University of Essex

Research Repository

Spider Pattern: An adaptive EEG feature extraction method for Alzheimer's disease and frontotemporal dementia classification

Accepted for publication in Psychiatry Research - Neuroimaging

Research Repository link: https://repository.essex.ac.uk/44274/

Please note:

Changes made as a result of publishing processes such as copy-editing, formatting and page numbers may not be reflected in this version. For the definitive version of this publication, please refer to the published source. You are advised to consult the published version if you wish to cite this paper.
https://doi.org/10.1016/j.pscychresns.2026.112366

www.essex.ac.uk

University of Essex

Spider Pattern: An Adaptive EEG Feature Extraction Method for Alzheimer's Disease and Frontotemporal Dementia Classification

Abstract

Dementia is a major and growing health challenge, and electroencephalography (EEG) provides a non-invasive and comparatively accessible modality for computational investigation of dementia-related brain activity. This study evaluates an adaptive time-domain feature extraction technique named Spider Pattern (SP), combined with Social Spider Algorithm (SSA) feature selection and feedforward neural network (FFNN) classification, for Alzheimer's disease (AD), frontotemporal dementia (FTD), and normal control (NC) discrimination. The preprocessed public EEG dataset contains 88 participants (36 AD, 23 FTD, and 29 NC). Using 15-second non-overlapping segmentation, 4,535 segments were analysed and represented by 608 SP features. The primary evaluation used nested leave-one-subject-out (LOSO) validation in which SSA feature selection was performed only within each outer training fold. Subject-level accuracies were 78.46% for AD vs. NC (AUC 0.810), 71.15% for FTD vs. NC (AUC 0.712), and 52.54% for AD vs. FTD (AUC 0.562). Exact paired McNemar tests showed no significant difference between SSA and either full-feature or mRMR comparators. A secondary segment-wise 10-fold analysis using all 608 SP features produced higher accuracies of 90.83%, 93.08%, and 90.87%, respectively; however, every fold contained test participants also represented in training, so these values are not estimates of unseen-subject generalisation. The gap between the two estimates quantifies the effect of participant leakage on this widely used public dataset. Sensitivity analyses showed task-dependent effects of SP design choices, while repeated SSA runs showed limited feature-subset stability. The results support further investigation of SP for case-control EEG classification but do not establish reliable AD-vs.-FTD discrimination or clinical deployment.

Keywords: Alzheimer's disease, frontotemporal dementia, EEG, Spider Pattern, Social Spider Algorithm, subject-independent validation.

1. Introduction

Alzheimer's disease (AD) is a progressive neurodegenerative disorder that primarily affects cognitive functions and memory and is a leading cause of dementia in old people. The early symptoms of AD are difficulty in remembering recent events, names, or conversations [1]. With further progression, AD patients have more clear symptoms such as disorientation, mood changes, language deficits, and reduced ability to do routine activities [2].

The number of AD patients has been significantly increasing, which is because of the increasing population of old people around the world, especially in developed countries [1]. According to the World Health Organization (WHO), there are about 55 million people living with dementia around the world, with 10 million new cases diagnosed every year [3]. AD accounts for approximately 60-70% of all dementia cases. In fact, AD is the most prevalent type of dementia [4]. Dementia is the seventh leading cause of death worldwide and a major source of disability in aged people. Epidemiological projections indicate a rapid increase, with the number of AD patients expected to reach 75 million by 2030 and rise to approximately 131 million by 2050 [2, 4].

Currently, there is no curative treatment or medical solution for AD, and available therapies only try to provide modest symptomatic relief [5]. However, early and accurate diagnosis of AD has a key role in controlling and slowing the disease progression [6]. The early diagnosis of AD might increase the quality of life for patients. The main challenge in early detection is distinguishing pathological symptoms of AD from normal age-related cognitive changes.

Traditional diagnostic approaches are often based on invasive methods such as brain tissue biopsy. On the other hand, non-invasive diagnostic techniques have been actively explored [7]. Current diagnostic protocols combine neuropsychological evaluation, brain imaging techniques, and neuronal signal recordings to support accurate diagnosis. However, the main challenge is still unclear clinical decision making based on these modalities [8]. In addition, the high cost of neuroimaging techniques such as Magnetic Resonance Imaging (MRI) and Positron Emission Tomography (PET) scans is another reason they are not widely used for AD detection [9].

Recently, electroencephalography (EEG) has become one of the most

popular and well-known neuroimaging techniques, due to its accessibility, non-invasive nature, and cost-effectiveness [10]. EEG signals record the brain's electrical activities generated by the physiological functions of neurons [11]. Traditional EEG analysis was based on visual inspection by neurologists; however, it was a complicated task and prone to human error [12]. Analyzing the nonlinear and complex dynamics of the brain requires advanced computation-based systems [13]. A computer-based system can provide higher precision and sensitivity than human visual analysis in analyzing EEG data.

In the last decades, several methodologies for EEG-based AD detection have been developed, which can be categorized into three main approaches: time-domain based methods, time-frequency domain-based methods, and deep learning-based methods. Time-domain based methods extract features directly from EEG signals, such as computing statistical components and entropies from EEG data [14]. These methods typically assume EEG data to be stationary, however, these signals are inherently non-stationary, depicting chaotic and complex behavior [15].

On the other hand, time-frequency domain-based methods decompose EEG data into specific frequency subbands before feature extraction. These methods have shown acceptable performance in EEG data processing [16, 17]. However, time-frequency domain-based methods increase the redundancy of the data during multi-level decomposition, which is one of their limitations [11]. Besides, tuning the parameters of decomposition methods is another challenge in time-frequency analyzing of EEG data [18]. When the level of decomposition is selected too high, the resulting subbands become more complex and redundant than the original EEG data. Likewise, when the decomposition level is selected too small, the resulting subbands cannot represent the complex and nonlinear patterns of EEG data. Recent neurological EEG work has also systematically compared discrete wavelet, continuous wavelet and synchrosqueezing time–frequency representations with pretrained CNNs, showing that the representation choice can materially affect both image-wise and subject-wise classification performance [19].

The performance of time-domain based methods and time-frequency domain-based methods are related to the feature extraction technique. If the extracted features are significant, the performance is high. Likewise, if the extracted features are not significant, the performance drops. Most feature extraction techniques measure the global trend in EEG data, and cannot decode local trends and variations in EEG data [20, 8].

Deep learning models have also been developed for EEG classification [21, 22]. Depending on the architecture, they may operate directly on multichannel time-series data or on transformed representations such as time–frequency maps; conversion of EEG to images is therefore one possible design rather than a universal requirement. A more general methodological concern is validation leakage: when multiple segments from the same participant are randomly distributed across training and test folds, reported performance may reflect participant-specific information rather than generalisation to unseen individuals [23, 24]. A recent validation-aware EEG treatment-response study likewise separated segment-level separability from subject-wise generalisation and confined learned processing to the appropriate training partitions, illustrating why the validation unit changes the evidential meaning of reported performance [25]. Deep models can also require substantial training data and their learned representations may be more difficult to trace back to specific input channels or local temporal patterns [26].

Three issues in the existing EEG-based dementia-classification literature motivate the present work. First, compact time-domain descriptors should capture local temporal transitions without requiring a large learned representation. Second, evaluation should separate participants at the subject level when the aim is to estimate generalisation to unseen individuals. Third, feature attribution should distinguish descriptive model importance from neurophysiological or connectivity claims that the analysis cannot support.

To address these issues, this study investigates the Spider Pattern (SP) feature extraction technique for EEG-based classification of AD, FTD and NC participants. SP is an adaptive time-domain descriptor designed to encode local temporal transitions through a fixed spider-like arrangement of neighbouring samples and a ternary coding step. The ternary thresholding component is related to established local ternary pattern ideas [27], and one-dimensional local binary pattern variants have previously been applied to EEG time series [28]; the contribution of SP lies in its specific temporal rearrangement, path construction and histogram representation rather than in ternary coding alone.

SP operates directly on the preprocessed time-domain EEG without additional time–frequency decomposition. Each SP feature remains traceable to its source EEG channel, which permits descriptive channel- and region-level summaries. These summaries are normalised for unequal numbers of electrodes per region and complemented by held-out permutation importance; they are not interpreted as functional connectivity.

The SP technique generates 608 features per EEG segment. SSA is used as a wrapper-based feature selector, but it is treated as the standard SSA algorithm applied to feature selection rather than as a newly modified optimisation algorithm. To prevent test-subject information from influencing feature selection, SSA is repeated entirely within each outer LOSO training fold. The selected features are then used to train a FFNN. Full-feature and mRMR-selected models are evaluated under the same outer subject-level splits for comparison.

Recent related studies include CNN–LSTM–attention classification [29], an explainable deep-learning framework [30], and attention-guided fusion of spectral, fractal and nonlinear EEG features [31]. Their published aggregate results provide useful context, but they are not treated as paired statistical comparators because the validation units and subject-level predictions are not identical to the present study. A closely related recent AHEPA study introduced Dynamic Geometric Pattern Maps (DGPM), which arrange channel-resolved nonlinear EEG features into a 10–20-informed image representation and evaluate both segment-wise and LOSO protocols [32]; it is particularly relevant here as an alternative geometric EEG representation and as further evidence that validation unit materially affects reported dementia-classification performance.

The main contributions of this study are as follows:

- Definition and evaluation of the SP time-domain feature extraction technique, with a non-negative adaptive threshold and a mutually exclusive ternary encoding.
- Leakage-aware nested LOSO evaluation in which SSA feature selection, scaling and FFNN training are confined to the outer training participants.
- Comparison of SSA-selected SP features with the full 608-feature representation and a training-only mRMR selector, together with confidence intervals and exact paired McNemar tests.
- Sensitivity analyses, SSA selection-stability assessment, normalised regional selection summaries, and held-out permutation importance that give a restrained account of model feature use.

Beyond the descriptor itself, the study contributes to evaluation practice in EEG-based dementia classification. Segment-level and subject-level

estimates are reported for the same features, data and classifier, so that the difference between them isolates the effect of participant leakage rather than of a change in method. The negative result for AD vs. FTD is reported in full. Together these provide a reference point against which accuracies reported on this public dataset under heterogeneous validation protocols can be interpreted.

The rest of the paper is organized as follows. Section 2 describes the datasets used in this study. Section 3.1 explains the steps applied to prepare the EEG data. The methodology is presented in Section 3, which describes the SP feature extraction technique, SSA, FFNN, and the interpretability analysis. Section 4 reports and discusses the results, analyzes the experimental findings, and compares them with existing methods. Finally, the conclusion in Section 5 summarizes the main contributions and discusses future directions.

