Hyperdimensional Computing With Local Binary Patterns: One-Shot Learning of Seizure Onset and Identification of Ictogenic Brain Regions Using Short-Time iEEG Recordings

Original
Hyperdimensional Computing With Local Binary Patterns: One-Shot Learning of Seizure Onset and Identification of Ictogenic Brain Regions Using Short-Time iEEG Recordings / Burrello, A., Schindler, K., Benini, L., Rahimi, A.. - In: IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING. - ISSN 0018-9294. - 67:2(2020), pp. 601-613. [10.1109/TBME.2019.2919137]

Availability:
This version is available at: 11583/2978552 since: 2023-05-16T15:01:59Z

Publisher:
IEEE-INST ELECTRICAL ELECTRONICS ENGINEERS INC

Published
DOI:10.1109/TBME.2019.2919137

Hyperdimensional Computing with Local Binary Patterns: One-shot Learning of Seizure Onset and Identification of Ictogenic Brain Regions using Short-time iEEG Recordings

Alessio Burrello, Kaspar Schindler, Luca Benini, Abbas Rahimi

◆

Abstract—*Objective:* We develop a fast learning algorithm combining symbolic dynamics and brain-inspired hyperdimensional computing for both seizure onset detection and identification of ictogenic (seizure generating) brain regions from intracranial electroencephalography (iEEG). *Methods:* Our algorithm first transforms iEEG time series from each electrode into symbolic local binary pattern codes from which a *holographic distributed representation* of the brain state of interest is constructed across all the electrodes and over time in a hyperdimensional space. The representation is used to quickly learn from few seizures, detect their onset, and identify the spatial brain regions that generated them. *Results:* We assess our algorithm on our dataset that contains 99 short-time iEEG recordings from 16 drug-resistant epilepsy patients being implanted with 36 to 100 electrodes. For the majority of the patients (10 out of 16), our algorithm quickly learns from one or two seizures and perfectly (100%) generalizes on novel seizures using k-fold cross-validation. For the remaining six patients, the algorithm requires three to six seizures for learning. Our algorithm surpasses the state-of-the-art including deep learning algorithms by achieving higher specificity (94.84% vs. 94.77%) and macroaveraging accuracy (95.42% vs. 94.96%), and 74× lower memory footprint, but slightly higher average latency in detection (15.9 s vs. 14.7 s). Moreover, the algorithm can reliably identify (with a p-value < 0.01) the relevant electrodes covering an ictogenic brain region at two levels of granularity: cerebral hemispheres and lobes. *Conclusion and significance:* Our algorithm provides: (1) a unified method for both learning and classification tasks with end-to-end binary operations; (2) one-shot learning from seizure examples; (3) linear computational scalability for increasing number of electrodes; (4) generation of transparent codes that enables post-translational support for clinical decision making. Our source code and anonymized iEEG dataset are freely available at http://ieeg-swez.ethz.ch.

Index Terms—iEEG, one-shot learning, local binary patterns, symbolic dynamics, hyperdimensional computing, seizure detection, localization of seizure onset zone

1 INTRODUCTION

Epilepsy is a severe and prevalent chronic neurological disorder affecting 1–2% of the world's population [1]. One third of epilepsy patients continue to suffer from seizures despite best possible pharmacological treatment [2]. For these patients with so-called drug-resistant epilepsy [3], various algorithms based on intracranial electroencephalography (iEEG) recording are proposed to detect the onset of seizures [1]. Complementary to this approach, efficient and robust algorithms are required to not only detect the seizure onset but also to identify the ictogenic (i.e. seizure-generating) brain regions for possible surgical removal [4], [5]. Such detection and identification has to be extremely accurate, based on relatively scarce data, as typically only few seizures are recorded in the epilepsy monitoring unit due to the limited duration of iEEG recordings (typically 1 to 3 weeks [6] to minimize patient's discomfort).

The iEEG currently provides the best spatial resolution and the highest signal-to-noise ratio (SNR) of electrical brain activity recordings [1]. Recent studies have shown successful application of machine learning methods [1], [7]–[9] using iEEG signals to detect two distinct states of brain activity in patients with epilepsy, i.e., interictal (= between seizures) and ictal (= during seizures). These methods are based on extracting useful features followed by traditional supervised machine learning methods (such as random forest [1], support vector machines [7], Bayesian analysis [9], artificial neural networks [7]), and more recently deep learning algorithms [8]. These methods are however seriously challenged by the need to reliably detect seizures from a small number of examples. This is due to the patient-specific nature of seizure dynamics (i.e. activity patterns at onset, propagation and termination of seizures), and to the inherent asymmetry in the iEEG recording, namely that the ratio of interictal to ictal segments is typically very large.

In addition, these conventional methods face other important challenges including: (1) The outcome of their learning is often a "black box" that is not *transparent* to an expert neurologist, hence cannot be analyzed for better diagnosis, e.g., precisely delineating the ictogenic brain regions. (2) Their high computational complexity and memory

---

Manuscript received August 21, 2018; revised October 10, 2018 and April 2, 2019; accepted May 7, 2019. Support was received from the ETHZ Postdoctoral Fellowship Program, the Marie Curie Actions for People COFUND Program, the Hasler Foundation under project 18082, and the EU's Horizon 2020 Research and Innovation Program under grant No 780215.

A. Burrello, L. Benini, and A. Rahimi are with the Department of Information Technology and Electrical Engineering at the ETH Zurich, 8092 Zurich, Switzerland. L. Benini is also with the Department of Electrical, Electronic and Information Engineering, University of Bologna, 40136 Bologna, Italy. E-mails: bualessi@student.ethz.ch, lbenini@iis.ee.ethz.ch, abbas@iis.ee.ethz.ch

K. Schindler is with the Sleep-Wake-Epilepsy-Center, Department of Neurology, Inselspital, Bern University Hospital, University Bern, 3010 Bern, Switzerland. Email: kaspar.schindler@insel.ch

demands render them unsuitable for real-time detection on resource-limited wearable or implantable devices. (3) Their offline training and slow (iterative) training time prevent them from online and incremental learning from new seizure occurrences, hence they cannot be quickly adapted to new dynamics. (4) They operate with few electrodes, e.g. 6 [9], 8 [10], and 16–72 [1] electrodes. However, a larger number of electrodes is mandatory to properly assess the spatio-temporal evolution and spreading of epileptic seizures [5], [11], [12] and to properly identify the borders of the seizure onset zone for surgical resection [13]. Furthermore, recent studies have demonstrated that iEEG recordings from outside the seizure onset zone provide important information about seizure generalization [14] and thus might be helpful to prevent its occurrence and thereby decrease the risk for sudden unexpected death [15]. To address the aforementioned challenges, we present the following contributions:

(1) We propose a single algorithm for both learning and classification tasks by jointly exploiting symbolic dynamics and brain-inspired vector-symbolic architectures. The proposed algorithm combines methods from symbolic dynamics [16]–[18] (Section 2.1) and brain-inspired computing [19] (Section 2.2) that supports one-shot or few-shot learning, i.e. the ability to learn object categories from one or few examples. Symbolic dynamics models a dynamical system by a discrete space consisting of sequences of abstract symbols, each of which corresponds to a state of the system. At the heart of our algorithm is a brain-inspired vector-symbolic computational theory called hyperdimensional (HD) computing [19] that learns quickly by computing with random vectors in a very high dimensionality, also referred to as *hypervectors*. Our proposed algorithm consists of analyzing the occurrence of symbols or patterns—that even bears similarity to classical visual EEG interpretation [16]—followed by one-/few-shot learning. First, as a *symbolization* method, we exploit local binary patterns (LBP) [18] to map a sequence of iEEG samples into a small bit string as a symbol. Second, these symbols are projected into an HD space that enables reliably combining them over time and across electrodes to encode a compact representation (i.e., a prototype vector) for one-shot learning from the state of interest. Further, we use the same algorithm for both learning and classification tasks: the algorithm initially learns from few ictal or few interictal segments by writing the corresponding prototype vector (ictal or interictal) into an associative memory, and then classifies new segments based on Hamming distance among these two *learned* prototype vectors.

(2) Our algorithm is based on binary representations and operators, and is computationally scalable: (i) For every iEEG electrode, the LBP feature extractor directly transforms the time series into symbols as bit strings with limited length. (ii) HD computing then projects the symbols to an HD space and computes a distributed long binary vector that encodes occurrences of the symbols among all electrodes (an approximation method to encode histograms of symbols). (iii) The training and classification are performed by simply bundling and comparing the binary vectors. (iv) The classification decision is followed by a patient-dependent voting to reduce the false alarms. Further, the computational complexity of the algorithm linearly scales for any number of input electrodes. This scalability provides a universal interface to homogeneously cover all patients with different numbers of implanted electrodes (e.g. 36 to 100) and seizure dynamics. The concurrent use of LBP and HD computing enables end-to-end execution of our algorithm with simple binary codes to avoid otherwise expensive operations such as costly floating-point arithmetic (Section 4).

