Brain Informatics

RESEARCH Open Access

Efficient emotion recognition using hyperdimensional computing with combinatorial channel encoding and cellular automata

Alisha Menon*, Anirudh Natarajan, Reva Agashe, Daniel Sun, Melvin Aristio, Harrison Liew, Yakun Sophia Shao and Jan M. Rabaey*

Abstract

In this paper, a hardware-optimized approach to emotion recognition based on the efficient brain-inspired hyperdimensional computing (HDC) paradigm is proposed. Emotion recognition provides valuable information for human–computer interactions; however, the large number of input channels (> 200) and modalities (> 3) involved in emotion recognition are significantly expensive from a memory perspective. To address this, methods for memory reduction and optimization are proposed, including a novel approach that takes advantage of the combinatorial nature of the encoding process, and an elementary cellular automaton. HDC with early sensor fusion is implemented alongside the proposed techniques achieving two-class multi-modal classification accuracies of > 76% for valence and > 73% for arousal on the multi-modal AMIGOS and DEAP data sets, almost always better than state of the art. The required vector storage is seamlessly reduced by 98% and the frequency of vector requests by at least 1/5. The results demonstrate the potential of efficient hyperdimensional computing for low-power, multi-channeled emotion recognition tasks.

Keywords: Brain-inspired, Hyperdimensional computing, Emotion recognition, Wearable, Memory optimization, Hardware efficient, Multi-modal sensor fusion

1 Introduction

Affective computing for informed human–computer interaction (HCI) is an area of growing research interest [1]. Traditional interfaces such as keyboards and mouse are limited to conveying explicit information; the HCI experience can be enhanced through the inclusion and interpretation of additional implicit information [2]. For example, context-dependent human behavioral patterns can be learned and used to inform feedback systems of user intention in a wide variety of applications, such as driver warning systems, smart environments, automated tutoring systems, etc. [2–4]. Providing computers with emotional skills will allow them to intelligently react to subtle user context changes, such as emotional state [5].

Though a common approach is interpreting audio–visual signals such as facial expressions and voices, these may not be the primary source of expression. Emotion is not always easily observable, rather it requires a combination of various behavioral observations and physiological indices that together can provide sufficient information [6]. Existing data sets collected for affective computing include various forms of physiological signals to create a comprehensive observation of emotional state [7, 8]. In the era of Internet-of-things (IoT), advances in wearable devices make the inclusion of various sensing modalities in intelligent HCI applications increasingly feasible [9].

*Correspondence: allymenon@berkeley.edu; jan_rabaey@berkeley.edu

Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA, USA

A representation of emotion used for affective computing is the circumplex model of affect [10]. This model proposes that all affective states come from two neurophysiological systems, valence (pleasure vs. displeasure) and arousal (alertness). Discrete emotional states can be described as a linear combination of these two dimensions. Joy, for example, can be described as positive valence system activation and moderate arousal system activation [10]. Other emotions consist of different degrees of activation of the two neurophysiological systems. For emotion recognition classification, these are reduced to high and low arousal and valence values which can, in combination, be used to define the nature of the emotion.

The emotion recognition system must also be able to address the challenge of multi-modal classification which results from the inclusion of diverse physiological sensors [11]. For this work, the AMIGOS and DEAP data sets were selected specifically for the large number of sensor channels and modalities. The AMIGOS data set contains electroencephalogram (EEG), galvanic skin response (GSR) and electrocardiogram (ECG) sensors [7]. The DEAP data set includes EEG, Electrooculography (EOG), Electromyography (EMG), GSR, blood volume pressure (BVP), temperature and respiration sensors [8].

Previous work on multi-modal fusion for the AMIGOS data set includes Fisher's linear discriminant with Gaussian Naive Bayes, which was shown to achieve F1 scores of 57% and 58.5% on valence and arousal [7, 9]. Wang et al. implemented recursive feature elimination (RFE) with a support vector machine (SVM) and obtained 68% and 66.3% accuracy on valence and arousal [11]. Wang et al. also implemented Extreme Gradient Boosting (XGBoost) for accuracies of 80.1% and 68.4% on valence and arousal. Siddharth et al. used extreme learning machines (ELM) for accuracies of 83.9% and 82.7% on valence and arousal [12]. Previous binary classification multi-modal fusion approaches for the DEAP data set include a restricted boltzmann machine (RBM) with an SVM classifier, with accuracies of 60.7% and 64.6% for valence and arousal, respectively [13]. Wang et al. used a deep belief network (DBN) through multi-layer RBMs for valence and arousal accuracies of 51.2% and 68.4% [14]. Yin et al. used a multiple-fusion-layer based ensemble classifier of stacked autoencoder (MESAE) for accuracies of 76.2% and 77.2% for valence and arousal [1].

Since emotion recognition can provide valuable information for HCI, a hardware-efficient platform that allows for extended-use, on-board classification, would increase the feasibility of long-term wearable monitoring and thus increase the scope of potential feedback applications. While previous work shows strong results for the AMIGOS and DEAP data sets in terms of classification accuracy, the ease of hardware implementation for training and inference are not considered while designing the models; these methods have high computational complexity that reduces implementation feasibility on resource-limited wearable platforms. SVMs, for example, while demonstrating high accuracy, are challenging to implement efficiently on hardware, and demonstrate a trade-off between precise accuracy and meeting hardware constraints [15, 16]. In addition, multi-modal fusion approaches require parallel encoding schemes prior to the fusion point which further increase the complexity creating a bottleneck for real-time wearable classification.

To address this, in this work brain-inspired Hyperdimensional Computing (HDC) is used for emotion recognition. HDC is an area of active research that has been successfully demonstrated for classifying physiological signals such as the wearable EMG classification system implemented from Moin et al. that achieves 97.12% accuracy in recognizing 13 different hand gestures [17], the iEEG seizure detection system developed by Burrello et al. [18], and the EEG error-related potentials classification for brain–computer interfaces implemented by Rahimi et al. [19]. It is based on the idea that human brains do not perform inference tasks using scalar arithmetic, but rather manipulate large patterns of neural activity. These patterns of information are encoded in binary hypervectors, with dimensions ranging in the thousands to ensure that any two random HVs are likely to be nearly orthogonal to each other [20]. There are three operations that are performed on these hypervectors: bundling, binding, and permutation. Bundling is a componentwise add operation across input vectors, binding is a componentwise XOR operation, and permutation is a 1-bit cyclical shift. The simplicity of these operations suggests that HDC is very hardware efficient, as confirmed in previous work [16, 21]. Montagna et al. demonstrated that HDC computing achieved 2× faster execution and lower power at iso-accuracy on an ARM Cortex M4 compared to an optimized SVM [16].

HDC has additional properties that demonstrate its potential for a wearable emotion recognition system. Previous work by Chang et al. developed a baseline, unoptimized architecture for emotion classification on the AMIGOS data set, which was able to achieve valence and arousal accuracies of 83.2% and 70.1%, respectively, demonstrating higher performance than SVM, XGB and gaussian naive bayes for all amounts of training data [9]. HDC encodes information in the same form no matter the type, number or complexity of the inputs. This is accomplished through basic vectors (items) that are random, and typically stored in an item memory (a codebook). Each channel is assigned a unique item memory vector, and feature values are typically encoded through a discretized mapping to additional unique hypervectors representing values within a set range. Each stream of information is encoded into this representation as shown in Fig. 1, which lends HDC well to sensor fusion.

HDC inherently binds features extracted from various physiological data streams. This suggests early fusion with reduction of parallel encoding schemes will have little effect on its accuracy, breaking the complexity–accuracy tradeoff. HDC offers a reduction of computation and memory requirements in contrast to traditional machine learning models, demonstrated by Montagna et al. [16]. It also offers the ability to use the same hardware for training & inference, rapid training, and robustness to noise/variations in input data making it a viable choice for wearable, hardware-constrained sensor fusion applications.

Datta et al. synthesized an implementation of a generic HDC application-specific integrated circuit (ASIC) processor that provided a power breakdown between the various blocks involved [21]. The item memory, which stores channel identification vectors, contributed the most, 42%, to the overall power of the processor. For the emotion recognition-specific application, the large number of channels (> 200) and modalities (> 3) requires advance storage of a correspondingly large number of unique vectors in the item memory. More channels translates into more memory. This would result in memory storage consuming ~50% of the overall processor power similar to [21]. Reduction of memory usage would allow HDC to meet stricter power/complexity constraints, improving its potential for implementation on wearable platforms.