2. Analysed Data

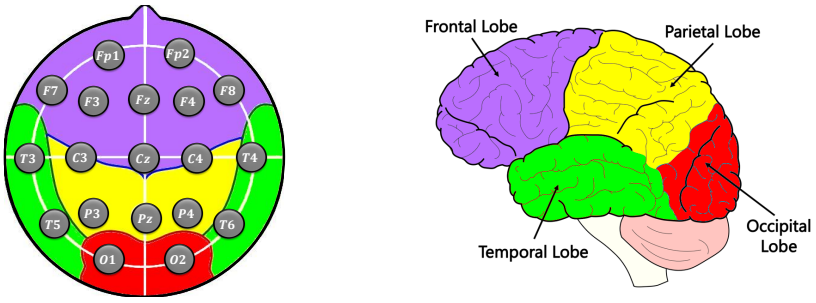
This study utilized the publicly available EEG dataset provided by Miltiadous et al. [33]. The dataset comprises EEG recordings from 88 individuals collected at the Department of Neurology, AHEPA General University Hospital, Thessaloniki, Greece. The participants were categorized into three groups: Alzheimer's disease (AD), Frontotemporal dementia (FTD), and normal control (NC) subjects.

The AD group consisted of 36 participants with a mean Mini-Mental State Examination (MMSE) score of 17.75. The FTD group consisted of 23 participants with a mean MMSE score of 22.17. The NC group included 29 participants with a maximum MMSE score of 30, which indicates normal cognitive function. The demographic information of the dataset is presented in Table 1. The groups were matched by age to prevent bias in classification performance. The significant difference in MMSE scores between groups confirms the cognitive impairment levels characteristic of AD and FTD conditions.

EEG data were acquired from 19 electrode channels positioned according to the international 10-20 system: Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, and O2. Figure 1 shows the spatial distribution of the EEG channels over the brain lobes. Two reference electrodes (A1 and A2) were used during the recording. The EEG signals were

Table 1: Demographic information of EEG dataset.

Group	N	Gender (M/F)	Age (years)	MMSE
AD	36	13/23	66.4 ± 7.9	17.75
FTD	23	14/9	63.6 ± 8.2	22.17
NC	29	11/18	67.9 ± 5.4	30.00



(a) 10-20 standard **(b)** brain lobes

Figure 1: Spatial distribution of electrodes based on the 10-20 standard and brain lobes.

recorded in the resting state with eyes closed at a sampling frequency of 500 Hz.

3. Methodology

This section presents the methodology of the proposed model for AD and FTD classification. Figure 2 shows the high-level workflow, comprising preprocessing, SP feature extraction, feature selection, classification, validation and interpretability analysis. In the primary LOSO analysis, SSA feature selection is performed *inside* each outer training fold, whereas the secondary segment-wise 10-fold analysis uses all 608 SP features without SSA; the validation details are given in Section 3.4.

3.1. Preprocessing

Preprocessing of EEG signals reduces noise and artifacts while preserving relevant information in data. In this study, a standard preprocessing pipeline is applied to remove noise and artifacts from the EEG recording and prepares

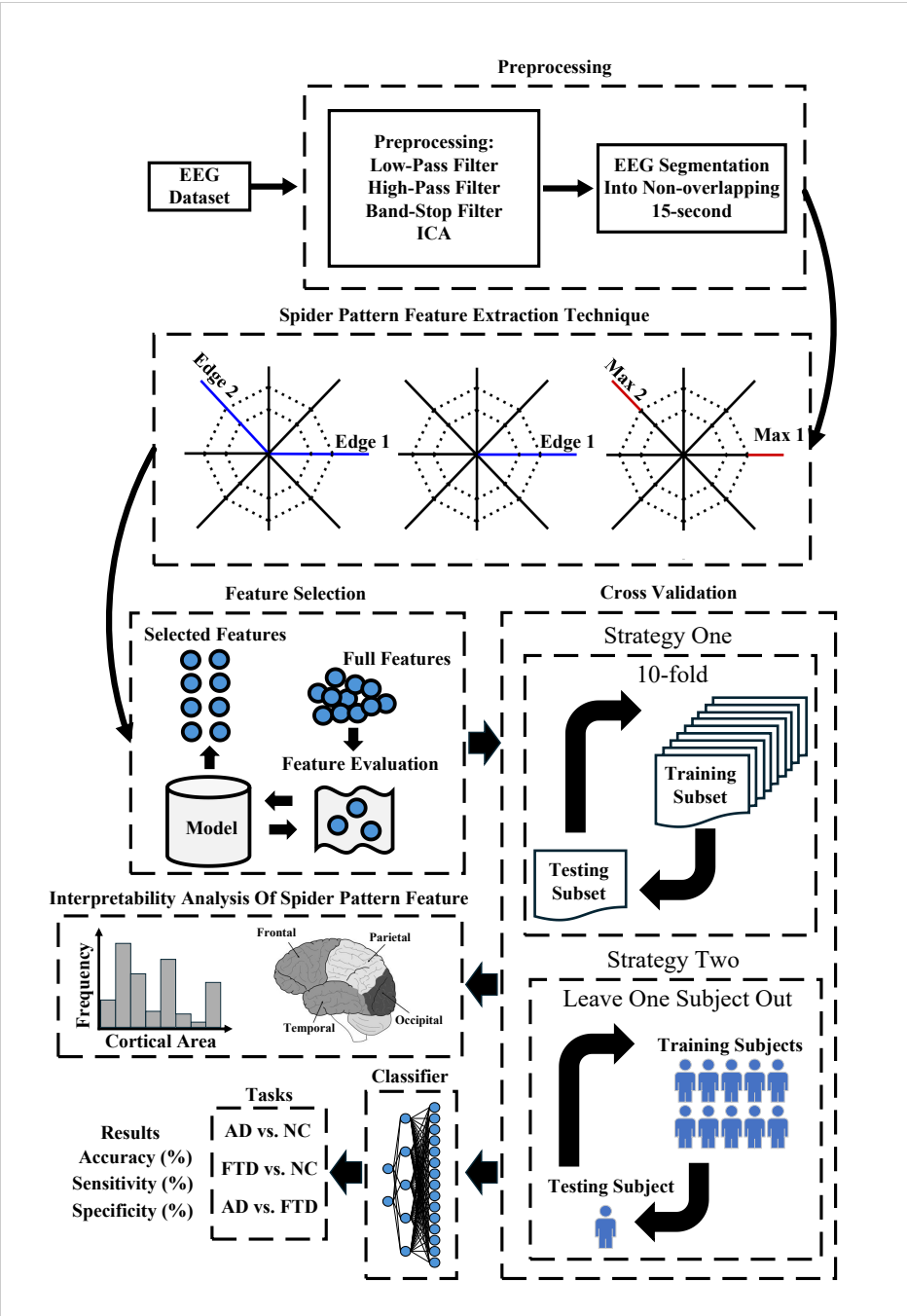


Figure 2: High-level workflow of the proposed framework. In the primary analysis, SSA feature selection is nested within each outer LOSO training fold; the secondary segment-wise 10-fold analysis uses all 608 SP features without SSA.

the data for feature extraction. The preprocessing steps include artifact removal, bandpass filtering, and segmentation of the EEG signals. Movement-related artifacts and other noise sources were identified and removed using the preprocessing toolbox provided in EEGLAB [34]. Independent Component Analysis (ICA) was applied to identify and remove components corresponding to eye blinks, muscle artifacts, and other non-neural sources. Visual inspection by an experienced EEG analyst was performed to verify artifact removal quality.

A series of finite impulse response (FIR) filters were applied to reduce noise. A high-pass filter with a cutoff frequency of 0.5 Hz was applied to remove the baseline DC offset. A low-pass filter with a cutoff frequency of 45 Hz was applied to remove high-frequency noise and muscle artifacts. In addition, a notch filter centered at 50 Hz with a bandwidth of 47 to 53 Hz was applied to remove power line interference. All filtering steps were implemented using zero-phase forward and backward filtering with MATLAB's `filtfilt` function to prevent phase distortion and preserve temporal information.

After preprocessing, the multichannel EEG recordings were separated into 15-second non-overlapping segments. Segments with residual artifacts or excessive noise were detected using amplitude thresholding and visual inspection and were excluded from further analysis. Table 2 summarises the 15-second EEG segments used in the analyses. Each preprocessed 15-second segment with 19 channels and a 500 Hz sampling rate results in a data matrix of size 7500×19 (i.e., 15 seconds $\times\, f_s \times 19$ channels). Figure 3 shows representative 15-second EEG segments for AD, FTD, and NC subjects.

All 88 participants were retained, yielding 1,889 AD, 1,079 FTD and 1,567 NC complete 15-second segments, for a total of 4,535 segments. The study considers three binary classification tasks: AD vs. NC, FTD vs. NC, and AD vs. FTD. Because the AD group has a mean MMSE of 17.75, the present dataset is not used to support a claim of early or prodromal AD detection.

3.2. Spider Pattern Feature Extraction

The Spider Pattern (SP) feature extraction technique is designed to encode local short-term temporal transitions and morphological variations in EEG data. The method analyses consecutive samples within a sliding window and summarises the resulting local transition codes as a compact histogram-based descriptor.

Table 2: Summary of 15-second EEG segments per group after preprocessing.

Group	Number of Subjects	Number of Segments
AD	36	1,889
FTD	23	1,079
NC	29	1,567
Total	88	4,535

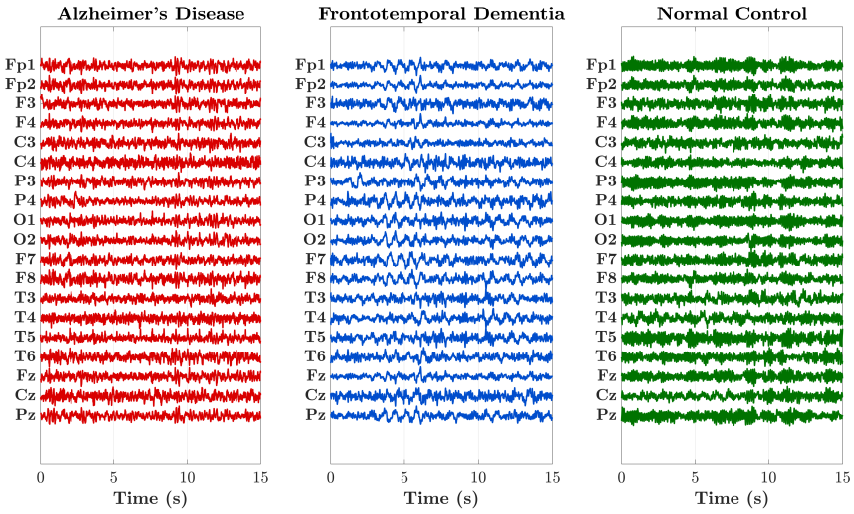


Figure 3: Representative 15-second preprocessed EEG segments across 19 channels for three groups: AD, FTD, and NC.

