

Premature Ventricular Contraction Beat Classification via Hyperdimensional Computing

Brandon W. Ung, Lulu Ge, and Keshab K. Parhi

Dept. of Electrical and Computer Engineering

University of Minnesota, Minneapolis, MN, USA

{ung00001, ge000567, parhi}@umn.edu

Abstract—Hyperdimensional computing (HD) is an emerging brain-inspired paradigm used for machine learning classification tasks. It manipulates ultra-long vectors—*hypervectors*—using simple operations, which allows for fast learning, energy efficiency, noise tolerance, and a highly parallel distributed framework. HD computing has shown a significant promise in the area of biological signal classification. This paper addresses group-specific premature ventricular contraction (PVC) beat detection with HD computing using the data from the MIT-BIH arrhythmia database. Temporal, heart rate variability (HRV), and spectral features are extracted, and minimal redundancy maximum relevance (mRMR) is used to rank and select features for classification. Three encoding approaches are explored for mapping the features into the HD space. The HD computing classifiers can achieve a PVC beat detection accuracy of 97.7% accuracy, compared to 99.4% achieved by more computationally complex methods such as convolutional neural networks (CNNs).

I. INTRODUCTION

Hyperdimensional (HD) computing is an emerging machine learning model inspired by the cognitive functions of the brain. It is based on the cognitive model proposed by Kanerva, where information is encoded and equally distributed across the high dimensional vectors, which are referred to as *hypervectors*, typically with the dimensionality $d \geq 1,000$ [1]. Rather than computing with real numbers like the typical cognitive computing model, HD computing manipulates hypervectors to perform cognitive tasks such as pattern recognition instead of conventional numeric computation [2]. Furthermore, the HD computing model can quickly learn from provided training data in a one-shot learning manner with acceptable test accuracy [3]. One of the strongest advantages of HD computing is that an energy-efficient classifier can be used on Internet of Things (IoT) devices deployed for real-time learning and classification [4]. The best performing HD models can be used with acceptable accuracy, but may be outperformed by low-level neural networks as illustrated in [5], [6].

HD computing has been applied to various cognitive recognition tasks during its emergence. These tasks can include language recognition [7], DNA sequencing [8], and text classification [9]. One of the most prominent applications of HD computing lies in biological signal classification. Literature has illustrated the success of HD computing in EEG event and

seizure detection [10]–[12], EMG-based gesture classification [13], and biological gender classification [14]. One of the less explored fields in which HD computing could be applied to is in classification of electrocardiography (ECG) signals. Specifically, there is a gap in literature illustrating the applicability of HD computing in classification of cardiac arrhythmia.

Cardiac arrhythmia is the variation of normal heart rate and rhythm which is characterized by abnormal pulses [15]. In a recent survey [16], cardiac rhythm abnormalities have been found present in 2.3% of the adult population and in 4.8% for those ages 65 to 73. Cardiac rhythm abnormalities have been associated with a number of substantial morbidity costs, including increased risks of stroke [17], hospitalization [18], and sudden death [19]. There are several beat classes recommended for detection by the ANSI/AAMI standard for detection of arrhythmia, and many classifier models have been developed to categorize recorded ECG data into these different classes. Many features can be extracted for automatic detection of beat type, including heart rate variability (HRV), temporal, and spectral features [20]–[23].

This paper addresses a group-specific arrhythmic beat classification using binary HD computing, specifically premature ventricular contractions (PVCs). PVCs are prevalent in 40% to 75% subjects on Holter monitoring [24], [25], and they represent an increased risk of sudden death and heart failure in the presence of structural heart disease [19], [26]. HD computing is used for binary classification between heartbeats classified as normal and PVC. A combination of HRV, temporal, and spectral features are extracted from the ECG data. Then, feature ranking and selection are performed through the minimal redundancy maximum relevance (mRMR) algorithm, which is based on maximum statistical dependency based on mutual information [27].

The remainder of this paper is organized as follows. Section II illustrates the theoretical framework and concepts behind binary HD computing and the proposed three encoding approaches. Section III presents the ECG data from MIT-BIH Arrhythmia database, feature extraction, and feature selection for classification. Section IV highlights the results using the proposed three HD computing classifiers. Finally, Section V concludes the paper.

