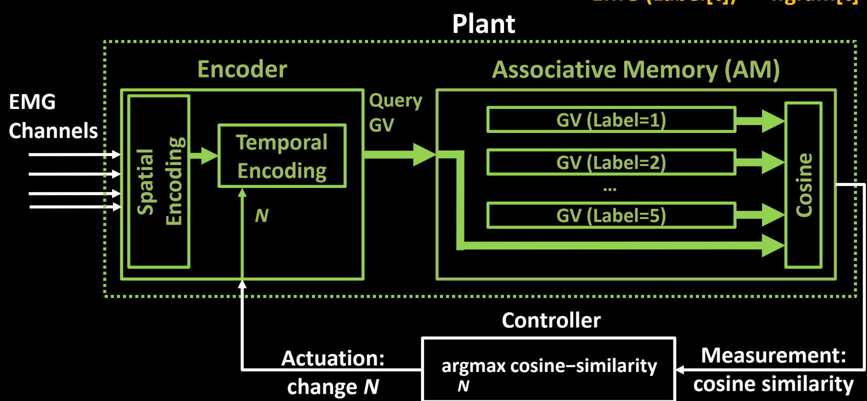


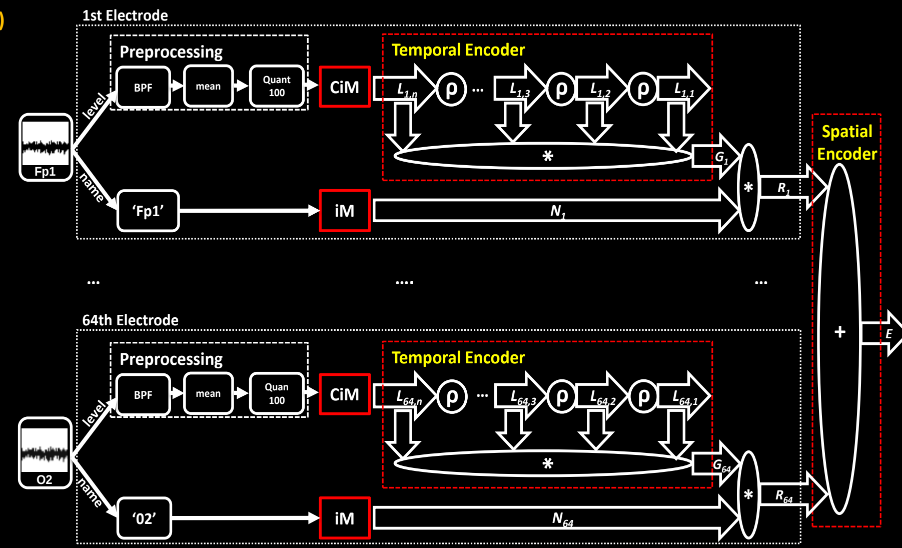


Applications	#I	#C	HD	Baseline
Language identification [ISLPED'16]	1	21	96.7%	97.9%
Text categorization [DATE'16]	1	8	94.2%	86.4%
Speech recognition [ICRC'17]	1	**26**	95.3%	93.6%
EMG gesture recognition [ICRC'16]	4	5	97.8%	89.7%
Flexible EMG [ISCAS'18]	**64**	5	96.6%	88.9%
EEG brain-machine interface [BICT'17]	**64**	2	74.5%	69.5%

Many variants of same …

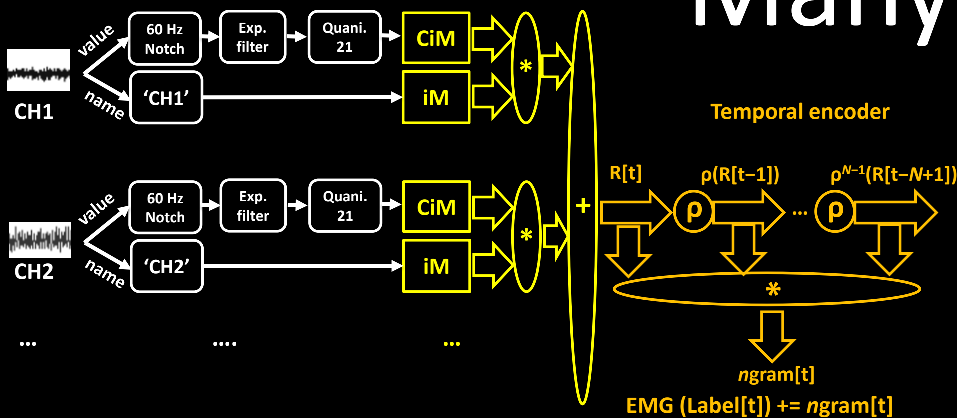


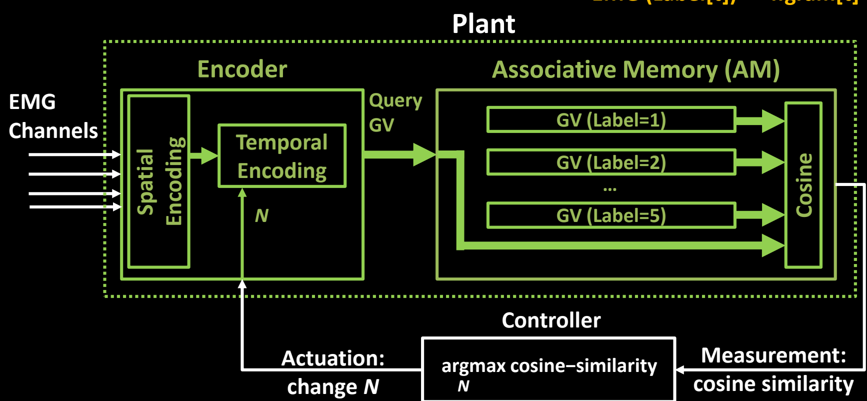


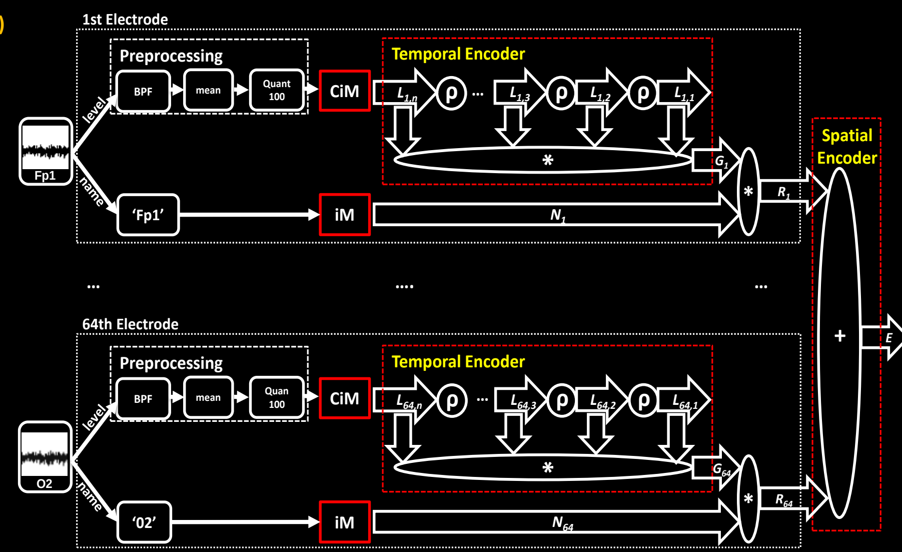


Applications	#I	#C	HD	Baseline
Language identification [ISLPED'16]	1	21	96.7%	97.9%
Text categorization [DATE'16]	1	8	94.2%	86.4%
Speech recognition [ICRC'17]	1	**26**	95.3%	93.6%
EMG gesture recognition [ICRC'16]	4	5	97.8%	89.7%
Flexible EMG [ISCAS'18]	**64**	5	96.6%	88.9%
EEG brain-machine interface [BICT'17]	**64**	2	74.5%	69.5%