In this work, use of pseudo-random vector generation through computation using a cellular automata is proposed and implemented for this purpose. A cellular automata consists of a grid of cells which evolve with complex, random behavior over time through a set of discrete computations using the current state and that of nearby cells [22]. Cellular automata rule 90 assigns the next state in a method shown to be equivalent to an XOR of the two nearest cells [22]. For a hypervector, each cell represents a single bit and rule 90 can be implemented through XOR of the cyclical left-shift and cyclical right-shift of the original vector. If $\mathrm{HV_n}$ is the hypervector state at step n, and ρ is the cyclical shift operation (+1 for right, −1 for left), then $\mathrm{HV_{n+1}}$ can be generated by

$$\mathrm{HV_{n+1}} = \rho^{+1}(\mathrm{HV_n}) \oplus \rho^{-1}(\mathrm{HV_n}) \tag{1}$$

These operations are vectorizable and computationally minimal. The emotion recognition architecture uses a fixed sequential channel (item) access pattern; therefore, this technique, with which the item memory vectors are sequentially evolving, can be used. Cellular automata grid sizes over 24 have been shown to generate new degrees of freedom for more than 10^3 steps before saturating [23]. Hypervectors, with tens of thousands of cells in the grid, provide linearly longer randomization periods; this is sufficient for most applications including emotion recognition. Using a single random seed vector, full-dimension random item memory hypervectors can be generated during the encoding process instead of being

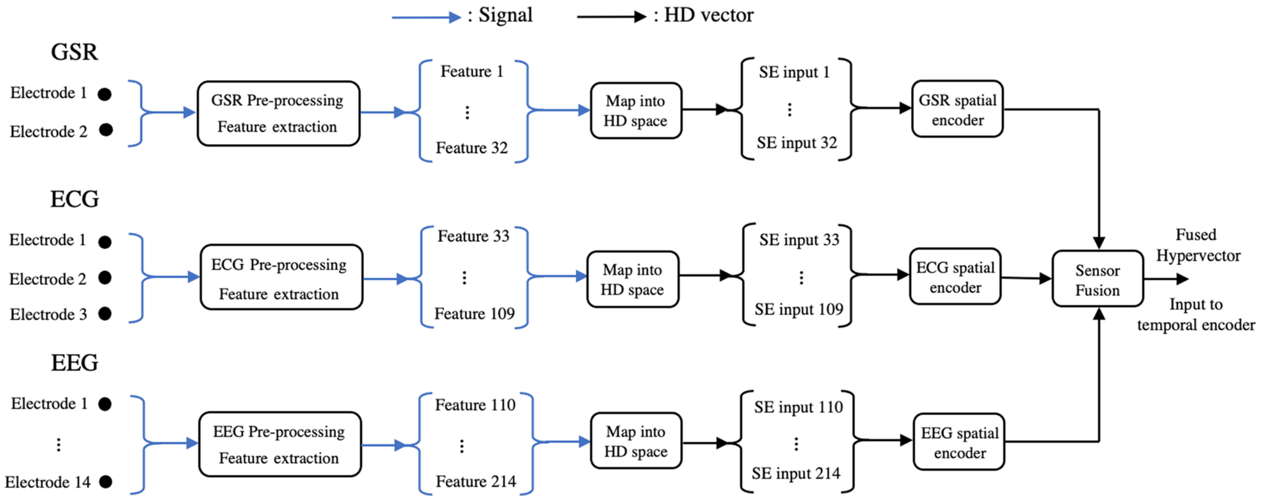


Fig. 1 Sensor fusion datapath from electrodes to a fused hypervector for the three-modality emotion recognition system used in AMIGOS with GSR, ECG, and EEG sensor inputs. The sensor inputs are pre-processed into a set of features which are then mapped into the HD space to create a set of spatial encoder (SE) inputs. These vectors are encoded within each modality, and then finally fused together to create one vector representing information from all of the channels. The process is detailed in Sect. 2

precomputed and stored. With this approach, the memory is constant regardless of the number of channels, increasing hardware efficiency.

Emotion recognition can provide valuable information for HCI. Long-term wearable monitoring of user's emotional state enables usage of implicit user information beyond the traditional keyboard and mouse. This has direct applications in feedback systems such as driver warning systems, smart environments, and automated tutoring systems. A hardware-efficient platform that allows for extended-use, on-board classification addresses these applications and also enables significant enhancement of the general interface between humans and computers. Towards this, this work presents the following contributions:

Design of an efficient sensor fusion HDC architecture. The early fusion approach reduces the parallel encoding paths previously used for HDC sensor fusion to a single one by taking advantage of HDC's inherent projection of features into large-capacity hypervector representations.

Algorithmic optimizations for memory reduction are proposed and implemented. This includes a novel approach that takes advantage of the combinatorial nature of the HDC encoding process, and the usage of an elementary cellular automata with rule 90 together to reduce vector storage and request frequency.

Reduction of hypervector dimension is explored. Dimension reduction is a method of comprehensive datapath reduction. The impact of this on the algorithm performance is explored in conjunction with all memory reduction and optimization techniques.

Results are reported for the DEAP and AMIGOS multimodal emotion recognition data sets for all experiments.

2 Methods

2.1 HDC early fusion architecture

The HDC physiological architecture includes four main blocks: the map into the hyperdimensional space (HDS), the spatial encoder, the temporal encoder, and the associative memory as shown in Fig. 2. The first block maps incoming data into HDS using an item memory or a generator. HDC depends on the pseudo-orthogonality of random vectors to be able to distinguish between various classes; a random vector will be nearly orthogonal to another random vector in the hyperdimensional space. Random vectors are used for the channel item memory vectors so that the source channel of a feature value can be included as information in the encoding process. These are stored in an item memory (iM).

To encode feature values, in this implementation, additional feature projection vectors are randomly generated for each channel and stored as well. In traditional architectures, the feature projection vector {−1, 0, 1} is

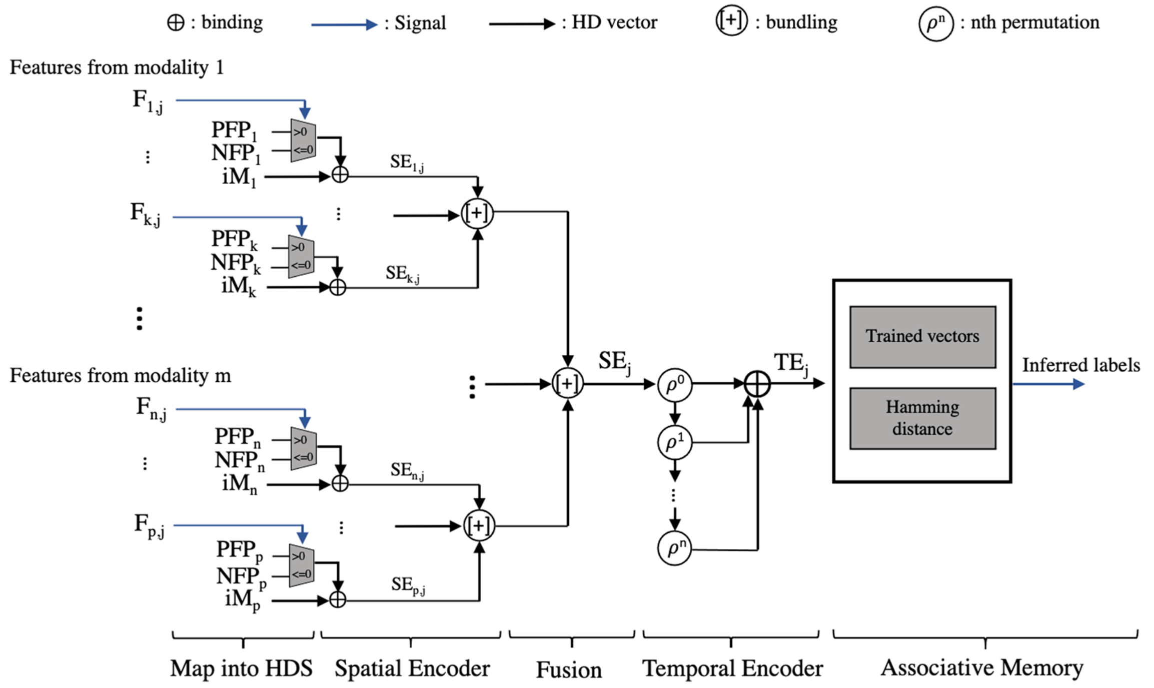


Fig. 2 HDC early fusion detailed architecture for m modalities with the four main blocks: map into HDS, spatial encoder, temporal encoder for n-gram of size n+1, and associative memory. Sensor fusion occurs early in the datapath, directly after the spatial encoder

multiplied by the feature value and then binarized by reducing the positive values in the vector to 1s, and the zeros and negative values to 0s. This process can be simplified to multiplexers selecting between a pre-generated random negative or positive binary feature projection vector depending on the feature value's sign to eliminate computationally expensive multipliers. This allows the feature projection vectors to maintain pseudo-orthogonality but have the same sparsity as the item memory vectors, making them interchangeable. As a result, the feature projection vectors can also be stored in the item memory instead of separately.