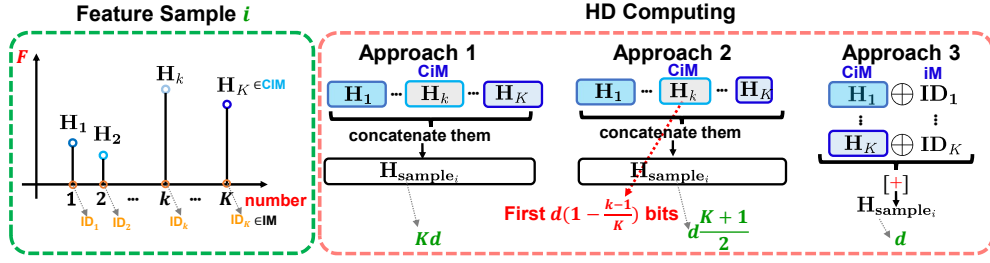


Fig. 1. Three HD encoding approaches for a feature sample. The corresponding dimensionality of the sample hypervector is Kd , $d\frac{K+1}{2}$ and d , respectively.

II. BACKGROUND AND METHODS

A. Basics

The components for hypervectors in HD computing can be binary, bipolar, integer, real, and complex values [1], [2]. However, this paper employs binary hypervectors, which are more energy efficient and require less hardware resources [28].

Three operations are typically used in HD computing to transform hypervectors: multiplication, addition, and permutation (MAP). In terms of multiplication, two hypervectors are bound through point-wise multiplication. For binary HD computing, the resulting hypervector takes on the form $\mathbf{X} = \mathbf{A} \oplus \mathbf{B}$, where $\mathbf{X}$ is exactly the bit-wise XOR of two input hypervectors $\mathbf{A}$ and $\mathbf{B}$. Addition in binary HD computing is a point-wise operation that generates the result using *majority rule*. Permutation is a shuffling operation that can be easily implemented as a circular shift in hardware. Different combinations of MAP operations can be used to transform input hypervectors into a single resulting hypervector for usage in classification.

B. Classification with HD Computing

In HD classification, *class* hypervectors are generated based on the training data, whereas *query* hypervectors are generated by the test data. Afterwards, a similarity measurement will be conducted between the query and class hypervectors. For binary hypervectors, the established check for similarity is the Hamming distance as computed by (1), where d represents the dimensionality of hypervectors. For identical hypervectors, the result $\text{Ham}(\mathbf{A}, \mathbf{B}) = 0$, indicating that no bits are different. In contrast, completely orthogonal and dissimilar hypervectors are represented by $\text{Ham}(\mathbf{A}, \mathbf{B}) = 0.5$ [2]. For classification, the predicted label of the test data is determined by the minimum Hamming distance.

$$\text{Ham}(\mathbf{A}, \mathbf{B}) = \frac{1}{d} \sum_{i=1}^d 1_{\mathbf{A}(i) \neq \mathbf{B}(i)}. \quad (1)$$

When building the HD classifiers, two memories are typically used to store the *seed* hypervectors: item memory (IM) and continuous item memory (CiM). IM stores the randomly generated hypervectors which are completely orthogonal to each other, whereas CiM stores correlated hypervectors. For data representation, hypervectors in IM are used to encode discrete information, while those in CiM are employed to represent correlated values in order to preserve similarity. A CiM can be built through generating a single binary hypervector

as the *base* hypervector and then flipping a distinct number of randomly selected bits based on the required quantization level M of the CiM. As described in [11], for an M -level CiM with M hypervectors, the minimum feature value will be mapped to hypervector $\mathbf{L}_1$ and for each consecutive hypervector $\mathbf{L}_m$, $\frac{d}{2M}$ bits will be flipped until the final vector $\mathbf{L}_M$ is orthogonal to the original $\mathbf{L}_1$. In this paper, M is 50 for all CiMs.

C. Encoding Approaches

Three approaches are employed in this paper to encode feature samples in HD computing.

