Journal of Integrative Neuroscience

J. Integr. Neurosci. **2026**; 25(10): 53849
https://doi.org/10.31083/JIN53849

Original Research

A Lightweight EEG-Based Alzheimer's Disease Detection Framework on an Edge-Computing Platform Toward Future Intelligent Digital Twin Systems

Amal Alqahtani[1,*], Saad Al-Ahmadi[1], Adel Soudani[1]

[1]Department of Computer Science, College of Computer and Information Sciences, King Saud University, 11451 Riyadh, Saudi Arabia
*Correspondence: 443203273@student.ksu.edu.sa (Amal Alqahtani)
Academic Editors: Parisa Gazerani and Bettina Platt
Submitted: 14 May 2026 Revised: 21 July 2026 Accepted: 5 August 2026 Published: 10 October 2026

Abstract

Background: Alzheimer's disease (AD) progresses slowly through several stages until the patient reaches full cognitive decline. The electroencephalography (EEG) alterations associated with AD often emerge before symptom onset, suggesting that accessible and repeated EEG-based assessment could facilitate earlier detection and longitudinal monitoring. However, this type of assessment requires an AD detection model operating on edge-computing platforms within the Internet of Medical Things (IoMT). At present, many existing AD detection models rely on complex deep learning architectures with numerous parameters, which significantly limit their practicality for on-edge deployment. **Methods:** We propose an on-edge AD detection framework comprising a lightweight dilated hybrid deep learning (DHDL) model with a channel selection approach to reduce the computational and power demands on the evaluated edge-computing platform. A key architectural component is the inclusion of dual-granularity dilated temporal convolutions. This method expands the receptive field at two scales to capture both short- and long-term temporal features from EEG signals without increasing the number of parameters. Subsequently, several hybrid lightweight layers extract distinct temporal and spatial features and are integrated with attention mechanisms. Additionally, our channel selection approach first forms an initial subset of EEG channels guided by individual channel classification accuracy, and then iteratively refines the selected channels using hill-climbing optimization. Here, we evaluated our framework on two publicly available EEG datasets recorded for the same group of subjects (n = 48 subjects): a conventional closed-eye (CE) dataset and a more challenging open-eye (OE) dataset. Validation followed a subject-independent Leave-2-Subjects-Out (L2SO) cross-validation protocol. Real-time deployment was tested on an edge-computing platform. **Results:** The proposed framework achieved 92.56% accuracy for the CE dataset and 75.15% accuracy for the OE dataset. Inference required 10.45 ms and 8.23 ms per segment for the CE and OE datasets, respectively, and consumed 9.69 mJ and 6.95 mJ of energy per segment, respectively. **Conclusions:** These results demonstrate the feasibility of the proposed framework as a real-time, on-edge inference module for AD detection. Although a complete intelligent digital twin system is not implemented in this work, the proposed framework can serve as an inference component for future digital twin systems, supporting early AD detection and progression monitoring.

Keywords: Alzheimer detection; intelligent digital twin; IoMT; edge computing platform; EEG signal; deep learning; channel selection

1. Introduction

Alzheimer's disease (AD), a progressive, irreversible neurological disorder that affects aging populations, is the most common form of dementia. Currently, among the 50 million people living with dementia worldwide, nearly 60% are AD patients [1]. To date, AD has no known cure, but early detection with appropriate treatment may help delay symptom progression, thereby preserving neurological functions and enhancing patients' quality of life [2,3]. Therefore, accessible tools are urgently needed to support earlier and more frequent neurological assessments. One promising paradigm that could alleviate this need is Intelligent Digital Twins (IDTs) [4,5]. For AD, IDTs can be viewed as intelligent virtual replicas of actual AD patients and would continuously collect and analyze real-time electroencephalography (EEG) signals to produce actionable insights into the patients' AD progression. Creating AD IDTs would require the following three components: EEG data, edge-computing platforms that can host the inference module, and a lightweight deep learning model that would serve as the inference module.

EEG is a noninvasive, accessible, and cost-effective tool that reveals AD biomarkers before the onset of clinical AD symptoms [6]. EEG also enables clinicians to determine the stage of AD and track its progression [7]. Recently, many deep learning models have been proposed for EEG-based AD detection and monitoring; however, these models were trained and evaluated on high-performance platforms. This approach has raised potential concerns regarding the real-world applicability of current models in edge-computing platforms with constrained computational, memory, and energy resources.

Edge-computing platforms provide accessible tools for the detection and monitoring of neurological diseases, such as epilepsy [8] and Parkinson's disease [9]. Accord-

Copyright: © 2026 The Author(s). Published by IMR Press.
This is an open access article under the CC BY 4.0 license.

Publisher's Note: IMR Press stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

ingly, these devices show promise for early AD detection and personalized AD progression tracking. Nevertheless, deploying AD detection models on these platforms remains challenging due to the conflict between the limited capabilities of these platforms and the high complexity of EEG data and AD detection models. On the one hand, edge-computing platforms are constrained by their low computing capability and limited memory and because they run on batteries with limited power. On the other hand, EEG signals are complex, nonstationary, and highly dimensional, so they demand high processing overhead. In this context, selecting an optimal set of EEG channels for AD detection is an essential step in on-edge deployment. This selection step would reduce data dimensionality and lower the overall computational and power demands while also improving model performance by excluding irrelevant and noisy channels.

Recent studies have demonstrated notable success in EEG-based AD detection. These efforts have explored a range of architectures, including standard convolutional neural network (CNN)-based models [10,11], transformer-based models [12], and hybrid CNN-long short-term memory (LSTM) models [13]. LSTM is a type of recurrent neural network (RNN) designed to process sequential data and remember long-term dependencies. However, the use of open-eye (OE) EEG recordings has proven more challenging, as these recordings are affected by noise, making classification harder [14,15]. Thus, despite the notable progress reported in previous AD detection studies, several issues remain unaddressed. First, although many researchers have suggested that their methods are appropriate for resource-constrained devices, to the best of our knowledge, no previous study has tested an AD detection model on an edge-computing platform [10,11,13]. Second, whereas some studies have been carried out on channel selection, their approaches have relied on handcrafted features to select the optimal channels, so the selected channels varied according to the extracted features [16,17]. Third, current deep learning models for AD detection often rely on complex architectures with a large number of parameters, limiting their suitability for on-edge deployment.

In the present study, we propose a novel framework for EEG-based AD detection implemented on an edge-computing platform to address the current challenges and limitations. Specifically, we propose a dilated hybrid deep learning (DHDL) model that can target diagnosis under edge deployment constraints. Our lightweight deep learning model is designed for edge deployment with a channel selection approach. The proposed model leverages dual-granularity dilated temporal convolutions that scan the raw signals to learn two-scale temporal features. The rationale of this dilation approach is to expand the receptive field at two scales without increasing the parameter count. The result is a reduction in the number of parameters and in the computational complexity of the subsequent layers. The model then learns deeper spatiotemporal features using hybrid layers with attention mechanisms. This approach differs from conventional temporal convolution networks (TCN) [18], which employ dilated causal convolutions in a single sequential path. It also differs from conventional multiscale temporal convolutions [19], which require significantly more parameters. To the best of our knowledge, this use of two parallel dilated temporal kernels with different receptive fields has not been previously explored in EEG-based models.

We also propose an effective channel selection approach that consists of two stages. In the first stage, we identify an initial channel subset by iteratively training and testing the DHDL model on individual channels and selecting the best-performing channels. Then, in the second stage, we refine this initial selection using hill-climbing optimization [20]. We then evaluate our framework using a conventional closed-eye (CE) dataset and a challenging OE dataset [15,21]. Both datasets were collected from the same subjects. The experimental results demonstrate the effectiveness of our framework under both the CE and OE EEG recording conditions. Notably, from a broader perspective, the proposed framework does not represent a complete IDT. Instead, it offers an on-edge inference module that would enable the future development of IDT for AD detection and monitoring. The following are the main contributions of this study:

- On-edge AD detection framework: We propose an on-edge AD detection framework developed for Internet of Medical Things (IoMT) environments. The framework combines a lightweight deep learning model with a channel selection approach. We demonstrate on-edge deployment readiness through real-time inference on an edge-computing platform.
- DHDL model: We develop a lightweight DHDL model that leverages dual dilated temporal convolutions that are integrated with hybrid spatial and temporal layers and attention.
- Channel selection approach: We propose a new channel selection approach that is guided by individual channel accuracy and refined by hill-climbing optimization.
- Comprehensive evaluation framework: We validate the model design and the effect of channel selection via extensive ablation experiments, followed by real-time on-edge measurements. The results demonstrate that the proposed framework achieves higher classification performance than reported for recent AD detection studies, while maintaining low resource utilization and minimal energy consumption on the edge-computing platform.

The remainder of this study is organized as follows. Section 2 presents the methods, including the proposed model, channel selection approach, datasets and preprocessing, training environment, edge environment, and performance metrics. Section 3 presents the results, including the classification performance and on-edge results. Section

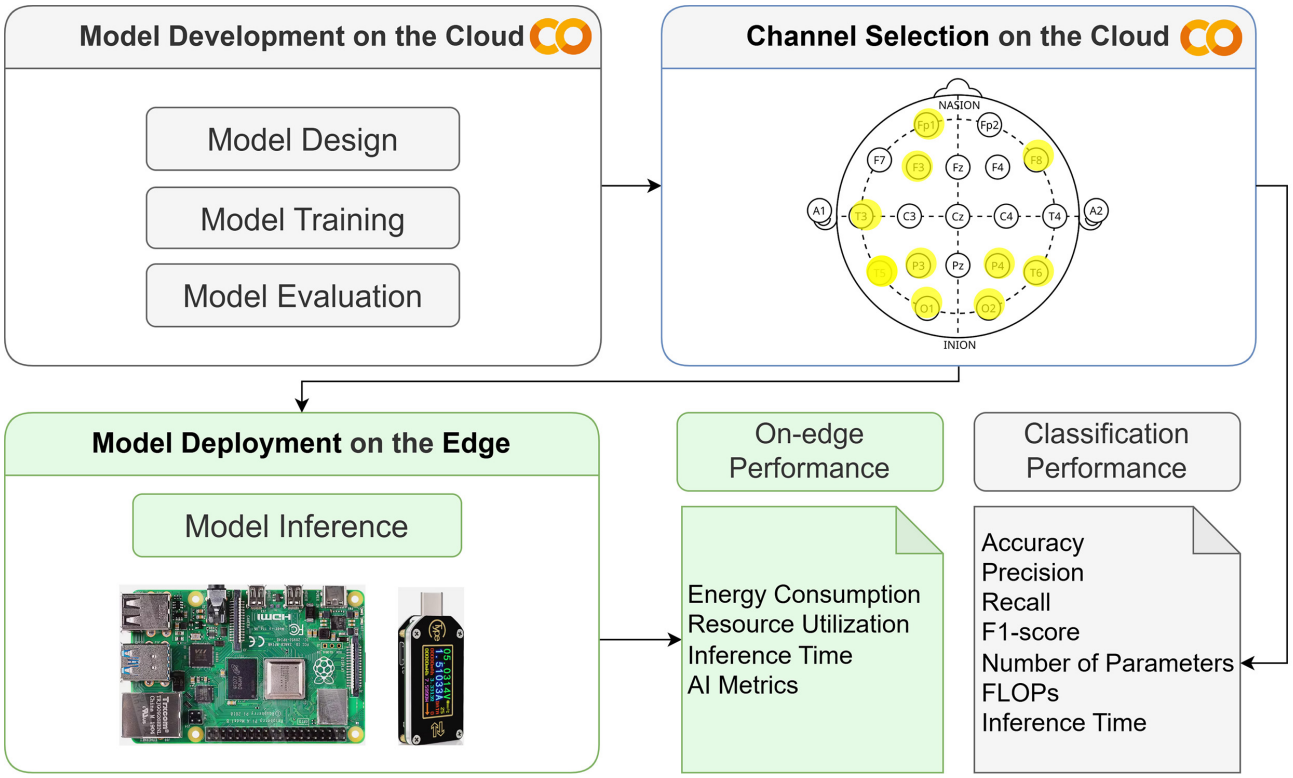


Fig. 1. The general workflow of the proposed framework. FLOPs, floating-point operations.