(3) One-shot learning, comparison, code and dataset release. We provide a dataset from 16 drug-resistant epilepsy patients that contains 99 iEEG recordings, each one consisting of a 3 minutes interictal (i.e. immediately pre-ictal) segment and the ictal segment followed by a 3 minutes postictal segment (Section 3). Using this dataset, we compare our algorithm with the state-of-the-art methods in [1], [7], [8] (Section 5). Our algorithm quickly learns from one seizure (for eight patients), or two seizures (for two more patients), and perfectly (100%) generalizes on detecting novel seizures with k-fold cross-validation. For the remaining six patients, the algorithm requires 3–6 seizure examples for learning. Overall, our algorithm surpasses all the state-of-the-art methods: e.g., compared to [8], it achieves higher specificity (94.84% vs. 94.77%) and macroaveraging accuracy (95.42% vs. 94.96%), and a 74× lower memory footprint. Our algorithm has slightly higher average latency in detection (15.9 s vs. 14.7 s), but it can raise an alarm in the first 8% of the ictal window. We also provide free access to our source code and anonymized iEEG dataset at http://ieeg-swez.ethz.ch.

(4) Our algorithm produces transparent codes for identifying seizure-generating brain regions. Due to the well-defined set of arithmetic operations with inverses in HD computing, the learned prototype vectors—i.e. the binary codes derived from the iEEG recordings during the ictal and interictal brain states—are transparent and analyzable with interpretable features. Our algorithm identifies the ictogenic brain regions by measuring the relative distances between the learned prototypes that are produced from different electrodes (Section 6). Such identification is done at two levels of spatial resolution, the cerebral hemispheres and lobes, with p-value < 0.01 (Section 7). This takes the application of our algorithm beyond the traditional scope of seizure onset detection by automatically identifying ictogenic brain regions that can provide more accurate data to better target surgical resection and thus potentially improve post-surgical seizure control. It enables post-translational support for clinical decision making and is in sharp contrast to those machine learning methods that produce "black boxes."

In the rest of this paper, we first provide a background (Section 2) in symbolization and brain-inspired hyperdimensional (HD) computing. We then provide more information about our iEEG dataset in Section 3. We describe in detail how to develop an HD-based algorithm to detect seizures from iEEG signals (Part I: Sections 4 and 5), and to identify ictogenic brain regions (Part II: Sections 6 and 7). A preliminary version of Part I has been reported in [20].

2 BACKGROUND

2.1 Symbolization using Local Binary Patterns (LBP)

A class of data-analysis methods is referred to as symbolization, which describes the process of transforming

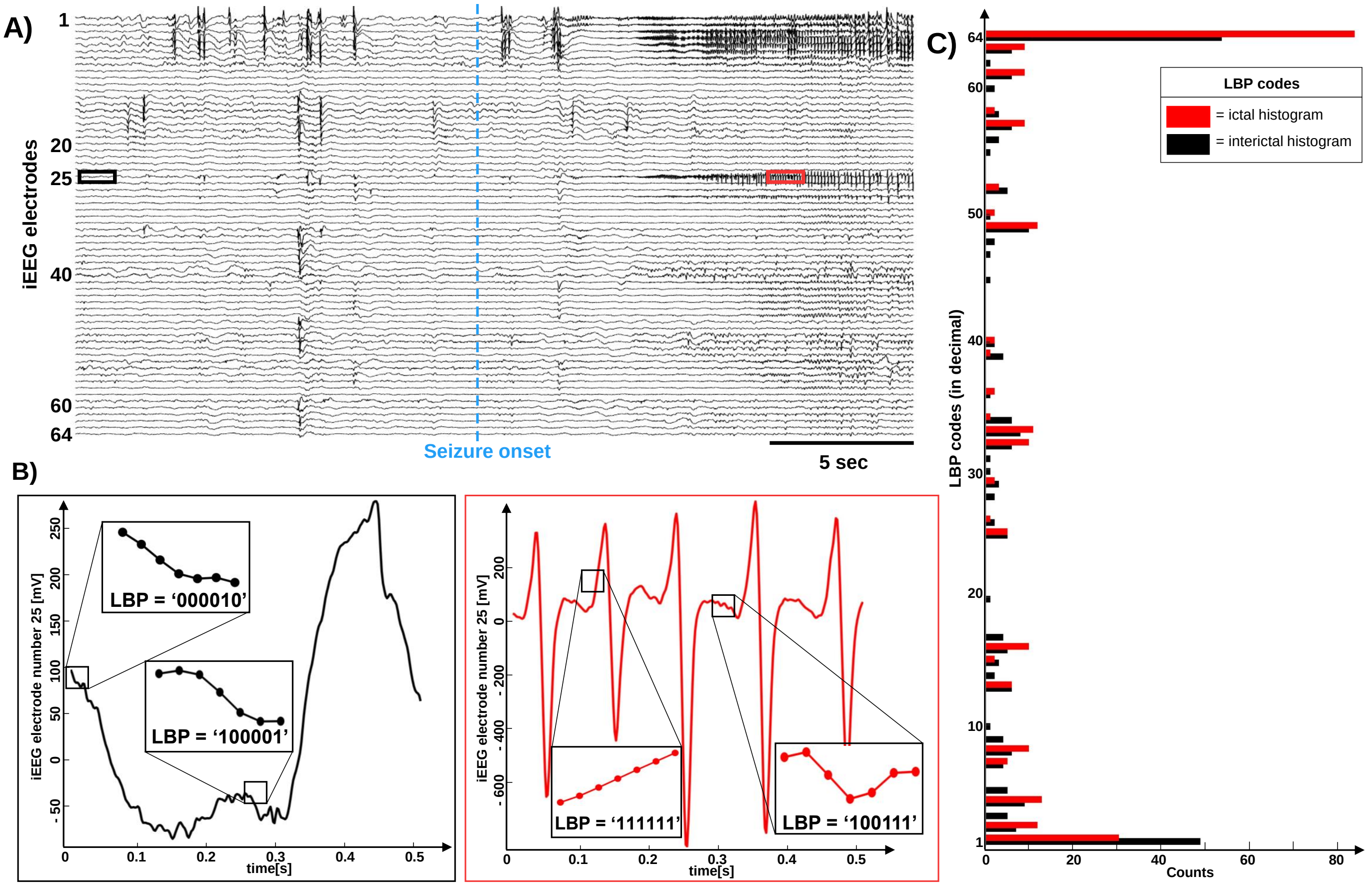


Fig. 1: A) iEEG signals before (interictal) and after (ictal) seizure onset. The blue dotted line marks seizure onset as determined by the visual inspection of an expert (K.S.). **B)** Zoomed in iEEG signals: (1) during the interictal state: the LBP codes are well distributed over almost all the possible codes; (2) during the ictal state, the strongly time-irreversible signals have a predominant portion of a single LBP code; examples of their LBP codes of l=6 are drawn. **C)** The corresponding histograms of the LBP codes inside the 0.5 s windows in **B**.

raw experimental measurements into a series of discrete symbols. Symbolization is particularly interesting for EEG analysis, because as recent experience has clearly demonstrated, it faithfully preserves dominant dynamical signal characteristics while significantly increasing the efficiency of detecting and quantifying information contained in real-world time series [21]. Symbolization may be efficiently achieved by mapping a sequence of iEEG samples into a bit string, i.e. a one-dimensional local binary pattern (LBP) [18], [22]. A LBP code reflects the relational aspects between consecutive values of the iEEG signals, i.e., whether their amplitudes increase or decrease. We observe that LBP codes are more efficient than other closer symbolization methods, e.g., directed horizontal graphs [23] that assign an integer input and output degree to each time point.

Computing a LBP code is straightforward: (1) The iEEG signal samples are converted into a bit string depending on the sign of the temporal difference of adjacent samples. If the difference is positive, we assign a 1 to the sampling point, otherwise a 0. (2) A LBP code of length l is associated with every sampling point by concatenating its bit with the successive $l-1$ bits. Fig. 1B shows the examples of LBP code with l=6. Fig. 1C illustrates how histograms of LBP codes differ between interictal and ictal states. During the interictal state the LBP codes are well distributed over almost all the possible codes. In contrast the ictal window has a predominant portion of a single LBP code and many LBP codes are missing due to the typically slow and asymmetric oscillations evolving during seizures [17], [23].

2.2 Hyperdimensional (HD) Computing

Inspired by the very size of the brain's circuits, we can model neural activity patterns with points of a hyperdimensional space. Hyperdimensional (HD) computing [19] explores this idea by computing with random vectors in a very high dimensionality (d), also referred to as *hypervectors*. To represent basic items, or symbols, HD computing starts by selecting a set of *atomic* vectors: d-dimensional (pseudo)random vectors with independent and identically distributed (i.i.d.) components. This thus conforms to a *holographic* or *holistic* representation: the encoded information is distributed *equally* over all the d components such that no component is more responsible to store any piece of information than another hence maximizing robustness. When the dimensionality is in the thousands, e.g. d=10,000, it yields a huge number of nearly orthogonal atomic vectors (see Section 2.2.1). This lets HD computing to combine two atomic vectors into a new *complex* fixed-width vector using well-defined vector-space operations (Section 2.2.2), while keeping the information of the two atomic vectors with high probability [24]. Hence, it is also called *holographic reduced representations* (HRRs) meaning that the reduced (fixed-width) descriptions have less information about components than the full descriptions [25]. This overall provides a novel perspective on data representations and associated operations with unique features in terms of robustness [26]–[28], speed of learning [29]–[32], and energy efficiency [33], [34].

In this paper, we consider d-dimensional binary random vectors[1] of equally probable 1s and 0s, i.e., dense binary elements of $\{0,1\}^d$. This dense binary coding is also known as binary spatter code [35]. In what follows, we first describe how such vectors can be compared (i.e. a measure of similarity among them), and how they can be combined using HD arithmetic operations.