1) *Approach 1 (Concatenation of Hypervectors)*: Inspired by [11], Approach 1 solely harnesses the seed hypervectors in CiMs to represent each feature sample. As shown in Fig. 1, a sample i contains K different features, indicating K different quantization schemes, where $1 \leq i \leq N_c$ and N_c is the total number of feature samples in the training data for a certain class c . An M -level CiM is built to contain level hypervectors $\{\mathbf{L}_1, \dots, \mathbf{L}_M\}$. Based on the statistics of a certain feature k , the corresponding feature value is quantified to the value bins of the CiM and mapped to the CiM as hypervector $\mathbf{H}_k$. Thus, K feature hypervectors are used and then concatenated to form the sample hypervector $\mathbf{H}_{\text{sample}_i}$ as shown in (2a), where $(\cdot)$ denotes the concatenation operation. The final class hypervector $\mathbf{H}_{\text{class}_c}$ for a certain class are generated by adding all N_c sample hypervectors as shown in (2b), where $[\cdot]$ indicates the majority rule.

$$\mathbf{H}_{\text{sample}_i} = (\mathbf{H}_1, \dots, \mathbf{H}_k, \dots, \mathbf{H}_K), \text{ where } \mathbf{H}_k \in \{\mathbf{L}_1, \dots, \mathbf{L}_M\}, \quad (2a)$$

$$\mathbf{H}_{\text{class}_c} = [\mathbf{H}_{\text{sample}_1} + \dots + \mathbf{H}_{\text{sample}_{N_c}}]. \quad (2b)$$

2) *Approach 2 (Ranked Hypervectors)*: Similar to Approach 1, a single CiM is built with M level hypervectors to represent the feature values. However, this proposed Approach 2 considers the feature ranking information to determine the hypervector size for given features. The dimension of the feature hypervectors decreases as the importance of that feature decreases. The highest ranked feature will be encoded by its level hypervector with the maximum bit length d . Then the k_{th} ranked feature will be represented by the first $d(1 - \frac{k-1}{K})$ bits of the corresponding CiM hypervector. The resultant sample hypervector is the concatenation of the K feature hypervectors.

3) *Approach 3 (Short Hypervectors)*: Approach 3 is exactly the record encoding method summarized in [2]. As shown in Fig. 1, to encode a given feature sample, the indexes of the features are represented by the feature *identifier* $\mathbf{ID}$

hypervectors in the IM, while the quantized feature values are encoded by the feature hypervectors in CiM. Equation (3) illustrates how to generate the sample hypervector $\mathbf{H}_{\text{sample}_i}$.

$$\mathbf{H}_{\text{sample}_i} = [\mathbf{H}_1 \oplus \mathbf{ID}_1 + \dots + \mathbf{H}_K \oplus \mathbf{ID}_K], \text{ where } \mathbf{H}_k \in \{\mathbf{L}_1, \dots, \mathbf{L}_M\}. \quad (3)$$

III. EXPERIMENTAL SETUP

A. MIT-BIH Arrhythmia Database

We use the ECG dataset provided by the MIT-BIH arrhythmia database, which contains 48 half-hour recordings of two-channel ambulatory ECG data recorded between 1975 and 1979 [29]. The data is recorded at a sampling frequency of 360 Hz per channel within a 10 mV range. Based on cardiologist review, annotations for beat types and their respective locations can be found in the annotation files. Refer to [29] for more details.

B. Flow Chart of Proposed Algorithm

Figure 2 shows the proposed algorithm for PVC detection. 1). The original 48 ECG recordings are divided into two groups as described in Table I. Recordings for patients with paced beats (patients 102, 104, 107, and 214) are thrown out based on AAMI standards. Within the 30-min recordings, the first 5-min of data are used for training while the remaining 25-min of data are used for test. 2). Since the recordings are a mix of normal ([N]), PVC ([V]) and other types of beats, we only extract the information for normal and PVC beats. All other types of beats are thrown out. The segmentation is conducted by a 400-ms window with 200-ms before and after the already marked R-wave peak location. Therefore, the original 30-min recordings are split into 400-ms segments. 3). Feature extraction is performed over all segments. Then, the feature set is subjected to a feature selection step using the mRMR algorithm [27], to select the top K features. Therefore, each 400-ms segment corresponds to a feature sample with K features. 4). These feature samples are fed into the HD classifiers using the proposed approaches in Section II-C.