Many variants of same …

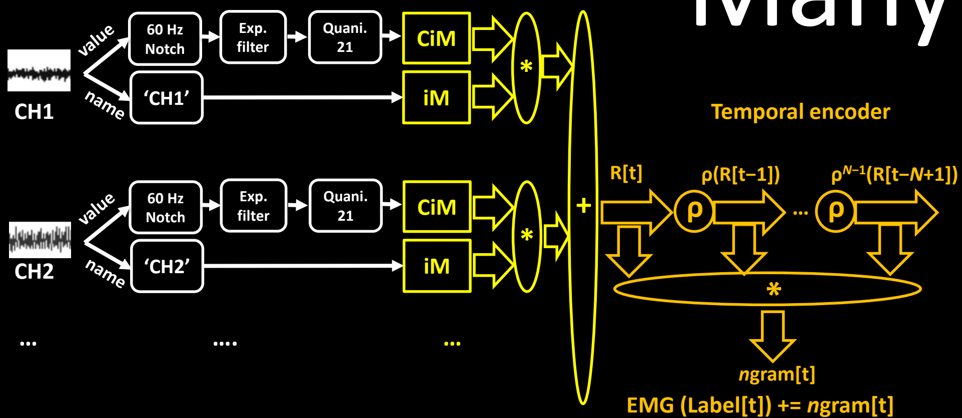


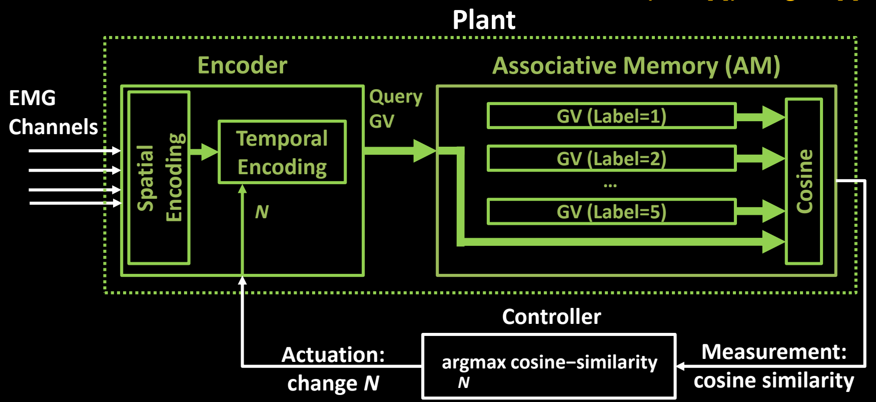


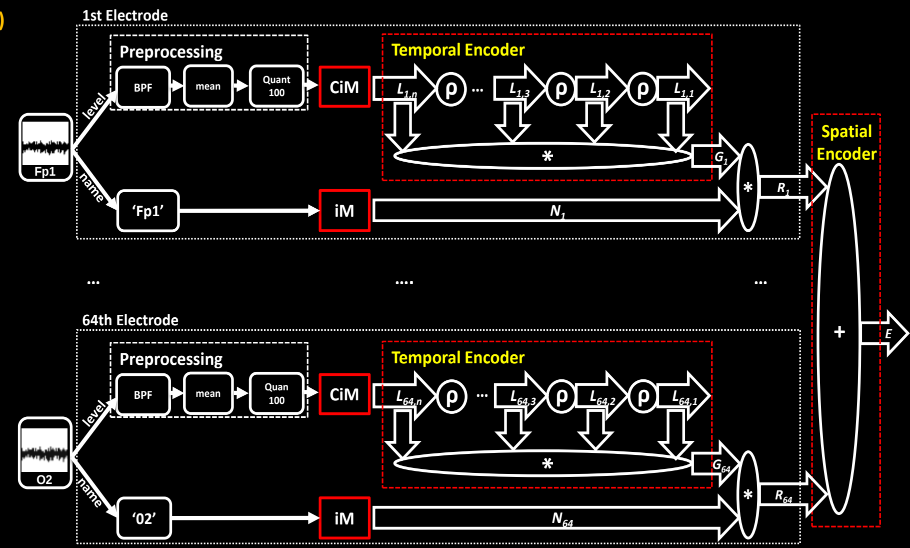


Applications	#I	#C	HD	Baseline
Language identification [ISLPED'16]	1	21	96.7%	97.9%
Text categorization [DATE'16]	1	8	94.2%	86.4%
Speech recognition [ICRC'17]	1	**26**	95.3%	93.6%
EMG gesture recognition [ICRC'16]	4	5	97.8%	89.7%
Flexible EMG [ISCAS'18]	**64**	5	96.6%	88.9%
EEG brain-machine interface [BICT'17]	**64**	2	74.5%	69.5%