In the spatial encoder, the binding operation (XOR) is utilized to generate a spatially encoded hypervector for each channel. If iM_i represents the item memory vector for channel i and $\mathrm{FP}_{i,j}$ represents the feature projection vector selected for channel i for sample j, then the spatially encoded hypervector for sample j $\mathrm{SE}_{i,j}$ is computed as

$$\mathrm{SE}_{i,j} = \mathrm{iM}_i \oplus \mathrm{FP}_{i,j} \tag{2}$$

To develop a complete hypervector, the bundling operation (vertical majority count across vectors) combines the many spatially encoded hypervectors within a sensor modality. If the sensor modality m has k channels and the bundling operation is represented as $+$, $\mathrm{SE}_{m,j}$ is computed as

$$\mathrm{SE}_{m,j} = (\mathrm{iM}_1 \oplus \mathrm{FP}_{1,j}) + \cdots + (\mathrm{iM}_k \oplus \mathrm{FP}_{k,j}) \tag{3}$$

Because emotion recognition involves various sensor modalities, it requires fusion. Previous sensor fusion implementations fused after the temporal encoder, but in this work, an early fusion approach is taken, which fuses the modalities directly after the spatial encoding process. Therefore, this architecture requires only a single temporal encoder as opposed to one per modality, as shown in Fig. 3. This reduces the parallel encoding paths while still allowing each modality to be weighted equally instead of by number of features. If there are m sensor modalities, the fused spatially encoded hypervector for sample j is

$$\mathrm{SE}_j = \mathrm{SE}_{1,j} + \mathrm{SE}_{2,j} + \cdots + \mathrm{SE}_{m,j} \tag{4}$$

HDC also has the ability to encode temporal changes through the use of n-grams based on a sequence of N samples. This is invaluable for physiological signals that are time-varying as it allows for the capturing of time-dependent emotional fluctuations within the same class or between segments of the same class. The permutation operation (cyclical shift, represented as ρ) is used to keep track of previous samples. Hypervectors coming from the spatial encoder are permuted and then bound with the next hypervector N times in the temporal encoder. This results in an output that observes changes over time, TE_j, that can be computed as

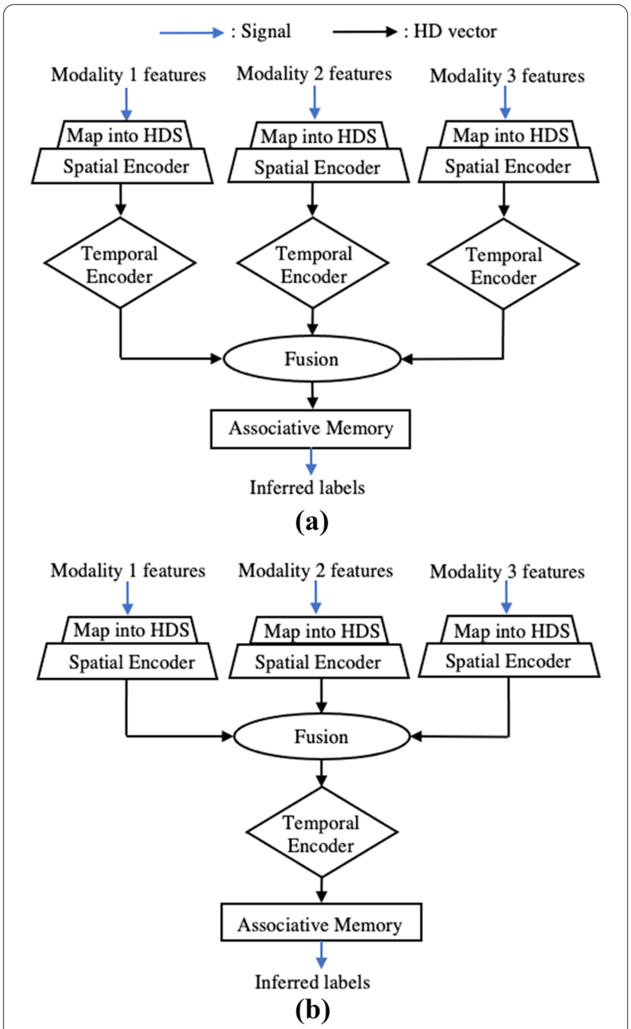


Fig. 3 HDC **a** late fusion and **b** early fusion architectures for a three-modality emotion recognition system. The late fusion architecture fuses after the temporal encoder, resulting in 3 parallel temporal encoders—one per modality. In comparison, the early fusion architecture fuses before the temporal encoder, resulting in only 1 temporal encoder

$$\mathrm{TE}_j = \mathrm{SE}_j \oplus \rho^{+1}(\mathrm{SE}_{j-1}) \oplus \cdots \oplus \rho^{+(N-1)}(\mathrm{SE}_{j-(N-1)}) \tag{5}$$

During the training process, many such encoded hypervectors are generated, bundled to represent a class and then stored into the final block, the associative memory. During inference, the encoded hypervector is compared against each trained hypervector using Hamming distance. For binary vectors, this involves an XOR and then popcount. The comparison with least distance is the inferred label.

2.2 Implementation

The HDC early fusion architecture is implemented on both the AMIGOS and DEAP data sets with a standard dimension of 10,000 for the full datapath in the baseline implementation. In the AMIGOS study, GSR recorded at 128 Hz (1 channel across middle and index fingers), ECG recorded at 256 Hz (2 channels on right and left arm crooks) and continuous EEG recorded at 128 Hz (14 channels: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4) were measured for 33 subjects as they watched 16 videos [7]. Each video for each subject was classified to have either led to a positive or negative emotion (valence), and the strength of the emotion was classified as either strong or weak (arousal). From the 3 sensor modalities, Wang et al. selected 214 time and frequency domain features relevant to accurate emotion classification [11]. GSR has 32 features, ECG has 77 features, and EEG has 105 features. Similar preprocessing and features are used in this work as this feature selection demonstrated excellent performance on the AMIGOS data set in previous work [9, 11]. The features used include GSR skin response/conductance and skin conductance slow response, ECG heart rate spectral power, variability and heart rate time series, and EEG average power spectral density and asymmetry of theta band, alpha band, beta band, and gamma band. The data for all 33 subjects was appended and a moving average of 15 s over 30 s was applied. The signals were scaled to be between −1 and +1 to meet the HDC encoding process and downsampled by a factor of 8 for more rapid processing and usage of the HDC classification algorithm. Previous work uses the leave-one-subject-out approach to evaluate performance, this was also implemented for the early fusion architecture [7, 11, 12]. The temporal encoder was tuned and an optimal n-gram of 3 feature windows was selected. For both data sets, transitionary ngrams (those with samples from both classes) were excluded from training and testing.