Let an EEG channel signal be represented as:

$$x = \{x_1, x_2, \ldots, x_L\}, \qquad x \in \mathbb{R}^L, \tag{1}$$

where L is the total number of samples in the segment. The first-order difference signal is computed to emphasize temporal changes:

$$d_i = x_{i+1} - x_i, \qquad i = 1, \ldots, L-1, \tag{2}$$

resulting in the differentiated sequence:

$$d = \{d_1, d_2, \ldots, d_{L-1}\}. \tag{3}$$

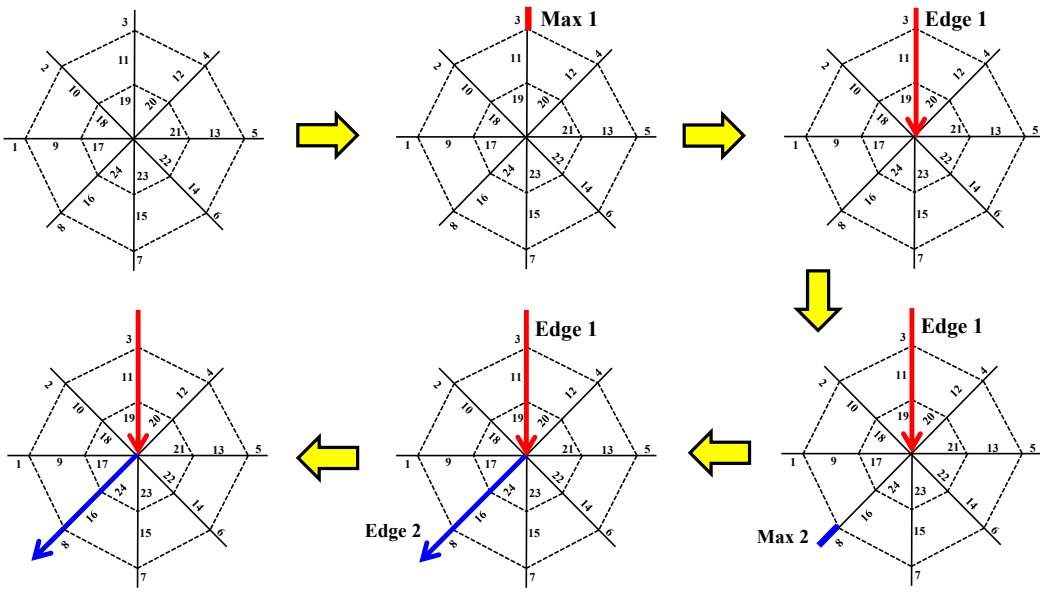


Figure 4: Spider Pattern organisation, showing three concentric rings and eight directional spokes. The geometry is a deterministic rearrangement of temporal samples for local pattern encoding and is not an anatomical or functional-connectivity model.

Differencing removes constant offsets and emphasizes local changes, but it also removes absolute amplitude information; this design choice is therefore examined in the sensitivity analysis rather than assumed to be uniformly beneficial.

A sliding window of length $W = 25$ samples is applied to the differentiated signal:

$$\mathbf{s}_j = \{d_j, d_{j+1}, \ldots, d_{j+24}\}, \qquad j = 1, \ldots, L - 25. \tag{4}$$

Within each window $\mathbf{s}_j$, the first 24 elements are deterministically arranged as three concentric rings with eight directional positions, as illustrated in Figure 4. The 25th element is reserved for the adaptive threshold. With $W = 25$, the 24-node arrangement therefore requires no interpolation. Alternative values of W are examined in the sensitivity analysis.

The two largest *signed* values in the innermost ring (the first eight mapped samples) are identified by sorting:

$$\text{sorted indices} = \text{argsort}(\{d_j, d_{j+1}, \ldots, d_{j+7}\}), \tag{5}$$

with

$$p_a = \text{sorted indices(end)}, \tag{6}$$

$$p_b = \text{sorted indices(end} - 1). \tag{7}$$

Here, p_a and p_b denote the locations of the largest and second-largest signed values. The first path is traversed from the inner ring to the outer ring,

$$E_1 = \{p_a,\ p_a + 8,\ p_a + 16\}, \tag{8}$$

and the second path is traversed from the outer ring to the inner ring,

$$E_2 = \{p_b + 16,\ p_b + 8,\ p_b\}. \tag{9}$$

The complete six-node path is

$$E = [E_1\ E_2] = \{E(1), E(2), E(3), E(4), E(5), E(6)\}. \tag{10}$$

Five consecutive differences are computed along this path:

$$\mathbf{e}_j = \{\Delta_1, \Delta_2, \Delta_3, \Delta_4, \Delta_5\}, \tag{11}$$

where

$$\Delta_k = \mathbf{s}_j[E(k+1)] - \mathbf{s}_j[E(k)], \qquad k = 1, \ldots, 5. \tag{12}$$

The adaptive threshold is defined by the absolute magnitude of the 25th local differentiated value:

$$T_j = |\mathbf{s}_j(25)| = |d_{j+24}|. \tag{13}$$

A small positive machine-precision lower bound relative to the local numerical scale is used when this value is exactly zero. The absolute value is required because a signed threshold can be negative, in which case the ternary conditions below would overlap.

Each edge difference is converted into a mutually exclusive ternary state:

$$q_{j,k} = \begin{cases} 1, & \text{if } \Delta_k \geq T_j, \\ -1, & \text{if } \Delta_k \leq -T_j, \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

This is a ternary local temporal encoding rather than a binary LBP operator. Ternary thresholding is related to local ternary patterns [27], and

one-dimensional local binary patterns have previously been applied to EEG signals [28].

Binary weights are used to convert the positive and negative ternary states into separate decimal codes:

$$\beta = \{2^0, 2^1, 2^2, 2^3, 2^4\} = \{1, 2, 4, 8, 16\}, \tag{15}$$

$$UP_j = \sum_{k=1}^{5} \mathbb{I}(q_{j,k} = 1) \cdot \beta_k, \tag{16}$$

$$DOWN_j = \sum_{k=1}^{5} \mathbb{I}(q_{j,k} = -1) \cdot \beta_k. \tag{17}$$

Each code lies in $[0, 31]$. For each channel, fixed histogram bins corresponding to integer codes 0–31 are used across the $L - 25$ local windows. The upward and downward histograms are summed:

$$SP_c = H_c^{\text{up}} + H_c^{\text{down}} \in \mathbb{R}^{32}, \qquad c = 1, \ldots, 19. \tag{18}$$

Finally, features from the 19 channels are concatenated:

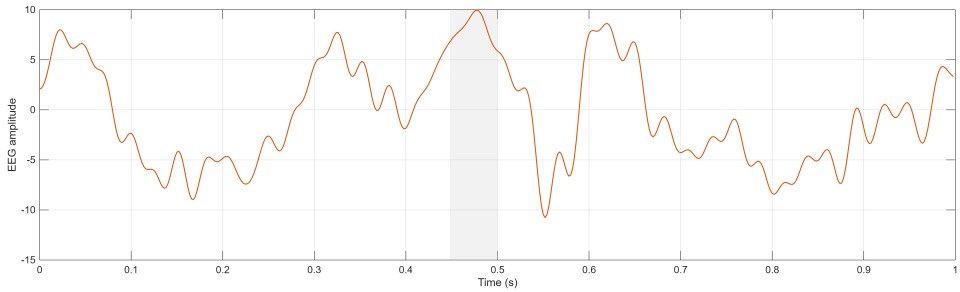
$$\mathbf{SP} = [SP_1; SP_2; \ldots; SP_{19}] \in \mathbb{R}^{608}. \tag{19}$$

Thus, each channel contributes 32 features and each 15-second EEG segment is represented by $19 \times 32 = 608$ SP features.

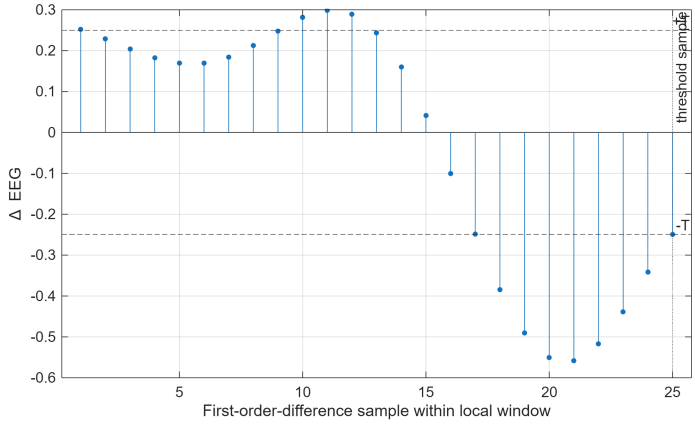
Algorithm 1 summarises the SP feature extraction procedure. Figure 5 illustrates the construction for one local window of a representative EEG excerpt: panel (a) marks the 25-sample window on the signal, panel (b) shows the 25 first-order differences, of which samples 1–24 define the spider geometry and the absolute value of sample 25 defines the threshold ($T_j = 0.2496$ in this example), and panel (c) shows the three-ring arrangement with the selected six-node path along which the five ternary-coded differences are computed.

3.3. The Social Spider Algorithm

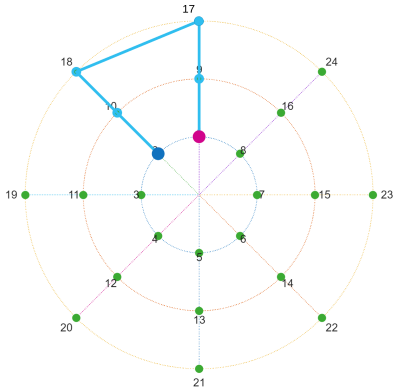
Spiders can be divided into two broad behavioral categories: solitary and social species. Most spiders live alone and are aggressive towards the other members of their own species. Only about 1% of spiders are social, which means that they live in large colonies. Social spiders cooperate in activities



(a) One-second EEG excerpt; the shaded interval is the local SP window.



(b) First-order differences within the window; dashed lines mark $\pm T_j$.



(c) Spider geometry and the selected six-node path.

Figure 5: Construction of a local Spider Pattern code from a representative preprocessed EEG segment (channel Fp1, AD participant). The spider geometry is a deterministic temporal rearrangement of samples and does not represent anatomical connectivity.

Algorithm 1 Spider Pattern (SP) Feature Extraction

Require: EEG segment $\mathbf{X} \in \mathbb{R}^{L\times 19}$, window length $W = 25$.
Ensure: Spider Pattern feature vector $\mathbf{SP} \in \mathbb{R}^{608}$.