Many variants of same …



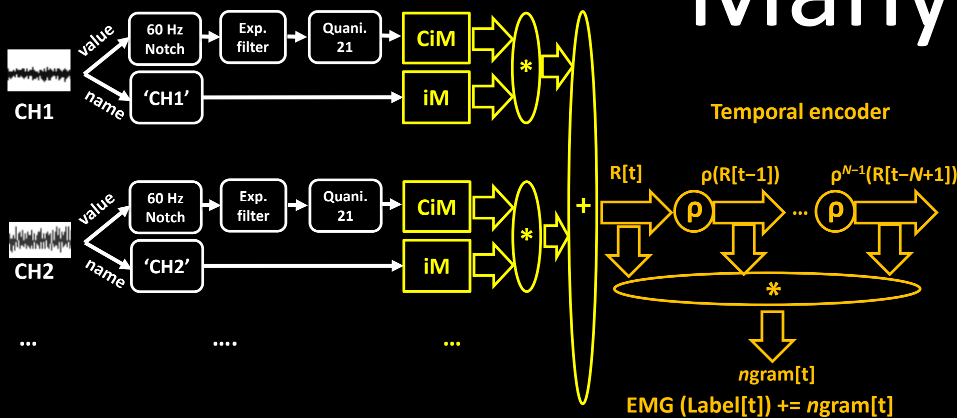


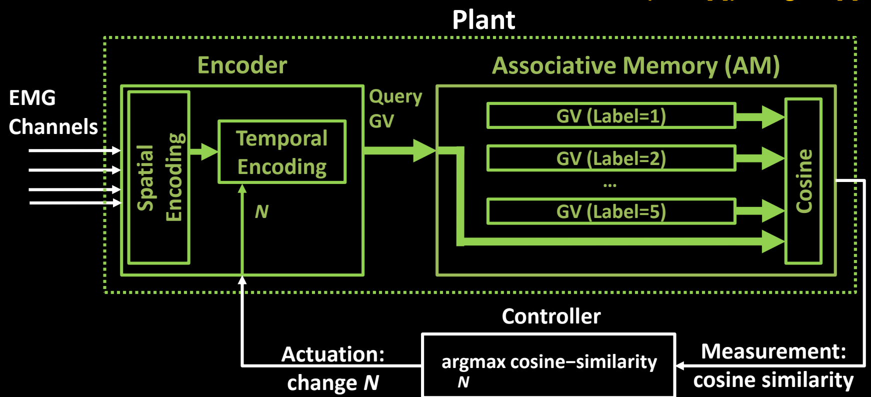


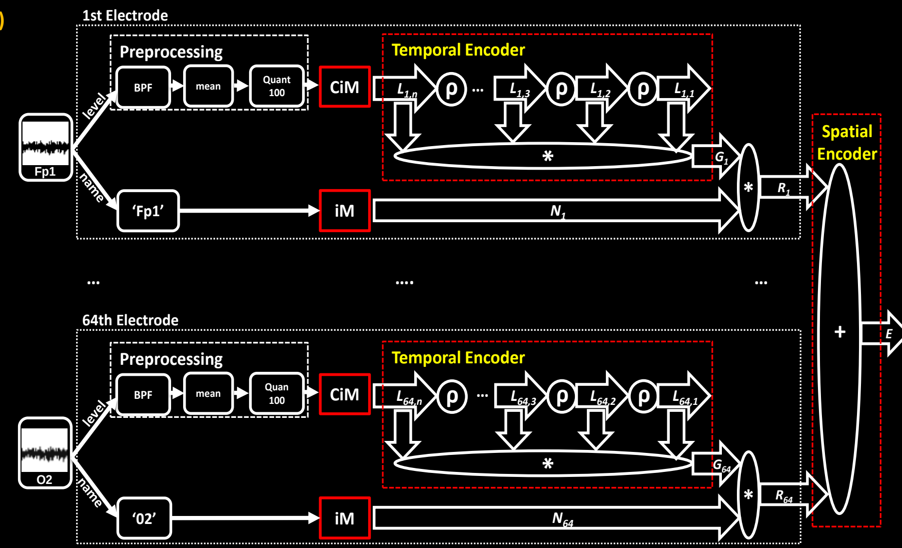
✓ 2× lower energy than SVM

Applications	#I	#C	HD	Baseline
Language identification [ISLPED'16]	1	21	96.7%	97.9%
Text categorization [DATE'16]	1	8	94.2%	86.4%
Speech recognition [ICRC'17]	1	**26**	95.3%	93.6%
EMG gesture recognition [ICRC'16]	4	5	97.8%	89.7%
Flexible EMG [ISCAS'18]	**64**	5	96.6%	88.9%
EEG brain-machine interface [BICT'17]	**64**	2	74.5%	69.5%

Many variants of same …







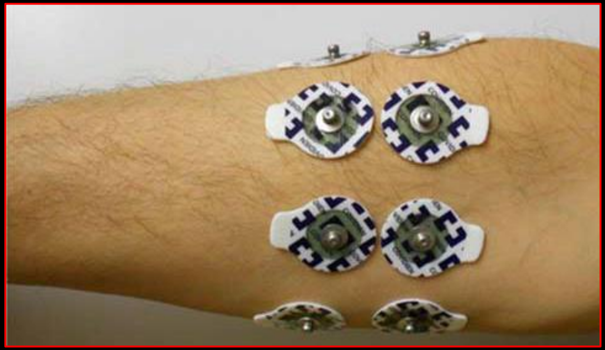
Applications	#I	#C	HD	Baseline
Language identification [ISLPED'16]	1	21	96.7%	97.9%
Text categorization [DATE'16]	1	8	94.2%	86.4%
Speech recognition [ICRC'17]	1	**26**	95.3%	93.6%
EMG gesture recognition [ICRC'16]	4	5	97.8%	89.7%
Flexible EMG [ISCAS'18]	**64**	5	96.6%	88.9%
EEG brain-machine interface [BICT'17]	**64**	2	74.5%	69.5%

- ✓ 2× lower energy than SVM

Embedded Accelerator (28nm, 1.5mm², 2mW) [DAC'18]

- ✓ **256 channels: 10 ms real-time constraints**
- ✓ 10× lower energy than ARM Cortex-M4

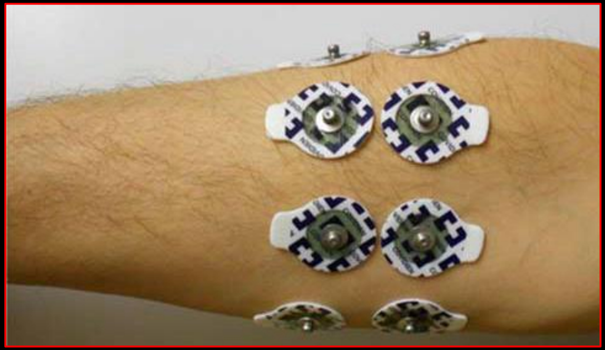
HD Learns 3× Faster



EMG (8 gel-based electrodes)

SVM needs **3.2×** more trials [ICRC'16]

HD Learns 3× Faster



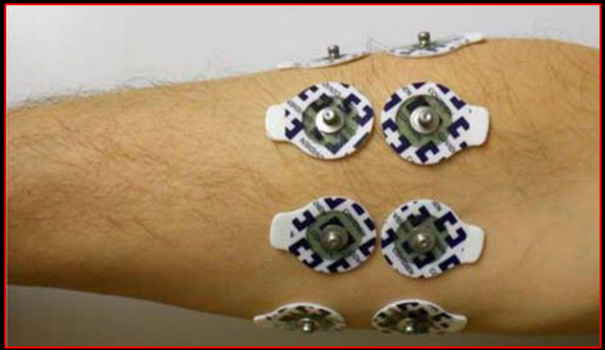
EMG (8 gel-based electrodes)

SVM needs **3.2×** more trials [ICRC'16]

EEG (64 electrodes)