The DEAP study was collected in a similar format as the AMIGOS with 32 subjects watching 40 one-minute highlight excerpts of music videos selected to trigger distinct emotional reactions; however, it contains more extensive sensor modalities all recorded at 512 Hz: continuous EEG (32 channels placed according to the international 10–20 system), EMG (2 channels: neck and corner of mouth), GSR (1 channel across middle and index fingers), BVP (1 channel on the thumb), EOG (4 channels above and below each eye), temperature (1 channel on skin) and respiration amplitude (1 channel) [8]. The arousal and valence scores were self-assessed by the participants on a scale between 1 and 9. A binary classification system is maintained for high and low valence and arousal by thresholding the scale at 5. Preprocessing and feature selection were done using the TEAP toolbox which selected time and frequency domain features for 5 of the modalities based on previous work in those areas [24]. These features have been shown to enable high performance on the DEAP data set in prior work and hence were selected for this work [25]. EMG has 10 features including power and statistical moments over two channels. EEG has 192 features across the 32 channels including power spectral density in delta, theta, slow alpha, alpha, beta and gamma bands. GSR has 7 features including number of peaks, amplitude of peaks, rise time and statistical moments. BVP's 17 features include inter-beat intervals, multiscale entropy at 5 levels, tachogram power, power spectral density in multiple bands, and statistical moments. Respiration has 12 features including main frequency, power spectral density and statistical moments. This results in 40 samples with a total of 238 features per video from 5 modalities per subject. The signals were then scaled to be between −1 and +1 for the HDC encoding scheme. Previous work for this data set does training and inference independently by subject which was adopted in this work as well [13, 1, 8]. Typically, 90% of the data set is used for training per subject with the remaining 4 videos used for testing. For HDC, due to the inclusion of the temporal encoder, this would result in limited number of inferences leading to imprecise classification accuracies. As a result, the size of the training set was decreased to be 80% of the data set with 20% used for testing. A temporal n-gram of 3 was selected for this data set as well.

2.3 Memory optimization

For both the AMIGOS and DEAP data sets, there are over 200 features that need to be spatially encoded. This requires advance storage of 214/238 iM vectors and 420/476 feature projection (FP) vectors—positive (PFP) and negative (NFP)—totalling to 642/714 vectors that need to be stored in the item memory. Use of a unique iM and FP vector set per channel is shown in first column of Fig. 4. Without significant reduction of the memory requirements, optimizations of other blocks will provide limited benefits to the overall efficiency.

In the spatial encoder, the iM vector and the FP vector are bound together to form a unique representation containing feature information that is specific to a feature channel. However, both the iM and FP vectors do not need to be unique to the feature channel in order to generate a unique combination of the two. The binding operation will inherently create a vector different, and pseudo-orthogonal to both of its inputs. Therefore,

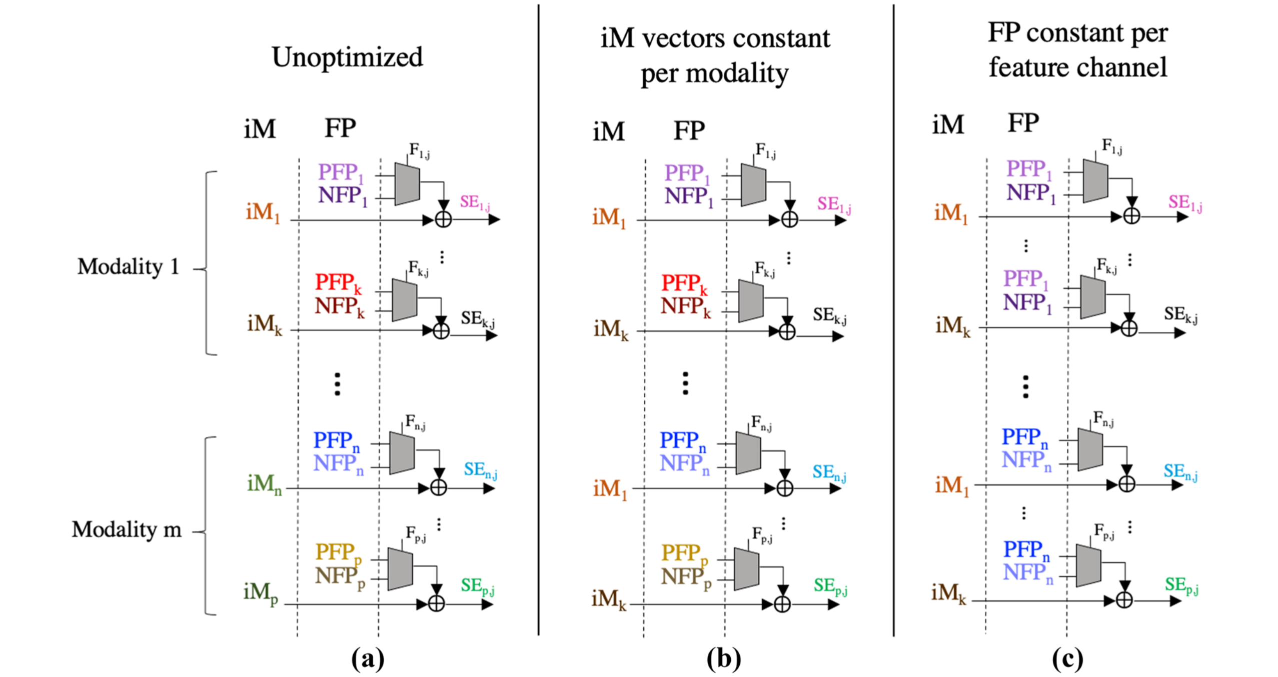


Fig. 4 iM and FP vectors used to map into HDS to generate unique SE vectors per channel for **a** 'unoptimized' with distinct iM and FP vectors for all channels, **b** 'iM vectors constant per modality' with the same iM vectors between different modalities, and **c** 'FP constant per feature channel' with the same FP vectors between channels of the same modality

as long as one of these inputs is different for a specific feature channel, the spatially encoded feature channel vector (represented by the SE vectors in Fig. 4) will be unique. Using this idea, a set of optimizations were developed and implemented on the DEAP and AMIGOS data sets:

'iM vectors constant per modality': the iM is replicated across the various modalities, shown in the second column of Fig. 4. If, between each modalities, the FP vectors are different, then orthogonality and input feature channel uniqueness are maintained even if the iM is the same.

'FP constant per feature channel': though the iM is now the same between each modalities, each feature channel within a modality still has a unique iM vector. Therefore, it is possible to re-use the same FP vectors for every feature channel within a modality, as shown in the third column of Fig. 4. This requires maintaining 2 unique FP vectors (PFP and NFP) for each modalities, and unique iM vectors within a modality.

'Combinatorial pairs': taking this combinatorial binding strategy to its limit, the 2-input binding operation can be used to generate many unique vectors from a smaller set of vectors by following an algorithmic process. Each feature channel requires a distinct set containing an iM vector, and two FP (positive & negative) vectors: {iM, PFP, NFP}. If the vectors for feature channel 1 are {A, B, C}, then the bound pairs that could result from spatial encoding (iM $\oplus$ PFP or iM $\oplus$ NFP) are:

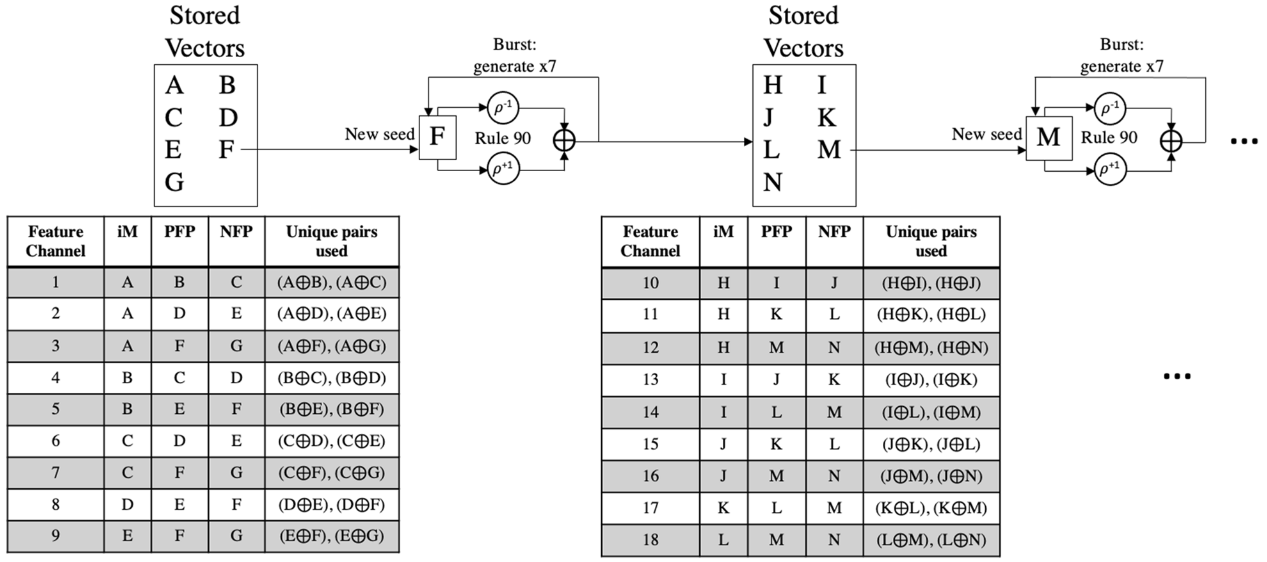
- A $\oplus$ B
- A $\oplus$ C

B $\oplus$ C will not occur, because they are both FP vectors. However, it is a unique pairing that could be re-used for another channel. For example, the set for feature channel 2 could be: {B, C, D}. The encoding process would use the following pairings:

- B $\oplus$ C
- B $\oplus$ D

This re-use strategy is the key to saving memory; it can be applied across all channels using a bank of the minimal required vectors, as shown in the first part of Fig. 5.

Each vector can be paired with every other vector only once to maintain orthogonality and paired uniqueness across all feature channel. For a feature channel, one vector (the iM) must have two other available vectors (PFP and NFP) to pair with. With $\lfloor x \rfloor$ defined as the floor function of x, the following equation can be used to calculate the total feature channels, TFC, possible given a bank of v vectors:



Feature Channel	iM	PFP	NFP	Unique pairs used
1	A	B	C	(A⊕B), (A⊕C)
2	A	D	E	(A⊕D), (A⊕E)
3	A	F	G	(A⊕F), (A⊕G)
4	B	C	D	(B⊕C), (B⊕D)
5	B	E	F	(B⊕E), (B⊕F)
6	C	D	E	(C⊕D), (C⊕E)
7	C	F	G	(C⊕F), (C⊕G)
8	D	E	F	(D⊕E), (D⊕F)
9	E	F	G	(E⊕F), (E⊕G)

Feature Channel	iM	PFP	NFP	Unique pairs used
10	H	I	J	(H⊕I), (H⊕J)
11	H	K	L	(H⊕K), (H⊕L)
12	H	M	N	(H⊕M), (H⊕N)
13	I	J	K	(I⊕J), (I⊕K)
14	I	L	M	(I⊕L), (I⊕M)
15	J	K	L	(J⊕K), (J⊕L)
16	J	M	N	(J⊕M), (J⊕N)
17	K	L	M	(K⊕L), (K⊕M)
18	L	M	N	(L⊕M), (L⊕N)

Fig. 5 'Combinatorial pairs' feature channel vector set generation demonstrated for 7 stored vectors. iM loops through vector bank after exhausting available sequential pairs for FP. Hybrid method follows by burst re-generating the vector bank with rule 90 so that new combinatorial pairs can be formed for more feature channels. Generation of 18 feature channel vector sets using a bank of only 7 vectors is shown

$$\mathrm{TFC} = \sum_{n=1}^{\nu-2} \left\lfloor \frac{\nu - n}{2} \right\rfloor \tag{6}$$

The formula can be derived by looping through each vector in the vector bank and sequentially grouping it with pairs of the other vectors. The generation of feature channel sets can be algorithmic, following the pattern shown in the tables in Fig. 5.

'Rule 90 generation': implementation of the cellular automata with rule 90 will allow trading off vector storage with vector generation. If there are m modalities, the first $2 \times m$ generated vectors would be used for the PFP and NFP vector for each modality. These would be maintained throughout training and inference resulting in $2 \times m + 1$ locally stored vectors including an initial seed vector. However, the rest of the iM vectors would be generated on the fly for each feature channel during the encoding process, requiring no additional vector storage. This is possible because of the fixed access pattern of the iM. The generation process requires use of rule 90 across the entire hypervector, and local storage of the most recently generated vector to use as the next seed. 1 vector is requested and then generated for each feature channel.

'Hybrid': to reduce vector requests and hence the computation for rule90, the last two schemes: 'combinatorial paired binding' and 'rule 90 generation', can be combined. This hybrid strategy could include burst generation of a small set of vectors which could be locally stored. From this set, combinatorial pairs are assigned to feature channels and spatially encoded. This set can be gradually re-populated with new vectors as the old vectors are exhausted in the encoding process providing new possible pairs. This provides further tradeoff between vector storage and computation. The vector request rate (vector generation requests per feature channel) is minimized when the vector storage is large enough for the combinatorial paired binding scheme alone at which point no generation is required.

'Dimensionality reduction': the final method of memory reduction is in the form of hypervector dimension reduction. The algorithm outlined in 2 stays exactly the same, but the length of the HD vectors used throughout is shortened. This changes the size of the entire datapath, impacting both the logic complexity and the memory storage approximately linearly. However, smaller hypervectors also have reduced pseudo-orthogonality—random lower dimensional vectors are less likely to be nearly orthogonal in the hyperdimensional space than higher dimensional vectors. The capacity for information that can be stored within a hypervector is reduced. This especially impacts the output of the bundling operation that occurs in the spatial encoder which no longer represents as much information about each input channel, impacting classification accuracy. This optimization is a tradeoff between algorithm accuracy performance and overall efficiency. The impact of changing dimensions on emotion recognition accuracy for the various memory optimizations is also explored.

Table 1 AMIGOS classification accuracy comparison table

Method	HV vs. LV (%)	HA vs. LA (%)
GaussianNB* [7, 9]	57	58.5
SVM [11]	68.0	66.3
ELM [12]	83.9	82.8
XGB [11]	80.1	68.4
HDC late fusion [9]	83.2	70.1
HDC early fusion	87.1	80.5

*F1 score. Accuracy value not available

Table 2 DEAP classification accuracy comparison table

Method	HV vs. LV (%)	HA vs. LA (%)
GaussianNB [8]	57.6	62.0
RBM with SVM [13]	60.7	64.6
MESAE [1]	76.2	77.2
DBN [14]	51.2	68.4
HDC early fusion	76.7	74.2

3 Results

The HDC early fusion architecture was implemented on the AMIGOS and DEAP data sets for classification of high vs. low arousal and high vs. low valence. HDC early fusion achieved the highest average valence and arousal accuracy on AMIGOS, with the Rule 90 generation encoding method. A comparison against other AMIGOS binary classification multi-modal work using SVM, XGB, Gaussian Naive Bayes (GaussianNB) and ELM is shown in Table 1. The early fusion encoding process provided a boost of 3.9% for valence and 10.4% for arousal from the late fusion HDC architecture previously implemented by Chang et al. [9] and demonstrates higher average accuracy than state of the art.

On the DEAP data set, HDC early fusion achieved the highest average valence and arousal accuracy with the FP constant per feature channel encoding method. A comparison against other DEAP binary classification multi-modal work using GaussianNB, RBM with SVM, MESAE, and DBN is shown in Table 2. HDC early fusion accuracy is very comparable with other high performance multi-modal approaches to the DEAP data set.

One of the key benefits of HDC is the hardware efficiency, which is further improved for large-channeled emotion recognition tasks through the memory optimizations discussed earlier. The results for both valence and arousal accuracy as well as the resulting stored vector count for AMIGOS and DEAP across all memory-optimizing encoding methods are shown in Fig. 6.