4 presents the discussion, including comparisons with state-of-the-art models and edge-oriented models, the framework rationale, and limitations and future work. Finally, Section 5 concludes the study.

2. Methods

The general workflow of the proposed framework, as illustrated in Fig. 1, involved three steps. The first step was to design, train, and evaluate the DHDL model using all EEG channels. The second step was to select the optimal channel subset using the proposed selection approach. The third step was deployment of the model on an edge-computing platform to test the model's real-time performance. The first and second steps were run on the cloud, whereas the third step was run on the edge-computing platform. During model development, we developed several techniques, namely dilated temporal convolutions, depthwise convolutions, attention mechanisms, and an LSTM network. Collectively, these layers were carefully designed and tuned to enhance the classification performance with a minimal number of parameters. A channel selection approach was subsequently applied to further reduce the data dimensionality and computational overhead.

We first examined the model's performance on individual channels and produced an initial subset of EEG channels. The subset was updated iteratively using hill-climbing optimization. We reported the conventional model performance evaluation metrics and assessed the effect of the proposed channel selection approach. Next, we optimized the trained model using a compression technique and deployed it to an edge-computing platform with the test data. The model inference was then conducted on the edge-computing platform. During model inference, we strictly monitored resource utilization, calculated energy consumption, and reported the inference time per segment. This section details the DHDL model and the proposed channel selection approach.

2.1 DHDL Model Architecture

The DHDL comprises five main blocks, each intended to extract distinct feature representations while keeping the overall architecture lightweight and adequate for on-edge deployment. The dilated dual-granularity feature fusion block (DDG-FF) extracts and fuses temporal features, the dilated deep feature extraction block (DDFE) captures diverse spatiotemporal feature representations, the spatial attention block (S-Att) emphasizes the most informative feature maps [22], the temporal attention block (T-Att) highlights relevant time segments, the LSTM block models long-range temporal dependencies, and a fully connected layer finally produces the classification output. The DHDL model is illustrated in Fig. 2, and its specifications are summarized in Table 1.

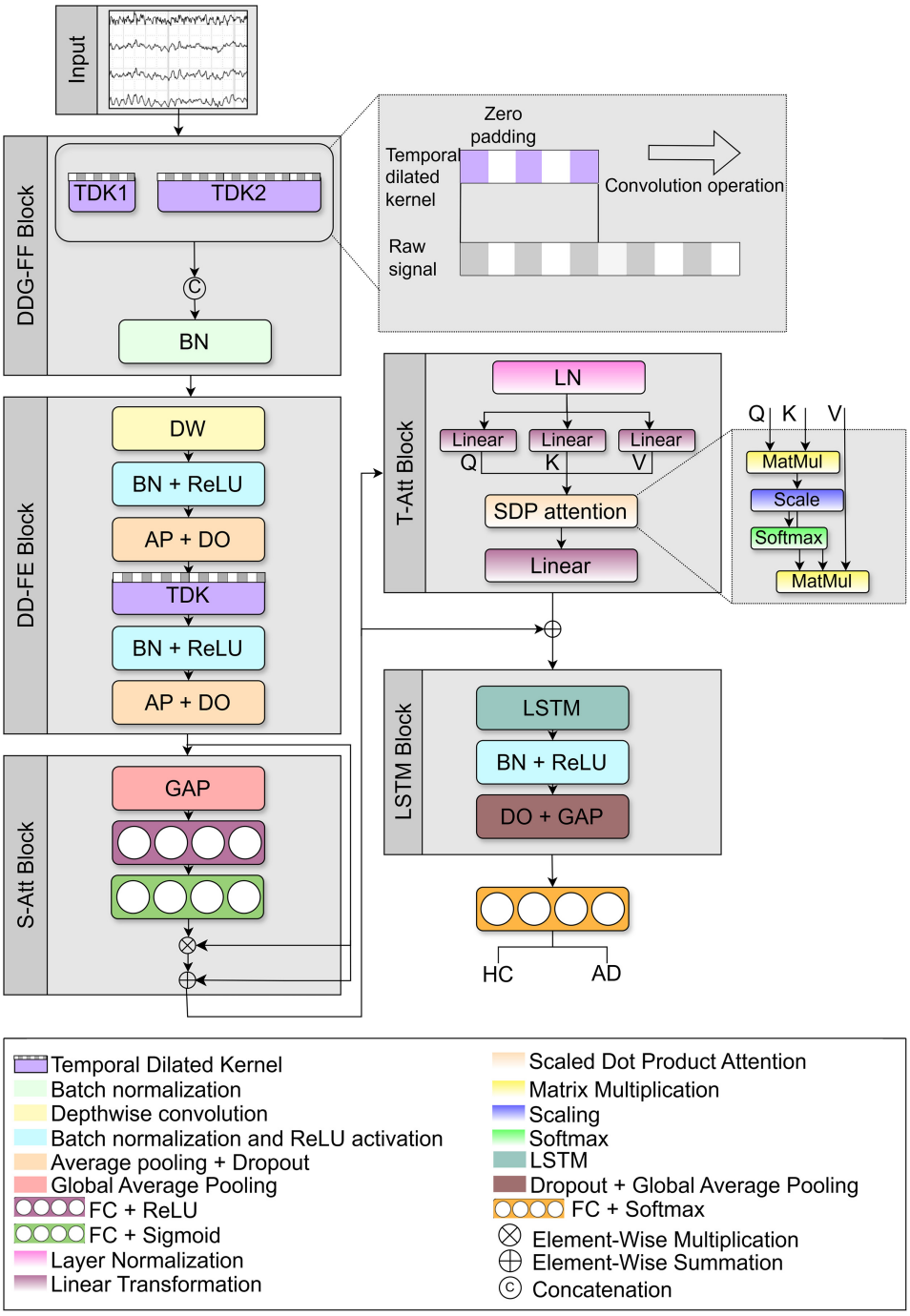


Fig. 2. The DHDL model architecture. DHDL, dilated hybrid deep learning; TDK, temporal dilated kernel; DDG-FF, dilated dual-granularity feature fusion block; BN, batch normalization; DW, depthwise convolution; ReLU, rectified linear unit; AP, average pooling; DO, dropout; S-Att, spatial attention block; T-Att, temporal attention block; GAP, global average pooling; LN, layer normalization; SDP, scaled dot product; LSTM, long short-term memory; HC, healthy controls; FC, fully connected layer; Q, query; K, key; V, value.

2.1.1 Dilated Dual-granularity Feature Fusion Block (DDG-FF)

The DDG-FF block extracts temporal features from raw EEG signals using two parallel dilated temporal convolution branches. Each branch kernel has a different size and dilation rate. The dilation mechanism inserts consecutive equal spaces between the kernel elements without increasing the number of parameters. The rationale for using the DDG-FF block is to learn richer temporal feature granularities while maintaining a low number of parameters in the current and subsequent layers. In this context, the receptive field (RF) is the number of consecutive input samples that a temporal convolutional kernel can capture during a convolution process. In the standard convolution, this is equal

Table 1. Specifications of the DHDL model.

Layer	#K & KS	Input shape	Output shape	#P
TDK 1	8 (1, 32)	19 × 1000 × 1	19 × 1000 × 8	256
TDK 2	8 (1, 63)	19 × 1000 × 1	19 × 1000 × 8	504
C	-	19 × 1000 × 8 + 8	19 × 1000 × 16	0
BN		19 × 1000 × 16	19 × 1000 × 16	64
DW	2 (19, 1)	19 × 1000 × 16	1 × 1000 × 32	608
BN	-	1 × 1000 × 32	1 × 1000 × 32	128
AP	(1, 4)	1 × 1000 × 32	1 × 250 × 32	0
TDK	16 (1, 8)	1 × 250 × 32	1 × 250 × 16	4096
BN	-	1 × 250 × 16	1 × 250 × 16	64
AP	(1, 4)	1 × 250 × 16	1 × 62 × 16	0
GAP	-	1 × 62 × 16	16	0
FC	-	16	2	34
FC	-	2	16	48
Reshape	-	16	1 × 1 × 16	0
Mult	-	1 × 62 × 16, 1 × 1 × 16	1 × 62 × 16	0
ADD	-	1 × 62 × 16	1 × 62 × 16	0
Reshape	-	1 × 62 × 16	62 × 16	0
LN	-	62 × 16	62 × 16	32
SHA	-	62 × 16	62 × 16	552
ADD	-	62 × 16	62 × 16	0
LSTM	-	62 × 16	62 × 32	6272
BN	-	62 × 32	62 × 32	128
GAP	-	62 × 32	32	0
FC	-	32	2	66
Total number of parameters				12,852
Number of trainable parameters				12,660

#K, number of kernels; KS, kernel size; #P, number of parameters; C, concatenation; AP, average pooling; Mult, element-wise multiplication; ADD, element-wise addition; SHA, single-head attention.

to the kernel size. In contrast, in dilated convolution, it is expressed as:

$$RF = (K - 1) \times DR + 1, \tag{1}$$

where K denotes the kernel size and DR represents the dilation rate, which controls the number of spaces between kernel elements to provide an independent expansion of the receptive field of each branch. The first branch employs eight kernels with a size of (1, 32) and a dilation rate of (1, 1), to yield a receptive field of 32 samples capable of capturing high-frequency rhythms. The second branch also has eight kernels with a size of (1, 63) and a dilation rate of (1, 2), resulting in a receptive field of 125 samples. This allows the model to capture low-frequency activity.

Formally, each branch in the DDG-FF block is denoted as $C_{t_i}(x)$ and applies eight convolutional kernels with the corresponding dilation rate d^i. The convolution is defined as:

$$C_{t_i}(x) = \sum_j w_k^{(i)}(j) \cdot x\left(c, t - d^i \cdot j\right), \tag{2}$$

where i ∈ {1, 2}, $w_k^{(i)}$ denotes the weights of the k^{th} filter, d^i is the dilation rate, and $x(c, t, 1)$ is the EEG signal at channel c and time t. The output feature maps of both branches are then fused using concatenation and normalized to form the following final unified features:

$$DDG\text{–}FF = (C_{t_1}(x) \parallel C_{t_2}(x)), \tag{3}$$

where || denotes concatenation. Here, $C_{t_1}(x)$ extracts short-range temporal features, whereas $C_{t_2}(x)$ captures long-range temporal features.

2.1.2 Dilated Deep Feature Extraction Block (DDFE)

The DDFE block learns deeper spatiotemporal features by combining two basic layers, namely, depthwise

convolution and dilated temporal convolution. A depthwise convolution layer is integrated to learn spatial features across EEG channels, whereas the dilated temporal convolution is introduced to learn deeper temporal features with a minimal number of parameters at a finer temporal scale. Our design ensures high feature representation with a limited parameter count. The block starts with a depthwise layer of size (1, 19), followed by batch normalization to stabilize training and rectified linear unit (ReLU) activation to introduce nonlinearity to the network. Next, an average pooling layer is applied to reduce the time dimension by a factor of 4, along with a dropout regularization layer to mitigate overfitting. The dropout rate is set to 30%, which randomly sets 30% of the activations to zero during training. The sequence of these three layers is inspired by the EEGNet model [23]. Following these layers, another dilated temporal layer is applied with 16 kernels, each of size (1, 8), and the dilation rate is empirically set to (1, 1). The output feature maps are then passed through another batch normalization, ReLU activation, an average pooling layer with a reduction ratio of 4, and a dropout regularization layer set to 30%. The operations of the DDFE block can be expressed as follows:

$$x' = DW(\text{Conv}(x)), \tag{4}$$

$$DDFE = DTConv\left(x'\right), \tag{5}$$

where $x \in R^{C\times T\times F_1}$ denotes the output from the DDG-FF block. Here, *DWConv* denotes the depthwise convolution across EEG channels, and *DTConv* denotes the dilated temporal convolution.