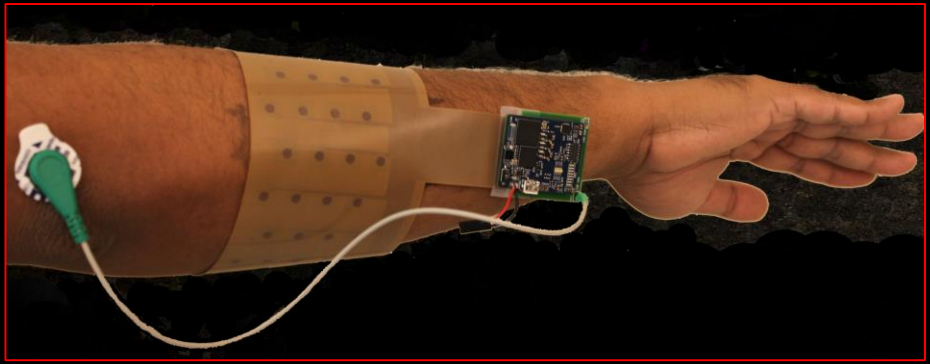
Gaussian needs **3×** more trials and preprocessing [BICT'17]

HD Learns 3× Faster



EMG (8 gel-based electrodes)

SVM needs **3.2×** more trials [ICRC'16]

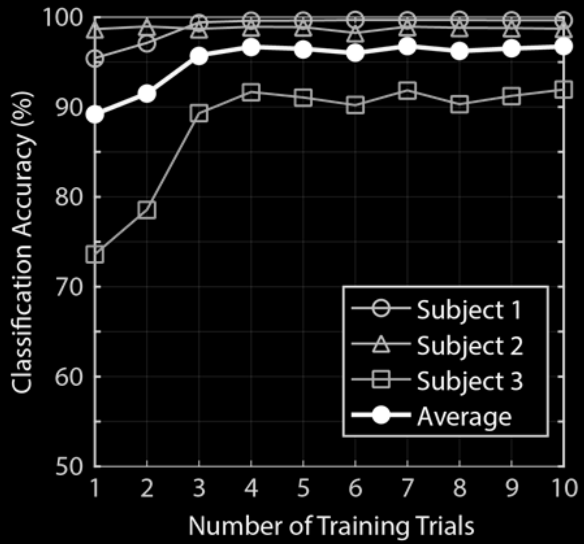


Flexible high-density electrode array [ISCAS'18]

EEG (64 electrodes)

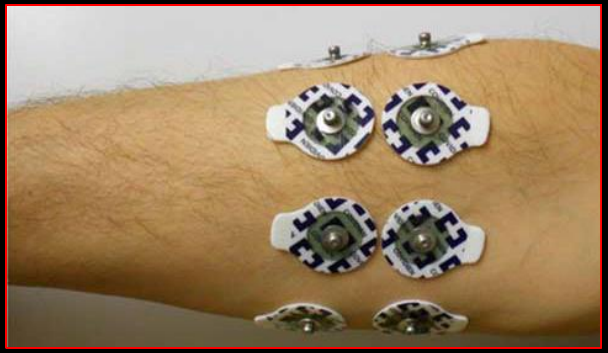
Gaussian needs **3×** more trials and preprocessing [BICT'17]

Train/Test **30 mins** apart



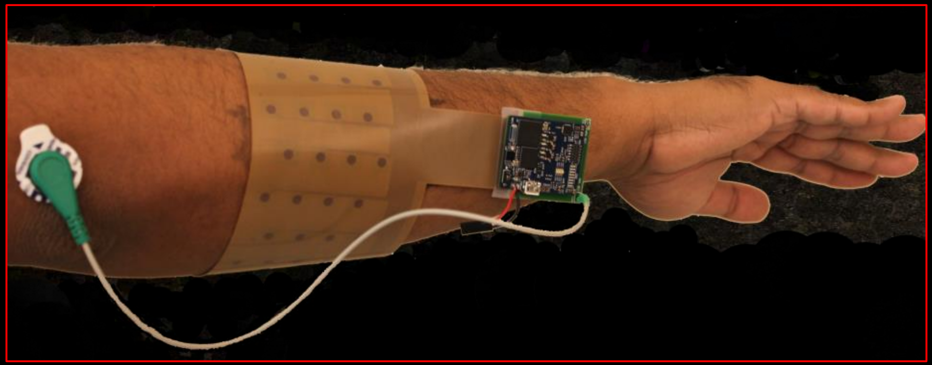
One-shot: 89.2%

HD Learns 3× Faster



EMG (8 gel-based electrodes)

SVM needs **3.2×** more trials [ICRC'16]

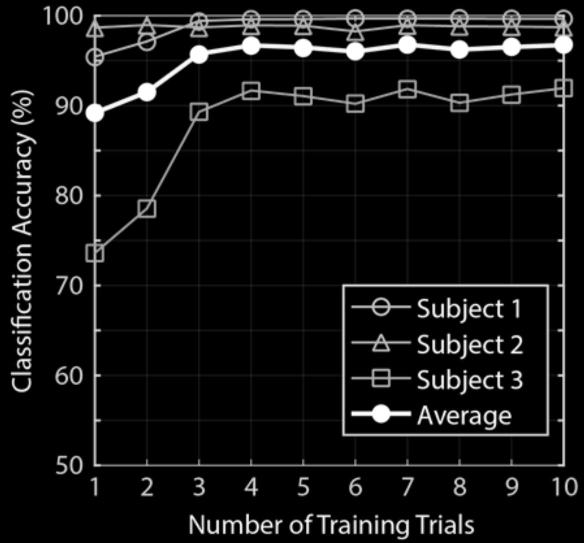


Flexible high-density electrode array [ISCAS'18]

EEG (64 electrodes)

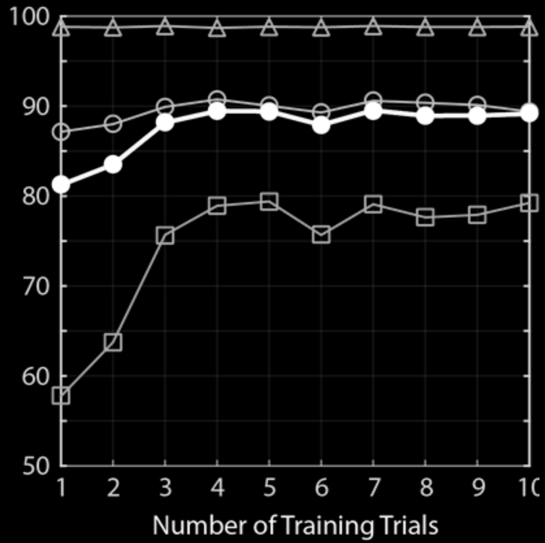
Gaussian needs **3×** more trials and preprocessing [BICT'17]