For AMIGOS, with 214 channels and 3 modalities, the unoptimized method requires 3 unique vectors per feature channel {iM, PFP, NFP}—a total of 642 vectors. The 'iM vectors constant per modality' scheme is limited by the largest modality which is EEG with 105 feature channels. This results in a total of $105 + 214 \times 2 = 533$ vectors. The 'FP constant per feature channel' reduces the total vector set to $105 + 2 \times 3 = 111$. The 'combinatorial pairs' method uses Eq. 6 and results in a required 31 vectors. Finally, the 'rule 90 generation' stores one FP pair {PFP, NFP} for each modality along with the seed vector, a total of $2 \times 3 + 1 = 7$. The memory optimizations result in an overall decrease in required vector storage by 98.91% from 642 vectors to 7, while the accuracy actually increased by 1.9% for arousal and 2.7% for valence. For DEAP the overall memory storage is higher due to increased feature channels, 238, and modalities, 5. The memory optimizations result in an overall decrease of 98.46% from 714 vectors to 11, while the accuracy actually increased by 0.6% for arousal and minimally decreased by 1.4% for valence.

Using the combinatorial pair method alone, the relationship between feature channel sets generated and number of stored vectors is shown in Fig. 7. Linear increases in number of stored vectors will cause result in a quadratically increasing number of available feature channel sets. This plot demonstrates that with 7 vectors, 9 feature channel sets are available, but with 50 vectors, 600 feature channel sets are available.

The combinatorial pair method can be used together with rule 90 in a hybrid scheme to provide options for tradeoff between memory and vector requests. In the solely rule 90 version, 7 vectors are stored for AMIGOS and 11 vectors for DEAP; a total of 214 and 238 vector requests are made during the encoding process for AMIGOS and DEAP for a single sample—a vector request rate of 1. However, using the burst generation technique, a small subset of vectors could be generated in one shot using rule 90, stored, and then used for spatially encoding a quadratically larger number of feature channels to reduce the total number of vector requests made. The relationships between vector request rate (total vector requests / number of feature channels) and vector storage for AMIGOS and DEAP are shown in Fig. 8.

The only rule 90 method stores 7 and 11 vectors regardless which, if used with the hybrid scheme, could be used to generate 9 feature channels with every burst instead of only 7 for AMIGOS, or 25 feature channels instead of just 11 for DEAP. This results in a reduction in frequency of vector requests by 22.22% and 56.00% for AMIGOS and DEAP, respectively, even while using the same number of stored vectors.

Finally, the impact of reducing dimension on the overall accuracy performances of the algorithm for emotion recognition tasks are shown for AMIGOS

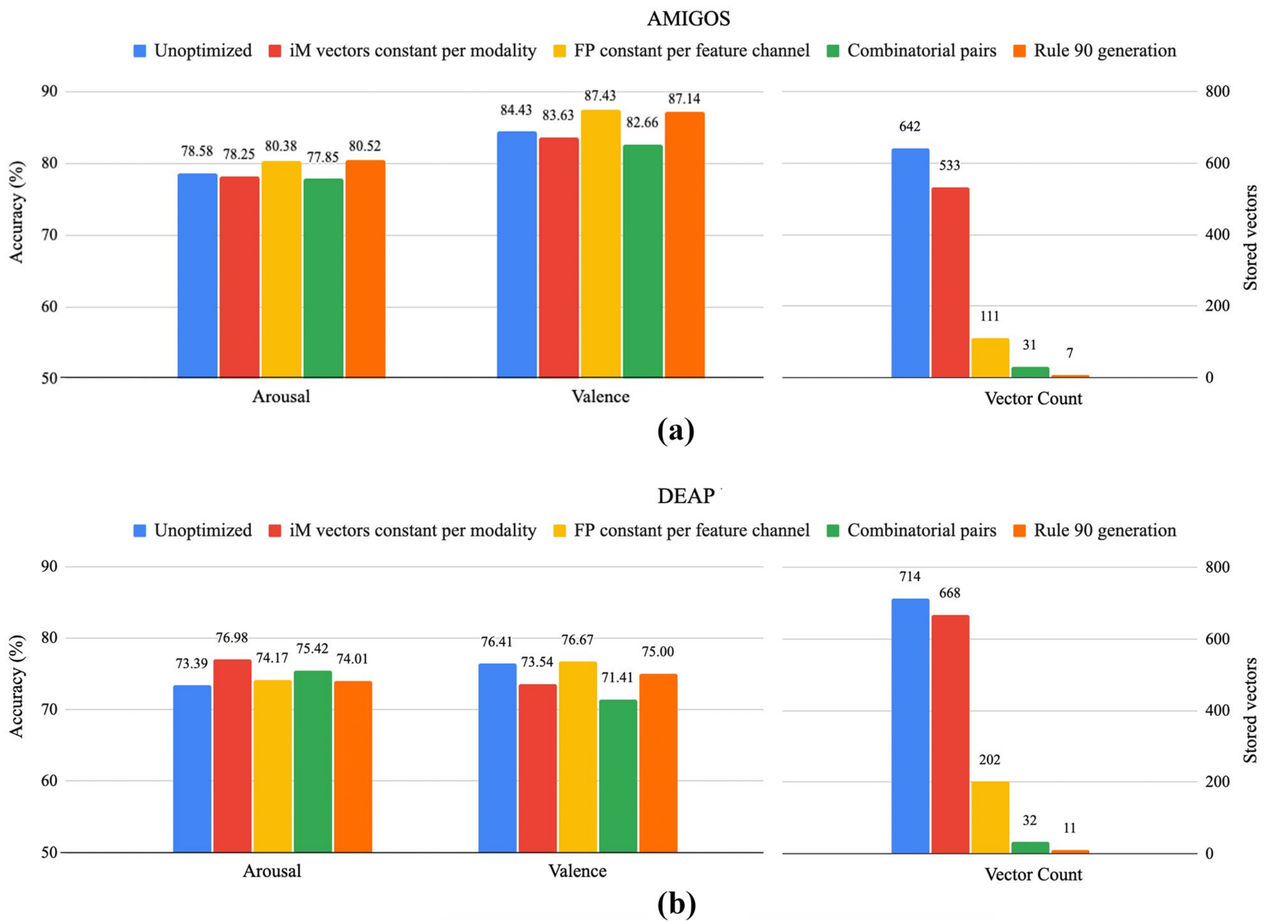


Fig. 6 Arousal and valence accuracies and required vector storage for the various memory optimization as compared to unoptimized for **a** AMIGOS and **b** DEAP data sets. Optimizations include 'unoptimized' with distinct iM and FP vectors for all channels, 'iM vectors constant per modality' with the same iM vectors between different modalities, 'FP constant per feature channel' with the same FP vectors between channels of the same modality, and 'Rule 90 generation' with generated FP and iM vectors on top of the previous optimizations

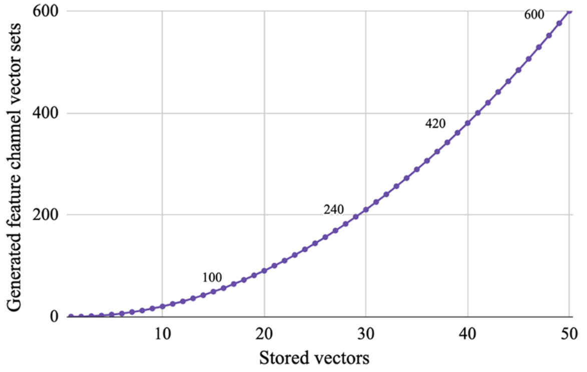


Fig. 7 Number of unique feature channel vector sets {iM, PFP, NFP} that can be generated, and hence the number of channels that can be encoded, using the combinatorial pair technique as the number of stored vectors increases

and DEAP in Fig. 9. For the AMIGOS data set, a gradual decrease in accuracy is observed particularly from dimensions of 7000 by which point the average accuracy has dropped by 1%. Steeper decreases of ~0.6% and ~4.4% are seen between dimensions 3000 and 2000 and between 2000 and 1000. Overall, between a dimension of 10,000 and 1000, there is a decrease in average accuracy of ~7%.

For the DEAP data set, there is greater variation in accuracy across the dimensions and between methods; however, an overall trend of decreasing accuracy can still be seen, particularly past dimensions of 5000 at which point the accuracy drops below 74.5% and continues to decrease rapidly including a ~4.4% drop between 2000 and 1000. Overall, between 10, 000 and 1000 there is a decrease in average accuracy of ~5.4%.