2.2.1 Measure of Similarity

Using the dense binary coding, the similarity between two vectors is defined by the Hamming distance as the number of components at which they differ. To express the distance on a real scale of 0 to 1, we divide the Hamming distance by d denoted as: $\Delta(X,Y) : \{0,1\}^d \times \{0,1\}^d \rightarrow [0,1]$. In high dimensions, e.g. $d \geq 1000$, most points are $d/2$ bits apart from each other, which yields a normalized Hamming distance of $\Delta \approx 0.5$, and stands for two nearly orthogonal vectors [36]. This stems from the binomial distribution for $p = 1/2$ and $n = d$, where $d/2$ is the mean. Correlated vectors yield $\Delta \approx 0$ whereas $\Delta \approx 1$ implies anti-correlation [19].

HD computing begins by randomly generating a set of atomic vectors that represent basic items in the cognitive system. The atomic vectors are nearly orthogonal to each other, and are stored in a so-called *item memory* (IM). The IM is like a symbol table or dictionary of the items defined in the system, and stays fixed throughout the computation. In our seizure detection system, the LBP codes and the names of electrodes are the basic items (or symbols) that are assigned to the atomic vectors. These atomic vectors, inside the IM, are used as building blocks from which more complex vectors are constructed. Such complex vectors stand for concepts or percepts. For a complex vector is composite in nature, it can be very similar to other complex vectors with similar composition and structure [37].

2.2.2 HD Arithmetic Operations

HD computing builds upon a well-defined set of arithmetic operations with the vectors. Here, we focus on two main operations for the dense binary codes that are used for encoding/decoding data: addition and multiplication. These two operations are powerful and versatile by forming an algebraic field. Addition, or bundling, is defined as the componentwise majority function: every component of the input vectors is summed followed by normalization that retains them in the binary space. This is essentially a bitwise thresholded sum of n vectors that results in 0 when $n/2$ or more arguments are 0, and 1 otherwise. When adding an even number of vectors, in case of disagreement for a component (equal number of 1s and 0s), the majority can be chosen randomly. To avoid this random tie-breaking, we can augment the addition operation with an *additional* vector that is computed by binding a pair of vectors that are chosen systematically from the even number of vectors [34]. Addition of three vectors A, B, and C is denoted as $[A + B + C]$ where the brackets $[\ldots]$ stand for normalization.

Multiplication, or binding, is defined as the componentwise Exclusive OR (XOR). Likewise, it is denoted as $A \oplus B \oplus C$. Both operations work on an arbitrary number of input vectors and produce a fixed-width d-bit vector with an important distinction: bundling produces a vector that is *similar* to the input vectors ($\Delta([A+B+C], A) \approx 0$), whereas binding produces a *dissimilar* vector ($\Delta(A \oplus B \oplus C, A) \approx 0.5$). Hence, bundling is well suited for representing sets, multisets and can combine field/value pairs to represent a larger structure (e.g., a record or a tuple). More importantly, representations of such composite structures are constructed directly from representations of the atomic vectors by applying the operations without requiring any learning.

Learning and classification with HD computing is composed of four main steps: 1) mapping symbols to atomic vectors; 2) combining atomic vectors with the arithmetic operations inside an *encoder* to produce a complex vector representing an item of interest; 3) combining the complex vectors from the same category of items to produce a *prototype* vector representing the entire class of category (i.e., learning); 4) finally comparing the prototype vectors with a query vector to put it into categories (i.e., inference/classification).

3 iEEG DATASET

3.1 Patients

We include the anonymized data sets of 16 patients (P1–P16) of the epilepsy surgery program of the Inselspital Bern in this study for a total of 99 recordings. Clinical data on the patients is summarized in Table 1. Except for the need for invasive EEG studies, there were no additional inclusion criteria. All the patients gave written informed consent that their data from iEEG might be used for research purposes. The decision on the necessity for iEEG recordings, the electrode implantation scheme and the decision on surgical therapy was made entirely on clinical grounds. These decisions were taken prior to and independently from the compilation of this dataset.

3.2 iEEG Acquisition and Data Format

iEEG signals were recorded intracranially by strip, grid, and depth electrodes (all manufactured by AD-TECH, Wisconsin, USA), using a Nicolet One recording system with a C64 amplifier (VIASYS Healthcare Inc., Madison, Wisconsin, USA). An extracranial electrode, localized between 10/20 positions Fz and Cz, was used as reference for signal recording. iEEG recordings were either sampled at 512 or 1024 Hz, depending on whether they were recorded with more or less than 64 contacts. The iEEG recorded with less than 64 contacts were down-sampled to 512 Hz prior to further analysis. iEEG signals were re-referenced against the median of all the channels free of permanent artifacts as judged by visual inspection. After 16-bit analog-to-digital conversion, the data were digitally band-pass filtered between 0.5 and 150 Hz using a fourth-order Butterworth filter prior to analysis and written onto disk at a rate of 512 Hz.

---

[1] We use capitalized italic letters to indicate d-bit vectors that can also appear with a subscript and superscript.

TABLE 1: Clinical characteristics of patients and seizures included in the dataset.
Abbreviations: FLE: frontal lobe epilepsy, TLE: temporal lobe epilepsy, PLE: parietal lobe epilepsy, L: left, R: right, MRI: magnetic resonance imaging, y/n: yes/no, n.a.: not applicable

ID	Electrodes [#]	Seizures [#]	Seizure duration [s]			Age[y]	Hemisphere	MRI findings	Syndrome	Engel outcome
			Mean	Min	Max					
P1	100	5	14	10	22	46	R	n	TLE	2
P2	64	4	146	89	179	48	L	y(hippocampal sclerosis)	TLE	1
P3	62	14	98	31	139	32	L	y(focal cortical dysplasia)	PLE	4
P4	42	4	223	96	301	19	L	y(hippocampal sclerosis)	TLE	1
P5	59	6	88	67	117	31	L	y(hippocampal sclerosis)	TLE	2
P6	36	2	15	14	16	31	R	y(tuberous sclerosis)	FLE	4
P7	74	7	587	154	1002	36	L	y(pilocytic astrocytoma)	PLE	1
P8	61	3	121	52	184	23	L	n	TLE	4
P9	92	6	79	19	100	49	R	y(focal cortical dysplasia)	FLE	2
P10	47	13	71	10	252	24	L	n	TLE	2
P11	59	2	57	52	61	38	L	n	TLE	4
P12	54	10	99	80	154	20	R	n	TLE	2
P13	98	2	99	73	125	25	R	n	TLE	1
P14	49	10	45	23	93	59	L	y(space occupying amygdala)	TLE	4
P15	56	9	144	104	198	27	L	n	TLE	1
P16	64	2	109	83	135	26	R	y(hippocampal sclerosis)	TLE	1
Mean	**64**	**6**	**124**	**60**	**192**	**33**				

Forward and backward filtering was applied to minimize phase distortions.

All the iEEG recordings were visually inspected by an EEG board-certified experienced epileptologist (K.S.) for seizure identification and exclusion of channels continuously corrupted by artifacts. Each recording consists of 3 minutes of interictal segments (immediately before the seizure onset), the ictal segment (ranging from 10 s to 1002 s), and 3 minutes of immediate postictal time. In addition to the iEEG data for each patient, the dataset includes the age, the indices of channels, the indices of resected channels, the MRI findings, the epilepsy syndrome and the post-surgical outcome.

4 PART I: SEIZURE ONSET DETECTION

In this section, we present the main contribution of the paper for seizure onset detection. We show how the LBP feature extractor and HD computing can be combined to efficiently learn from iEEG recordings and detect novel seizures. Our proposed algorithm exploits LBP codes to directly symbolize the iEEG signal of each electrode. Then a complex d-dimensional binary representation is constructed to capture the statistics of the LBP codes across all electrodes and over time. This representation is used for learning and classification followed by simple postprocessing. The entire processing chain is displayed in Fig. 2. This approach is computationally efficient and extracts symbols for analyzing the occurrence of patterns, a process even somewhat similar to classical EEG reading by a human expert who tries to integrate visually detectable local and global characteristics of the iEEG signals into a coherent interpretation; then our approach quickly learns from one or few examples of these patterns per patient.

4.1 Preprocessing and LBP Feature Extraction

After filtering and downsampling the iEEG signals (Section 3.2), a LBP code is computed for every sampling point. Table 2 shows the impact of LBP length ($l \in [4, 8]$) on various metrics averaged among patients. The LBP codes with different lengths produce almost similar performance (l=6 results in highest accuracy). However, using a larger code length impairs its applicability to non-stationary iEEG signals and increases the latency of classification since the code length determines the minimum duration of the statistical analysis window, i.e., the duration of the window should be large enough that all symbols can at least theoretically occur once [16], [17]. With l=6, the statistical window can be as short as 0.5 s (containing 256 samples) that meets $256 > 2^l$. This LBP code considers six consecutive samples, and moves by one sample, generating 2^6 different symbols that are fed into the next stage for learning and classification (see Fig. 2).

TABLE 2: The impact of LPB length (l) on sensitivity, specificity, accuracy and delay.