Train/Test **30 mins** apart



One-shot: 89.2%

Train/Test **1 day** apart

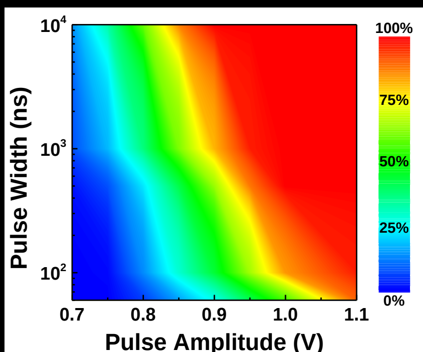


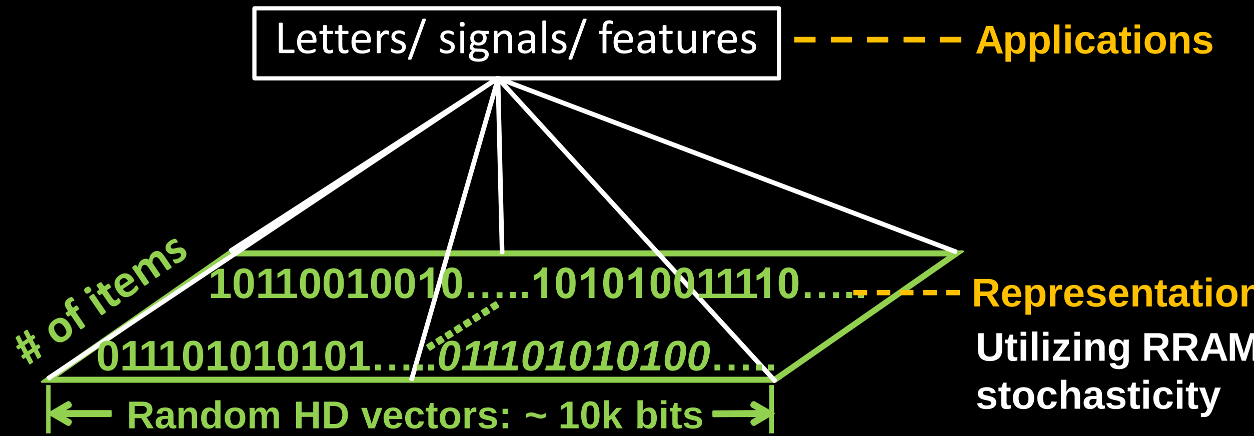
With repositioning: 82.%

SVM: 51%

The True Opportunity for HD

3D Integration and Nanoscale Devices

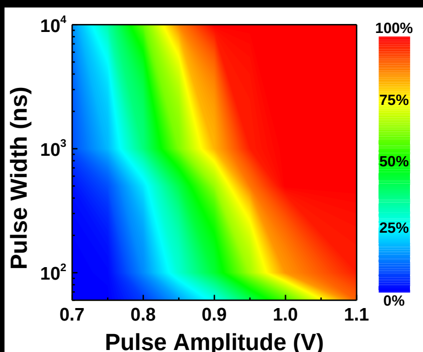


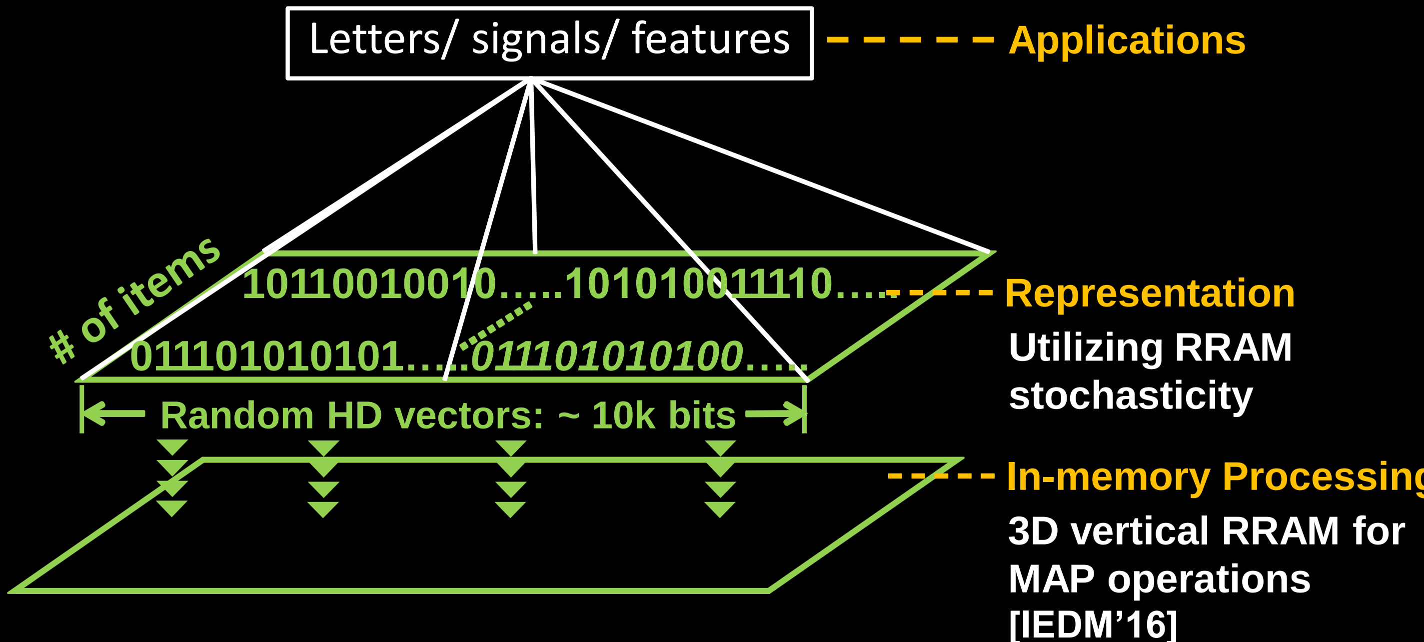


Utilizing RRAM stochasticity

The True Opportunity for HD

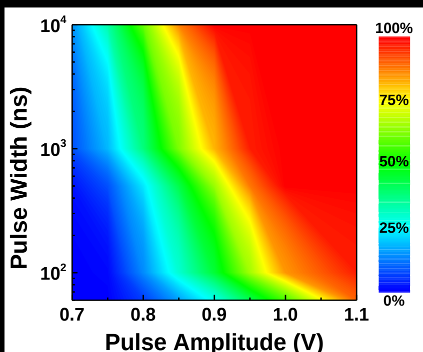