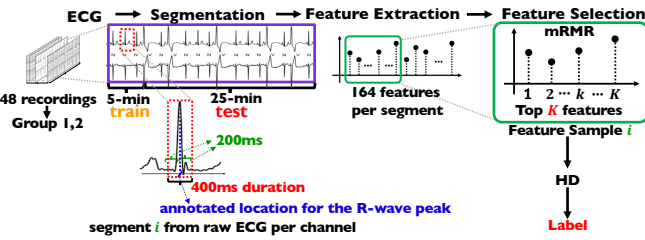


Fig. 2. Flow chart of proposed algorithm.

C. Feature Extraction

Literature has shown that a variety of features can be used for classification of the different types of beats present in MIT-BIH Arrhythmia database [30]–[32]. These can include HRV, temporal (morphological), and spectral (frequency) features. A total of 164 beat features are extracted in this paper consisting of 6 HRV, 22 temporal, and 136 spectral features.

HRV features are those determined from measuring the duration between adjacent R peaks, called the R-R interval. The annotated R-peaks provided by the database files are used

TABLE I
PATIENT GROUPINGS

Patient Group	Recording	Training Beat Count		Test Beat Count	
		PVC	Normal	PVC	Normal
Group 1	101,106,108,109, 110,112,114,115, 116,118,119,122, 124,201,203,205, 207,208,215,220, 223,230	554	6434	3107	31391
Group 2	100,103,105,111, 113,117,121,123, 200,202,210,212, 213,214,219,221, 222,228,231,232, 233,234	512	5743	2683	30449

to find adjacent R-peak locations surrounding a given beat. Examples of features extracted for HRV analysis include the R-R intervals, relative R-R intervals, and root mean square of successive differences (RMSSD) for beats near the test beats.

Temporal beat features are extracted from the time domain within the 400-ms segments. For each channel, statistical measurements of the beat window are extracted. These include the mean, maximum, minimum, peak-to-RMS, kurtosis, and skewness values. The spectral features are extracted from each lead using the same 400-ms beat window as during temporal feature extraction. These features come from the orthogonal wavelet transform features described in [32], 180-point Fast Fourier Transform (FFT), and band powers. Similar to the temporal features, statistical measurements of the absolute values for the single-sided real FFT are extracted. Also, the absolute values of each 1-20 Hz values generated from the FFT are recorded. For band power features, the band power and relative band power are extracted for band ranges 0.5-12 Hz, 12-25 Hz, 25-40 Hz, 40-57 Hz, and 63-90 Hz.

D. Performance Evaluation

For this group-specific PVC classification, the data is divided into two folds since there are two groups. Within each fold, the classification model is trained from one group and then tested over the whole two groups. The final performance shown in this paper is the average results of the two folds. Conforming to the state-of-the-art methods for this PVC detection [33]–[35], test accuracy, sensitivity, specificity and precision are used as the main metrics to evaluate the performance.

IV. EXPERIMENTAL RESULTS

The number of selected features utilized for encoding in the HD space are varied between 5, 10, and 20. Table II highlights the top twenty features ranked by the mRMR feature selection algorithm collected from the training data of each patient group. As demonstrated, there are 10 common selected features, although their ranking differs between the two groups of training data. The top three features for both training and test data are illustrated in Figs. 3 and 4. Furthermore, the impact of hypervector size is explored by varying the dimensionality d to be 100, 200, 500, 1,000, and 2,000 for all approaches.