Compute first-order differences $\mathbf{D} = \mathrm{diff}(\mathbf{X})$ independently for all channels.
for each channel $c = 1$ to 19 **do**
    Initialize UP_list and $DOWN_list$.
    for $j = 1$ to $L - 25$ **do**
        Extract $\mathbf{s}_j = \{d_j, \ldots, d_{j+24}\}$.
        Map the first 24 values to three rings of eight positions.
        Select the two largest signed values in the first ring.
        Construct $E = \{p_a, p_a + 8, p_a + 16, p_b + 16, p_b + 8, p_b\}$.
        Compute five consecutive differences along E.
        Set $T_j = |d_{j+24}|$ with a positive numerical lower bound.
        Encode the five differences into mutually exclusive ternary states.
        Convert positive and negative states into UP_j and $DOWN_j$ codes.
        Append codes to the corresponding lists.
    end for
    Build fixed 32-bin histograms for integer codes 0–31 and sum them.
end for
Concatenate the 19 channel histograms into $\mathbf{SP} \in \mathbb{R}^{608}$.
return SP.

such as tending eggs and juveniles, finding food, and defending the colony. Social spiders have a developed nervous system and use vibrational signals for communication. These vibrations of the web, caused by the movement and activity of its constituents, provide important information. Therefore, social spiders coordinate with each other by receiving vibrations in various cases, from hunting prey to defending the web [35, 36]. Hence, this vibration-based communication method has led to a technique known as the Social Spider Algorithm.

The Social Spider Algorithm (SSA) is a relatively new metaheuristic technique [36]. SSA is a variant of Social Spider Optimization (SSO) [35]. The main difference between SSO and SSA is in modeling spider behavior: SSO models spiders into two categories of different sexes and then assigns different search strategies to each group. It was inspired by the complex mating rituals within natural colonies. In SSA, all spiders fall under the same category,

and their actions depend on a single factor: the intensity of the vibration. Therefore, all agents share the same searching mechanism, and SSA is easier to implement and often converges faster [35, 36]. Due to its advantages mentioned above, SSA has been known as a very efficient optimizer when dealing with complex nonlinear optimization problems, and has been widely adopted in engineering optimizations, communications, energy management, machine learning, artificial intelligence, etc. [35, 37, 38, 39, 36, 40, 41].

From a mathematical standpoint, SSA models the search space as a multidimensional spider web. The position of each spider corresponds to a candidate solution to the optimization problem. Spiders may move across the web but are not permitted to leave it, as positions outside the web represent invalid solutions. The fitness or quality of each candidate solution is evaluated using the objective function, analogous to the likelihood of finding a food source at that location. Whenever a spider moves to a new position, it generates a vibration whose intensity depends on the fitness at that point. This vibration then propagates throughout the web. Each vibration contains information about the source spider, and others access this information upon detecting it. Social spiders are capable of distinguishing between different vibrations with high precision and can distinguish multiple signals traveling simultaneously along the web [35, 36].

As mentioned earlier, in SSA, the main mechanism of information exchange is vibration, and hence it is one of the core concepts of the algorithm. Each vibration has two properties: the position of the source spider and the intensity of vibration. If the position of the spider a at time t is denoted by $P_a(t)$ or simply P_a, then $I(P_a, P_b, t)$ denotes the intensity of the vibration felt by the spider b positioned at P_b at time t caused by a spider a. It takes values in $[0, +\infty)$. More precisely, $I(P_s, P_s, t)$ is the vibration intensity sensed by a spider s at its own position. The intensity at the source depends on the fitness of that position $f(P_s)$, and is given by the following equation [36]:

$$I(P_s, P_s, t) = \log\left(\frac{1}{f(P_s) - C} + 1\right) \tag{20}$$

where C is a small constant introduced to avoid division by zero. Under this relation, the smaller the fitness function value, the stronger the vibration; hence, the other spiders would be attracted to move toward that position. Moreover, similar to natural spider webs, the strength of vibration decreases

with distance. This attenuation is modeled as following [36]:

$$I(P_a, P_b, t) = I(P_a, P_a, t) \times \exp\left(-\frac{D(P_a, P_b)}{\bar{\sigma} \times r_a}\right) \tag{21}$$

Here, σ is the standard deviation in each dimension about the spider position, and $D(\cdot)$ is the Manhattan distance between two spiders, which follows Equation 22. The parameter r_a is the attenuation rate. The smaller the r_a, the stronger the attenuation; hence, the weaker the influence of long distance vibrations on the decision [36].

$$D(P_a, P_b) = \|P_a - P_b\|_1 \tag{22}$$

SSA works through three main phases: initialization, iteration, and termination. These stages are performed in sequence, driving the optimization process from the creation of the initial population to the finding of the final solution. In fact, the algorithm first creates an initial population and then runs iteratively until the stopping criteria are reached and the optimal solution is obtained [36].

In this phase, the basic parameters used by SSA are set, which include the number of spiders, the maximum number of iterations, and other control parameters. An initial population of spiders is then created randomly and scattered over the web (i.e., the search space). In all cases, the total number of spiders remains the same throughout the simulation. After placing the spiders randomly within the search space, the fitness function is evaluated at each position. The best fitness value out of all the individuals is recorded as the initial global optimum. Then, for each spider, an initial target vibration is assigned to its current position, while all auxiliary attributes—including initial vibration intensity, change counters, and similar internal variables—are initialized to zero. Each spider holds a memory, storing the aforementioned individual information. By the end of the initialization phase, the algorithm is prepared to proceed to the iteration phase [36, 37].

During each iteration of SSA, the algorithm performs a cyclical sequence of operations. These operations are carried out in the following sub-steps: fitness evaluation, vibration generation, update the target vibration, mask changing, following position, random walk, and constraint handling [36, 37]. The sub-steps of SSA are explained in detail in the following:

- **Fitness Evaluation**: At the beginning of every iteration, all spiders update their positions and compute their fitness values at the new

position. If a superior fitness value is found, the global optimum is correspondingly updated.

- **Vibration Generation**: In this step, each spider generates a vibration at its current position according to Equation 20, and this vibration is propagated across the web following Equation 21. In addition, the spider s receives vibrations emitted by all other spiders in the population ($|pop|$). Each received vibration contains information about the source position and its attenuation adjusted intensity after propagation. The set of vibrations received by the spider s is denoted by V.

- **Update the target vibration**: Once all the received vibrations have been collected, each spider finds the strongest vibration V_s^{best} and compares its intensity to that of the target vibration V_s^{tar} stored in its memory. If the new V_s^{best} exceeds the intensity of V_s^{tar}, the spider replaces V_s^{tar} with the new V_s^{best} vibration in its memory and resets C_s (the number of iterations since the target vibration was last updated) to zero. Otherwise, the target vibration remains unchanged, and C_s is incremented by one.

- **Mask Changing**: Once the V_s^{tar} is determined, the spider s must perform a random walk toward it. SSA uses a dimensional mask, which is a binary vector of length D (the number of problem dimensions), to guide this movement. This mask determines which dimensions of the spider's position will be updated according to the received vibrational information and which dimensions will remain unchanged. This mask helps the spider to modify only the relevant parts of its positions based on the optimal position or stronger vibration signals.

  For each dimension, a value of 1 means that the dimension will be updated, and a value of 0 means the previous value will be preserved. As noted earlier, during the initialization phase, all mask elements are set to zero, allowing the spiders to initially move in a completely random and unbiased manner. In the subsequent iterations, however, each spider updates its mask with a probability of $1 - p_c^{(C_s)}$, where $p_c \in (0, 1)$ is a user-defined parameter that controls the likelihood of keeping the current mask, and C_s is the number of iterations since the last target vibration change. Thus, as C_s increases, the probability of modifying the mask also increases, thereby strengthening the exploratory capacity of the algorithm.

If the algorithm decides to modify the mask, each bit is updated independently such that it becomes 1 with probability p_m and 0 with probability $1 - p_m$. The parameter p_m is also user-defined, which controls the variability of movement across dimensions. If all bits are zero, one bit is randomly set to 1, and if all bits are one, one bit is randomly set to 0.

- **Following position**: After the dimensional mask is determined, each spider s must generate a new position for itself, referred to as the following position P_s^{fo}. Suppose the position of the source of the received vibrations is denoted by P_s^i and the position of the target vibration source is denoted by P_s^{tar}, where $i = \{1, 2, \ldots, |pop|\}$ represents the number of members in the population. To enable each spider to utilize not only its own position but also the vibrational information from other spiders for its next move, the value of each dimension i of this position is determined based on the dimensional mask as follows [36]:

$$P_{s,i}^{fo} = \begin{cases} P_{s,i}^{tar}, & m_{s,i} = 0, \\ P_{r,i}, & m_{s,i} = 1, \end{cases} \tag{23}$$

  where r is a random integer selected independently from the interval $[1, |pop|]$, and $m_{s,i}$ denotes the i-th dimension of the dimensional mask of spider s.

- **Random Walk**: With the following position determined, the spider s proceeds to perform its random walk, moving toward a new position in the search space by combining information from both its current position and its previous movement. In practice, the spider first advances by a random fraction of its prior movement in the same direction as before. It then moves toward the following position, but this second component of motion is applied dimension-wise, meaning that each coordinate is updated independently using a separate random coefficient between zero and one. In other words, every dimension progresses toward the target with its own randomly generated scaling factor. Therefore, the random walk formula for spider s is given as follows [36]:

$$P_s(t+1) = P_s + (P_s - P_s(t-1)) \times r + (P_s^{fo} - P_s) \odot R \tag{24}$$

  Here, $\odot$ denotes element-by-element multiplication, R is a vector of random numbers between zero and one, and r represents the proportion

of movement in the previous direction. After completing the random movement, the spider stores its new position to be used as the previous movement direction in the next iteration.

- **Constraint Handling**: Since spiders may move outside the web during the random walk step, invalid positions must be corrected to remain consistent with the search space boundaries. In SSA, the reflection method is used to correct such positions. In this approach, any dimension of the position vector that exceeds the upper or lower bound is reflected back toward the permissible space. The corrected position (boundary-constraint-free position) of spider s in dimension i is calculated as follows [36]:

$$P_{s,i}(t+1) = \begin{cases} (\bar{x}_i - P_{s,i}) \times r, & \text{if } P_{s,i}(t+1) > \bar{x}_i, \\ (P_{s,i} - \underline{x}_i) \times r, & \text{if } P_{s,i}(t+1) < \underline{x}_i, \end{cases} \tag{25}$$

  Here, $\bar{x}_i$ and $\underline{x}_i$ denote the upper and lower bounds of the search space in dimension i, respectively, and r is a random floating-point number in the interval $(0, 1)$. In this way, the final position of the spider remains within the valid range without violating the constraints.