LBP length (l)	4-bit	5-bit	6-bit	7-bit	8-bit
Sensitivity (%)	95.21	95.91	96.01	95.93	**96.33**
Specificity (%)	94.54	94.34	**94.84**	94.25	93.82
Accuracy (%)	94.88	95.12	**95.42**	95.09	95.08
Delay (s)	15.83	15.09	15.96	**14.88**	14.90

4.2 HD Learning and Classification

We use HD computing with holographic distributed representation to ease learning and classification from the LBP codes. HD computing first projects the LBP codes to the HD space via the IM, which assigns an orthogonal binary vector to every LBP code, i.e., $C_1 \perp C_2 \ldots \perp C_{64}$. To combine these vectors across all electrodes, HD computing generates a spatial record (S), in which an electrode *name* is treated as a field, and its LBP code as the value of this field. The IM also maps the name of electrodes to orthogonal binary vectors: $E_1 \perp E_2 \ldots \perp E_n$ for a patient with n electrodes. This allows to bind ($\oplus$) the name of each electrode ($E_j \mid j \in [1, n]$) to its corresponding code ($C_{i(j)} \mid i \in [1, 64]$). This binding ($E_j \oplus C_{i(j)}$) generates a

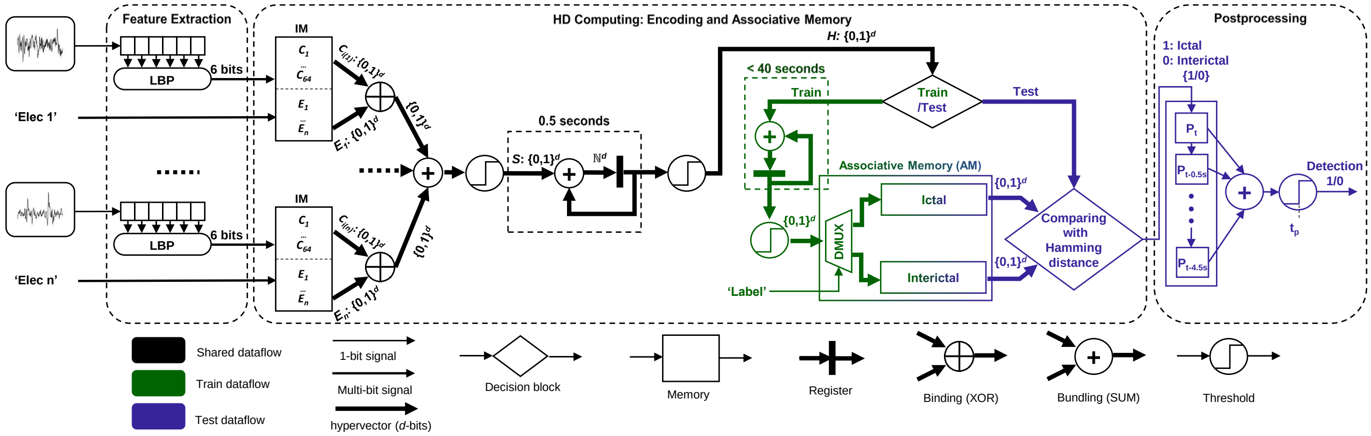


Fig. 2: Processing chain of our proposed algorithm: (1) Feature extraction generates a 6-bit LBP code for each electrode; (2) HD computing projects these codes into d-dimensional atomic vectors and constructs a complex vector H to represent the histogram of 0.5 s recording. During training the associative memory (AM) learns from this vector, and during inference provides a label for it; (3) As postprocessing, a simple patient-dependent (t_p) voting decides based on the last 10 labels.

new set of nearly orthogonal vectors to represent LBP codes per electrode that effectively reduces the size of IM from $64 \times n$ vectors to $64+n$ vectors. The spatial record (S) is then constructed by bundling the bound vectors of all electrodes: $S = [E_1 \oplus C_{i(1)} + E_2 \oplus C_{i(2)} + ... + E_n \oplus C_{i(n)}]$.

The binary vector S is computed for every new sample, and holographically represents the spatial information about the LBP codes of all electrodes. The next step is to compute the histogram of LBP codes for a moving window of 0.5 s with 256 samples (see Section 4.1). To estimate the histogram of LBP codes inside this window, a multiset of temporally generated S vectors is computed as $H = [S^1 + S^2 + ... + S^{256}]$. The bundling is applied in the temporal domain through accumulation (i.e., componentwise addition) of S^t vectors $t \in \{1, ..., 256\}$, that are produced within the window, and then thresholding at half (i.e, normalization).

As we briefly mentioned in Section 2.1, the interictal and ictal states show different distributions of LBP codes inside the window: during an interictal segment, we have a nearly random signal, with almost evenly distributed histograms; conversely, during a seizure we typically observe rhythmic signals, i.e., slow and often temporally asymmetric oscillations, which yield polarized histograms (see Fig. 1). This shows that the distribution of LBP codes, not necessarily their sequence, is an important indicator to distinguish ictal vs. interictal states. The HD space naturally encodes such histograms in the binary vector H by accumulating and thresholding the spatial vectors. Appendix A explains the HD encoding of histograms in detail.

The output of the encoding is H, a binary distributed vector that is updated every 0.5 s. To train our classifier, we use this vector to build an associative memory (AM) containing two *prototype* vectors representing ictal and interictal labels. To train the interictal prototype, all H vectors computed over an interictal window of 40 s are accumulated (summed), and then thresholded (normalized) to be stored in the AM as a binary prototype vector. Similarly, an ictal prototype vector is generated from a smaller window of 10–30 s depending on seizure duration. For classification, the label of an unseen 0.5 s window is determined by comparing its H to every prototype of the AM: the prototype that results in the minimum Hamming distance is the label.

4.3 Postprocessing

The last part of the algorithm postprocesses the labels produced by the HD classifier every 0.5 s. It defines a window of 5 s where a final decision is made based on the last 10 labels collected from the HD classifier (shifting labels of 0.5 s at a time). The decision is made based on a patient-specific threshold (t_p): the algorithm detects the seizure onset when the number of ictal labels inside the 5 s window is equal or greater than t_p. During the training, t_p is initially set to 10 (out of 10, to reduce the false alarms) and is decreased such that the algorithm can detect the training ictal segment. After training, for all the patients, $t_p \in [8, 10]$.

Overall, our algorithm has five parameters: the dimension of vectors (d), the size of LBP code (l), the duration of the two windows, and t_p. Only the last parameter is patient-dependent, whereas the others are fixed for all patients. Nevertheless, to reduce the memory load, d can be adjusted to the individual patient depending on the number of electrodes and seizure dynamics. We observed that the algorithm works with d=10,000 for all patients. For some patients it may even be reduced to 1000 without impairing its performance.

5 PART I: EXPERIMENTAL RESULTS

In this section, we first provide a concise review of the state-of-the-art methods for seizure onset detection (Section 5.1). We then compare these methods with our proposed algorithm (Section 5.2).

5.1 State-of-the-Art Methods

We focus on recent machine-learning based seizure onset detection methods [1], [7], [8] that may be partitioned into two groups. The first group is based on extracting time or frequency features or a mixture of them from the iEEG signals followed by traditional supervised machine learning methods (Section 5.1.1). For implementation of this

TABLE 3: Comparisons of our algorithm (LBP+HD) with the state-of-the-art methods: LBP + SVM [7], LGP + MLP [7],LSTM [8], Mixed + RF [1]. The upper part of the table shows the results in the one-shot learning setting, while the lower part shows the results of few-shot learning. The eight patients in the one-shot group present significantly fewer seizures with slightly longer durations than the other eight patients in the few-shot group. The last part of table shows the memory requirements to store each model (i.e. the learned weights).
Abbreviations and symbols used in this table: TrS: number of trained seizures, k: number of folds in cross-validation, ℓ: latency as delay of seizure onset detection, Spe.: specificity, Sen.: sensitivity, elec.: electrodes, and Accuracy: macroaveraging accuracy.