TABLE II
TOP 20 SELECTED FEATURES USING mRMR

Rank	Group 1	Group 2
1	Ch. 1 Temporal Kurtosis	Ch. 1 Temporal Peak-to-RMS
2	Ch. 2 DWT Detail Lvl. 7 Coef.	Prev. RR Interval / Ave. RR Intervals of 10s
3	Ch. 2 Rel. Power Band 40-57 Hz	Ch. 2 DWT Detail Lvl. 4, 3rd Coef.
4	Prev. RR Interval / Ave. RR Intervals of 10s	Ch. 1 DWT Detail Lvl. 4, 6th Coef.
5	Ch. 1 DWT Detail Lvl. 6, 1st Coef.	Next RR Interval / Ave. RR Intervals of 10s
6	Ch. 1 DWT Detail Lvl. 6, 2nd Coef.	Ch. 1 DWT Detail Lvl. 5, 1st Coef.
7	Next RR Interval / Ave. RR Intervals of 10s	Ch. 2 DWT Detail Lvl. 5 St. Dev.
8	Ch. 2 Rel. Power Band 0.5-12 Hz	Ch. 2 Rel. Power Band 40-57 Hz
9	Ch. 2 DWT Detail Lvl. 5 St. Dev.	Ch. 1 DWT Detail Lvl. 6, 1st Coef.
10	Ch. 1 Rel. Power Band 40-57 Hz	Ch. 1 Temporal Min.
11	Ch. 1 FFT Amp., 1 Hz	Ch. 2 DWT Detail Lvl. 4, 5th Coef.
12	Ch. 1 Temporal Peak-to-RMS	Ch. 2 DWT Detail Lvl. 5, 3rd Coef.
13	Ch. 2 DWT Detail Lvl. 4, 5th Coef.	Ch. 1 DWT Detail Lvl. 4 Variance
14	Ch. 2 DWT Detail Lvl. 6 Variance	Ch. 1 Temporal Kurtosis
15	Ch. 1 Rel. Power Band 12-25 Hz	Ch. 1 DWT Detail Lvl. 5 St. Dev.
16	Ch. 2 Temporal Peak-to-RMS	Ch. 2 DWT Detail Lvl. 4 Skewness
17	Ch. 2 DWT Detail Lvl. 5, 3rd Coef.	Ch. 1 FFT Kurtosis
18	Ch. 1 DWT Detail Lvl. 4 Variance	Ch. 1 DWT Detail Lvl. 6, 2nd Coef.
19	Ch. 1 DWT Detail Lvl. 6 St. Dev.	Ch. 1 DWT Detail Lvl. 4 Max
20	Ch. 1 DWT Detail Lvl. 7 Coef.	Ch. 1 FFT Amp., 2 Hz

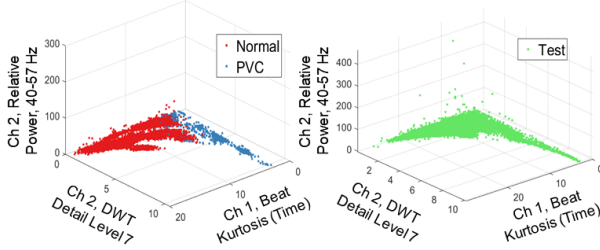


Fig. 3. 3D scatter plot of the PVC and normal beats top three mRMR training features for Group 1 (left panel) and the test features for Group 2.

Simulation results are shown in Tables III. 1). For Approach 1, modifying feature hypervector size does not affect performance as demonstrated in Table III. However, the number of features used does impact performance. The highest accuracy measured for Approach 1 is using 10 features with 97.5% accuracy. 2). Approach 2 achieves the best performance among these three approaches: 97.7% accuracy, 90.4% sensitivity, 98.4% specificity, 85.1% precision, and AUC of 0.944 using the maximum number of selected features and dimensions. With just 10 features, the AUC of the model increases beyond 0.950 starting with maximum dimension of 1000 3). Approach 3 achieves its best performance in regards to AUC when the hypervector size is 1000 bits and beyond. This is indicative that lower overall dimensionality of hypervectors may not store enough information for an accurate classifier.

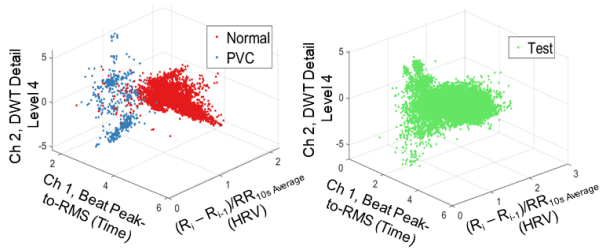


Fig. 4. 3D scatter plot of the PVC and normal beats top three mRMR training features for Group 2 (left panel) and the test features for Group 1.