The iteration phase continues until one of the stopping criteria is satisfied. These criteria may include reaching the maximum number of iterations, consuming the allowed CPU time, reaching a desired error, exceeding a predefined number of iterations without any new improvement in the fitness value, or any other termination condition. Once the iteration phase concludes, the SSA enters its final stage, in which the best solution-along with its corresponding fitness value-is obtained. The pseudo code of SSA is presented in Algorithm 2.

In this study, SSA is employed as a wrapper-based feature selection method to identify informative SP features. The standard SSA search space is configured with one continuous variable in $[0, 1]$ for each of the 608 SP features. A feature is retained when the corresponding position value is at least 0.5; this binarisation is an application of the standard SSA representation rather than a new SSA algorithm.

For the primary subject-independent analysis, feature selection is nested inside every outer LOSO fold. For each held-out participant, only the remaining training participants are available to SSA. To avoid overweighting

Algorithm 2 Social Spider Algorithm (SSA)

Initialize all SSA parameters.
Generate the population of spiders (*pop*) and allocate memory for their attributes.
Initialize the target vibration V_s^{tar} for each spider.
while the stopping criterion is not satisfied **do**
    for each spider s in *pop* **do**
        Evaluate the fitness of s.
        Generate a vibration at the current position of s.
    end for
    for each spider s in *pop* **do**
        Compute the intensity of all vibrations V generated by the population.
        Select the strongest vibration V_s^{best} from V.
        if the intensity of V_s^{best} is greater than the stored V_s^{tar} **then**
            Update the stored target vibration V_s^{tar} with V_s^{best}.
        end if
        Update the counter c_s.
        Generate a random number $r \in [0, 1)$.
        if $r > p_c^{(c_s)}$ **then**
            Update the dimension mask m_s.
        end if
        Construct the following position P_s^{fo}.
        Perform a random walk.
        Handle any violated constraints.
    end for
end while
return the best solution obtained.

participants with more EEG segments, segment-level SP vectors are first averaged within each training participant for the SSA fitness calculation. Five stratified subject-wise inner folds are then created within the outer training set. A $K = 5$ nearest-neighbour classifier is evaluated on these inner folds, and the feature-selection fitness is

$$f(P_s) = 1 - BA_{\text{inner}}(P_s) + \lambda \frac{m}{608}, \tag{26}$$

where BA_{inner} is inner subject-level balanced accuracy, m is the number of

selected features, and $\lambda = 0.01$ is a sparsity penalty. The SSA population size is 10, the maximum number of iterations is 50, $r_a = 1.0$, $p_m = 0.1$, and $p_c = 0.7$. Five independent SSA runs are performed per outer fold, and the mask with the lowest training-only fitness is retained. The outer test participant therefore does not influence feature selection.

Two training-only comparators are evaluated under the identical outer LOSO splits: (i) the full 608-feature SP representation and (ii) minimum-redundancy maximum-relevance (mRMR) feature selection [42]. The mRMR ranking is computed using only outer-training data, with the number of retained features matched to the SSA-selected count in the corresponding fold.

3.4. Feedforward Neural Network

After feature selection, the selected SP features are fed into a feedforward neural network (FFNN) for classification [43]. The FFNN contains one hidden layer with 10 neurons and is trained with scaled conjugate-gradient backpropagation for a maximum of 100 epochs. A completely fresh network is created in every fold. Standardisation parameters are estimated from training data only and applied unchanged to the corresponding test data, and no network weights are carried between folds.

The primary evaluation uses outer LOSO CV. All segments from one participant are held out, while the remaining participants are used for nested feature selection and FFNN training. During FFNN fitting, segment weights are adjusted so that each training participant contributes equal total weight despite unequal segment counts. For the held-out participant, positive-class probabilities are averaged across that participant's segments, and the participant is assigned to the positive class when the mean score is at least 0.5.

A secondary 10-fold experiment is also performed at the *segment* level to quantify the effect of the validation unit. It uses all 608 SP features, no SSA, a fresh FFNN in every fold, and training-only standardisation. Participant identity is deliberately not used to form the folds; consequently, segments from the same participant may appear in both training and test data. The overlap audit subsequently confirmed 100% test-participant overlap with training in every fold, so this analysis is not interpreted as subject-independent generalisation.

Table 3: FFNN settings and evaluation configurations.

Setting	**Secondary segment-wise 10-fold**	**Primary nested LOSO**
Architecture	Input – hidden (10 neurons) – output	Input – hidden (10 neurons) – output
Training	Scaled conjugate gradient, max. 100 epochs	Scaled conjugate gradient, max. 100 epochs
Input features	All 608 SP features	Nested SSA-selected SP features
Feature selection	None	Inside each outer training fold
Scaling	Training-fold only	Outer-training only
Split unit	Segments	Participants
Subject overlap	Allowed; 100% observed	None
Test decision	Segment-level	Mean held-out subject score, threshold 0.5
Interpretation	Secondary segment discrimination	Unseen-subject generalisation

The trained classifiers are evaluated from true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Accuracy, sensitivity and specificity are defined as

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%, \tag{27}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \times 100\%, \tag{28}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \times 100\%. \tag{29}$$

For the primary subject-level analysis, balanced accuracy, precision, F1 score, Matthews correlation coefficient (MCC), and ROC area under the curve (AUC) are also reported. Wilson 95% confidence intervals are used for accuracy, sensitivity and specificity, and AUC confidence intervals are obtained from 1,000 subject-level bootstrap resamples. Exact two-sided paired McNemar tests compare SSA with the full-feature and mRMR models using predictions from the same held-out participants.

3.5. Interpretability Analysis

Each SP feature is traceable to its source EEG channel. The channel-to-region mapping in Table 4 is used for descriptive analysis [44, 45, 46]. However, the order of selected feature labels is not treated as physiological connectivity, and raw regional counts are not used because regions contain unequal numbers of electrodes.

Three complementary analyses are used. First, channel-level selection rate is calculated as the proportion of corresponding SP bins selected across outer LOSO folds, and region-level selection rate is the mean channel rate within each region. Second, SSA stability is assessed from five independent SSA runs within each outer fold using selected-feature counts and pairwise Jaccard similarity. Third, held-out permutation importance is calculated after model training by permuting selected SP bins belonging to a channel across the held-out participant's segments and measuring the reduction in probability assigned to the participant's true class. Region-level permutation importance is the mean of the corresponding channel values. These analyses describe model feature use and are not interpreted as functional connectivity or direct disease biomarkers.

3.6. Sensitivity Analyses

Sensitivity analyses examine five families of SP design choices: window length W of 15, 20, 25, 30 or 35 samples; difference order 0, 1 or 2; three threshold definitions (absolute final sample, median absolute node value, and median-absolute-deviation based); signed versus absolute dominant-direction ranking; and segment durations of 5, 10, 15 and 30 seconds. Each condition is evaluated using subject-disjoint stratified 5-fold CV, all 608 SP features, equal participant weighting in FFNN training, and a fresh network in each fold. SSA is omitted so that the descriptor choice is not confounded by stochastic feature selection. For $W \neq 25$, the first $W - 1$ local values are linearly resampled to the fixed 24-node spider geometry; $W = 25$ remains the primary non-interpolated configuration. These analyses are exploratory and are not used to select a new primary configuration after inspecting test performance. A dedicated synthetic-noise perturbation experiment and an ablation of the number of dominant spider directions were not performed.

Table 4: Brain regions, associated EEG channels, and functional descriptions.

Symbol	Region	Channels	Functional Description
FL	Frontal Left	Fp1, F3, F7	Planning, organization, logical reasoning, and controlled cognitive processing.
FR	Frontal Right	Fp2, F4, F8	Emotional interpretation, creative reasoning, and intuitive decision-making.
Fz	Frontal Midline	Fz	Attention regulation, executive control, and conflict monitoring.
CL	Central Left	C3	Voluntary motor control, somatosensory integration, and movement execution.
CR	Central Right	C4	Motor coordination, feedback integration, and sensorimotor rhythm generation.
Cz	Central Midline	Cz	Motor planning, body coordination, and bilateral motor activity integration.
PL	Parietal Left	P3	Spatial reasoning, sensory integration, and multisensory information interpretation.
PR	Parietal Right	P4	Spatial awareness, visual-spatial orientation, and body position perception.
Pz	Parietal Midline	Pz	High-level sensory integration and spatial decision-making.
TL	Temporal Left	T3, T5	Language comprehension, auditory perception, and memory encoding.
TR	Temporal Right	T4, T6	Emotional processing, non-verbal auditory recognition, and episodic memory.
OL	Occipital Left	O1	Early visual processing, shape and pattern recognition.
OR	Occipital Right	O2	Visual-spatial analysis, motion interpretation, and complex visual perception.

4. Results and Discussion

The framework was implemented in MATLAB R2023a. The three binary tasks were AD vs. NC, FTD vs. NC, and AD vs. FTD, with 608 SP features extracted from each 15-second segment. The primary evidence is the nested subject-independent LOSO analysis. The segment-wise 10-fold experiment is retained as a secondary analysis because it intentionally allows segments from the same participant in training and testing.

Table 5: Secondary segment-wise 10-fold results using all 608 SP features and FFNN. Every fold had 100% test-participant overlap with training; these results are not subject-independent.

Task	Segments	TP	TN	FP	FN	Accuracy (%)	Bal. Acc. (%)	AUC
AD vs. NC	3,456	1,713	1,426	141	176	90.83	90.84	0.975
FTD vs. NC	2,646	996	1,467	100	83	93.08	92.96	0.980
AD vs. FTD	2,968	1,773	924	155	116	90.87	89.75	0.964

For the secondary segment-wise analysis, sensitivity and specificity were 90.68% and 91.00% for AD vs. NC, 92.31% and 93.62% for FTD vs. NC, and 93.86% and 85.63% for AD vs. FTD. Because segments are clustered within participants and every test participant is also represented in training, ordinary segment-level confidence intervals are not reported as if the segments were independent observations.

Table 6: Primary subject-level performance of SP + nested SSA + FFNN under outer LOSO validation. Confidence intervals (CI) are subject-level: Wilson intervals for accuracy, sensitivity and specificity, and bootstrap intervals for AUC.