2.1.3 Spatial Attention Block (S-Att)

Different feature maps contribute unequally to AD detection. To tackle this issue, the S-Att block utilizes the Squeeze-and-Excitation (SE) attention mechanism [22]. This mechanism enables the model to selectively emphasize the significant feature maps that are more relevant to AD detection and to suppress less relevant ones. The goal of this block is to enhance the spatial feature representation. The output features from the DDFE block, denoted as $x \in R^{1\times \frac{T}{16}\times F_2}$, first pass through a global average pooling (GAP) to produce a compact descriptor per feature map. Subsequently, this descriptor passes through two fully connected layers. The first layer reduces the descriptor size by a factor of eight and applies ReLU activation to learn the relationships between the feature maps. The second layer restores the original descriptor size and applies a sigmoid activation to produce attention weights. This process is expressed as:

$$a = \sigma\left(W_2 \cdot \text{ReLU}(W_1 \cdot \text{GAP}(x))\right), \tag{6}$$

where W_1 and W_2 are the weights of the first and second fully connected layers, respectively, σ denotes the sigmoid activation, and a represents the attention weights. The attention weights are then multiplied by the original block input:

$$\widetilde{x} = x \cdot a. \tag{7}$$

To maintain model stability and improve generalization, a residual connection adds the original block input to the recalibrated feature map, yielding the final feature representation:

$$S\text{–}Att = \widetilde{x} + x. \tag{8}$$

2.1.4 Temporal Attention Block (T-Att)

The T-Att block enhances the temporal feature representation using a single-head attention (SHA) mechanism [24]. First, the output from the S-Att block is reshaped from (1 × T/16 × F_2) to (T/16 × F_2), denoted as $x \in R^{\frac{T}{16}\times F_2}$. The T-Att block then applies layer normalization (LN) to stabilize the feature representation. Using linear projections, the normalized features are then mapped into three distinct representations: query (Q), key (K), and value (V):

$$Q = LN(x)W_Q, \tag{9}$$

$$K = LN(x)W_K, \tag{10}$$

$$V = LN(x)W_V, \tag{11}$$

where W_Q, W_k, and $W_V \in R^{F_2\times d_k}$ are learnable weight matrices, and d_k is the dimension of the attention head. In this design, we use a single head with d_k = 8 to maintain computational efficiency on edge-computing platform. Subsequently, the scaled dot product attention is computed as:

$$a(Q,K,V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{12}$$

Here, the Softmax (·) computes the attention scores as a probability distribution over the temporal positions. The obtained attention scores are then multiplied by the value matrix to determine the significance of each temporal position. The resulting matrix is projected back to the original feature representation using the output projection matrix $W^O \in R^{d_k \times F_2}$:

$$SHA(Q, K, V) = a(Q, K, V)W^O. \quad (13)$$

We then use a residual connection to add the original block input to the output:

$$T\text{–}Att = x + SHA(Q, K, V). \quad (14)$$

2.1.5 LSTM Block

The final block utilizes an LSTM network. The LSTM network is particularly beneficial for EEG signals because it can record the impact of preceding time sequences on subsequent sequences within the signal. This capability enables the model to learn long-range temporal dependencies through its gating mechanisms [25]. These gating mechanisms preserve important past information while discarding less relevant content. For an input feature sequence $x \in R^{\frac{T}{16} \times F_2}$, the LSTM operation is defined as:

$$[h_t, c_t] = LSTM(x), \quad (15)$$

where h_t and c_t are the hidden and cell states that represent temporal dependencies. Following the LSTM network, a GAP layer computes the average value across the temporal dimension to summarize each feature map into a single representative value, thereby preparing the features for the final classification layer. Finally, a fully connected layer with Softmax activation is applied for classification to differentiate healthy controls (HC) from patients with AD.

2.2 Channel Selection Approach

We propose an effective channel selection approach to reduce the dimensionality of EEG data and eliminate redundant information. Our approach consists of two stages. In the first stage, we form an initial channel subset by iteratively identifying the best-performing channels based on the corresponding classification accuracy. Then, in the second stage, we optimize this initial subset through an enhanced hill-climbing optimization algorithm, which yields the final optimized channel subset. Hill-climbing optimization is a well-known local search algorithm that has demonstrated effectiveness in addressing optimization problems [20]. It begins with a random solution, which it then iteratively modifies to search for the optimal solution. The search process is repeated until the algorithm becomes stuck in a state in which no further improvements are possible. However, channel selection is a combinatorial problem, and the number of possible channel subsets is $2^{19} - 1 = 524{,}287$. Therefore, the typical hill-climbing optimization behavior is not practical for solving the channel selection problem. In contrast, our approach employs the initial channel subset obtained from the first stage as the initial solution for the enhanced hill-climbing optimization, which has a defined stopping condition. The rationale of applying this approach is to enable a more rapid convergence with the optimized channel subset.

We prevent channel selection bias by applying the channel selection approach exclusively to an independent set of 10 subjects (5 AD and 5 HC) who are not included in the model evaluation experiments in Section 3. Fig. 3 illustrates the flowchart for the proposed approach. In the first stage, the DHDL model is trained and evaluated using a single channel. The Leave-2-Subjects-Out (L2SO) protocol is utilized for evaluation to ensure generalization among all 10 subjects. The accuracy of the model using every single channel is stored. The channels are subsequently ranked in descending order based on the stored accuracies. This ranking is then used to build channel subsets by iteratively adding the next top-ranked channel to the channel subset. After each channel addition, the model is retrained, and its accuracy is evaluated. The first stage converges when the model achieves a predefined accepted accuracy threshold (A_acc). We set A_acc to reflect an acceptable marginal accuracy loss relative to the full 19-channel model accuracy, given that fewer channels are used. A_acc was set to approximately 2% below the full 19-channel baseline L2SO accuracy of the 10 subjects. This ~2% margin falls within the 95% CI of the full 19-channel baseline (±2.63% for CE and ±8.64% for OE). Hence, the accepted accuracy loss lies within the fold-to-fold variability of the full 19-channel model. This means the resulting channel subset from Stage 1 remains within the natural performance variation of the full 19-channel model rather than representing a harmful performance degradation. This margin allows a meaningful reduction in the number of channels while keeping the resulting accuracy loss minimal. For the CE dataset, A_acc was set to 89% relative to a baseline of 91.71 ± 2.63%. For the OE dataset, A_acc was set to 68% relative to a baseline of 70.85 ± 8.64%. At the end of Stage 1, the corresponding selected channels are considered to represent the initial channel subset. This subset is then used as the initial solution for the enhanced hill-climbing optimization.

In this sense, Algorithm 1 describes the enhanced hill-climbing optimization algorithm. The inputs to the algorithm are the complete 19 channels, denoted as $\mathcal{C}$, the initial channel subset, denoted as S_0, and the maximum number of iterations, denoted as max_iter. The value of max_iter was set to 10 to allow sufficient channel exploration while preventing excessive computational cost caused by repeated model retraining and re-evaluation. Specifically, in each it-

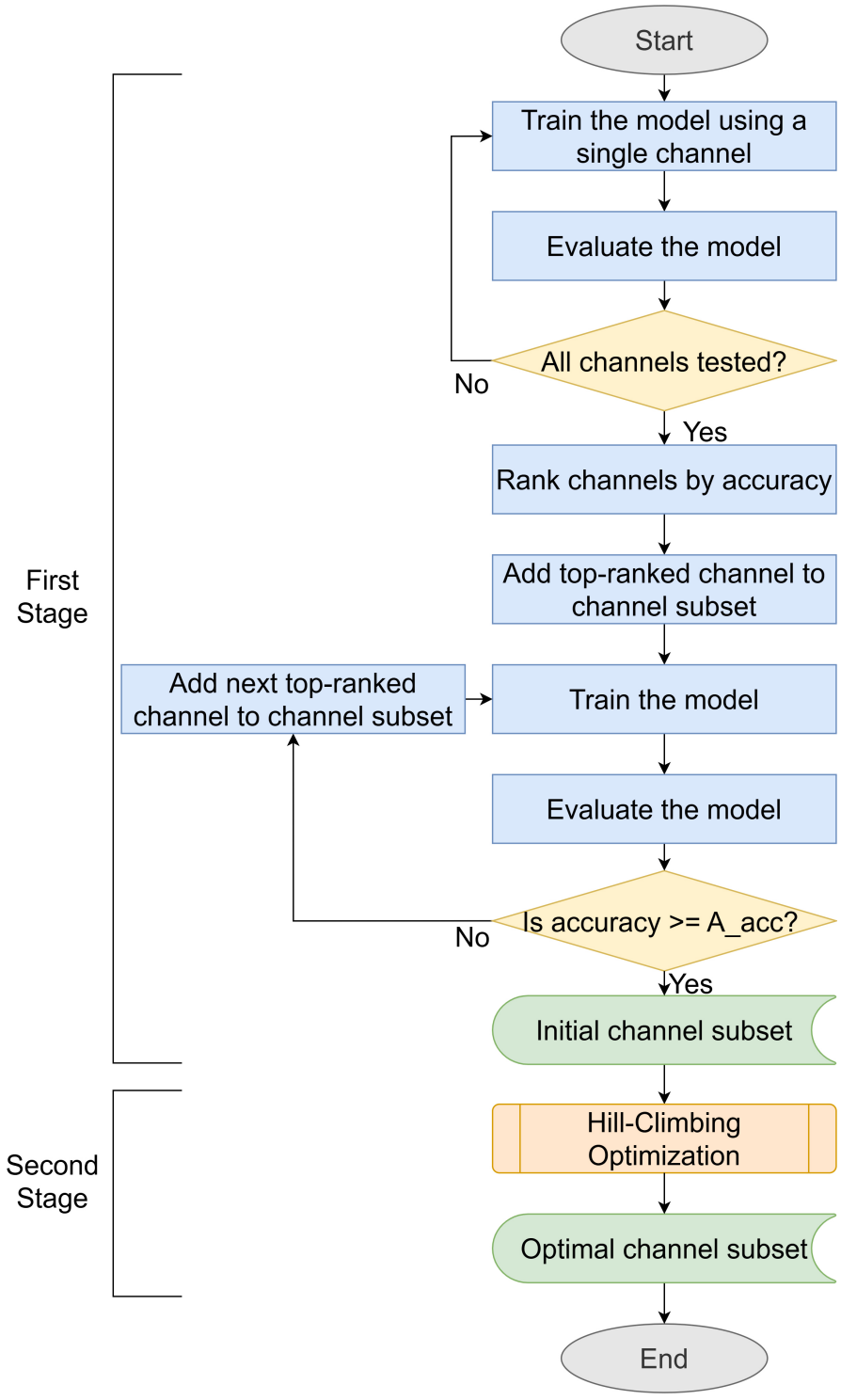


Fig. 3. Flow chart of the proposed channel selection approach.

eration, the model is trained and evaluated across all L2SO folds. Since 10 subjects were used for channel selection, the L2SO procedure produced 5 folds. Ten iterations therefore require 50 training-and-evaluation cycles per dataset, each running up to 100 epochs. This resulted in 10,000 training epochs across both datasets. This repeated evaluation requires considerable computational resources; hence, setting max_iter to 10 provides a practical balance between channel exploration and computational efficiency. Similar to the first stage, we utilize the model accuracy via L2SO as an objective function. The first step consists of initializing three main parameters: current channel subset (S), current accuracy (Acc_{curr}), and an iteration counter (iter) set to zero. The algorithm then iteratively optimizes the channel subset by replacing one channel at a time. Specifically, in each iteration, one channel from the current channel subset (c_{out})

Algorithm 1: Hill-Climbing Optimization.