3D Integration and Nanoscale Devices

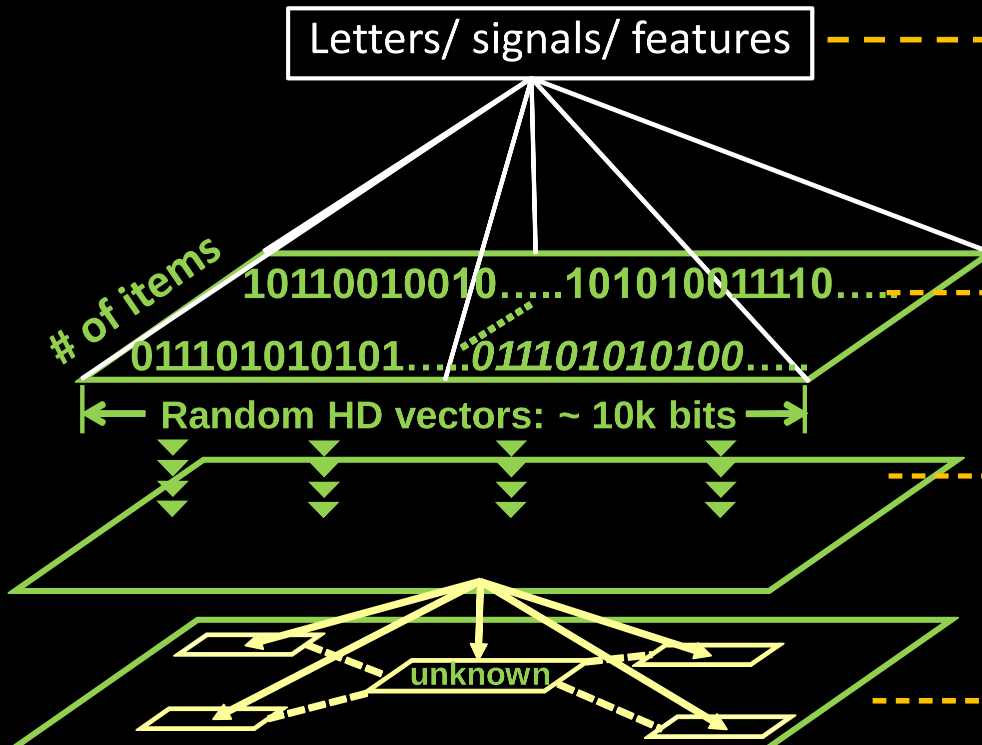




The True Opportunity for HD

3D Integration and Nanoscale Devices





Applications

Representation

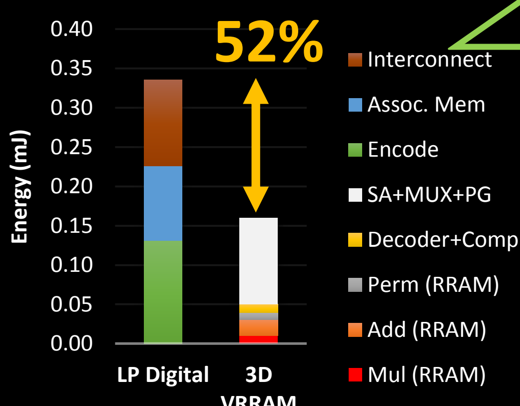
Utilizing RRAM stochasticity

In-memory Processing

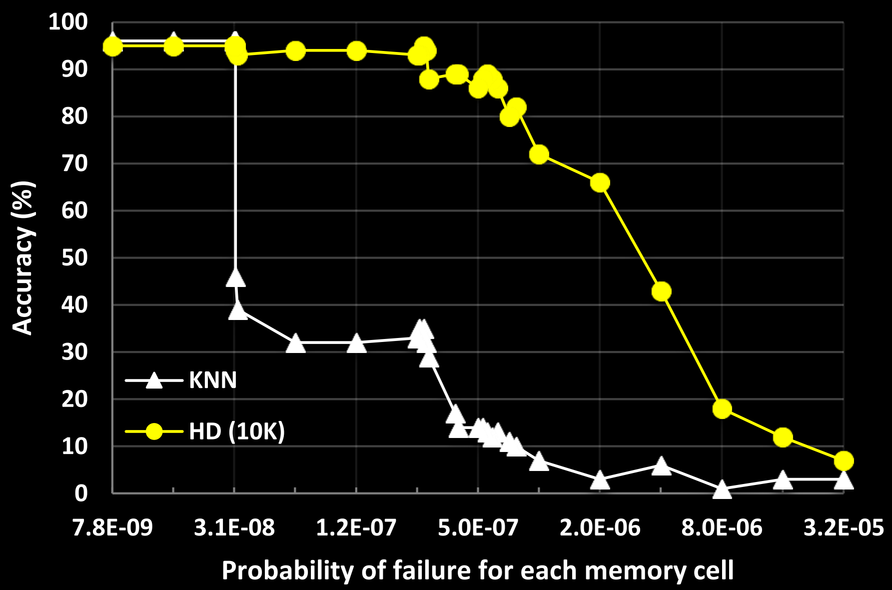
3D vertical RRAM for MAP operations [IEDM'16]

Learning/ Inference

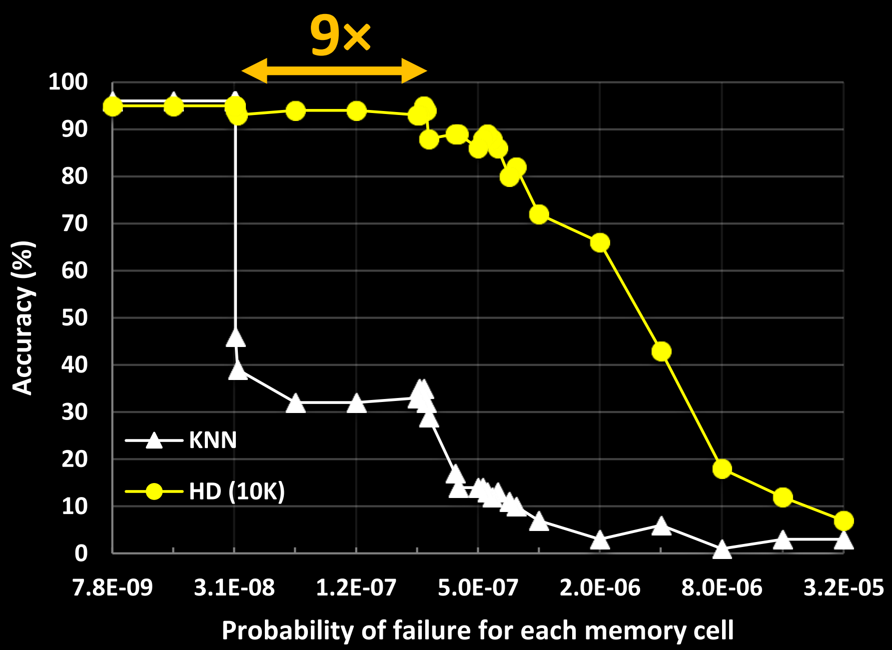
Analog resistive associative memory [HPCA'17]