								One-shot learning									
			LBP + HD Computing			**LBP + Linear SVM**			**LGP + MLP**			**LSTM**			**Mixed + RF**		
ID	**TrS**	k	ℓ **[s]**	**Spe. [%]**	**Sen. [%]**	ℓ **[s]**	**Spe. [%]**	**Sen. [%]**	ℓ **[s]**	**Spe. [%]**	**Sen. [%]**	ℓ **[s]**	**Spe. [%]**	**Sen. [%]**	ℓ **[s]**	**Spe. [%]**	**Sen. [%]**
P2	1	4	15.1	100	100	10.1	91.74	75	12.2	98.26	100	10.6	100	100	10.8	100	100
P4	1	4	34.5	100	100	29.3	100	100	35.2	100	100	29.5	100	100	19.4	100	100
P5	1	6	20.9	100	100	14.7	92.09	100	14.6	84.54	100	17.1	99.31	100	24.0	100	100
P6	1	2	6.3	100	100	9.0	100	100	7.5	100	100	8.9	100	100	9.0	100	100
P8	1	3	13.2	100	100	11.9	100	100	10.3	100	100	10.9	100	100	13.4	100	100
P11	1	2	7.0	100	100	6.5	100	100	6.5	100	100	8.5	100	100	5.5	100	100
P13	1	2	10.0	100	100	16.3	100	100	9.8	100	100	11.3	100	100	7.0	95.76	100
P16	1	2	32.3	100	100	29.3	100	100	29.5	96.81	100	27.3	100	100	16.3	100	100
	mean		**17.4**	**100.0**	**100.0**	**15.9**	**97.98**	**96.88**	**15.7**	**97.45**	**100.0**	**15.5**	**99.91**	**100**	**13.2**	**99.47**	**100.0**
								Few-shot learning									
P1	2	4	6.3	100	100	6.9	100	100	6.9	96.76	100	5.9	99.98	100	5.8	99.91	100
P15	2	8	36.4	100	100	31.3	99.86	100	30.8	91.71	100	31.4	99.99	100	24.4	89.88	100
P3	3	12	21.8	79.97	91.03	15.6	81.33	100	16.8	77.04	100	16.1	86.85	55.30	17.2	73.28	69.64
P7	3	5	5.0	49.9	88.57	5.0	51.78	88.57	10.9	65.53	91.42	14.1	50.16	85.00	3.2	48.71	88.57
P9	3	4	16.2	96.31	96.43	12.4	89.05	96.42	14.4	88.93	89.28	7.5	82.98	100	6.0	91.04	100
P10	3	11	3.9	98.41	94.41	9.2	97.39	92.31	13.3	98.78	88.81	8.9	98.67	99.09	8.5	98.98	88.46
P12	6	5	15.9	96.88	80	13.6	82.62	88	17.3	85.01	90	17.6	99.56	90	2.4	33.90	87.50
P14	4	7	10.5	95.94	85.71	9.1	98.02	95.71	9.9	94.58	94.28	9.6	98.85	92.86	17.4	93.47	100
	mean		**14.5**	**89.68**	**92.02**	**12.9**	**87.51**	**95.13**	**15.0**	**87.29**	**94.22**	**13.9**	**89.63**	**90.28**	**10.6**	**78.65**	**91.77**
	Total mean		**15.9**	**94.84**	**96.01**	**14.4**	**92.74**	**96.00**	**15.4**	**92.37**	**97.11**	**14.7**	**94.77**	**95.14**	**11.9**	**89.06**	**95.89**
	Accuracy			**95.42**			**94.37**			**94.74**			**94.96**			**92.47**	
								Memory footprint									
	Weights [bytes]			2496			256 × #.elec.			10240 × #.elec. + 160			185408			36000	
				1			4–10×			148–410×			74×			14×	

group, we use Python 2.7 with the Scikit-learn library. The other group consists of recent deep learning algorithms that automatically extract features and construct classifiers (Section 5.1.2). The deep neural networks are implemented in Python 2.7 using Keras library with Tensorflow backend.

5.1.1 Feature Extraction and Supervised Learning

We consider three algorithms using different combination of feature extraction and classifier:

(1) Mixed+RF. We consider the first place algorithm from the crowdsourcing kaggle.com competition for development of seizure detection algorithms using iEEG [1]. This algorithm considers three sets of mixed features: the pairwise cross-correlation between channels, the cross-correlation between normalized channel power spectra, and the frequency magnitudes of each channel in the range of 1–47 Hz. For classification, a random forest classifier of 3000 trees is used. Random forest is an ensemble method that combines several individual classification trees. From the training dataset several bootstrap subsets are drawn, and a classification tree is fit to each of them. The response variable is predicted as majority vote of the predictions of all trees.

(2) LBP+SVM. The histogram of LBPs is used as features to a support vector machine (SVM) using a linear kernel (i.e., separation). The SVM is a discriminative classifier that is defined by a separating hyperplane. The SVM model is trained using the box constraint with regularization parameter (c) equal to 1.0; more details in [7].

(3) LGB+MLP. A slightly different version of LBP is called local gradient pattern (LGP). The histogram of LGP as features, is fed to a multilayer perceptron (MLP) [7]. The MLP is a class of feedforward artificial neural networks, consisting of at least three fully connected layers: one input layer, one or multiple hidden layer(s), and an output layer. The backpropagation algorithm is used to train the model. We use a single hidden layer with 40 neurons, and 1000 as the maximal number of iterations for convergence. The MLP performs better with LGPs than LBPs.

5.1.2 Deep Learning

We consider a long-short term memory (LSTM) network that is fed with raw iEEG samples (all channels are concatenated) [8]. LSTM is a cell for layers of a recurrent neural network enabling to process sequences of inputs. The LSTM network is composed of the first layer with 100 neurons, a subsequent dense layer that reduces the dimensionality to 50, and a max-pooling stage with 50 output features followed by a softmax layer that performs the classification; more details are discussed in [8].

We also consider a convolutional neural network (CNN), as a deep feedforward neural network commonly used in analyzing visual inputs. Recent work [38] proposes to use CNNs coupled with short time Fourier transform (STFT) as a universal method for seizure prediction. We apply their method to our iEEG based seizure detection that results in lower performance compared to other state-of-the-art methods: 83.61% specificity, 97.73% sensitivity, and 17.9 s delay.

5.2 Comparisons

We compare our algorithm (LPB+HD) with the aforementioned state-of-the-art methods using the dataset described in Section 3. Our algorithm is implemented in Python 2.7 using PyTorch. To have an identical setup for the state-of-the-art methods we also add the postprocessing step that is tuned for each of them to increase their specificity. We report performance metrics including specificity, sensitivity, macroaveraging accuracy, and delay of seizure onset detection given a limited number of trained seizure examples. Sensitivity is defined as the percentage of correctly detected seizures in the test dataset. Specificity is defined as the percentage of misclassified interictal 0.5 s windows. Macroaveraging accuracy is the mean of sensitivity and specificity that gives them equal weights to address the different durations of ictal and interictal segments. Latency is measured as the time the algorithm takes to classify an unseen seizure after the seizure onset time point marked by the expert; note that it is not the implementation latency but the *working* latency of an algorithm. Table 3 illustrates the full comparison results. We divide all available seizures for a patient into only two sets: training and testing (no evaluation set). We train all the methods only on the training set, and measure the metrics on the test set using k-fold cross-validation, where k is the total number of seizures minus the number of seizures in the training set (i.e., we rotate the trained seizures among all available seizures).

Based on the number of seizures used for training, we observe that the patients may be roughly partitioned into two groups: patients with one-shot learning (in the upper part of Table 3), and patients that need few more training seizures (few-shot learning) in the lower part of the table. For half of the patients (8 out of 16), our algorithm exhibits one-shot learning, i.e. training with only one seizure is possible. Our algorithm achieves perfect (100%) specificity and sensitivity in detecting novel seizures. The other methods cannot exhibit such perfect generalization in the one-shot setting: the LSTM is the closest method that achieves 99.91% specificity. Overall, all the methods achieve better performance for these patients in the upper part of the table.

The remaining eight patients, listed in the lower part of Table 3, are more challenging due to their fast and very localized seizures (i.e., only 2 or 3 electrodes out of 70 are involved in the ictal activity). Hence for these patients few more seizures (2–6) are required for training. Our algorithm trained with two seizures still maintains perfect generalization for two more patients (P1 and P15) while the other methods are behind. Only for two patients (P3 and P7), our algorithm shows a low specificity in the few-shot learning; on average, it achieves 89.68% specificity (vs. 89.63% in the LSTM) and 92.02% sensitivity (vs. 90.28% in the LSTM).

Considering both one-shot and few-shot settings across Table 3, our algorithm achieves higher macroaveraging accuracy on average (95.42%) than the other methods. The LSTM comes closest reaching 94.96% macroaveraging accuracy due to our added postprocessing method to avoid otherwise lower accuracy of 90.46%. Although other methods reach slightly higher sensitivity for some of the patients, our algorithm achieves the highest specificity (94.84% on average) that clearly shows the limitation of the other methods for long-time recordings. We further assess our algorithm on long-time recording from P15 using the same training setup. Our algorithm achieves 0 false alarms in 191 hours interictal recording; further results about long-term recording are provided in [39].

The latency of seizure onset detection of our algorithm is slightly larger than the one yielded by the LSTM (15.9 s vs. 14.7 s). Given that the average duration of ictal segments is 123.6 s (see Table 3), our algorithm still detects the seizures in the first 8% of the ictal window. Such a delay of detection below 20 s is well suited for several important applications considering that iEEG seizure onset often precedes clinical onset by more than 20 s [40]. Furthermore, considering that the median durations of mesial temporal lobe seizures and of neocortical extratemporal seizures have been found to be 106 s and 78 s, respectively [41], it may be helpful for therapeutic interventions aimed at early seizure termination or prevention of generalization of seizure activity [42].

More importantly, our algorithm operates with simple binary operations and a lower memory footprint than all other methods, which employ floating-point operations. As the output of training, our algorithm requires to store only the contents of the AM: $2 \times d$ bits for the two prototypes. The IM can be efficiently *rematerialized* by a cellular automaton from a random seed [43], hence there is no need to store the IM [34]. The last part of Table 3 shows the number of weights to be stored for each model. The number of weights for all the state-of-the-art methods (except the LSTM) grows with the number of electrodes. Though the number of weights in the LSTM are constant, its computational time grows with the number of electrodes, since it is a sequential model. Considering d=10,000, our method is stored on 2,500 bytes resulting in at least 4–10× lower memory requirements with respect to the simplest SVM classifier, and up to 74× with respect to the LSTM. Moreover, this is a conservative memory estimation for our algorithm, as the dimensionality can be reduced to 1000 for some iEEG recordings without affecting the performance.