Inputs:

$\mathcal{C}$: complete set of 19 channels

$S_0 \subset C$: initial channel subset

max_iter: maximum number of iterations

Outputs:

Optimized channel subset S^* and its corresponding accuracy Acc^*

Initialize:

$S \leftarrow S_0$

$Acc_{curr} \leftarrow Acc(S)$ // baseline L2SO accuracy

$iter \leftarrow 0$

Compute:

While ($iter <$ max _iter) do:

  Randomly select $c_{out} \in S$ and $c_{in} \in (\mathcal{C} - S)$

  Form a new subset: $S' = (S - c_{out}) \cup c_{in}$

  Train and evaluate DHDL model using S': $Acc' \leftarrow Acc(S')$

  If $Acc' > Acc_{curr}$:

    $S \leftarrow S'$

    $Acc_{curr} \leftarrow Acc'$

    End

  iter ← iter + 1

  End

Return:

$S^* = S$, $Acc^* = Acc_{curr}$

Acc_{curr}, current accuracy.

is randomly replaced with another channel that is not included in the subset (c_{in}) to form a new candidate channel subset (S′). The model is then retrained and evaluated with S′ subset. If the new candidate subset yields a higher accuracy (Acc(S′) > Acc_{curr}), the replacement is accepted and the current channel subset is updated. Otherwise, the replacement is rejected, the current subset remains unchanged, and the algorithm proceeds to the next iteration. This procedure continues until the iteration counter reaches the max_iter value. The final outputs of the algorithm are the optimized channel subset (S*) and its corresponding accuracy (Acc*).

2.3 Datasets and Preprocessing

We trained and evaluated the proposed model using two publicly available benchmark EEG datasets: the CE and OE datasets [15,21]. Both datasets have a sampling frequency of 500 Hz and were obtained from the same group of 65 subjects (29 HC and 36 AD). The CE dataset was recorded under a resting-state closed-eye protocol, whereas the OE dataset was recorded under an open-eye protocol following photic stimulation. The stimulation consisted of visual light delivered at frequencies of 5 Hz up to 30 Hz in 5 Hz increments. However, the stimulation frequencies were not consistent across all subjects. Upon our inspection, the majority of subjects had recordings at 5, 10, 15, and 20 Hz. For model development and evaluation, only subjects with this common set of frequencies were retained. In addition, an equal number of subjects per class was required to strictly apply the L2SO cross-validation protocol, detailed in Section 3.2, yielding 48 subjects (24 AD and 24 HC). Another independent holdout set of 10 subjects (5 AD and 5 HC) was used exclusively for channel selection. For channel selection, all available visual frequencies were used, which varied from 5 up to 30 Hz. The remaining seven subjects (all AD) were not included to strictly apply the L2SO cross-validation protocol. In Table 2, the subjects clinical characteristics are listed including age, sex, Mini-Mental State Examination (MMSE) score and comorbidities. The mean ages of the AD and HC groups were 66.67 ± 8.78 and 68.13 ± 5.68 years, respectively. The sex distributions for both the AD and HC groups were not balanced (AD: 67% female; HC: 38% female). The MMSE score was used to measure cognitive function. The score was 16.88 ± 4.89 for AD subjects and 30 ± 0.0 for HC subjects. There were no dementia-related comorbidities for the AD subjects.

For the CE dataset, 9 minutes of EEG recording were included for each subject in the 48-subject group to maintain balanced data duration across subjects from distinct classes. The signals were segmented into two-second segments without overlapping. Thus, the total number of segments per subject was 270. They were then standardized per channel using the RobustScaler because it is less sensitive to outliers than z-scoring standardization [26]. Instead of centering the data around the mean and scaling by standard deviation, Robust scaling subtracts the median and

Table 2. Clinical characteristics of subjects in the CE and OE datasets.

Clinical characteristics	AD	HC
Age	66.67 ± 8.78	68.13 ± 5.68
Sex	16 F / 8 M	9 F / 15 M
MMSE score	16.88 ± 4.89	30.0 ± 0.0
Comorbidities	No dementia-related comorbidities	Not reported

± SD is provided for age and MMSE score. F, female; M, male; CE, closed-eye; OE, open-eye; AD, Alzheimer's disease; MMSE, mini-mental state examination.

scales by the interquartile range (IQR). The transformation is given by:

$$x' = \frac{x - Q_2}{Q_3 - Q_1}, \tag{16}$$

where x is the original signal, Q_2 is the median, Q_1 and Q_3 are the 25th and 75th percentiles, respectively, and $Q_3 - Q_1$ represents the IQR.

The use of the OE dataset was challenging, as the cognitive impairment of many subjects prevented them from following the recording instructions [15]. Common violations included eye closure, head movement, swallowing, muscle movement, and speech during the recording sessions. All of these suspected events were annotated by the dataset authors. Several preprocessing steps were applied to improve the signal quality. First, all suspected events were removed using an empirically defined buffer of 0.5 seconds before and 2.5 seconds after each event onset. The cleaned signals were then concatenated per subject and segmented into two-second segments using a sliding window with 70% overlap. Each segment was subsequently standardized per channel using Eqn. 13. To address the class imbalance and increase the segment diversity, data augmentation was applied through oversampling with random Gaussian noise addition, followed by a 50% time shift per segment. To avoid data leakage between the training and the test set, the augmentation was applied only to training data per fold.

2.4 Training Environment

The DHDL model was implemented in Python (version 3.11, Python Software Foundation, Wilmington, DE, USA) using the TensorFlow framework (version 2.19, Google LLC, Mountain View, CA, USA). The model was trained on the Google Colab (Google LLC, Mountain View) platform with access to an NVIDIA T4 GPU (NVIDIA Corporation, Santa Clara, CA, USA) and 15 GB of high-memory runtime. To guarantee reproducibility, all random seeds were set to 42. The model utilized the Adam optimizer (TensorFlow Keras API version 2.19, Google LLC), with a learning rate of 0.0001. Batch sizes were set to 256 and 128 for the CE and OE datasets, respectively, over a maximum of 100 epochs. The model was trained using categorical cross-entropy loss as a loss function, and early stopping with patience = 10 was used to avoid overfitting. We evaluated the model's performance in a subject-independent manner by applying an L2SO cross-validation protocol using a previously detailed approach [27]. In each fold, all segments from one AD subject and one HC subject were held out as the test set. All segments from another separate AD–HC pair were used for the validation set. The remaining subjects formed the training set. There was no overlap of subjects between the training, validation, and test sets. The procedure was repeated across all folds, while ensuring that each subject appeared in the test set exactly once. For on-edge deployment, the model was compressed using post-training float16 quantization [28].

2.5 Edge Environment

Model inference was conducted on the Raspberry Pi 5 (8 GB RAM model, Raspberry Pi Ltd, Cambridge, UK) for both the CE and OE datasets. This step is essential for validating the performance of the DHDL model and assessing the effect of the proposed channel selection approach on edge-computing platforms. The device features the Broadcom BCM2712 quad-core Arm Cortex-A76 CPU (Broadcom Inc., Palo Alto, CA, USA) operating at 2.4 GHz, 8 GB of LPDDR5-7500 SDRAM, and an external 64 GB microSD card (SanDisk ultra microSDXC, SanDisk Corporation, Milpitas, CA, USA). The CPU governor was set to "ondemand" during all experiments. In addition, the device requires 5V/5A DC power supply via USB-C and 64-bit operating system (Debian version 13.2, Raspberry Pi Ltd). The on-edge metrics were calculated across all folds, and the average results were reported, together with the standard deviation. The CPU temperature was monitored during experiments using the built-in 'vcgencmd' utility. No thermal throttling was observed, and the CPU temperature did not exceed 37.2 °C during any of the experiments. The model inference was performed using Python (version 3.11, Python Software Foundation) with the TensorFlow Lite (version 2.15, Google LLC) framework, using a single thread. The TensorFlow Lite CPU runtime enables the XNNPACK delegate by default, which in turn uses ARM (version 8.2, Arm Ltd, Cambridge, UK) Advanced Single Instruction Multiple Data (ASIMD) vector instructions automatically, with no manual configuration required. The

quantized model size was 215 KB. Inference was performed with a batch size of 1, consistent with real-time edge deployment. All experiments were conducted using Visual Studio Code (version 1.106.2, Microsoft Corporation, Redmond, WA, USA) as the development environment. Energy consumption during inference was calculated based on the mean power, which was derived from real-time current and voltage measurements recorded using a TC66C digital power meter (Hangzhou Ruideng Technology Co. Ltd., Hangzhou, Zhejiang, China).

2.6 Performance Metrics

2.6.1 Classification Performance

The classification performance, including accuracy, precision, recall, and F1-score, are defined in terms of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), as follows:

- Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{17}$$

- Precision:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{18}$$

- Recall:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{19}$$

- F1-score:

$$F1\text{–}score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{20}$$

We also reported the test-set inference time (i.e., the time required to classify all segments in the test set per fold), the number of trainable parameters, and floating-point operations (FLOPs) to assess the model complexity.

2.6.2 On-Edge Performance

The practicality of the DHDL model with the proposed channel selection approach was evaluated on the edge-computing platform using four indicators: classification performance (assessed as described in 2.6.1), energy consumption, resource utilization, and inference time per segment.

- Energy Consumption: Energy consumption was computed based on the recorded voltage and current measurements. The power was first calculated as:

$$P = V \times C, \tag{21}$$

where V and C are the voltage and current values recorded during model inference. The dynamic energy consumed during model inference was then derived as:

$$E = (P_{\text{inf}} - P_{\text{idle}}) \times T, \tag{22}$$

where P_{inf} and P_{idle} are the power during model inference and the idle state, respectively, and T denotes the inference time. The measured idle power was 1.59 W.

Assuming the device is under duty-cycle mode, the edge-computing platform performs one inference approximately every two seconds (the segment duration) and stays idle to the next cycle. The battery life was estimated based on the duty-cycle power, which accounts for both the inference period and the idle period between consecutive inferences. The energy consumed during one operating cycle was estimated as

$$E_{cycle} = P_{\text{inf}} \times T_{\text{inf}} + P_{\text{idle}}(T_{cycle} - T_{\text{inf}}), \tag{23}$$

where T_{cycle} = 2 s is the interval between two consecutive EEG segments. The corresponding duty-cycled power was calculated as:

$$P_{cycle} = \frac{E_{\text{cycle}}}{T_{cycle}}. \tag{24}$$

The expected duty-cycle battery life was then estimated, assuming a nominal battery voltage of 3.7 V, as [29]:

$$E_{\text{battery}}\,(\text{Wh}) = \frac{\text{Capacity}_{\text{battery}}(\text{mAh}) \times \text{V}}{1000}, \tag{25}$$

$$\text{Battery life (hours)} = \frac{E_{\text{battery}}\,(\text{Wh})}{P_{cycle}\,(\text{W})}, \tag{26}$$

where $E_{battery}$ denotes the available battery energy in watt-hours and P_{cycle} represents the power consumption during the duty-cycle, calculated using Eqn. 21. The number of segments that can be classified per single full charge was estimated as:

$$\text{Number of segments} = \frac{\text{Battery life (hours)} \times 3600}{2}. \tag{27}$$

● Resource Utilization and Inference Time: Resource utilization was assessed using CPU utilization, random access memory (RAM) utilization, and inference time. The inference time was reported at three levels: the per-segment inference time, the per-subject inference time and the test-set inference time. The per-segment inference time refers to the time required to classify a single EEG segment, in milliseconds. The per-subject inference time refers to the time estimated to classify 9 minutes recording for a single subject, in seconds, calculated as:

$$\text{Per} - \text{subject inference time} = \frac{\text{per-segment inference time (ms)} \times 270}{1000}, \quad (28)$$

where 270 is the number of segments in a 9-minute recording. The test-set inference time refers to the time required to classify all the test segments per fold.