Metric	AD vs. NC	FTD vs. NC	AD vs. FTD
Participants (N)	65	52	59
TP / TN / FP / FN	30 / 21 / 8 / 6	15 / 22 / 7 / 8	23 / 8 / 15 / 13
Accuracy, % (95% CI)	78.46 (67.03–86.71)	71.15 (57.73–81.67)	52.54 (40.04–64.73)
Sensitivity, % (95% CI)	83.33 (68.11–92.13)	65.22 (44.89–81.19)	63.89 (47.58–77.52)
Specificity, % (95% CI)	72.41 (54.28–85.30)	75.86 (57.89–87.78)	34.78 (18.81–55.11)
Balanced accuracy, %	77.87	70.54	49.34
MCC	0.562	0.413	−0.014
AUC (95% CI)	0.810 (0.704–0.910)	0.712 (0.558–0.853)	0.562 (0.395–0.711)

Under the primary nested LOSO analysis, AD vs. NC achieved 78.46% accuracy and AUC 0.810, while FTD vs. NC achieved 71.15% accuracy and

AUC 0.712. AD vs. FTD was substantially weaker: accuracy was 52.54%, balanced accuracy 49.34%, MCC approximately zero, and the AUC 95% confidence interval included 0.5. The results therefore do not support reliable subject-independent discrimination between AD and FTD.

The contrast with Table 5 is substantial. Segment-wise 10-fold accuracy exceeded nested LOSO accuracy by 12.37 percentage points for AD vs. NC, 21.93 points for FTD vs. NC, and 38.33 points for AD vs. FTD. The overlap audit showed 100% test-participant overlap with training in every segment-wise fold, illustrating why segment-level random splits should not be interpreted as performance on unseen participants [23, 24].

Table 7: Sensitivity analyses under subject-disjoint 5-fold CV with all 608 SP features and no SSA. Each cell reports accuracy (%) / balanced accuracy (%) / AUC. Rows marked † correspond to the primary SP configuration.

Condition	**AD vs. NC**	**FTD vs. NC**	**AD vs. FTD**
Window length			
$W = 15$	70.77 / 70.59 / 0.733	65.38 / 64.47 / 0.681	47.46 / 44.38 / 0.446
$W = 20$	67.69 / 67.48 / 0.723	75.00 / 73.99 / 0.780	50.85 / 47.95 / 0.523
$W = 25^{\dagger}$	69.23 / 68.87 / 0.722	75.00 / 74.44 / 0.756	55.93 / 52.90 / 0.559
$W = 30$	72.31 / 72.32 / 0.754	71.15 / 70.54 / 0.742	59.32 / 54.89 / 0.531
$W = 35$	78.46 / 77.87 / 0.814	69.23 / 68.82 / 0.727	71.19 / 68.54 / 0.722
Difference order			
0	70.77 / 71.26 / 0.771	67.31 / 65.74 / 0.798	62.71 / 59.24 / 0.601
1†	69.23 / 68.87 / 0.722	75.00 / 74.44 / 0.756	55.93 / 52.90 / 0.559
2	61.54 / 60.92 / 0.673	75.00 / 73.99 / 0.759	61.02 / 58.64 / 0.632
Threshold definition			
Absolute final sample†	69.23 / 68.87 / 0.722	75.00 / 74.44 / 0.756	55.93 / 52.90 / 0.559
Median absolute node value	63.08 / 62.64 / 0.673	69.23 / 68.82 / 0.760	50.85 / 48.73 / 0.536
MAD of node values	70.77 / 70.59 / 0.670	71.15 / 70.54 / 0.672	49.15 / 46.56 / 0.484
Dominant-direction ranking			
Signed†	69.23 / 68.87 / 0.722	75.00 / 74.44 / 0.756	55.93 / 52.90 / 0.559
Absolute magnitude	69.23 / 69.20 / 0.686	73.08 / 72.71 / 0.736	57.63 / 55.07 / 0.565
Segment duration			
5 s	72.31 / 71.98 / 0.728	75.00 / 74.89 / 0.729	52.54 / 49.34 / 0.548
10 s	72.31 / 71.98 / 0.712	76.92 / 77.06 / 0.721	49.15 / 46.56 / 0.513
15 s†	69.23 / 68.87 / 0.722	75.00 / 74.44 / 0.756	55.93 / 52.90 / 0.559
30 s	70.77 / 70.26 / 0.735	76.92 / 77.06 / 0.774	55.93 / 52.90 / 0.560

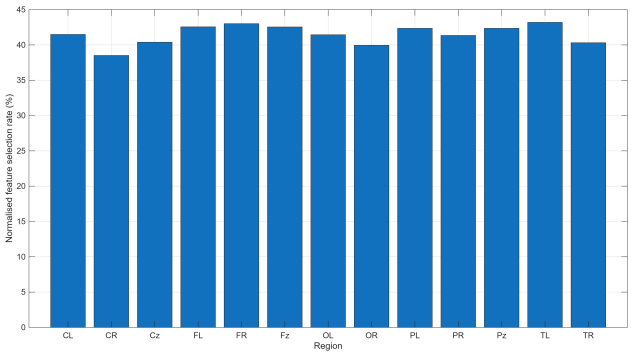
The sensitivity analysis shows that no single alternative dominates all

three classification tasks. In particular, first-order differencing is not uniformly superior to the undifferenced signal, and the effects of window length and segment duration are task-dependent. The $W = 25$ configuration is retained as the primary SP specification because it maps 24 temporal values directly to the three-ring eight-direction geometry plus one threshold sample; the higher numerical results for some alternative settings are not used for post hoc redefinition of the primary method.

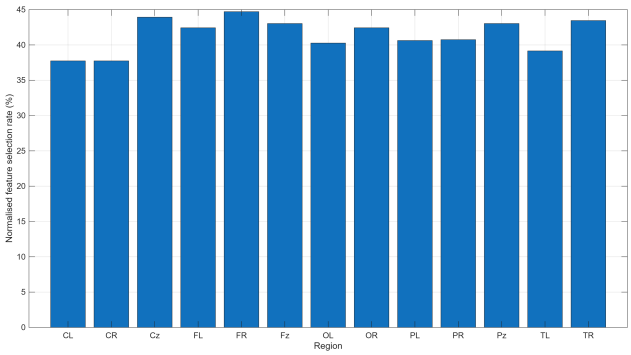
The normalised selection rates in Figure 6 are broadly distributed rather than showing a compelling frontal predominance. Held-out permutation effects are generally small and several are negative, indicating that perturbing a region's selected features does not consistently reduce confidence in the held-out participant's true class. Consequently, no connectivity, symbolic-sequence or entropy interpretation is drawn from these results.

The performance of the proposed model is compared with selected existing approaches in Table 8. These studies use different preprocessing, input representations, segment lengths, clinical variables and validation units, so the table provides context rather than a controlled ranking. Segment-wise k-fold values are not treated as equivalent to subject-level LOSO estimates.

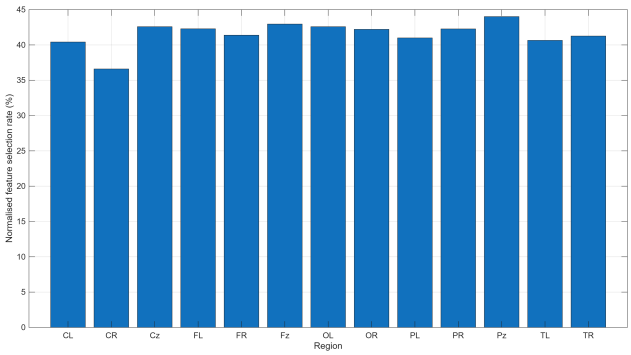
DICE-Net reported 83.28% AD-vs.-NC accuracy under LOSO [47], while Ma et al. [51] reported 76.9% for AD vs. NC and 90.4% for FTD vs. NC using electrode-pair communication features and demographic information. Lal et al. [48] and Chen et al. [49] reported high k-fold values under protocols that are not directly comparable to the present subject-level evaluation. Zheng et al. [50] reported LOSO values of 87.69%, 82.69% and 72.9% for the three tasks. Akbari et al. [32] evaluated DGPM on the same AHEPA cohort and reported LOSO accuracies of 69.23%, 75.00% and 64.41% for AD vs. NC, FTD vs. NC and AD vs. FTD, respectively, while segment-wise 10-fold accuracies exceeded 96%; this alternative representation therefore shows a similar qualitative gap between segment-level and unseen-subject evaluation. The present results are discussed as protocol-specific measurements rather than evidence of superiority. This interpretation is also consistent with validation-aware EEG modelling in a different clinical setting, where segment-level separability and subject-wise generalisation were deliberately treated as distinct evidential questions [25]. Three recent deep-learning studies [29, 30, 31] were also considered, but paired statistical comparison is not possible without predictions for the same held-out participants.



(a) AD vs. NC

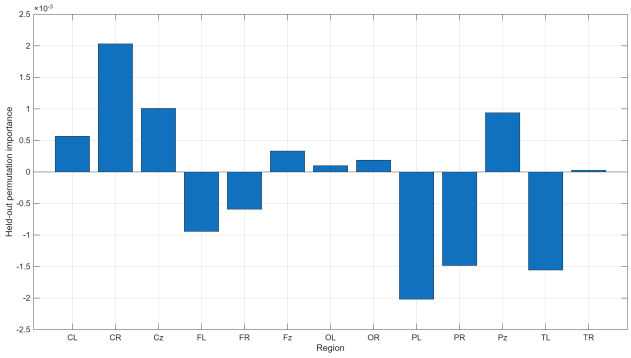


(b) FTD vs. NC

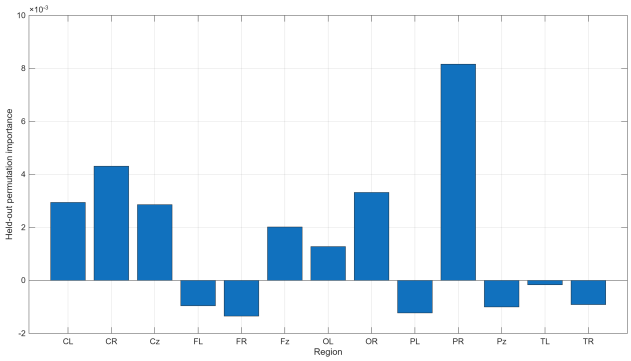


(c) AD vs. FTD

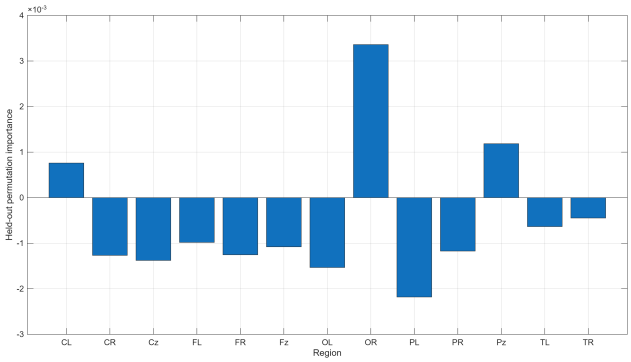
Figure 6: Regional interpretability analysis. (a)–(c) Normalised regional SSA selection rates across outer LOSO folds for AD vs. NC, FTD vs. NC and AD vs. FTD; rates are normalised within channels and regions so that regions with more electrodes do not automatically receive larger values. (d)–(f) Held-out regional permutation importance for the same tasks (next page). These quantities describe model feature use and are not measures of functional connectivity.