TABLE III
SIMULATION RESULTS FOR THREE APPROACHES

Approach 1: Concatenation of Hypervectors						
Selected Features	Dimension per Feature	Measure				
		Sensitivity	Specificity	Precision	Accuracy	AUC
5	100	91.8%	97.3%	79.6%	96.8%	0.945
	200	91.8%	97.3%	79.6%	96.8%	0.945
	500	91.8%	97.3%	79.6%	96.8%	0.945
	1000	91.8%	97.3%	79.6%	96.8%	0.945
	2000	91.8%	97.3%	79.6%	96.8%	0.945
10	100	82.6%	99.0%	88.5%	97.5%	0.908
	200	82.6%	99.0%	88.5%	97.5%	0.908
	500	82.6%	99.0%	88.5%	97.5%	0.908
	1000	82.6%	99.0%	88.5%	97.5%	0.908
	2000	82.6%	99.0%	88.5%	97.5%	0.908
20	100	84.5%	98.4%	83.9%	97.2%	0.915
	200	84.5%	98.4%	83.9%	97.2%	0.915
	500	84.5%	98.4%	83.9%	97.2%	0.915
	1000	84.5%	98.4%	83.9%	97.2%	0.915
	2000	84.5%	98.4%	83.9%	97.2%	0.915
Approach 2: Ranked Hypervectors						
Selected Features	Maximum Dimension	Measure				
		Sensitivity	Specificity	Precision	Accuracy	AUC
5	100	92.9%	96.3%	74.9%	96.1%	0.946
	200	93.6%	95.5%	71.9%	95.3%	0.946
	500	93.9%	95.6%	72.3%	95.5%	0.947
	1000	93.8%	96.0%	73.5%	95.8%	0.949
	2000	94.0%	95.5%	71.6%	95.4%	0.947
10	100	92.9%	96.5%	76.4%	96.2%	0.947
	200	91.3%	97.3%	79.2%	96.8%	0.943
	500	92.3%	97.4%	80.2%	97.0%	0.949
	1000	93.1%	97.3%	79.2%	96.9%	0.952
	2000	92.5%	97.4%	80.1%	97.0%	0.950
20	100	92.0%	98.0%	82.7%	97.5%	0.950
	200	90.5%	98.2%	84.0%	97.5%	0.944
	500	90.2%	98.4%	85.0%	97.7%	0.943
	1000	90.0%	98.5%	85.5%	97.7%	0.942
	2000	90.4%	98.4%	85.1%	97.7%	0.944
Approach 3: Short Hypervectors						
Selected Features	Dimension	Measure				
		Sensitivity	Specificity	Precision	Accuracy	AUC
5	100	91.5%	95.5%	71.1%	95.1%	0.935
	200	90.8%	96.5%	71.6%	96.0%	0.936
	500	90.0%	95.6%	70.0%	95.1%	0.928
	1000	91.8%	96.2%	71.6%	95.8%	0.940
	2000	92.4%	96.0%	73.3%	95.7%	0.942
10	100	67.6%	89.4%	52.9%	87.5%	0.785
	200	82.4%	95.8%	72.6%	94.7%	0.891
	500	79.5%	95.5%	68.7%	94.2%	0.875
	1000	90.5%	94.8%	68.0%	94.5%	0.927
	2000	90.9%	93.8%	67.3%	93.6%	0.923
20	100	71.2%	97.2%	70.2%	94.9%	0.842
	200	89.7%	91.7%	52.9%	91.6%	0.907
	500	78.7%	98.4%	82.8%	96.7%	0.886
	1000	90.9%	96.7%	75.1%	96.2%	0.938
	2000	86.5%	97.5%	78.4%	96.6%	0.920

TABLE IV
COMPARISON WITH LITERATURE

Method	Features	Accuracy	Sensitivity	Specificity	Precision
HD Computing with Feature Ranked Hypervectors	20	97.7	90.4	98.4	85.1
Kth Nearest-Neighbors [33]	26	-	96.7	97.2	-
Variation of Principal Directions [34]	150	98.77	96.12	98.96	86.48
CNN + LSTM + Rules Inference [35]	150	99.41	97.59	99.54	93.55

The HD computing model using mRMR algorithm with variable hypervector sizes for group-specific PVC classification is compared against the state-of-the-art methods as described in Table IV. These methods include a Kth nearest-neighbor classifier [33], a PCA-based classifier [34], and combination of CNN and LSTM [35]. These methods achieve high accuracy with over 96% sensitivity. The HD model is able to perform at a similar accuracy level, albeit at the cost of sensitivity and precision.