6 PART II: IDENTIFICATION OF ICTOGENIC BRAIN REGIONS

In this section, we present the main contribution of the paper for identification of ictogenic brain regions. Precise identification of ictogenic brain regions followed by surgical resection often improves seizure control and can even eliminate the occurrence of seizures completely [5]. An important practical challenge is that with presurgical iEEG recordings (or any other current diagnostic method) the brain tissue of the so-called "epileptogenic zone," i.e. neuroanatomical areas that are necessary and sufficient to generate epileptic seizures, cannot be mapped directly and completely. Therefore, in clinical practice, the seizure onset zone/"ictogenic zone" (SOZ, i.e. the area where the first ictal iEEG signal changes are recorded), is used as a proxy for the epileptogenic zone [44]. To date, the seizure onset zone is delineated by clinical interpretation of iEEG recordings, mostly based on visual analysis, which is time consuming and may yield high inter-rater variability. This demands an algorithm that can learn *transparent* codes from the iEEG recordings, hence the codes are analyzable to locate the SOZ. HD computing produces such codes with interpretable features due to its well-defined set of arithmetic operations with inverses. In the following, we describe our algorithm that can automatically identify the SOZ at two levels of spatial resolution: the cerebral hemispheres and cerebral lobes.

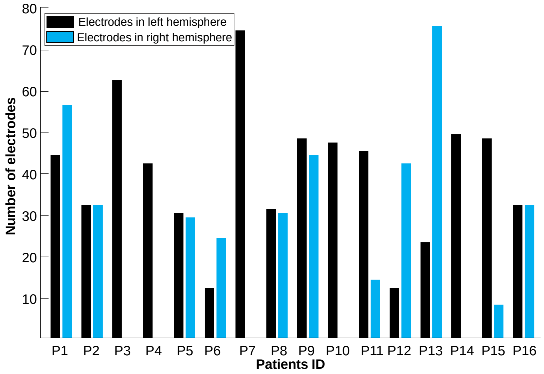


Fig. 3: Number of electrodes implanted in left and right cerebral hemisphere.

6.1 Algorithm for Identification of Ictogenic Hemispheres

Fig. 3 shows the number of electrodes implanted in the left or right cerebral hemispheres. As shown, for 11 patients out of 16, the clinical experts are uncertain whether the seizures start from the left, from the right, or from both cerebral hemispheres, and hence the number of electrodes implanted into both hemispheres is almost equal. Our first aim is to identify the location of SOZ at the spatial scale of the cerebral hemispheres.

Our proposed algorithm is mainly based on the seizure onset detection algorithm (Section 4) followed by statistical hypothesis testing. The algorithm first generates two sets of ictal (P_j^S) and interictal (P_j^I) prototypes for every electrode j by bundling the vectors representing the LBP symbols ($C_{i(j)}^t$) extracted from the related segments:

$$P_j^S = [\sum_{t \in ictal(10\,s)} C_{i(j)}^t] \mid i \in [1, 64]\ j \in [1, n]$$

$$P_j^I = [\sum_{t \in interictal(10\,s)} C_{i(j)}^t] \mid i \in [1, 64]\ j \in [1, n]$$

Note that the prototypes are computed solely for the relevant electrode. For example, for 'electrode 1', one prototype vector (P_1^I) is associated with one interictal window of 10 s, and another prototype (P_1^S) with one ictal window starting right after the seizure onset for 10 s (the duration of the shortest seizure is 10 s c.f. Table 1). If the algorithm requires more ictal/interictal examples, multiple prototypes will be generated (one per example). As is immediately apparent, when two different electrodes k and l continuously receive the same input stimuli, their corresponding encoded prototypes are identical, i.e., $P_k^I = P_l^I$ and $P_k^S = P_l^S$.

Then, the algorithm computes at least one score for every electrode j as the normalized Hamming distance between its related prototypes ($\Delta(P_j^I, P_j^S)$): the larger the identification score, the closer the proximity of the electrode to the SOZ. When the two prototypes encode iEEG windows with very similar dynamics (i.e. the same distribution of C_i), they yield a $\Delta \approx 0$. Conversely, a $\Delta \approx 0.5$ implies no correlation between the two prototypes due to a different distribution

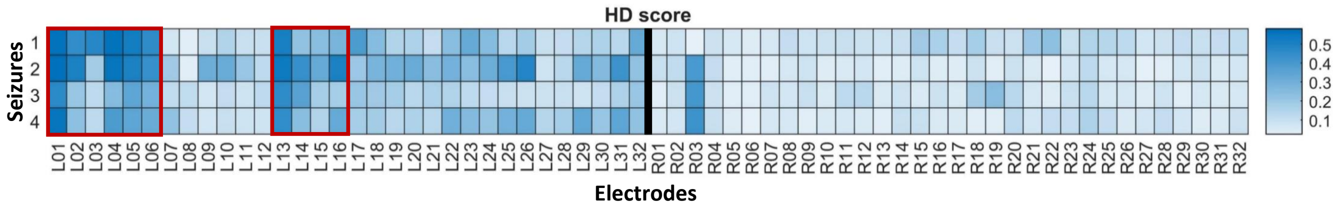


Fig. 4: Electrodes' scores of P2, generated by four subsequent seizures. Considering only the first seizure (the first row of scores) is sufficient for the algorithm to identify the correct hemisphere. The red squares mark the electrodes in the SOZ. The box-plot of scores is shown in Fig. 5.

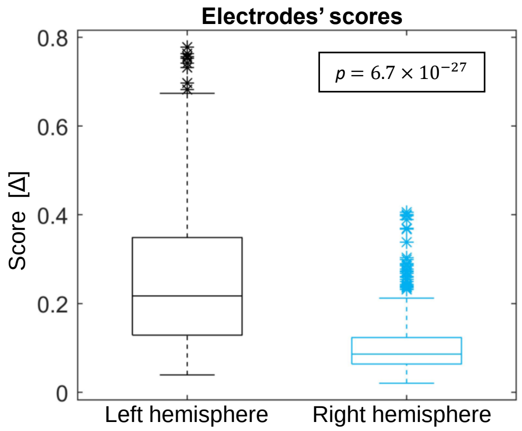


Fig. 5: Box plot with scores of P2 by supplying all four seizures and one interictal window. On each box, the central mark indicates the median, and the bottom/top edges of the box indicate the 25th/75th percentiles. The dotted line extends to the most extreme data points not considering outliers. Outliers are marked with '*'.

of the C_i. In other words $\Delta \approx 0$ indicates that the electrodes are not involved in the seizure activity whereas an increasing value of the score suggests an increasing proximity of the electrodes to the SOZ. Fig. 4 shows an example of the scores computed for each electrode from four seizures of P2.

Finally, the algorithm incorporates statistical analysis with the computed scores of individual electrodes for identifying the seizure-generating hemisphere. The electrodes are divided into two groups, i.e. electrodes in the right hemisphere and electrodes in the left hemisphere. Then we use the Kolmogorov-Smirnov test to assess the distribution of the scores. Since the distribution is normal, we perform a t-Student test to determine if the two groups of scores are significantly different from each other. If the result of the test is statistically *significant*, i.e., the resulting p-value of the test is lower than 0.01, we accept it, and identify the hemisphere with highest score as the ictogenic one. Fig. 5 shows an example of the box-plot of the scores for P2, clearly indicating that the left hemisphere is ictogenic with p=6.7×10^{-27}.

If the result of the test is not significant ($p \geq 0.01$), we reject it; this means that the algorithm requires more examples to be able to infer the hemisphere correctly, hence we supply more ictal and/or interictal examples. For each exemplar window a separate prototype will be generated that effectively increases the number of scores per electrode and therefore the significance of the test. Therefore our algorithm keeps computing new prototypes and related scores over time until $p < 0.01$. For most of the patients a single ictal window and five interictal windows are sufficient (see Table 4).

6.2 Algorithm for Identification of Ictogenic Lobe

Here, our aim is to further investigate the location of the SOZ at the spatial resolution of cerebral lobes. We straightforwardly extend our proposed algorithm (Section 6.1) by dividing the electrodes into a larger number of groups (hence using a finer spatial resolution), and accordingly invoke a proper statistical testing method. Since the generation of prototypes and computation of scores are the same as in the previous section, we present the statistical analysis only.

First, the electrodes are divided into different groups according to the lobes that they cover. However, the exact electrode's membership in the lobes is not available, hence we use strips/grids (series of close electrodes) as a proxy. The strips with six or less electrodes are associated to a single group whereas the strips with eight or more electrodes are divided into two different groups. Then, one-way ANOVA test (with Bonferroni post-hoc test) is computed to determine if at least one group of scores (i.e. one lobe) is significantly different from the others. We use all the available ictal segments to compute the scores for this fine-grained identification.

7 PART II: EXPERIMENTAL RESULTS

In this section, we report the identification accuracy and the p-values of our algorithm for determining the ictogenic regions. For identifying the ictogenic hemisphere, we assess how our algorithm performs on the patients with bilaterally localized electrodes (11 out of 16 patients; see Fig 3).

For these patients, ictogenic hemispheres are indicated by the experts as previously reported in [5]. The second column of Table 4 lists these hemispheres as ground truth, and shows the mean score of our algorithm for the left and right hemispheres. When the score of one side significantly differs from the other side ($p < 0.01$), the algorithm classifies the hemisphere with the highest score as the ictogenic hemisphere.

Using only one interictal window, the algorithm often requires one or more ictal examples to meet $p < 0.01$.

TABLE 4: Patients with bilaterally localized electrodes are analyzed to identify their ictogenic hemisphere. The first part shows the minimum number of seizures for training (TrS.) to identify the ictogenic hemisphere with $p < 0.01$. Five interictal prototypes are used to attain robustness of identification. In addition, the mean scores of the hemispheres and the p-value of the t-test are reported. The experiments are also repeated by training with all available seizures.