(d) AD vs. NC



(e) FTD vs. NC



(f) AD vs. FTD

Figure 6: (Continued.) (d)–(f) Held-out regional permutation importance for AD vs. NC, FTD vs. NC and AD vs. FTD, defined as the mean reduction in the probability assigned to the held-out participant's true class when the selected SP bins of each channel are permuted, averaged within regions.

Table 8: Contextual comparison with selected EEG-based approaches on the same AHEPA dataset. Values are not treated as controlled head-to-head comparisons because validation and processing protocols differ.

Study	Data	Method	Segment	CV	Task	Accuracy
[47]	EEG	DICE-Net, band power/coherence	30 s	LOSO	AD vs. NC	83.28%
[48]	EEG	Handcrafted features + KNN	1 s, 90% overlap	15-fold	AD vs. NC	91.00%
				15-fold	FTD vs. NC	93.00%
				15-fold	AD vs. FTD	81.60%
[49]	EEG + MMSE	CNN–ViT fusion	4 s	10-fold	AD vs. NC	87.30%
				10-fold	FTD vs. NC	82.90%
				10-fold	AD vs. FTD	81.60%
[50]	EEG	MTRRP + SVM	5 min	LOSO	AD vs. NC	87.69%
				LOSO	FTD vs. NC	82.69%
				LOSO	AD vs. FTD	72.90%
[32]	EEG	DGPM + pretrained CNNs	5 s	LOSO	AD vs. NC	69.23%
				LOSO	FTD vs. NC	75.00%
				LOSO	AD vs. FTD	64.41%
[51]	EEG + demographics	Electrode-pair communication + SVM	1 min	LOSO	AD vs. NC	76.90%
				LOSO	FTD vs. NC	90.40%
Ours	**EEG**	**SP + full features + FFNN**	**15 s**	**segment-wise 10-fold**	**AD vs. NC**	**90.83%**
				segment-wise 10-fold	**FTD vs. NC**	**93.08%**
				segment-wise 10-fold	**AD vs. FTD**	**90.87%**
	EEG	**SP + nested SSA + FFNN**	**15 s**	**LOSO**	**AD vs. NC**	**78.46%**
				LOSO	**FTD vs. NC**	**71.15%**
				LOSO	**AD vs. FTD**	**52.54%**

Table 9: Subject-level accuracies of the SSA-selected, full-feature and mRMR-selected models under the same outer LOSO splits, with exact paired McNemar tests.

	Accuracy (%)			McNemar p	
Task	**SSA**	**Full**	**mRMR**	**SSA vs. Full**	**SSA vs. mRMR**
AD vs. NC	78.46	76.92	72.31	1.000	0.344
FTD vs. NC	71.15	65.38	76.92	0.375	0.375
AD vs. FTD	52.54	54.24	55.93	1.000	0.791

SSA does not consistently produce the highest numerical accuracy. It is highest for AD vs. NC, mRMR is highest for FTD vs. NC, and all three approaches are weak for AD vs. FTD. None of the exact paired McNemar comparisons is significant at $p < 0.05$, so no claim is made that SSA significantly outperforms the full representation or mRMR.

Table 10: Number of SSA-selected features and within-fold repeat stability (pairwise Jaccard similarity of five SSA runs) across the primary outer LOSO analysis.

	Selected features				Jaccard
Task	**Mean**	**SD**	**Min**	**Max**	**mean ± SD**
AD vs. NC	254.12	16.84	226	309	0.2678 ± 0.0140
FTD vs. NC	254.77	17.89	229	325	0.2676 ± 0.0126
AD vs. FTD	252.63	17.67	218	295	0.2633 ± 0.0134

The low pairwise Jaccard similarities indicate limited agreement in the exact subsets selected by repeated SSA runs. This restricts biological interpretation of individual selected SP bins even when similar classification objectives can be achieved by different subsets.

In summary, SP provides a compact channel-resolved time-domain representation based on local temporal transitions and a non-negative ternary threshold. Nested subject-level validation removes feature-selection leakage and yields conservative estimates of unseen-participant performance. Internal comparators, confidence intervals and paired tests show that SSA is not consistently or significantly superior to the full-feature and mRMR alternatives, and the sensitivity, stability and held-out importance analyses indicate

which aspects of the method are robust and which should not be given a neurophysiological interpretation.

Two implications extend beyond the specific descriptor. First, the 12–38 percentage-point difference between segment-wise and subject-level accuracy obtained here with identical features, data and classifier provides a direct, within-study measurement of participant leakage on the AHEPA dataset, which is widely used for method development. Reported accuracies on this dataset should therefore be read together with their validation unit, and subject-level estimates with confidence intervals should be regarded as the minimum standard for claims of generalisation. Second, the AD-vs.-FTD result shows that a descriptor can separate each dementia group from controls at the subject level while providing no usable discrimination between the two dementia groups; differential diagnosis of dementia subtypes from resting-state EEG features therefore requires separate evidence and should not be inferred from case-control performance.

This study has several limitations. The cohort contains 88 participants from a single centre and no external dataset was used for independent validation. The AD group has a mean MMSE of 17.75, so the current data do not establish early or prodromal AD detection. Although the preprocessing pipeline described in Section 3.1 was applied, the experiments did not compare alternative preprocessing pipelines or quantify sensitivity to residual preprocessing choices. The sensitivity analyses are exploratory subject-disjoint 5-fold experiments without SSA and should not be interpreted as a second confirmatory test set. A dedicated synthetic-noise perturbation experiment and an ablation of the number of dominant spider directions were not performed. Held-out permutation importance was used instead of SHAP, and neither the regional analysis nor the feature order supports causal or functional-connectivity claims. Finally, computational runtime was not benchmarked in a way that supports claims of real-time or clinical efficiency.

5. Conclusion

This study evaluated Spider Pattern as an adaptive time-domain EEG feature extraction technique for AD, FTD and NC classification. In the primary analysis, feature selection was nested entirely within each outer LOSO training fold, yielding subject-level accuracies of 78.46% for AD vs. NC, 71.15% for FTD vs. NC and 52.54% for AD vs. FTD. The first two results indicate moderate case-control discrimination, whereas the AD-vs.-FTD result

does not support reliable subject-independent subtype discrimination. SSA did not significantly outperform the full 608-feature representation or mRMR and showed limited selection stability. A secondary segment-wise 10-fold analysis produced higher accuracies of 90.83%, 93.08% and 90.87%, but every fold contained test participants also represented in training; those values therefore describe segment-level discrimination rather than unseen-subject generalisation. The interpretability analysis is limited to normalised feature-selection distributions and held-out predictive importance, without interpreting selected-feature order as brain connectivity. By reporting leakage-free subject-level estimates, their uncertainty and a negative result alongside segment-level estimates on the same data, the study provides a calibrated reference for interpreting EEG dementia-classification accuracies reported on this public dataset. Future work should evaluate SP on external cohorts, compare preprocessing pipelines systematically, and investigate more stable feature representations for differential dementia classification.

Data and Code Availability

The analysed EEG dataset is publicly available as described by Miltiadous et al. [33]. The analyses were implemented in MATLAB R2023a. The analysis code is available from the corresponding author on reasonable request.

Funding

This research received no funding.

Competing Interests

The authors declare that they have no competing interests.

Ethics and Consent to Participate

This study utilized a publicly available EEG dataset provided by Miltiadous et al. [33]. The original data collection was approved by the Scientific and Ethics Committee of AHEPA General University Hospital, Thessaloniki, Greece. As this study involved secondary analysis of publicly available de-identified data, no additional ethics approval was required.

Declarations

Clinical trial registration: Not applicable. **Conflict of interest:** The authors declare that they have no conflict of interest.

CRediT Authorship Contribution Statement

Hesam Akbari: Conceptualization, Methodology, Software, Data curation, Formal analysis, Investigation, Validation, Visualization, Writing – original draft, Writing – review & editing, Supervision, Project administration.
Muhammad Tariq Sadiq: Conceptualization, Methodology, Formal analysis, Validation, Investigation, Writing – original draft, Writing – review & editing, Supervision, Project administration.
Hamed Shamsi: Methodology, Writing – original draft, Writing – review & editing, Supervision.
Kuldeep Singh: Validation, Writing – original draft, Writing – review & editing, Supervision.
Mahdieh Ghazvini: Writing – original draft, Visualization.
Siuly Siuly: Conceptualization, Writing – original draft, Writing – review & editing, Supervision.
Mutlu Mete: Resources, Writing – review & editing, Supervision, Project administration.

References

[1] S. Sonko, M. I. Houssam, K. D. Bissadu, B. O'Connor, G. Hossain, Scoping the landscape of deep learning for alzheimer's disease stage classification: Methods, challenges, and opportunities, BME frontiers 6 (2025) 0202.

[2] P. A. Mir, N. Kumar, G. T. Bhutia, P. Chaudhary, G. Kaur, S. K. Gupta, The aging gut–glia–immune axis in alzheimer's disease: microbiome-derived mediators of neuroinflammation and therapeutic innovation, GeroScience (2026) 1–41.

[3] A. Association, 2019 alzheimer's disease facts and figures, Alzheimer's & dementia 15 (3) (2019) 321–387.

[4] B. Dubois, A. Padovani, P. Scheltens, A. Rossi, G. Dell'Agnello, Timely diagnosis for alzheimer's disease: a literature review on benefits and challenges, Journal of Alzheimer's disease 49 (3) (2015) 617–631.

[5] M. S. Rafii, P. S. Aisen, Detection and treatment of alzheimer's disease in its preclinical stage, Nature aging 3 (5) (2023) 520–531.

[6] L. Wang, J. Glass, L. Kourtis, R. Au, Multi-modal data analysis for early detection of alzheimer's disease and related dementias, The Journal of Prevention of Alzheimer's Disease 13 (1) (2026) 100399.

[7] S. K. Khare, U. R. Acharya, Adazd-net: Automated adaptive and explainable alzheimer's disease detection system using eeg signals, Knowledge-Based Systems 278 (2023) 110858.

[8] S. K. Khare, V. Bajaj, U. R. Acharya, Spwvd-cnn for automated detection of schizophrenia patients using eeg signals, IEEE Transactions on Instrumentation and Measurement 70 (2021) 1–9.

[9] S. K. Chandrasekar, J. Arthanari, K. K. Chandrasekar, E. Gaviria, S. Shashank, A. Jethi, J. John, S. Bopparaju, I. Potulapati, M. A. Mateen, et al., Advanced imaging techniques (pet, fmri, dti) in early detection of neurodegenerative diseases: a systematic review, Health Science Reports 8 (7) (2025) e70855.