3. Results

3.1 Classification Performance

We applied several ablation experiments to assess the significance of the main blocks and design choices in the DHDL architecture. To ensure a fair comparison, all the EEG channels were involved in the ablation experiments. Individual components were systematically removed or modified, and the resulting impact on the overall model performance was assessed. Fig. 4 illustrates the performance of three model variants against the complete model. Removing the S-Att block caused the classification accuracy to decline to 87.24% for the CE dataset and 73.09% for the OE dataset, demonstrating the effectiveness of the S-Att block in enhancing spatial feature representation through feature map reweighting. Omitting the T-Att block reduced the accuracy to 89.09% for the CE dataset and to 68.30% on the OE dataset, underscoring the role of the T-Att block in selectively emphasizing temporally salient regions. Excluding the LSTM block reduced the accuracy to 88.22% for the CE dataset and to 63.83% for the OE dataset, confirming the importance of the LSTM block in capturing long-range temporal dependencies. In contrast, the complete model achieved an accuracy of 92.49%, with an F1-score of 93.02%, for the CE dataset, and an accuracy of 74.39%, with an F1-score of 75.69%, for the OE dataset. Taken together, these results confirm the contribution of each architectural component to the overall model performance.

We evaluated the impact of the design choices in the DDG-FF block by examining the effect of tuning two components: the number of branches and the dilation rates. We reported the model performance while varying the number of branches from a single branch to five parallel branches, as shown in Table 3. As expected, the addition of more branches causes a linear increase in the number of parameters. Furthermore, employing two branches significantly enhances the accuracy from 87.89% to 92.49% (an increase of 4.60 percentage points) for the CE dataset. A consistent trend was observed for the OE dataset, where two branches again yielded the best performance, with accuracy increasing from 66.95% to 74.39%. This accuracy gain is due to the richer temporal features captured by the two branches than by a single branch. Notably, the addition of more than two branches did not enhance the classification performance. Instead, it elevated the computational costs in terms of trainable parameters. These results suggest that the dual-branch architecture strikes a good balance between model complexity and accuracy.

We also used five different configurations of dilation rates to assess the effectiveness of the dilation mechanism. When evaluated using both the CE and OE datasets, the optimal accuracy was obtained when the dilation rate was set to 1 in the first branch and 2 in the second branch, as shown in Table 4. Overall, the results demonstrate that choosing two branches with appropriate kernel sizes and dilation rates allowed the model to capture both fine-grained and coarse-grained temporal features.

Table 5 summarizes the performance obtained after applying the proposed channel selection approach. Fourteen channels were chosen from the CE dataset (Fp2, F4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, and Cz), while 10 channels were chosen for the OE dataset (Fp1, P3, O1, O2, F7, F8, T4, T5, T6, and Pz). The difference in the number of selected channels between the two conditions reflects the distinct neural dynamics of each recording condition. In the CE condition, the discriminative AD features span across wider brain regions [30]. Therefore, a larger subset of 14 channels is required to discriminate AD from HC. Under the OE condition, visual processing concentrates discriminative features within a more spatially localized set of 10 channels. These findings confirm that the optimal channel subset depends on the recording condition. In both datasets, as expected, the optimized channel subset preserves the accuracy while reducing model complexity when compared to the original 19-channel.

Fig. 5 presents the confusion matrix for the DHDL model when evaluated using the CE and OE datasets and the optimized set of channels. For the CE dataset, the optimized 14-channel model achieves high performance (Fig. 5A). The diagonal shows the sensitivity of the HC and AD classes. The model correctly identified 91.90% of the HC segments and 93.36% of the AD segments. These results indicate a strong discriminative capability between the two classes. For the OE dataset, when using the optimized 10-channel model, the performance was relatively lower than for the CE dataset (Fig. 5B). As the HC sensitivity dropped to 70.19% and the AD sensitivity decreased to 80.38%. Notably, the performance drop is attributed to the noisier nature of the OE recordings. The preserved diagonal dominance in both confusion matrices confirms that most of the

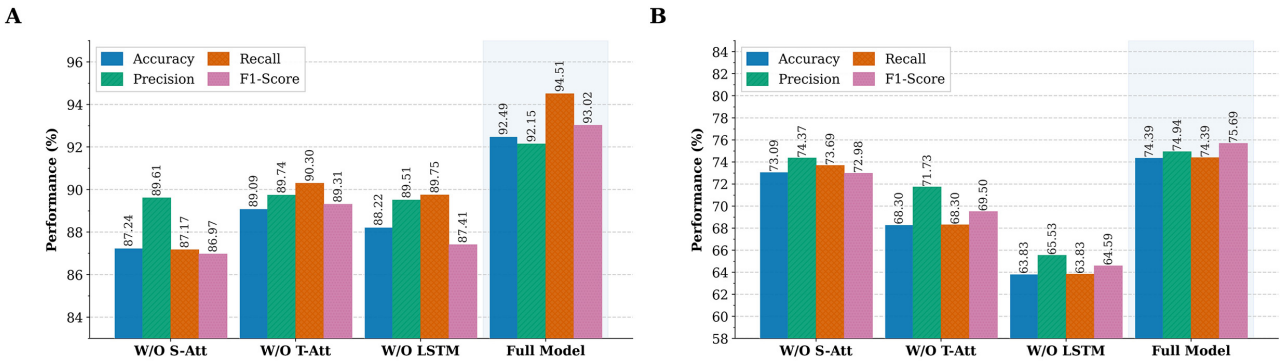


Fig. 4. Comparative ablation results. (A) CE dataset. (B) OE dataset. W/O, without.

Table 3. The effect of number of branches in the DDG-FF block.

Number of branches	CE dataset			OE dataset		
	Accuracy (%) ± 95% CI	F1-score (%) ± 95% CI	#trainable parameters	Accuracy (%) ± 95% CI	F1-score (%) ± 95% CI	#trainable parameters
1	87.89 ± 3.38	90.09 ± 3.01	9956	66.95 ± 8.47	69.00 ± 7.46	9956
2	**92.49 ± 2.90**	**93.02 ± 2.46**	**12,660**	**74.39 ± 7.40**	**75.69 ± 6.86**	**12,660**
3	90.39 ± 2.76	90.11 ± 2.81	15,364	71.63 ± 9.02	72.54 ± 8.76	15,364
4	90.02 ± 4.28	89.96 ± 3.54	18,316	69.20 ± 10.63	70.05 ± 10.62	18,316
5	89.21 ± 3.29	89.21 ± 3.39	21,020	70.05 ± 9.27	69.85 ± 10.13	21,020

Bold values indicate the optimal configuration. #, number of.

Table 4. The effect of dilation rate in the DDG-FF block.

Dilation rates	CE dataset		OE dataset	
	Accuracy (%) ± 95% CI	F1-score (%) ± 95% CI	Accuracy (%) ± 95% CI	F1-score (%) ± 95% CI
D1 = 1, D2 = 3	90.46 ± 3.22	90.49 ± 3.18	66.95 ± 8.63	68.18 ± 8.11
D1 = 2, D2 = 2	89.86 ± 3.17	90.56 ± 2.77	73.72 ± 8.03	74.98 ± 7.57
D1 = 4, D2 = 1	90.71 ± 3.23	90.83 ± 3.13	73.91 ± 7.64	75.55 ± 6.88
D1 = 3, D2 = 4	90.22 ± 2.97	90.60 ± 3.08	72.26 ± 9.16	74.00 ± 8.35
D1 = 1, D2 = 2	**92.49 ± 2.90**	**93.02 ± 2.46**	**74.39 ± 7.40**	**75.69 ± 6.86**

D1: Dilation rate for the first branch, D2: Dilation rate for the second branch. Bold values indicate the optimal configuration.

Table 5. Classification performance and computational cost of DHDL model across channel configurations.

Channel configuration	Accuracy (%) ± 95% CI	F1-score (%) ± 95% CI	#trainable parameters	FLOPs (M)
CE dataset				
19 channels	92.49 ± 2.90	93.02 ± 2.46	12,660	33.09
Optimized 14-channel	**92.67 ± 2.85**	**92.96 ± 2.58**	**12,500**	**25**
OE dataset				
19 channels	74.39 ± 7.40	75.69 ± 6.86	12,660	33.09
Optimized 10-channel	**75.27 ± 7.91**	**76.41 ± 7.60**	**12,372**	**18.54**

Bold values indicate the optimal configuration.

HC and AD segments were classified correctly. Combining the two results demonstrates that the proposed channel selection approach preserved the model's discriminative ability between the HC and AD classes even under the challenging OE conditions.

We examined the stability of the proposed channel selection approach by running the hill-climbing algorithm (Stage 2) five times with different random seeds (2, 123, 456, 789, and 2024) for both the CE and OE datasets. Table 6 reports the overlap rate between the final optimized subsets from these five runs and the final optimized subset

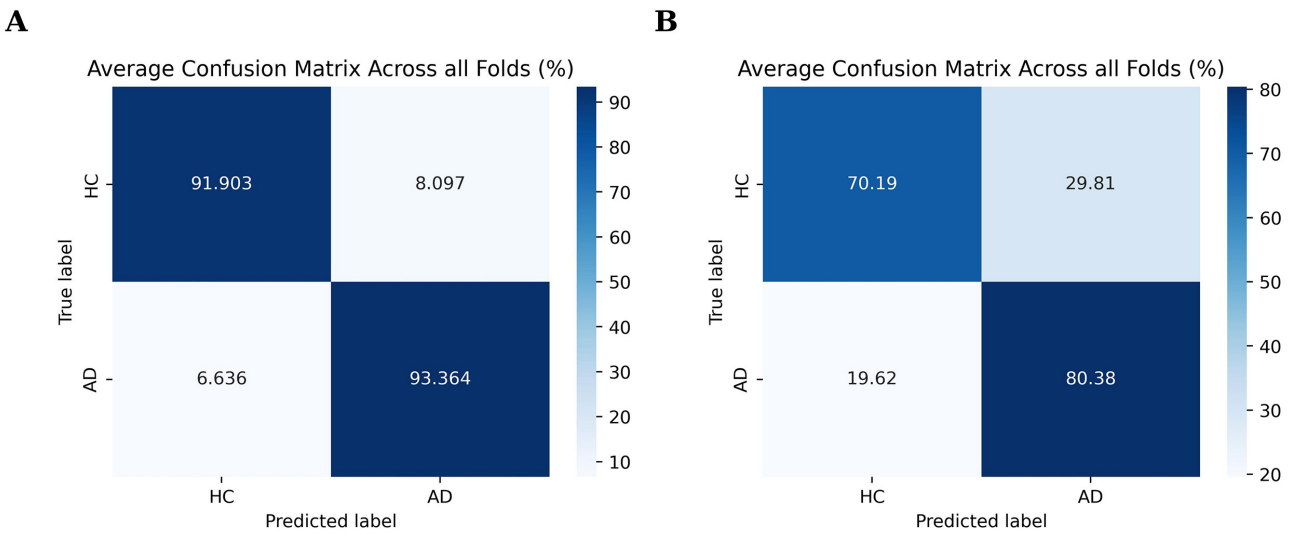


Fig. 5. Average confusion matrices across all folds. (A) Confusion matrix for the DHDL model using the optimized 14-channel subset (CE dataset). (B) Confusion matrix for the DHDL model using the optimized 10-channel subset (OE dataset).

used in all experiments, with seed 42. For the CE dataset, the overlap rates range from 85.71% (12/14 channels in common) to 92.86% (13/14 channels in common), with a mean overlap rate of 89.05% ± 3.69% across all pairwise comparisons. For the OE dataset, the overlap rate is either 80% (8/10 channels in common) or 90% (9/10 channels in common), with a mean overlap rate of 83.33 ± 4.88%. In the worst case, for both the CE and OE datasets, only two channels differ. The high overlap rate results show that the algorithm terminates with the same core channels and with minor differences across different runs, demonstrating the stability of the proposed channel selection approach. This stability is due to the algorithm starting from a non-random subset generated by Stage 1, which is guided by L2SO cross-validation accuracy.