		Trained with min. # of seizures s.t. $p < 0.01$				**Trained with all available seizures**			
ID	**Hemisphere**	**TrS. [#]**	**Mean Score Left Hem.**	**Mean Score Right Hem.**	**p-value**	**TrS. [#]**	**Mean Score Left Hem.**	**Mean Score Right Hem.**	**p-value**
P1	R	1	0.1306	0.1836	< 0.00001	5	0.1206	0.1809	< 0.00001
P2	L	1	0.2825	0.1436	< 0.00001	4	0.2574	0.1009	< 0.00001
P5	L	1	0.3423	0.1938	< 0.00001	6	0.3433	0.2093	< 0.00001
P6	R	1	0.1273	0.1789	0.001	2	0.1004	0.1627	< 0.00001
P8	L	1	0.1191	0.0815	< 0.00001	3	0.1170	0.0783	0.00002
P9	R	2	0.1152	0.1460	< 0.00001	6	0.1030	0.1441	< 0.00001
P11	L	1	0.1740	0.1174	< 0.00001	2	0.2090	0.1541	< 0.00001
P12	R	1	0.0611	0.0968	< 0.00001	10	0.0665	0.0923	< 0.00001
P13	R	1	0.1402	0.1904	0.00007	2	0.1329	0.2086	< 0.00001
P15	L	1	0.2043	0.1216	0.001	9	0.2297	0.1552	< 0.00001
P16	R	1	0.1185	0.2958	< 0.00001	2	0.1941	0.2791	< 0.00001

TABLE 5: Patients of Table 4 are also analyzed for ictogenic lobe identification. The table shows the ictogenic lobe, the identified lobe (if present) and p-value of the identification.

ID	Ictogenic Lobe	Identified Lobe	p-value
P1	right temporal	right temporal	< 0.00001
P2	left temporal	left temporal	< 0.00001
P5	left temporal	left temporal	< 0.00001
P6	right temporal	right temporal	< 0.00001
P8	left temporal	n.a.	n.a.
P9	right temporal	right temporal	< 0.00001
P11	left temporal	left temporal	< 0.00001
P12	right temporal	right temporal	< 0.00001
P13	right temporal	right temporal	< 0.00001
P15	left temporal	left temporal	< 0.00001
P16	left temporal	left temporal	< 0.00001

On the other hand, the number of interictal windows is naturally very large, a fact that can be exploited to produce the same reliable results with fewer ictal examples. Hence, the algorithm randomly selects five interictal windows and one ictal window for all patients (except P9, where two ictal windows were used) resulting in the correct classification of the ictogenic hemisphere with an accuracy of 100% (11 of 11) and $p < 0.01$. Moreover, we perform further analysis by increasing the number of ictal windows (see the second half of Table 4). The p-value decreases, implying a more robust identification. The randomly chosen interictal windows should be within an hour of seizure onset—though, this choice might influence the results.

For identifying the ictogenic lobe, we consider the same patients of Table 4. Table 5 lists the ictogenic lobe identified by visual inspection by an expert (K.S.). The algorithm identifies the same lobes for 10 out of 11 patients with $p < 0.00001$. For one patient (P8) the statistical analysis is ineffective as the scores of multiple lobes are almost similar. This result can be compared to the post-surgical outcome: the algorithm correctly classifies the SOZ for all the patients who remained seizure-free after surgery, i.e. Engel outcome (EO) = 1, and for all patients with rarely occurring disabling seizures after surgery for at least 2 years (EO = 2), whereas it can't identify a single ictogenic lobe for a patient with EO = 4 (no improvement after surgery).

8 CONCLUSIONS AND FUTURE WORK

Our algorithm learns from one or few iEEG seizure recordings; it exploits LBP codes and HD computing that enable full binary operations during training and inference. Its learning procedure is transparent and thus allows to translate the learned codes into information about the spatial localization of the seizure-generating brain regions, for example to better target surgical resection or neuromodulatory interventions. Our algorithm also provides a universal and linearly scalable interface with a minimal set of parameters that ease analyzing all iEEG recordings from different patients with 36 to 100 implanted electrodes. Using our dataset with 16 patients and 99 seizures, our algorithm requires a total of 34 seizures for training: eight seizures for eight patients (with one-shot learning) and 26 seizures for the other eight patients (with few-shot learning). We test the algorithm on 65 unseen seizures using k-fold cross-validation: the algorithm outperforms LBP+SVM, LGP+MLP, LSTM, and Mixed+RF with higher specificity and macroaveraging accuracy, and a lower memory footprint. Its performance however on other datasets is not yet reported.

Future work will focus on efficient hardware implementation of our algorithm in order to analyse long-term iEEG recordings.

REFERENCES

[1] S. N. Baldassano *et al.*, "Crowdsourcing seizure detection: algorithm development and validation on human implanted device recordings," *Brain*, vol. 140, no. 6, pp. 1680–1691, 2017.

[2] D. Schmidt *et al.*, "Evidence-based review on the natural history of the epilepsies." *Current opinion in neurology*, vol. 25 2, pp. 159–63, 2012.

[3] J. F. Tellez-Zenteno *et al.*, "Long-term outcomes in epilepsy surgery: antiepileptic drugs, mortality, cognitive and psychosocial aspects," *Brain*, vol. 130, no. Pt 2, pp. 334–345, Feb 2007.

[4] S. Wiebe *et al.*, "A randomized, controlled trial of surgery for temporal-lobe epilepsy," *N. Engl. J. Med.*, vol. 345, no. 5, pp. 311–318, Aug 2001.

[5] C. Rummel *et al.*, "Resected Brain Tissue, Seizure Onset Zone and Quantitative EEG Measures: Towards Prediction of Post-Surgical Seizure Control," *PLoS ONE*, vol. 10, no. 10, p. e0141023, 2015.

[6] Y. Nagahama *et al.*, "Intracranial EEG for seizure focus localization: evolving techniques, outcomes, complications, and utility of combining surface and depth electrodes," *J. Neurosurg.*, pp. 1–13, May 2018.

[7] A. K. Jaiswal *et al.*, "Local pattern transformation based feature extraction techniques for classification of epileptic EEG signals," *Biomedical Signal Processing and Control*, vol. 34, pp. 81 – 92, 2017.

[8] R. Hussein *et al.*, "Robust detection of epileptic seizures using deep neural networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 2546–2550.

[9] W. Zhou *et al.*, "Epileptic Seizure Detection Using Lacunarity and Bayesian Linear Discriminant Analysis in Intracranial EEG," *IEEE Trans Biomed Eng*, vol. 60, no. 12, pp. 3375–3381, Dec 2013.

[10] F. T. Sun *et al.*, "The RNS system: responsive cortical stimulation for the treatment of refractory partial epilepsy," *Expert Review of Medical Devices*, vol. 11, no. 6, pp. 563–572, 2014.

[11] C. C. Jouny *et al.*, "Characterization of epileptic seizure dynamics using Gabor atom density," *Clin Neurophysiol*, vol. 114, no. 3, pp. 426–437, Mar 2003.

[12] C. Geier *et al.*, "How important is the seizure onset zone for seizure dynamics?" *Seizure*, vol. 25, pp. 160–166, Feb 2015.

[13] N. M. Wetjen *et al.*, "Intracranial electroencephalography seizure onset patterns and surgical outcomes in nonlesional extratemporal epilepsy," *J. Neurosurg.*, vol. 110, no. 6, pp. 1147–1152, Jun 2009.

[14] J. S. Naftulin *et al.*, "Ictal and preictal power changes outside of the seizure focus correlate with seizure generalization," *Epilepsia*, vol. 59, no. 7, pp. 1398–1409, Jul 2018.

[15] C. Harden *et al.*, "Practice guideline summary: Sudden unexpected death in epilepsy incidence rates and risk factors: Report of the Guideline Development, Dissemination, and Implementation Subcommittee of the American Academy of Neurology and the American Epilepsy Society," *Neurology*, vol. 88, no. 17, pp. 1674–1680, Apr 2017.

[16] K. Schindler *et al.*, "On seeing the trees and the forest: Single signal and multi signal analysis of periictal intracranial EEG," *Epilepsia*, vol. 53, no. 9, pp. 1658–1668, 2012.

[17] K. Schindler *et al.*, "Forbidden ordinal patterns of periictal intracranial EEG indicate deterministic dynamics in human epileptic seizures," *Epilepsia*, vol. 52, no. 10, pp. 1771–1780, 2011.

[18] Y. Kaya *et al.*, "1d-local binary pattern based feature extraction for classification of epileptic eeg signals," *Applied Mathematics and Computation*, vol. 243, pp. 209 – 219, 2014.

[19] 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.

[20] A. Burrello *et al.*, "One-shot learning for iEEG seizure detection using end-to-end binary operations: Local binary patterns with hyperdimensional computing," in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Oct 2018, pp. 1–4.

[21] C. S. Daw *et al.*, "A review of symbolic analysis of experimental data," *Review of Scientific Instruments*, vol. 74, no. 2, pp. 915–930, 2003.

[22] T. S. Kumar *et al.*, "Classification of seizure and seizure-free EEG signals using local binary patterns," *Biomedical Signal Processing and Control*, vol. 15, pp. 33 – 40, 2015.

[23] K. Schindler *et al.*, "Ictal time-irreversible intracranial EEG signals as markers of the epileptogenic zone," *Clinical Neurophysiology*, vol. 127, no. 9, pp. 3051 – 3058, 2016.