[10] D. Puri, P. Kachare, S. Khare, I. Al-Shourbaji, A. Jabbari, A. Alameen, Hybrid reptile-snake optimizer based channel selection for enhancing alzheimer's disease detection, Journal of Bionic Engineering 22 (2) (2025) 884–900.

[11] S. K. Khare, S. March, P. D. Barua, V. M. Gadre, U. R. Acharya, Application of data fusion for automated detection of children with developmental and mental disorders: A systematic review of the last decade, Information Fusion (2023) 101898.

[12] S. Bagherzadeh, A. Shalbaf, Eeg-based schizophrenia detection using fusion of effective connectivity maps and convolutional neural networks with transfer learning, Cognitive Neurodynamics 18 (5) (2024) 2767–2778.

[13] S. Bagherzadeh, A. Shalbaf, A. Shoeibi, M. Jafari, R. San Tan, U. R. Acharya, Developing an eeg-based emotion recognition using ensemble deep learning methods and fusion of brain effective connectivity maps, IEEE Access (2024).

[14] H. Akbari, W. Korani, P. M. Kumar, R. Rostami, R. Kazemi, Dynamical pattern of successive bits to predict the outcomes of the ssri and rtms depression therapies using eeg signals, Computers in Biology and Medicine 196 (2025) 110500.

[15] H. Akbari, W. Korani, S. Muhammad, R. Rostami, R. Kazemi, M. T. Sadiq, Predicting depression therapy outcomes using eeg-derived amplitude polar maps, Brain Sciences 15 (9) (2025) 977.

[16] H. Akbari, W. Korani, Early detection of depression and alcoholism disorders by eeg signal, in: International Conference on Neural Information Processing, Springer, 2023, pp. 439–452.

[17] H. Akbari, M. T. Sadiq, Detection of focal and non-focal eeg signals using non-linear features derived from empirical wavelet transform rhythms, Physical and Engineering Sciences in Medicine 44 (1) (2021) 157–171.

[18] H. Akbari, M. T. Sadiq, A. U. Rehman, Classification of normal and depressed eeg signals based on centered correntropy of rhythms in empirical wavelet transform domain, Health Information Science and Systems 9 (2021) 1–15.

[19] A. Azadnouran, H. Akbari, M. T. Sadiq, D. Smith, M. Mete, Comparative evaluation of time–frequency transformations and pretrained cnn models for eeg-based parkinson's disease detection, BioMedInformatics 6 (2) (2026) 12. doi:10.3390/biomedinformatics6020012.

[20] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, H. Adeli, D. P. Subha, Automated eeg-based screening of depression using deep convolutional neural network, Computer methods and programs in biomedicine 161 (2018) 103–113.

[21] S. Bagherzadeh, M. R. Norouzi, S. B. Hampa, A. Ghasri, P. T. Kouroshi, S. Hosseininasab, M. A. G. Zadeh, A. M. Nasrabadi, A subject-

independent portable emotion recognition system using synchrosqueezing wavelet transform maps of eeg signals and resnet-18, Biomedical Signal Processing and Control 90 (2024) 105875.

[22] Y. Liu, W. Xue, L. Yang, M. Li, Deep learning-based eeg emotion recognition: A review, Brain Sciences 16 (1) (2025) 41.

[23] G. Brookshire, J. Kasper, N. M. Blauch, Y. C. Wu, R. Glatt, D. A. Merrill, S. Gerrol, K. J. Yoder, C. Quirk, C. Lucero, Data leakage in deep learning studies of translational eeg, Frontiers in Neuroscience 18 (2024) 1373515.

[24] A. Miltiadous, A. Ntetska, V. Aspiotis, E. Moustakli, M. G. Tsipouras, A. T. Tzallas, N. Giannakeas, E. Glavas, P. Angelidis, K. D. Tzimourta, The ahepa eeg benchmark: setting the standard for machine learning in dementia diagnosis, a scoping review, Cognitive Neurodynamics 20 (2026) 95. doi:10.1007/s11571-026-10464-w.

[25] H. Akbari, M. Mete, R. Rostami, R. Kazemi, M. T. Sadiq, Validation-aware retrospective eeg treatment-response modelling using chaotic pattern of prime numbers features: Segment-level separability and subject-wise generalisation, Bioengineering 13 (6) (2026) 659. doi:10.3390/bioengineering13060659.

[26] S. Kinahan, P. Saidi, A. Daliri, J. Liss, V. Berisha, Achieving reproducibility in eeg-based machine learning, in: Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency, 2024, pp. 1464–1474.

[27] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, IEEE Transactions on Image Processing 19 (6) (2010) 1635–1650. doi:10.1109/TIP.2010.2042645.

[28] P. P. M. Shanir, K. A. Khan, Y. U. Khan, O. Farooq, H. Adeli, Automatic seizure detection based on morphological features using one-dimensional local binary pattern on long-term eeg, Clinical EEG and Neuroscience 49 (5) (2018) 351–362. doi:10.1177/1550059417744890.

[29] C. Bunterngchit, T. Chearanai, Y. Bunterngchit, Advanced eeg-based classification of alzheimer's disease using cnn-lstm-attention architec-

ture, in: 2024 22nd International Conference on Research and Education in Mechatronics (REM), IEEE, 2024, pp. 107–112. doi:10.1109/REM63063.2024.10735597.

[30] C. Bunterngchit, L. H. Baniata, A. Rasool, NeuroXAI: Explainable deep learning for eeg-based detection of alzheimer's and parkinson's diseases, in: 2025 IEEE 25th International Conference on Bioinformatics and Bioengineering (BIBE), IEEE, 2025, pp. 612–619. doi:10.1109/BIBE66822.2025.00107.

[31] C. Bunterngchit, C. Jamroen, S. Aslam, A. Rasool, O. W. Samuel, Attention-guided electroencephalography fusion with fractal and nonlinear dynamics decodes early alzheimer's biomarkers, Computers and Electrical Engineering 133 (2026) 111076. doi:10.1016/j.compeleceng.2026.111076.

[32] H. Akbari, A. Azadnouran, M. T. Sadiq, S. Siuly, E. Ghaderpour, M. Mete, Alzheimer's disease classification using dynamic geometric pattern maps: A novel image representation of eeg signals, Biomedical Signal Processing and Control 130 (2027) 111554.

[33] A. Miltiadous, K. D. Tzimourta, N. Giannakeas, M. G. Tsipouras, T. Afrantou, P. Ioannidis, A. T. Tzallas, Alzheimer's disease and frontotemporal dementia: A robust classification method of eeg signals and a comparison of validation methods, Diagnostics 11 (8) (2021) 1437.

[34] A. Delorme, S. Makeig, Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis, Journal of neuroscience methods 134 (1) (2004) 9–21.

[35] E. Cuevas, M. Cienfuegos, D. Zaldívar, M. Pérez-Cisneros, A swarm optimization algorithm inspired in the behavior of the social-spider, Expert Systems with Applications 40 (16) (2013) 6374–6384.

[36] J. James, V. O. Li, A social spider algorithm for global optimization, Applied soft computing 30 (2015) 614–627.

[37] L. Abualigah, A. A. Al Turk, A. M. Ikotun, R. A. Zitar, A. R. Alsoud, N. Khodadadi, A. G. Hussien, H. Jia, Social spider optimization algorithm: survey and new applications, in: Metaheuristic optimization algorithms, Elsevier, 2024, pp. 15–31.

[38] S. O. Frimpong, I. E. Agbehadji, A. Abayomi, R. C. Millham, E. Freeman, M. M. Ujakpa, Economic and reliability determination of sustainable renewable energy mix based on social spider prey optimization algorithm., Soft Computing-A Fusion of Foundations, Methodologies & Applications 27 (15) (2023).

[39] H. Nashaat, M. Salah, R. Rizk, A novel downlink scheduler based on social spider algorithm for enhanced lte networks, in: International Conference on Advanced Intelligent Systems and Informatics, Springer, 2025, pp. 319–333.

[40] K. Suruli, V. Ila, Social spider optimization algorithm-based optimized power management schemes, Electric Power Components and Systems 48 (11) (2020) 1111–1124.

[41] C. UmaRani, S. Ramalingam, S. Dhanasekaran, K. Baskaran, An hybrid machine learning and improved social spider optimization based clustering and routing protocol for wireless sensor network, Wireless Networks 31 (2) (2025) 1885–1910.

[42] H. Peng, F. Long, 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 27 (8) (2005) 1226–1238. doi:10.1109/TPAMI.2005.159.

[43] D. Yu, F. Cao, Construction and approximation rate for feedforward neural network operators with sigmoidal functions, Journal of Computational and Applied Mathematics 453 (2025) 116150.

[44] G. Boztas, T. Tuncer, O. Aydogmus, M. Yildirim, A dcslbp based intelligent machine malfunction detection model using sound signals for industrial automation systems, Computers and Electrical Engineering 119 (2024) 109541.

[45] H. W. Loh, C. P. Ooi, E. Aydemir, T. Tuncer, S. Dogan, U. R. Acharya, Decision support system for major depression detection using spectrogram and convolution neural network with eeg signals, Expert Systems 39 (3) (2022) e12773.

[46] E. Aydemir, T. Tuncer, S. Dogan, R. Gururajan, U. R. Acharya, Automated major depressive disorder detection using melamine pattern with eeg signals, Applied Intelligence 51 (9) (2021) 6449–6466.

[47] A. Miltiadous, E. Gionanidis, K. D. Tzimourta, N. Giannakeas, A. T. Tzallas, Dice-net: a novel convolution-transformer architecture for alzheimer detection in eeg signals, IEEe Access 11 (2023) 71840–71858.

[48] U. Lal, A. V. Chikkankod, L. Longo, A comparative study on feature extraction techniques for the discrimination of frontotemporal dementia and alzheimer's disease with electroencephalography in resting-state adults, Brain sciences 14 (4) (2024) 335.

[49] Y. Chen, H. Wang, D. Zhang, L. Zhang, L. Tao, Multi-feature fusion learning for alzheimer's disease prediction using eeg signals in resting state, Frontiers in neuroscience 17 (2023) 1272834.

[50] H. Zheng, X. Xiong, X. Zhang, Multi-threshold recurrence rate plot: a novel methodology for eeg analysis in alzheimer's disease and frontotemporal dementia, Brain Sciences 14 (6) (2024) 565.

[51] Y. Ma, J. K. S. Bland, T. Fujinami, Classification of alzheimer's disease and frontotemporal dementia using electroencephalography to quantify communication between electrode pairs, Diagnostics 14 (19) (2024) 2189.