3.2 Energy Consumption

Table 7 details the energy consumption of the DHDL model when configured with the original and optimized channel configurations for the CE and OE datasets. For the CE dataset, the average power consumption during inference was 2.64 W for the 19-channel configuration and 2.52 W for the 14-channel configuration (a reduction of 4.55%). The OE results follow the same trend, as the optimized 10-channel model reduced the mean power consumption by 6.54% (from 2.60 W to 2.43 W). This reduction in mean power is attributed to the decreased architectural complexity in terms of both the number of trainable parameters and FLOPs. When combined with the shorter inference time achieved by the optimized configurations, this leads to a more substantial reduction in dynamic energy consumption. For the CE dataset, the energy decreased from 14.85 mJ per segment in the 19-channel model to 9.69 mJ per segment in the 14-channel model. Similarly, for the OE dataset, the dynamic energy consumption decreased from 14.21 mJ per segment in the 19-channel model to 6.95 mJ per segment in the optimized 10-channel model.

The battery lifetime was also estimated for two typical power-bank capacities at 10,000 mAh and 20,000 mAh, corresponding to approximately 37 Wh and 74 Wh, respectively. As expected, for both the CE and OE datasets, the estimated battery life is slightly longer for the optimized channel subsets than for the 19-channel model, with the values up to 46.40 h for the CE dataset and 46.44 h for the OE dataset. Moreover, the number of segments classified per full charge shows an enhancement for both the CE and OE datasets. The optimized 14-channel model classifies ~41,759 and ~83,519 segments per full charge on 10,000 mAh and 20,000 mAh batteries, compared to ~41,692 and ~83,384 for the 19-channel model. For the OE dataset, the 10-channel model reaches ~41,795 and ~83,590 segments versus ~41,700 and ~83,400 for the 19-channel model. These results demonstrate that the DHDL model has low energy consumption. The consistent energy consumption results across different datasets support the suitability of the proposed framework for edge deployment. Furthermore, the proposed channel selection approach enhances the framework's energy efficiency and supports a longer operating time.

3.3 Resource Utilization and Inference Time

Table 8 shows the resource utilization and inference time results for the proposed framework when evaluated for both the CE dataset and the OE dataset. As expected, the models trained on the optimized channel subsets outperform the 19-channel models for both the CE and OE datasets in all metrics. In depth, for the CE dataset, the 14-channel model reduces RAM usage from 211.36 MB to

Table 6. Overlap rate between the final optimal channel subsets across five randomized runs, and against the Table 5 subset (final), for the CE and OE datasets.

Run	Selected channels	Compared runs	Common channels count	Overlap rates
		CE dataset		
1	Fp1, Fp2, F3, F4, P3, P4, O1, O2, F8, T3, T4, T5, T6, Fz	Run 1 vs Run 2	12/14	85.71%
		Run 1 vs Run 3	12/14	85.71%
		Run 1 vs Run 4	12/14	85.71%
		Run 1 vs Run 5	13/14	92.86%
		Run 1 vs Final	12/14	85.71%
2	Fp2, F3, F4, P3, P4, O1, O2, F7, F8, T3, T5, T6, Fz, Pz	Run 2 vs Run 3	13/14	92.86%
		Run 2 vs Run 4	12/14	85.71%
		Run 2 vs Run 5	13/14	92.86%
		Run 2 vs Final	12/14	85.71%
3	F3, F4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Pz	Run 3 vs Run 4	12/14	85.71%
		Run 3 vs Run 5	13/14	92.86%
		Run 3 vs Final	12/14	85.71%
4	Fp2, F3, F4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Cz	Run 4 vs Run 5	13/14	92.86%
		Run 4 vs Final	13/14	92.86%
5	Fp2, F3, F4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz	Run 5 vs Final	13/14	92.86%
Final	Fp2, F4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz	—	—	—
		OE dataset		
1	Fp1, F3, O1, O2, F7, F8, T3, T4, T5, T6	Run 1 vs Run 2	8/10	80.00%
		Run 1 vs Run 3	9/10	90.00%
		Run 1 vs Run 4	8/10	80.00%
		Run 1 vs Run 5	8/10	80.00%
		Run 1 vs Final	8/10	80.00%
2	Fp1, F3, P3, O1, O2, F7, F8, T5, T6, Cz	Run 2 vs Run 3	9/10	90.00%
		Run 2 vs Run 4	8/10	80.00%
		Run 2 vs Run 5	8/10	80.00%
		Run 2 vs Final	8/10	80.00%
3	Fp1, F3, P3, O1, O2, F7, F8, T4, T5, T6	Run 3 vs Run 4	9/10	90.00%
		Run 3 vs Run 5	9/10	90.00%
		Run 3 vs Final	9/10	90.00%
4	Fp1, F3, F4, P3, O1, O2, F7, T4, T5, T6	Run 4 vs Run 5	8/10	80.00%
		Run 4 vs Final	8/10	80.00%
5	Fp1, F3, C3, P3, O1, O2, F7, F8, T4, T6	Run 5 vs Final	8/10	80.00%
Final	Fp1, P3, O1, O2, F7, F8, T4, T5, T6, Pz	—	—	—

199.64 MB, CPU utilization from 25.01% to 24.96%, inference time per segment from 14.11 ms to 10.45 ms, and inference time per subject from 3.81 s to 2.82 s. Similar improvements are also observed for the OE dataset. The 10-channel model reduces RAM usage from 211.37 MB to 190.63 MB, CPU utilization from 25.04% to 24.91%, inference time per segment from 14.01 ms to 8.23 ms, and inference time per subject from 3.78 s to 2.22 s. Taken together, all the model configurations for both datasets maintain low CPU and RAM utilization. This setup maintains the availability of approximately 75% of the CPU capacity and consumes less than 3% of the 8 GB RAM.

Figs. 6,7 show the real-time resource utilization and inference time per segment for the DHDL model on the CE and OE datasets. Both figures clearly show that the model variants maintain low and stable CPU utilization of approximately 25% across all segments. The observed fluctuations can be associated with background operating system activities. Similarly, the RAM usage remains stable across all processed segments for both models. The inference time also remains consistent across all classified segments. As expected, reducing the number of EEG channels clearly reduces the RAM consumption and effectively accelerates the inference time per segment for both the CE and OE datasets. Overall, this result consistency indicates a reliable real-time behavior on the edge-computing platform. This feature is particularly essential for AD monitoring, where sustained real-time inference is necessary. It also verifies the decrease

Table 7. Energy consumption of the DHDL model.

CE dataset		
Metric	19-channel model	Optimized 14-channel model
Mean power	2.64 ± 0.02 W	**2.52 ± 0.03 W**
Dynamic power	1.05 ± 0.02 W	**0.93 ± 0.03 W**
Dynamic energy consumption	14.85 ± 0.30 mJ/segment	**9.69 ± 0.29 mJ/segment**
Battery life estimation (10,000 mAh)	~23.16 ± 0.002 h	**~23.20 ± 0.002 h**
Battery life estimation (20,000 mAh)	~46.32 ± 0.004 h	**~46.40 ± 0.004 h**
#segments classified per full charge (10,000 mAh)	~41,692 ± 3.85 segments	**~41,759 ± 3.79 segments**
#segments classified per full charge (20,000 mAh)	~83,384 ± 7.70 segments	**~83,519 ± 7.60 segments**
OE dataset		
Metric	19-channel model	Optimized 10-channel model
Mean power	2.60 ± 0.02 W	**2.43 ± 0.04 W**
Dynamic power	1.01 ± 0.02 W	**0.84 ± 0.04 W**
Dynamic energy consumption	14.21 ± 0.32 mJ/segment	**6.95 ± 0.30 mJ/segment**
Battery life estimation (10,000 mAh)	~23.17 ± 0.002 h	**~23.22 ± 0.002 h**
Battery life estimation (20,000 mAh)	~46.33 ± 0.005 h	**~46.44 ± 0.004 h**
#segments classified per full charge (10,000 mAh)	~41,700 ± 4.21 segments	**~41,795 ± 3.91 segments**
#segments classified per full charge (20,000 mAh)	~83,400 ± 8.41 segments	**~83,590 ± 7.81 segments**

Bold values indicate the optimal results. Standard deviation is reported for mean results across all folds. #, number of.

Table 8. Resource utilization and inference time results of the DHDL model.

CE dataset		
Metric	19-channel model	Optimized 14-channel model
RAM usage	211.36 ± 0.02 MB	**199.64 ± 0.03 MB**
CPU utilization	25.01 ± 0.37%	**24.96 ± 0.42%**
Per-segment inference time	14.11 ± 0.02 ms	**10.45 ± 0.02 ms**
Per-subject inference time	3.81 ± 0.01 s	**2.82 ± 0.01 s**
OE dataset		
Metric	19-channel model	Optimized 10-channel model
RAM usage	211.37 ± 0.01 MB	**190.63 ± 0.08 MB**
CPU utilization	25.04 ± 0.47%	**24.91 ± 0.44%**
Per-segment inference time	14.01 ± 0.04 ms	**8.23 ± 0.03 ms**
Per-subject inference time	3.78 ± 0.01 s	**2.22 ± 0.01 s**

Bold values indicate the optimal results. Standard deviation is reported for mean results across all folds. RAM, random access memory.

in computational complexity and in the number of operations needed during the forward pass resulting from the reduction in the number of channels. These findings demonstrate the suitability of the proposed framework for enabling real-time AD detection on edge-computing platforms.

4. Discussion

4.1 Comparison With State-of-the-Art Models

Table 9 (Ref. [10,11,12,13,14,23,31,32]) presents a comparison of the DHDL model with state-of-the-art models using the CE and OE datasets. To ensure a fair evaluation, we reproduced the results of four baseline models (EEGNet [23], ShallowNet [31], DeepConvNet [31], and lightweight convolution neural network for AD detection (LCADNet) [11]). For the remaining models, we included the results as documented in their original publications to show a relative comparison rather than head-to-head results.