[24] R. W. Gayler, "Vector symbolic architectures answer Jackendoff's challenges for cognitive neuroscience," in *Proceedings of the Joint International Conference on Cognitive Science. ICCS/ASCS*, 2003, pp. 133–138.

[25] T. Plate, *Holographic Reduced Representations*. CLSI Publications, 2003.

[26] A. Rahimi *et al.*, "A robust and energy efficient classifier using brain-inspired hyperdimensional computing," in *Low Power Electronics and Design (ISLPED), 2016 IEEE/ACM International Symposium on*, August 2016.

[27] A. Rahimi *et al.*, "High-dimensional computing as a nanoscalable paradigm," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 9, pp. 2508–2521, Sept 2017.

[28] T. Wu *et al.*, "Brain-inspired computing exploiting carbon nanotube FETs and resistive RAM: Hyperdimensional computing case study," in *IEEE International Solid-State Circuits Conference, ISSCC*, 2018.

[29] A. Rahimi *et al.*, "Hyperdimensional biosignal processing: A case study for EMG-based hand gesture recognition," in *IEEE International Conference on Rebooting Computing*, October 2016.

[30] A. Moin *et al.*, "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*, In press 2018.

[31] A. Rahimi *et al.*, "Hyperdimensional computing for noninvasive brain–computer interfaces: Blind and one-shot classification of EEG error-related potentials," *10th ACM/EAI International Conference on Bio-inspired Information and Communications Technologies (BICT)*, 2017.

[32] A. Rahimi *et al.*, "Hyperdimensional computing for blind and one-shot classification of EEG error-related potentials," *Mobile Networks and Applications*, Oct 2017.

[33] F. Montagna *et al.*, "PULP-HD: accelerating brain-inspired high-dimensional computing on a parallel ultra-low power platform," in *Proceedings of the 55th Annual Design Automation Conference*, ser. DAC '18. New York, NY, USA: ACM, 2018, pp. 111:1–111:6.

[34] M. Schmuck *et al.*, "Hardware Optimizations of Dense Binary Hyperdimensional Computing: Rematerialization of Hypervectors, Binarized Bundling, and Combinational Associative Memory," *ACM Journal on Emerging Technologies in Computing (JETC)*, 2019.

[35] P. Kanerva, "Binary spatter-coding of ordered k-tuples," in *ICANN'96, Proceedings of the International Conference on Artificial Neural Networks*, ser. Lecture Notes in Computer Science, , Ed., vol. 1112. Springer, 1996, pp. 869–873.

[36] P. Kanerva, *Sparse Distributed Memory*. Cambridge, MA, USA: The MIT Press, 1988.

[37] M. A. Kelly *et al.*, "Encoding structure in holographic reduced representations." *Canadian Journal of Experimental Psychology*, vol. 67, no. 2, pp. 79–93, 2013.

[38] N. D. Truong *et al.*, "Convolutional neural networks for seizure prediction using intracranial and scalp electroencephalogram," *Neural Networks*, vol. 105, pp. 104 – 111, 2018.

[39] A. Burrello *et al.*, "Laelaps: An energy-efficient seizure detection algorithm from long-term human ieeg recordings without false alarms," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2019*, 2019.

[40] H. Martin *et al.*, "Latencies from intracranial seizure onset to ictal tachycardia: A comparison to surface eeg patterns and other clinical signs," *Epilepsia*, vol. 56, no. 10, 2015.

[41] P. Afra *et al.*, "Duration of complex partial seizures: an intracranial EEG study," *Epilepsia*, vol. 49, no. 4, pp. 677–684, Apr 2008.

[42] G. K. Bergey, "Neurostimulation in the treatment of epilepsy," *Exp. Neurol.*, vol. 244, pp. 87–95, Jun 2013.

[43] S. Wolfram, "Random sequence generation by cellular automata," *Advances in Applied Mathematics*, vol. 7, no. 2, pp. 123 – 169, 1986.

[44] F. Rosenow *et al.*, "Presurgical evaluation of epilepsy," *Brain*, vol. 124, no. 9, pp. 1683–1700, 2001.

APPENDIX A
HD ENCODING OF HISTOGRAMS

We describe how the HD operations can be applied to encode histograms. The aim is to store an approximated version of a histogram in HD space to save memory and ease computation with only binary components. Below, we illustrate an example to better explain the procedure.

We use the LBP codes as basic symbols, and their counts in a 1 s window (i.e. 512 symbols) as a histogram. First, the LBP codes are mapped into atomic vectors through the IM. Then, the atomic vectors generated during the 1 s window are bundled to produce a complex vector (H) representing the histogram. In this way, the histogram of LBP codes is *holistically* represented in a single binary vector H. The original LBP distribution can be recalled by comparing H with the individual atomic vectors. For every symbol, we compute the similarity as $1 - \Delta$ that recovers the count of the symbol. The set of the computed similarities represents the approximated histogram.

Fig. 6 shows the Pearson correlation coefficient between the similarity values (extracted from the approximated histograms) and the exact count of the symbols in the his-

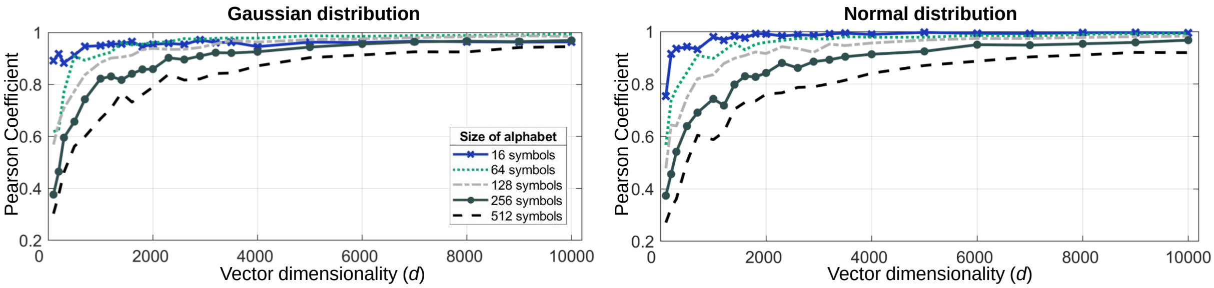
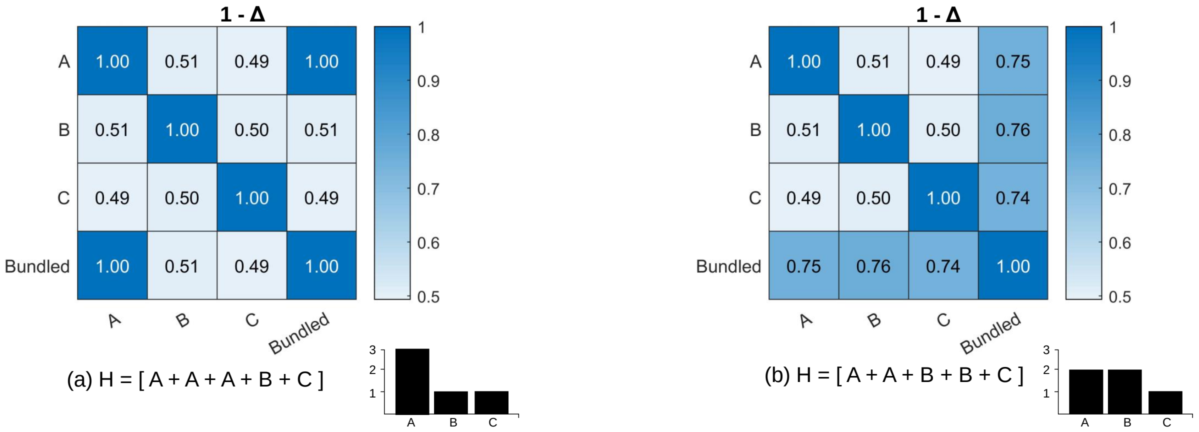


Fig. 6: Pearson coefficient between exact histograms and their approximated HD versions. The procedure consists of the following steps: (1) creations of Gaussian and uniform distributions from an alphabet with m symbols where $m \in \{16, 64, 126, 256, 512\}$; (2) association of a random atomic vector to every symbol; (3) creation of the histogram vectors H for a window of 512 symbols by bundling their corresponding atomic vectors; (4) computing the similarities between H and the the atomic vectors; (5) Plotting Pearson correlation between similarities and the exact counts.



(a) H = [A + A + A + B + C]

(b) H = [A + A + B + B + C]

Fig. 7: Similarity between three atomic vectors and two different encoded histograms. **(a)** symbol A occurs more than 50% of time, whereas **(b)** no symbol occurs more than 50% of time.

tograms. We consider five alphabets with different sizes of 16, 64, 128, 256, and 512 symbols. We use two different distributions to generate histograms from these alphabets for a window of 512 symbols: a Gaussian distribution to mimic polarized ictal histograms, and a uniform distribution to mimic randomly distributed interictal histograms. The experiments are repeated for vectors while varying the dimension (d). Using a $d > 2000$ with the 64-symbol alphabet, a Pearson correlation > 0.9 is observed for both distributions. The Pearson correlation further grows toward 1 with larger d. However, if a single symbol occurs more than 50% of the total symbols inside the histogram, the encoded H is a copy of that specific symbol (Fig. 7). This is due to the characteristics of the bundling operation (majority sum): if several copies of any vector are included in the bundling, then the resultant vector is closer to the dominating vector than to other vectors.