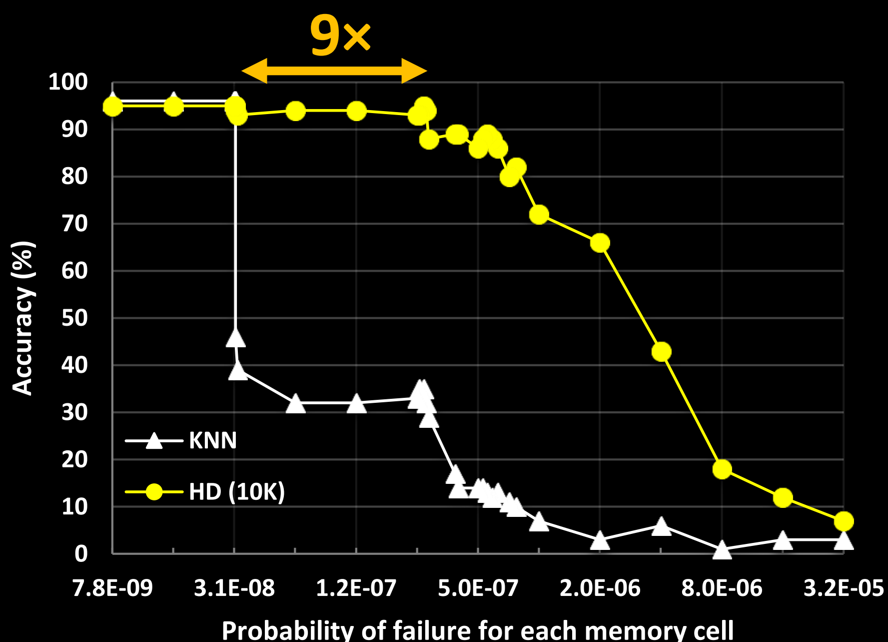
HD is Extremely Robust Against Errors

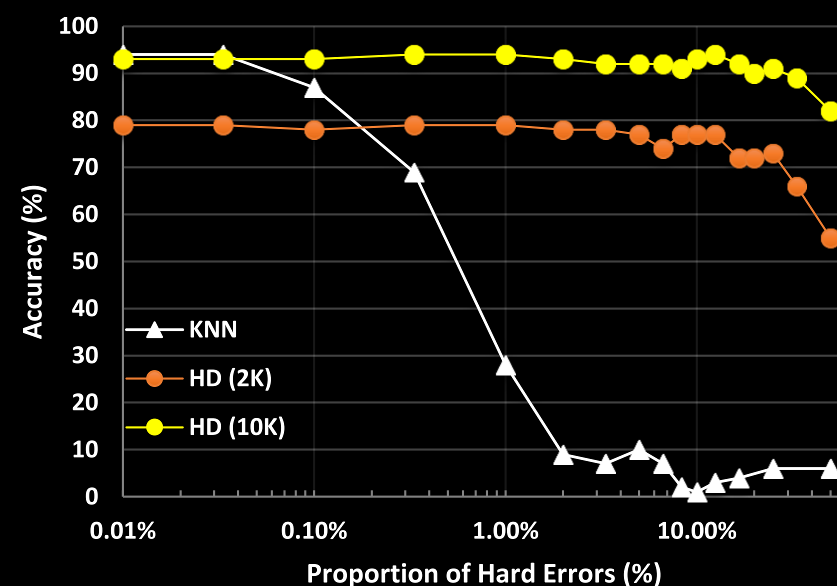


HD is Extremely Robust Against Errors

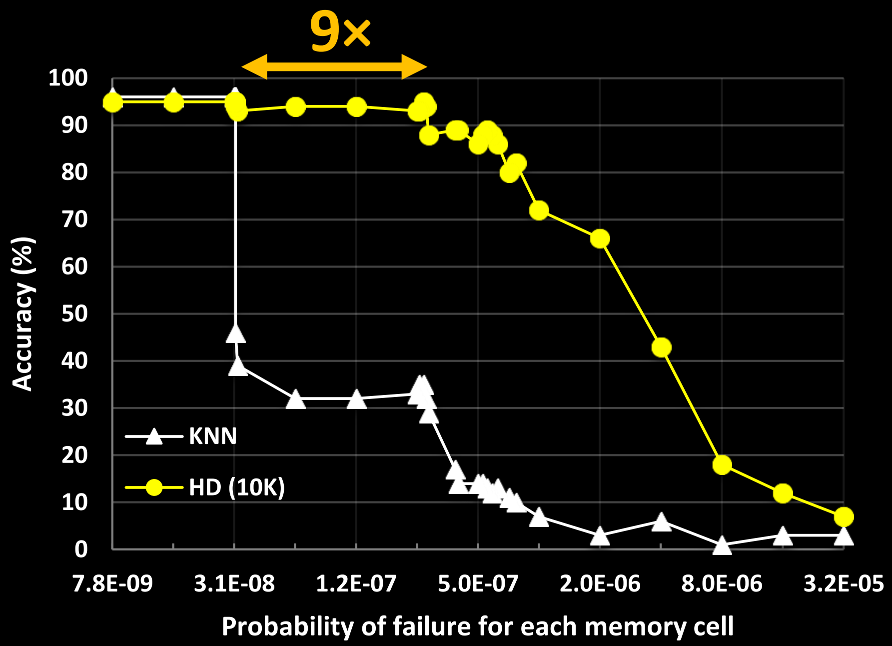


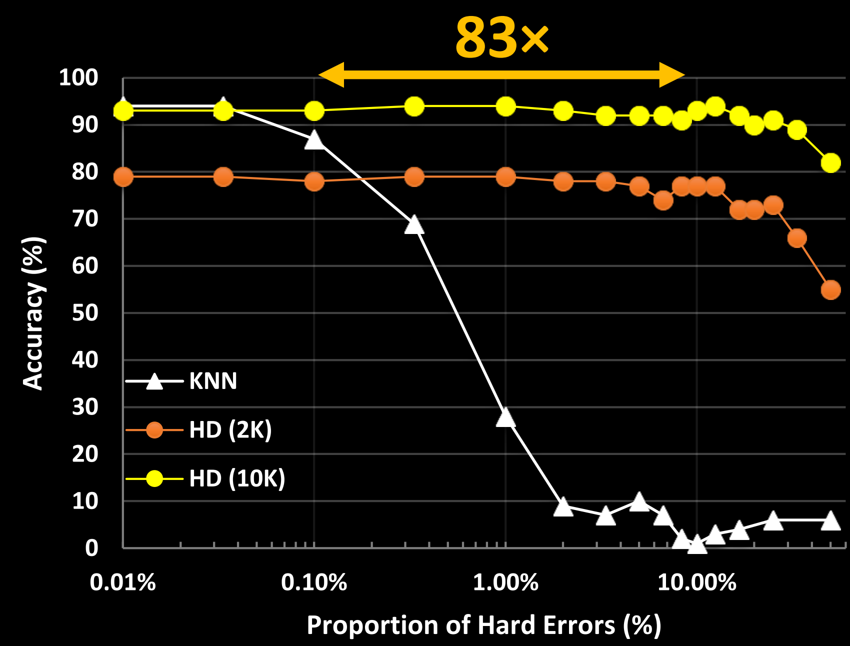
HD is Extremely Robust Against Errors



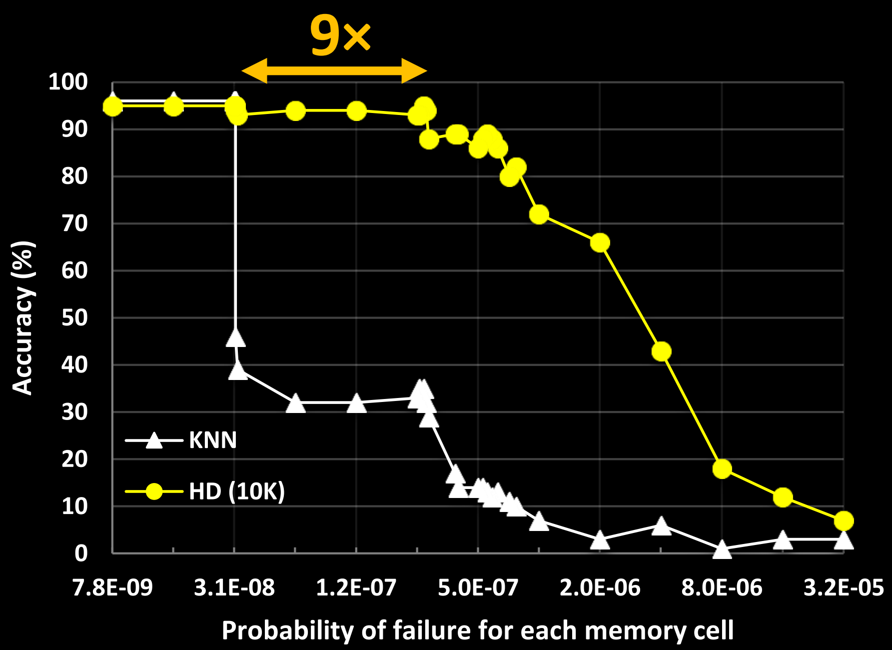


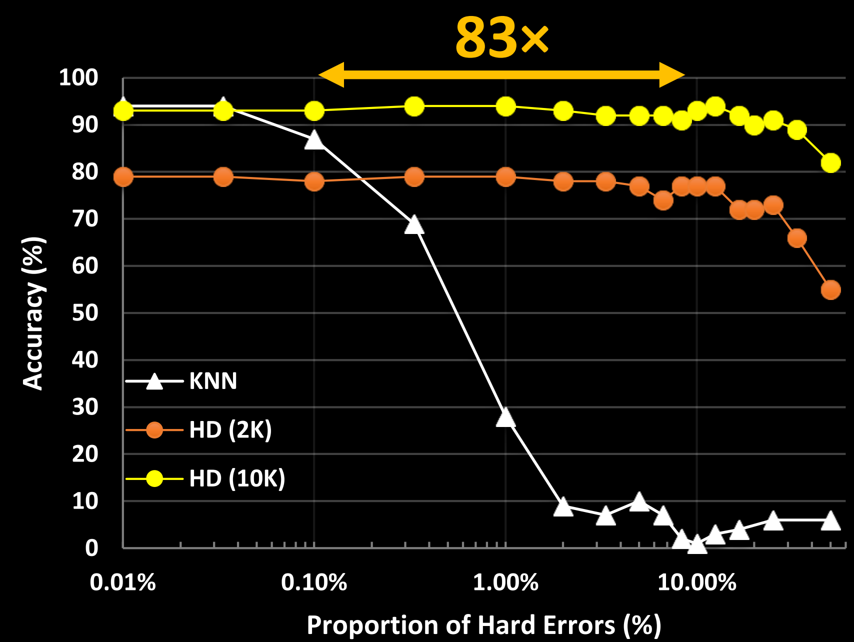
HD is Extremely Robust Against Errors





HD is Extremely Robust Against Errors





Robustness in low SNR:

- Seed hypervectors with **i.i.d. components**
- MAP operations are nearly **i.i.d.-preserving**
- **Holographic:** a failure in a component is not "contagious"
- HD algorithm is **data-driven** with **(almost) no control flow** conditions

High Order Bits

- Simple HD architectural templates to encode analog input signals for various biosignal applications
- Fully scalable
- Identical hardware for leaning and inference
- Fast learning under low SNR conditions
  - Enabling online and continuous learning!

Relevant publications

- A. Rahimi, A. Tchouprina, P. Kanerva, J. del R. . Millan, J. M. Rabaey, **"Hyperdimensional Computing for Blind and One-Shot Classification of EEG Error-Related Potentials,"** In *ACM/Springer Mobile Networks & Applications (MONET)*, Special Issue on Biologically Inspired Networking, 2017. [PDF]
- A. Rahimi, S. Datta, D. Kleyko, E. P. Frady, B. Olshausen, P. Kanerva, J. M. Rabaey, **"High-dimensional Computing as a Nanoscalable Paradigm,"** In *IEEE Transactions on Circuits and Systems (TCAS-I)*, 2017. [PDF]
- A. Rahimi, P. Kanerva, J. del R. Millan, J. M. Rabaey, "**Hyperdimensional Computing for Noninvasive Brain-Computer Interfaces: Blind and One-Shot Classification of EEG Error-Related Potentials,"** In *10th EAI International Conference on Bio-inspired Information and Communications Technologies (BICT)*, March 2017. **[Best Paper]** [PDF] [PPTX][Artifact]