Our quantitative results confirm that the DHDL model achieves higher or comparable performance across all configurations on both datasets. When evaluated using the CE dataset, the 19-channel configuration yields 92.49% accuracy, while the optimized 14-channel variant achieves 92.67%. When deployed on an edge-computing platform, the compressed model achieves an accuracy of 92.56%, indicating a negligible accuracy drop (0.12%). However, when evaluated using the challenging OE dataset, the accuracy degrades due to the underlying artifacts described in

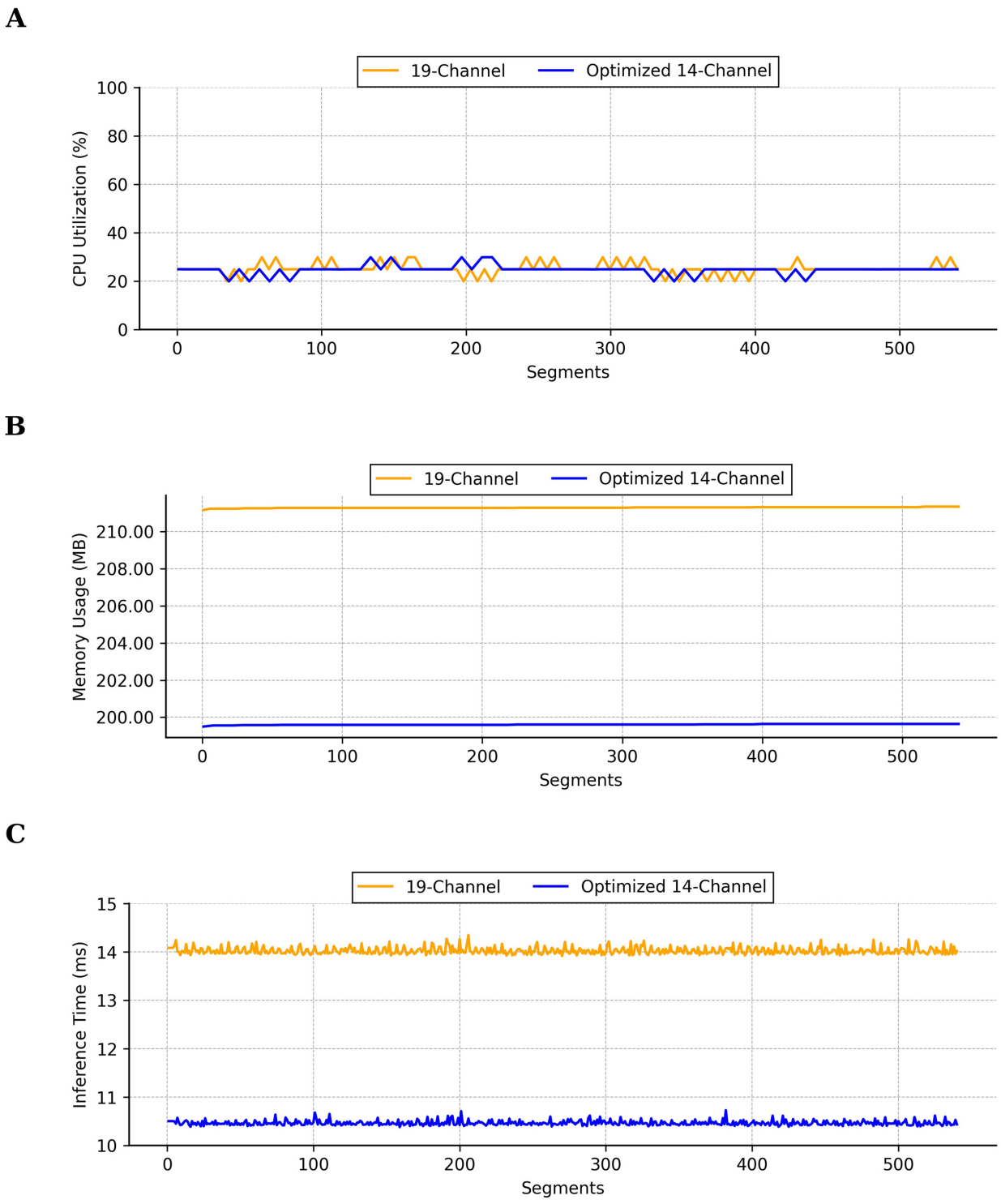


Fig. 6. Resource consumption and inference time per segment during on-edge inference for the DHDL model using the CE dataset. (A) CPU utilization. (B) RAM usage. (C) Inference time.

Section 3.1. The reported accuracy is 74.39% for the 19-channel model and 75.27% for the optimized 10-channel model. The compressed on-edge model preserved this accuracy with a marginal accuracy loss, achieving an accuracy of 75.15%. After applying the channel selection approach, the DHDL model requires only 12,500 trainable parameters and 25 million FLOPs for the CE dataset. For the OE dataset, the model requires 12,372 trainable parameters and 18.54 million FLOPs.

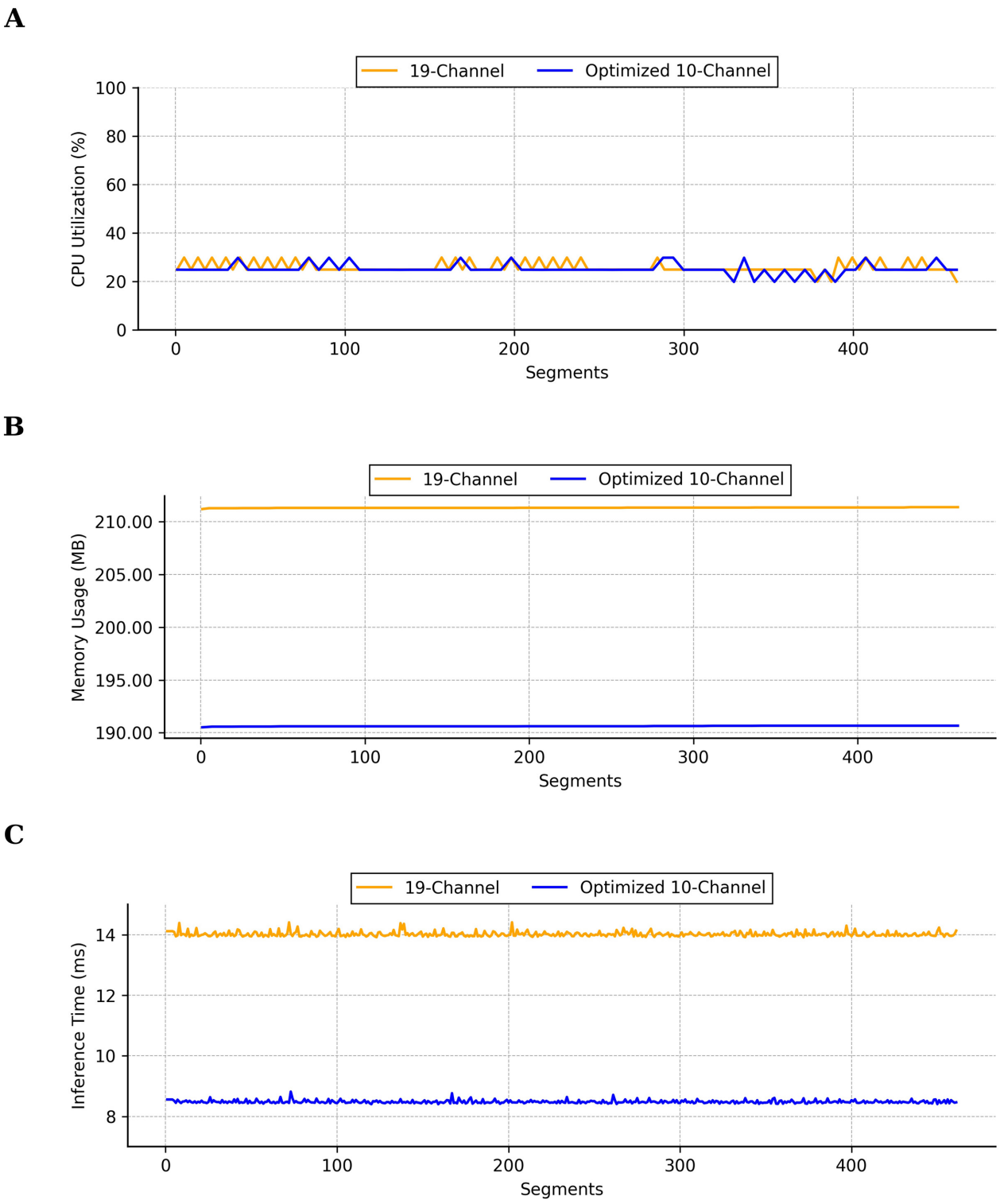


Fig. 7. Resource consumption and inference time per segment during on-edge inference for the DHDL model using the OE dataset. (A) CPU utilization. (B) RAM usage. (C) Inference time.

In contrast, spatiotemporal EEG analysis for dementia detection using convolutional neural network (STEADYNet) and LCADNet employ 446,963 and 141,302 parameters, respectively. Interestingly, EEGNet employs only 2890 parameters, yet it demands 39.23 million FLOPs. This contrast can be attributed to the model architecture. Specifically, EEGNet performs operations on high-dimensional feature maps using temporal and separable

Table 9. Comparison with state-of-the-art models.

Paper	Input	#C	Accuracy (%) ± 95% CI	F1-score (%) ± 95% CI	Test-set inference time (s) ± SD	#parameters	FLOPs (M)
CE dataset							
EEGNet [23]*	Raw EEG	19	79.95 ± 6.94	78.76 ± 7.86	0.48 ± 0.01	**2890**	39.23
DeepConvNet [31]*	Raw EEG	19	64.30 ± 5.95	65.42 ± 6.03	0.56 ± 0.02	167,602	71.02
ShallowNet [31]*	Raw EEG	19	82.85 ± 5.54	83.84 ± 4.54	0.42 ± 0.02	42,002	80.69
LCADNet [11]*	Raw EEG	19	69.81 ± 5.53	72.70 ± 5.53	0.36 ± 0.01	141,302	80.16
STEADYNet [10]	Raw EEG	19	88.00 —	88.00 —	**0.35** —	446,963	**5.353**
ADformer [12]	Raw EEG	19	75.8 —	75.19 —	—	—	—
LEADNet [32]	Raw EEG	19	91.43 —	91.34 —	—	3,410,000	5270
CNN-aLSTM [13]	Temporal & spectral features	19	92.31 —	**93.15 —**	—	24,406	—
DHDL	Raw EEG	19	92.49 ± 2.90	93.02 ± 2.46	0.84 ± 0.02	12,660	33.09
DHDL	Raw EEG	14	**92.67 ± 2.85**	92.96 ± 2.58	0.83 ± 0.01	12,500	25
DHDL (on-edge)	Raw EEG	14	92.56 ± 2.56	93.06 ± 2.20	5.64 ± 0.01	12,500	25
OE dataset							
EEGNet [23]*	Raw EEG	19	66.27 ± 8.22	71.11 ± 8.15	0.30 ± 0.14	**2890**	39.23
DeepConvNet [31]*	Raw EEG	19	58.15 ± 6.24	61.40 ± 8.91	0.39 ± 0.09	167,602	71.02
ShallowNet [31]*	Raw EEG	19	65.59 ± 7.89	68.06 ± 8.22	**0.24 ± 0.15**	42,002	80.69
LCADNet [11]*	Raw EEG	19	53.85 ± 5.93	53.29 ± 7.57	**0.24 ± 0.15**	141,302	80.16
3DCNN [14]	DMD	19	74.23 ± 2.38	74.83 —	—	—	—
DHDL	Raw EEG	19	74.39 ± 7.40	75.69 ± 6.86	0.68 ± 0.08	12,660	33.09
DHDL	Raw EEG	10	**75.27 ± 7.91**	**76.41 ± 7.60**	0.66 ± 0.10	12,372	**18.54**
DHDL (on-edge)	Raw EEG	10	75.15 ± 7.18	75.91 ± 6.45	3.71 ± 0.15	12,372	**18.54**

* Asterisks indicate results reproduced under the same experimental settings used in this study. #C, number of channels. Bold values indicate the optimal results. EEG, electroencephalography; DMD, dynamic mode decomposition; CNN-aLSTM, convolution neural network-attention LSTM; 3DCNN, three-dimensional convolution neural network.

convolutions, but this increases the number of FLOPs, despite the compact parameter count.