V. CONCLUSION

This paper conducts a group-specific binary classification for PVC detection by analyzing the ECG data. Out of the totally extracted 164 features, up to 20 features are selected by the mRMR algorithm. Based on the top-ranked features, three HD encoding approaches are employed in this paper. Among them, Approach 2 achieves the best performance: 97.7% accuracy, 90.4% sensitivity, 98.4% specificity and 85.1% precision. This performance is comparable to the state-of-the-art results using 20 features, with an acceptable performance loss based on the sparse representation of the model. Future work will address multi-class arrhythmia classification using HD computing. Additionally, the low energy property of HD computing for biological signal classification will be explored through synthesis and implementation on an FPGA.

REFERENCES

- [1] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive computation*, vol. 1, no. 2, pp. 139–159, 2009.
- [2] L. Ge and K. K. Parhi, "Classification using hyperdimensional computing: A review," *IEEE Circuits and Systems Magazine*, vol. 20, no. 2, pp. 30–47, 2020.
- [3] A. Rahimi, S. Datta, D. Kleyko, E. P. Frady, B. Olshausen, P. Kanerva, and J. M. Rabaey, "High-dimensional computing as a nanoscale paradigm," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 9, pp. 2508–2521, 2017.
- [4] M. Imani, Z. Zou, S. Bosch, S. A. Rao, S. Salamat, V. Kumar, Y. Kim, and T. Rosing, "Revisiting hyperdimensional learning for FPGA and low-power architectures," in *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 2021, pp. 221–234.
- [5] Y.-C. Chuang, C.-Y. Chang, and A.-Y. A. Wu, "Dynamic hyperdimensional computing for improving accuracy-energy efficiency trade-offs," in *2020 IEEE Workshop on Signal Processing Systems (SiPS)*. IEEE, 2020, pp. 1–5.
- [6] T. Yu, Y. Zhang, Z. Zhang, and C. De Sa, "Understanding hyperdimensional computing for parallel single-pass learning," *arXiv preprint arXiv:2202.04805*, 2022.
- [7] A. Rahimi, P. Kanerva, and J. M. Rabaey, "A robust and energy-efficient classifier using brain-inspired hyperdimensional computing," in *Proceedings of the 2016 International Symposium on Low Power Electronics and Design*, 2016, pp. 64–69.
- [8] M. Imani, T. Nassar, A. Rahimi, and T. Rosing, "HDNA: Energy-efficient DNA sequencing using hyperdimensional computing," in *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 2018, pp. 271–274.
- [9] F. R. Najafabadi, A. Rahimi, P. Kanerva, and J. M. Rabaey, "Hyperdimensional computing for text classification," in *Design, automation test in Europe conference exhibition (DATE), University Booth*, 2016, pp. 1–1.
- [10] A. Burrello, L. Cavigelli, K. Schindler, L. Benini, and A. Rahimi, "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, 2019, pp. 752–757.
- [11] L. Ge and K. K. Parhi, "Applicability of hyperdimensional computing to seizure detection," *IEEE Open Journal of Circuits and Systems*, vol. 3, pp. 59–71, 2022.
- [12] A. Rahimi, A. Tchouprina, P. Kanerva, J. d. R. Millán, and J. M. Rabaey, "Hyperdimensional computing for blind and one-shot classification of EEG error-related potentials," *Mobile Networks and Applications*, vol. 25, no. 5, pp. 1958–1969, 2020.
- [13] S. Benatti, F. Montagna, V. Kartsch, A. Rahimi, D. Rossi, and L. Benini, "Online learning and classification of EMG-based gestures on a parallel ultra-low power platform using hyperdimensional computing," *IEEE transactions on biomedical circuits and systems*, vol. 13, no. 3, pp. 516–528, 2019.
- [14] R. Billmeyer and K. K. Parhi, "Biological gender classification from fmri via hyperdimensional computing," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 578–582.
- [15] C. Antzelevitch and A. Burashnikov, "Overview of basic mechanisms of cardiac arrhythmia," *Cardiac electrophysiology clinics*, vol. 3, no. 1, pp. 23–45, 2011.