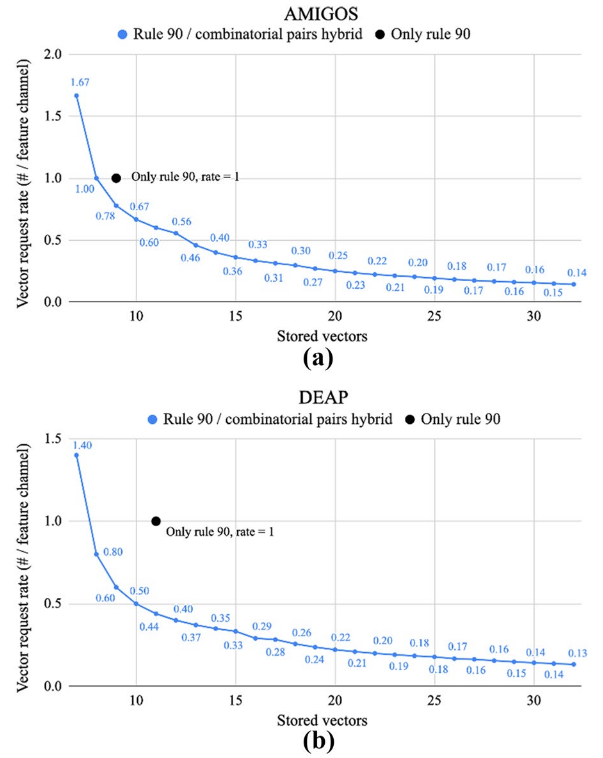


Fig. 8 Vector request rate (number of vectors requested per input channel) as the number of vectors stored increases while using rule 90 on a small set of continuously re-populating vectors as compared to only rule 90 for **a** AMIGOS and **b** DEAP

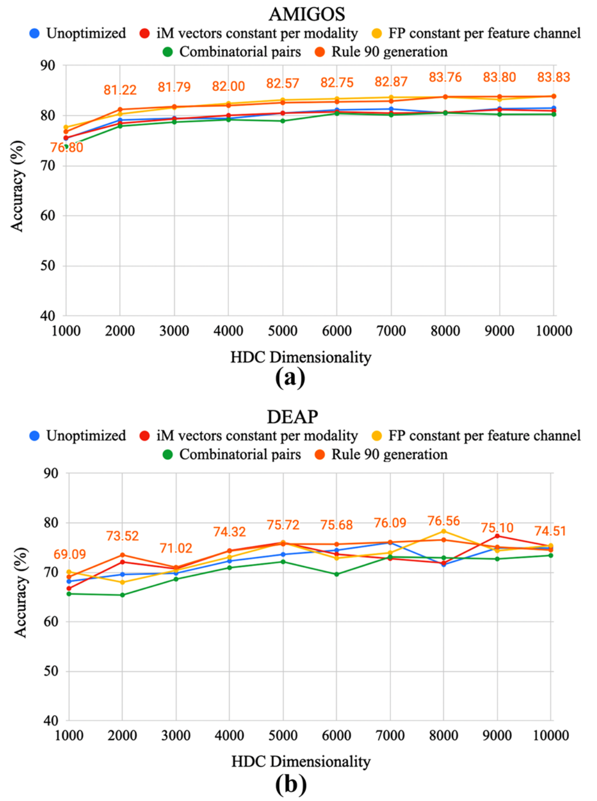


Fig. 9 Average valence and arousal accuracy for the various memory optimizations proposed as the hypervector length decreases for **a** AMIGOS and **b** DEAP. The data labels are shown for the most optimized version: rule 90 generation

4 Discussion

The first change that was implemented was an overall architecture shift from late to early fusion. The results demonstrate an improvement in performance on the AMIGOS data set despite moving the fusion point to combine the parallel data streams earlier in the encoding process. The boost in accuracy may come from the fact that different modalities may have different temporal behavior, which may lead to different optimal *n*-grams. For late fusion, an n-gram of 4 was used for all modalities without individual tuning. For early fusion, an optimal *n*-gram of 3 was selected for the fused modalities temporal behavior, improving the overall performance. The early fusion method requires tuning of only a single temporal encoder and still achieves higher accuracy even with reduction of the overall encoding complexity. This indicates the potential, benefits, and feasibility of early fusion encoding processes in HDC. Information is retained in the high-capacity vectors even with only a single encoding path after the spatial encoder.

In addition, compared to other works, as shown in Table 1, HDC early fusion performed better than GaussianNB, SVM, XGB, ELM and HDC late fusion on AMIGOS. It also performed better, as shown in Table 2 than GaussianNB, RBM with SVM and DBN and showed similar performance to MESAE on the DEAP data set. Given its high performance, HDC early fusion appears well-suited for emotion recognition tasks.

The difference in performance between valence and arousal classification accuracies of 2.5–6.6% for AMIGOS and and DEAP may be attributed to selection of n-gram size based on maximizing overall performance instead of selecting different *n*-gram sizes per category which would result in duplicate datapaths for valence and arousal classifications. The difference in performance between AMIGOS and DEAP (6.3% and 10.4% for arousal and valence) may be attributed to the difference in features and modalities present in each data set, their class separability, and the ability of HDC to differentiate while using the selected early fusion encoding scheme.

The performance of various memory optimizations were explored and shown in Fig. 6. HDC depends on near-orthogonality between different data streams and feature values to ensure that samples from different classes that vary in these ways are encoded into sufficiently orthogonal class vectors. Each optimization reduces the total number of unique vectors that need to be stored in advance; however, there was no decrease in accuracy on AMIGOS between the most unoptimized and most optimized. There was actually an average increase of ~2.3%; this accuracy change may be attributed to the random element of HDC vector initialization/generation which may result in either beneficial or detrimental random patterns. This is further demonstrated by the DEAP data set for which the optimizations increased the arousal accuracy by 0.6%, yet decreased valence accuracy by 1.4%. With an overall memory reduction of > 98%, the optimizations have a significant impact on the hardware requirements while displaying little to no performance degredation for both AMIGOS and DEAP, demonstrating that the techniques generalize across data sets.

A hybrid, burst generation technique was proposed, in which a small vector set would be used maximally, as shown in Fig. 7, and then re-generated. Using this method, the total number of vectors that need to be generated during training or inference of a single sample can be decreased, as shown in Fig. 8. Rule 90 alone requires generation of at least one vector per feature channel and doesn't take advantage of the combinatorial pairs available with its existing storage, hence implementation of this hybrid technique decreases the overall required vector generation. The benefit is higher for the DEAP data set with more modalities due to the prior storage of a larger number of vectors for rule 90.

This technique also allows for scalability while still maintaining memory size; existing vectors pairs can be used to their highest capacity and then the vector bank can be re-generated using rule 90 for the further capacity required by additional channels or modalities. This could be done until the limits of the cellular automata are reached ($>>10^3$). The trade-off between the computation for vector generation and additional storage provides options. The optimal performance point based on power or memory constraints can be determined for specific applications/platforms.

The dimension reduction shown in Fig. 9 demonstrates the trade-off between accuracy and comprehensive datapath size reduction. An optimal point could be selected that provides the accuracy needs of the system with minimum HDC dimensionality. With an accuracy tolerance of ~2%, the dimension can be reduced by 70% to hypervectors of 3000 bits for AMIGOS and by 80% to hypervectors of 2000 bits for DEAP. These techniques allow reduction of overall power due to significantly reduced computation and memory storage. For applications such as smart environments and enhanced human–computer interactions, this enables ease-of-use through longer battery life for low-power wearable systems.

5 Conclusions

In conclusion, this work proposed a solution to the many-channeled (> 200) memory-expensive emotion recognition task in the form of a brain-inspired early fusion hyperdimensional computing architecture alongside several optimization techniques that make emotion recognition feasible for hardware-constrained, low-power wearable applications. The various methods explored were able to achieve significant reduction >98% in required memory and > 20% decrease in frequency of vector requests. Finally, the impact of hypervector dimension on emotion recognition accuracy demonstrated < 2% performance degradation for datapath reductions of 70–80%.

Though this work focuses on emotion recognition, all the proposed techniques maintain the properties required for successful hyperdimensional computing and, therefore, could generalize to other applications, and will be particularly useful for those with many, varied streams of input information.