- M. Imani, D. Kong, A. Rahimi, T. Rosing, "**VoiceHD: Hyperdimensional Computing for efficient Speech Recognition,"** In *IEEE International Conference on Rebooting Computing (ICRC)*, 2017. [PDF]
- A. Rahimi, S. Benatti, P. Kanerva, L. Benini, and J. M. Rabaey, **"Hyperdimensional Biosignal Processing: A Case Study for EMG-based Hand Gesture Recognition,"** In *IEEE International Conference on Rebooting Computing (ICRC)*, 2016. [PDF] [PPTX] [Artifact] [Video]
- A. Rahimi, P. Kanerva, and J. M. Rabaey, **"A Robust and Energy-Efficient Classifier Using Brain-Inspired Hyperdimensional Computing,"** In *ACM/IEEE International Symposium on Low-Power Electronics and Design (ISLPED)*, 2016. [PDF] [PPTX] [Artifact]
- H. Li, T. Wu, A. Rahimi, K.-S. Li, M. Rusch, C.-H. Lin, J.-L. Hsu, M. Sabry, S. Burc Eryilmaz, J. Sohn, W.-C. Chiu, M.-C. Chen, T.-T. Wu, J.-M. Shieh, W.-K. Yeh, J. M. Rabaey, S. Mitra and H.-S. P. Wong, **"Hyperdimensional Computing with 3D VRRAM In-Memory Kernels: Device-Architecture Co-Design for Energy-Efficient, Error-Resilient Language Recognition,"** In *IEEE International Electron Devices Meeting (IEDM)*, 2016. [PDF] [PPTX]
- F. R. Najafabadi, **A. Rahimi**, P. Kanerva, and J. M. Rabaey, **"Hyperdimensional Computing for Text Classification,"** In *ACM/IEEE Design, Automation, and Test in Europe Conference (DATE), University Booth*, March 2016. [PDF] [Poster]
- A. Moin, A. Zhou, A. Rahimi, S. Benatti, A. Menon, S. Tamakloe, J. Ting, N. Yamamoto, Y. Khan, F. Burghardt, L. Benini, A. C. Arias, J. M. Rabaey, **"An EMG Gesture Recognition System with Flexible High-Density Sensors and Brain-Inspired High-Dimensional Classifier,"** In *IEEE International Symposium on Circuits and Systems (ISCAS), 2018.* [PDF]
- F. Montagna, A. Rahimi, S. Benatti, D. Rossi, L. Benini, **"PULP-HD: Accelerating Brain-Inspired High-Dimensional Computing on a Parallel Ultra-Low Power Platform,"** *IEEE/ACM Design Automation Conference (DAC),* 2018.