- [16] S. Khurshid, S. H. Choi, L.-C. Weng, E. Y. Wang, L. Trinquart, E. J. Benjamin, P. T. Ellinor, and S. A. Lubitz, "Frequency of cardiac rhythm abnormalities in a half million adults," *Circulation: Arrhythmia and Electrophysiology*, vol. 11, no. 7, p. e006273, 2018.
- [17] P. A. Wolf, R. D. Abbott, and W. B. Kannel, "Atrial fibrillation: a major contributor to stroke in the elderly: the Framingham study," *Archives of internal medicine*, vol. 147, no. 9, pp. 1561–1564, 1987.
- [18] D. H. Murman, A. J. McDonald, A. J. Pelletier, and C. A. Camargo Jr, "US emergency department visits for supraventricular tachycardia, 1993–2003," *Academic Emergency Medicine*, vol. 14, no. 6, pp. 578–581, 2007.
- [19] F. C. Messineo, "Ventricular ectopic activity: prevalence and risk," *The American journal of cardiology*, vol. 64, no. 20, pp. J53–J56, 1989.
- [20] A. S. Alvarado, C. Lakshminarayanan, and J. C. Principe, "Time-based compression and classification of heartbeats," *IEEE transactions on biomedical engineering*, vol. 59, no. 6, pp. 1641–1648, 2012.
- [21] P. De Chazal, M. O'Dwyer, and R. B. Reilly, "Automatic classification of heartbeats using ECG morphology and heartbeat interval features," *IEEE transactions on biomedical engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.
- [22] R. Benali, F. Berekci Reguig, and Z. Hadj Slimane, "Automatic classification of heartbeats using wavelet neural network," *Journal of medical systems*, vol. 36, no. 2, pp. 883–892, 2012.
- [23] F. I. Alarsan and M. Younes, "Analysis and classification of heart diseases using heartbeat features and machine learning algorithms," *Journal of big data*, vol. 6, no. 1, pp. 1–15, 2019.
- [24] G. A. Ng, "Treating patients with ventricular ectopic beats," *Heart*, vol. 92, no. 11, pp. 1707–1712, 2006.
- [25] H. L. Kennedy, J. A. Whitlock, M. K. Sprague, L. J. Kennedy, T. A. Buckingham, and R. J. Goldberg, "Long-term follow-up of asymptomatic healthy subjects with frequent and complex ventricular ectopy," *New England Journal of Medicine*, vol. 312, no. 4, pp. 193–197, 1985.
- [26] J. B. KoSTIS, K. McCrone, A. Moreyra, S. Gotzoyannis, N. M. Aglitz, N. Natarajan, and P. Kuo, "Premature ventricular complexes in the absence of identifiable heart disease," *Circulation*, vol. 63, no. 6, pp. 1351–1356, 1981.
- [27] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.
- [28] A. Patyk-Łońska, M. Czachor, and D. Aerts, "A comparison of geometric analogues of holographic reduced representations, original holographic reduced representations and binary spatter codes," in *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2011, pp. 221–228.
- [29] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.
- [30] Z. Ebrahimi, M. Loni, M. Daneshdalan, and A. Gharehbaghi, "A review on deep learning methods for ECG arrhythmia classification," *Expert Systems with Applications: X*, vol. 7, p. 100033, 2020.
- [31] B. M. Asl, S. K. Setarehdan, and M. Mohebbi, "Support vector machine-based arrhythmia classification using reduced features of heart rate variability signal," *Artificial Intelligence in Medicine*, vol. 44, no. 1, pp. 51–64, 2008.
- [32] M. H. Song, J. Lee, S. P. Cho, K. J. Lee, and S. K. Yoo, "Support vector machine based arrhythmia classification using reduced features," *Artificial Intelligence in Medicine*, 2005.
- [33] I. Christov, I. Jekova, and G. Bortolan, "Premature ventricular contraction classification by the Kth nearest-neighbours rule," *Physiological measurement*, vol. 26, no. 1, pp. 123–130, 2005.
- [34] R. Zarei, J. He, G. Huang, and Y. Zhang, "Effective and efficient detection of premature ventricular contractions based on variation of principal directions," *Digital Signal Processing*, vol. 50, pp. 93–102, 2016.
- [35] F.-y. Zhou, L.-p. Jin, and J. Dong, "Premature ventricular contraction detection combining deep neural networks and rules inference," *Artificial Intelligence in Medicine*, vol. 79, pp. 42–51, 2017.