To demonstrate the impact of the memory optimizations, the energy per prediction of an ASIC realization of the emotion-classification engine was reduced by 93% for the cellular automata rule 90 over prior HDC processors in a recent implementation study [26]. Future work could include efforts to improve the accuracy of the HDC arousal classification to be higher by modifying the encoding scheme. The overall hardware could remain similar by reusing existing blocks for minimally different functions depending on whether the classification is for valence or arousal. In addition, not all input channels may be relevant for the emotion recognition classification task. Reduction in overall input features would reduce the number of unique iM vectors needed. Future work could explore feature reduction/optimization for the HDC algorithm to determine which channels of information are truly necessary to maintain high accuracy for this task. Next steps could also include implementation of the proposed techniques for other applications with significantly larger numbers of channels and modalities to explore generalizability and scalability.

6 Limitations

In this work, the focus was the implementation of hyperdimensional computing for a multi-modal sensor fusion task and algorithm exploration of improvements in memory storage and encoding complexity. Towards this, preexisting emotion recognition data sets were used with features selected based on prior work. Future work could include feature optimization for performance with the HDC algorithm towards integration of sensors and HDC processing into a user interface.

Abbreviations

ASIC: Application-specific integrated circuit; BVP: Blood volume pressure; DBN: Deep belief network; ECG: Electrocardiogram; EEG: Electroencephalogram; EMG: Electromyography; EOG: Electrooculography; FP: Feature projection; GaussianNB: Gaussian Naive Bayes; GSR: Galvanic skin response; HCI: Human–computer interaction; HD: Hyperdimensional; HDC: Hyperdimensional computing; HDS: Hyperdimensional space; iM: Item memory; IoT: Internet-of-things; MESAE: Multiple-fusion-layer based ensemble classifier of stacked autoencoder; NFP: Negative feature projection; PFP: Positive feature projection; RBM: Restricted boltzmann machine; RFE: Recursive feature elimination; SVM: Support vector machine; XGBoost: Extreme gradient boosting.

Acknowledgements

The authors would like to thank Abbas Rahimi and Sohum Datta for their interest and support of this work.

Author contributions

AM, AN, and RA developed and implemented the learning algorithm with optimizations. AM, DS, MA and HL designed the architectures. YSS and JMR oversaw the project. AM and JMR wrote and edited the manuscript. All authors read and approved the final manuscript.

Funding

This work was supported in part by the CONIX Research Center, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA, as well as the VIP and HYDDENN programs sponsored by DARPA. This work was supported in part by the National Science Foundation Graduate Research Fellowship Program under Grant No. 1752814. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation, DARPA or the U.S. government. Support was also received from sponsors of Berkeley Wireless Research Center.

Availability of data and materials

Data sharing is not applicable to this article as no new data sets were generated or analysed during the current study.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 29 March 2021 Accepted: 15 June 2022
Published online: 27 June 2022

References

1. Yin Z, Zhao M, Wang Y, Yang J, Zhang J (2017) Recognition of emotions using multimodal physiological signals and an ensemble deep learning model. Comput Methods Programs Biomed 140:93–110
2. Zeng Z, Pantic M, Roisman GI, Huang TS (2008) A survey of affect recognition methods: audio, visual, and spontaneous expressions. IEEE Trans Pattern Anal Mach Intell 31(1):39–58
3. Shojaeilangari S, Yau WY, Nandakumar K, Li J, Teoh EK (2015) Robust representation and recognition of facial emotions using extreme sparse learning. IEEE Trans Image Process 24(7):2140–2152
4. Kapoor A, Burleson W, Picard RW (2007) Automatic prediction of frustration. Int J Hum Comput Stud 65(8):724–736
5. Picard RW, Vyzas E, Healey J (2001) Toward machine emotional intelligence: analysis of affective physiological state. IEEE Trans Pattern Anal Mach Intell 23(10):1175–1191
6. Cohn JF (2007) Foundations of human computing: Facial expression and emotion. In: Artifical Intelligence for Human Computing, Springer, pp 1–16
7. Correa JAM, Abadi MK, Sebe N, Patras I (2018) Amigos: A dataset for affect, personality and mood research on individuals and groups. IEEE Transactions on Affective Computing
8. Koelstra S, Muhl C, Soleymani M, Lee JS, Yazdani A, Ebrahimi T, Pun T, Nijholt A, Patras I (2011) Deap: a database for emotion analysis; using physiological signals. IEEE Trans Affect Comput 3(1):18–31
9. Chang EJ, Rahimi A, Benini L, Wu AYA (2019) Hyperdimensional computing-based multimodality emotion recognition with physiological signals. In: 2019 IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), IEEE, pp 137–141
10. Posner J, Russell JA, Peterson BS (2005) The circumplex model of affect: an integrative approach to affective neuroscience, cognitive development, and psychopathology. Dev Psychopathol 17(3):715
11. Wang SH, Li HT, Chang EJ, Wu AYA (2018) Entropy-assisted emotion recognition of valence and arousal using xgboost classifier. In: IFIP International Conference on Artificial Intelligence Applications and Innovations, Springer, pp 249–260
12. Siddharth S, Jung TP, Sejnowski TJ (2019) Utilizing deep learning towards multi-modal bio-sensing and vision-based affective computing. IEEE Transactions on Affective Computing
13. Shu Y, Wang S (2017) Emotion recognition through integrating eeg and peripheral signals. 2017 IEEE International Conference on Acoustics. Speech and Signal Processing (ICASSP), IEEE, pp 2871–2875
14. Wang D, Shang Y (2013) Modeling physiological data with deep belief networks. Int J Inf Educ Technol 3(5):505
15. Afifi S, GholamHosseini H, Sinha R (2020) Fpga implementations of svm classifiers: a review. SN Comput Sci 1:1–17
16. Montagna F, Rahimi A, Benatti S, Rossi D, Benini L (2018) Pulp-hd: Accelerating brain-inspired high-dimensional computing on a parallel ultra-low power platform. In: 2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC), IEEE, pp 1–6
17. Moin A, Zhou A, Rahimi A, Menon A, Benatti S, Alexandrov G, Tamakloe S, Ting J, Yamamoto N, Khan Y, et al (2020) A wearable biosensing system with in-sensor adaptive machine learning for hand gesture recognition. Nature Electronics pp 1–10
18. Burrello A, Cavigelli L, Schindler K, Benini L, Rahimi A (2019) Laelaps: An energy-efficient seizure detection algorithm from long-term human ieeg recordings without false alarms. In: 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), IEEE, pp 752–757
19. Rahimi A, Kanerva P, Millán JdR, Rabaey JM (2017) Hyperdimensional computing for noninvasive brain-computer interfaces: Blind and one-shot classification of eeg error-related potentials. In: 10th EAI Int. Conf. on Bio-inspired Information and Communications Technologies, CONF
20. Kanerva P (2009) Hyperdimensional computing: an introduction to computing in distributed representation with high-dimensional random vectors. Cognit comput 1(2):139–159
21. Datta S, Antonio RA, Ison AR, Rabaey JM (2019) A programmable hyper-dimensional processor architecture for human-centric iot. IEEE J Emerg Select Topics Circuits Syst 9(3):439–452
22. Kleyko D, Osipov E (2017) No two brains are alike: Cloning a hyperdimensional associative memory using cellular automata computations. In: First International Early Research Career Enhancement School on Biologically Inspired Cognitive Architectures, Springer, pp 91–100
23. Kleyko D, Frady EP, Sommer FT (2020) Cellular automata can reduce memory requirements of collective-state computing. arXiv preprint arXiv:2010.03585
24. Soleymani M, Villaro-Dixon F, Pun T, Chanel G (2017) Toolbox for emotional feature extraction from physiological signals (teap). Fronti ICT 4:1
25. Zhang X, Liu J, Shen J, Li S, Hou K, Hu B, Gao J, Zhang T (2020) Emotion recognition from multimodal physiological signals using a regularized deep fusion of kernel machine. IEEE Trans Cybern. 51(9):4386–4399
26. Menon A, Sun D, Aristio M, Liew H, Lee K, Rabaey JM (2021) A highly energy-efficient hyperdimensional computing processor for wearable multi-modal classification. In: 2021 IEEE Biomedical Circuits and Systems Conference (BioCAS), IEEE, pp 1–4

Publisher's Note