In contrast, the DHDL model employs dilated temporal convolutions in the DDG-FF block, and this reduces data dimensionality and convolutional operations from the first layer onward. This efficiency then extends throughout all subsequent layers. Moreover, incorporating a GAP layer before the final fully connected layer further reduces the computational burden at the last layer. When evaluated using the CE dataset, the LEADNet model achieves an accuracy of 91.43% using subject-level evaluation. Here, the classification for a subject is decided by a majority vote that aggregates the predicted class labels of all samples from the same subject. However, the model requires 3.41 million parameters, making it computationally expensive. Hybrid CNN-aLSTM [13] reported accuracy of 92.31%. Only a single deep learning study has been conducted on the OE dataset [14], and the approach achieved 74.23% accuracy. To assess the significance of our results, we applied a paired *t*-test using the results of all four baseline models across all L2SO folds. The results confirmed that our model significantly outperformed the compared models on both datasets ($p < 0.05$ in all comparisons).

4.2 Comparison With Edge-Oriented Models

Beyond the comparison of AD-detection systems, we also compared our model against compact edge-oriented models used for similar classification tasks. Table 10 (Ref. [33,34,35,36]) presents a comparison of the proposed model against four recent models for biosignal classification tasks, including seizure detection [33], electrocardiogram (ECG) anomaly detection [34], ventricular fibrillation detection [35], and drowsiness detection [36]. It shows the commonly used on-edge metrics, namely dynamic power, RAM usage, CPU utilization, accuracy, and per-segment inference time. This comparison can be considered to represent a relative comparative assessment rather than a controlled benchmarking, as the reported studies used different datasets, edge-computing platforms, and experimental configurations. Nevertheless, the comparison serves as a valuable reference for situating the proposed model within the wider domain of on-edge models for biosignal classification tasks.

In terms of power consumption, the ECG anomaly detection model depicted in [34] consumes a notably low power of 2.40×10^{-5} W, whereas the seizure detection model in [33] requires 4.00 W. Our model shows a modest power requirement of 0.93 W and 0.84 W for the op-

Table 10. Comparison with edge-oriented models.

Paper	Application	Edge computing platform	Dynamic power (W)	RAM usage (MB)	CPU utilization (%)	Accuracy (%)	Per-segment inference time (ms)
[33]	Seizure detection	Raspberry Pi 4B	4.00	499.400	35.40	**99.20**	2.06
[34]	ECG Anomaly Detection	Raspberry Pi 4	**2.40×10^{-5}**	—	**15.40**	92.89	**0.20**
[35]	Ventricular Fibrillation Detection	Raspberry Pi 4/STM32F7	1.42/0.952	_/0.055	—	—	1.06/615.70
[36]	Drowsiness detection	nRF52840 MCU	6.11×10^{-2}	**0.004**	—	91.31	18.42
DHDL (CE 14-channels)	Alzheimer detection	Raspberry Pi 5	0.93	199.641	24.96	92.56	10.45
DHDL (OE 10-channels)	Alzheimer detection	Raspberry Pi 5	0.84	190.630	24.91	75.15	8.23

Bold values indicate the optimal results.

timized CE and OE configurations, respectively, compared to other studies. The RAM usage varies substantially across the studies due to differences in the hardware architecture rather than model efficiency alone. Microcontroller-based deployments, by design, operate under strict memory constraints. The model in [36] requires only 0.004 MB on the nRF52840, while [35], running on the STM32F7 device, reports 0.055 MB. For single-board computers, the DHDL models consume 199.641 MB and 190.630 MB on the Raspberry Pi 5 for the CE and OE configurations, respectively. Our results are lower than the 499.400 MB reported by the seizure detection model in [33]. In terms of CPU utilization, our model achieves 24.96% and 24.91% for the CE and OE configurations, respectively, falling between the 15.40% reported by [34] and the 35.40% reported by [33], thereby indicating a moderate computational load. For the four comparison models, the reported classification accuracy ranges between approximately 91% and 92% (the exception is [33], which achieves 99.20%). The DHDL models achieve per-segment inference times of 10.45 ms and 8.23 ms, so they fall between the faster times reported in [34] (0.20 ms) and [33] (2.06 ms) and the slower times in [36] (18.42 ms) and [35] (615.70 ms). Overall, despite the use of different hardware setups and datasets across the considered models, our model demonstrates competitive on-edge performance across all metrics.

4.3 Framework Rationale

The proposed DHDL model offers several key advantages over existing AD detection models, particularly for inference on resource-constrained devices. Rigorous ablation experiments reveal that each block is significant to the overall model performance, justifying our design choices. One key contribution of the DHDL model is the design of dual dilated convolutions aimed at capturing richer features by expanding the receptive field without increasing the number of parameters. This design enables temporal feature modeling under limited computational resources. Consequently, this choice reduces the number of parameters and FLOPs required by the subsequent blocks. Additionally, the design of the hybrid lightweight blocks is intended for edge deployment, where each block enhances the learned feature representation. This lightweight architecture allows effective training of the DHDL model using a limited number of EEG segments, thereby reducing the risk of overfitting. The proposed channel selection approach reduces data dimensionality while also preserving the classification accuracy. Unlike existing AD detection models, the proposed model was successfully deployed on an edge-computing platform. The deployment results demonstrate that the proposed framework strikes a good balance between classification performance, computational complexity, and resource-energy efficiency on edge platforms.

4.4 Limitations and Future Work

Despite the strong performance and promising findings reported for the proposed model, certain limitations should be acknowledged. First, both the CE and OE datasets were obtained from the same source and involved a small cohort of 48 subjects. Also, both datasets were recorded using the international 10–20 EEG system. Consequently, the generalization of the proposed model to other populations, EEG acquisition systems, and recording protocols remains to be demonstrated. The collection of a large-scale EEG dataset and validation on an external dataset or prospectively collected clinical data are necessary to demonstrate the generalization and robustness of the proposed model. Second, potential demographic confounding

cannot be completely excluded. Age may influence EEG characteristics or interact with disease status. The similarity in mean age (66.67 ± 8.78 and 68.13 ± 5.68 years) does not guarantee that age-related EEG effects were fully controlled. Moreover, the AD and HC groups in the datasets were not sex-balanced (67% female in AD vs. 38% female in HC). Sex variations would influence EEG characteristics and therefore affect the model training process. Typically, age- and sex-related differences may affect EEG spectral, temporal, and spatial characteristics. Future studies should validate the framework using large-scale, age- and sex-balanced cohorts and apply age- and sex-stratified analyses to quantify and minimize the influence of demographic and clinical confounders. Third, in real clinical deployment, the AD diagnosis is a clinical decision at the patient level. The decision could be obtained using a majority-vote analysis of all segments belonging to the patient. Future work should account for such a method. Fourth, this study utilized pre-recorded EEG datasets for training and testing the model. Real-world conditions may introduce additional challenges beyond those captured in the analyzed datasets. In depth, EEG recordings can be affected by body motion, cardiac (ECG) interference, eye blinks, power line interference, environmental noise, changes in electrode impedance, electrode displacement and other uncontrolled artifacts. These artifacts are typically more frequent and more severe in unsupervised or home-based recording settings than in the controlled acquisition protocols used to construct the CE and OE datasets. The robustness of the proposed model to these factors was not empirically assessed in this study and should be evaluated in future work using data collected under real-world conditions of use. Fifth, different EEG acquisition devices have different numbers and placements of electrodes, sampling frequencies, amplifier characteristics, filtering settings, and susceptibility to electrical noise. These device-related differences may introduce distribution shifts between the data used for model development and those collected during practical deployment, potentially affecting classification performance. Future work should systematically evaluate robustness across different EEG devices and electrode setups. Sixth, the kernel sizes and dilation rates in the DDG-FF block are tuned to capture high and low frequency patterns for data sampled near 500 Hz. Rescaling the model for data sampled at substantially lower sampling rates is recommended. Seventh, this study considered only one stage of AD, but AD is known to progress through stages ranging from mild cognitive impairment to moderate and severe dementia. Because each stage requires different clinical decisions regarding treatment and patient care, extending the work to multistage AD classification is essential for longitudinal disease monitoring.

The findings of this study suggest several future research directions. First, integrating the optimized channel configurations with a scalp topology map is recommended to examine how these channels correlate with clinical neuropathological regions associated with Alzheimer's disease. Second, although the current on-edge experiments demonstrate efficient per-segment inference, practical IoMT environments may simultaneously perform EEG acquisition, preprocessing, wireless communication, storage, encryption, battery management, and synchronization with digital twin systems. These concurrent processes may affect inference time, energy consumption, and edge-computing platform resources. Accounting for such concurrent processing is an important direction for future work. Third, this study developed an on-edge inference component to enable future IDT for AD monitoring. In future work, we aim to extend the current framework to wider aspects of IDT, such as real-time updating of the patient's digital twin.

5. Conclusion

This study proposed a novel framework for AD detection using EEG signals implemented on an edge-computing platform. The framework consists of a lightweight DHDL model with an effective channel selection approach. Extensive ablation and on-edge experiments were conducted using two well-known EEG datasets (CE and OE) to justify the design choices and confirm the design efficiency. By applying a subject-independent L2SO protocol, the framework achieved 92.56% accuracy on the CE dataset and 75.15% on the OE dataset. When deployed on an edge-computing platform, the optimized models require only 10.45 ms and 8.23 ms per inference for the CE and OE datasets, respectively, while consuming 9.69 mJ and 6.95 mJ of energy per segment, respectively, and maintaining the CPU utilization at approximately 25% and the RAM utilization below 3%. Real-time experimental results showed that the proposed channel selection approach reduced energy consumption while also preserving classification performance. These results demonstrate that AD detection can be accomplished reliably on resource-constrained devices. Although this study is motivated by the potential role of intelligent digital twins in AD detection and monitoring, a complete IDT system was not implemented in this work. Specifically, this study does not include continuous synchronization with patient data and adaptive digital twin updating. Instead, the proposed framework serves as an on-edge inference component for the future IDT systems for neurological monitoring of AD.

Availability of Data and Materials

The datasets used in this study are publicly available in the OpenNeuro repository at https://openneuro.org/datasets/ds004504 (version 1.0.8) and https://openneuro.org/datasets/ds006036 (version 1.0.5). BIDS validation returned a valid status with no errors for both datasets. For ds006036, the warnings reflect missing optional metadata only, and these warnings have no effect on the proposed pipeline.

Author Contributions

AA: Conceptualization, Methodology, Formal Analysis, Software, Validation, Writing—Original Draft, Writing—Review and Editing, Visualization. SAA: Conceptualization, Methodology, Formal Analysis, Writing—Review and Editing, Investigation, Supervision, Project Administration. AS: Conceptualization, Methodology, Formal Analysis, Writing—Review and Editing, Investigation, Supervision, Project Administration. All authors read and approved the final manuscript. All authors have participated sufficiently in the work and agreed to be accountable for all aspects of the work.

Ethics Approval and Consent to Participate

As this study involved secondary analysis of data from a publicly available database, no additional ethics approval or informed consent was required.

Acknowledgment

The authors express their sincere gratitude to King Abdulaziz City for Science and Technology (KACST), represented by the Science Technology and Innovation Unit at King Saud University (STU-KSU) for their valuable support.

Funding

This research project is supported by a grant from King Abdulaziz City for Science and Technology (KACST), represented by the Science Technology and Innovation Unit at King Saud University (STU-KSU), Riyadh, Saudi Arabia (Grant No.STU-54).

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] World Health Organization. The top 10 causes of death. 2025. Available at: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death (Accessed: 17 June 2026).

[2] Passeri E, Elkhoury K, Morsink M, Broersen K, Linder M, Tamayol A, et al. Alzheimer's Disease: Treatment Strategies and Their Limitations. International Journal of Molecular Sciences. 2022; 23: 13954. https://doi.org/10.3390/ijms232213954

[3] Rasmussen J, Langerman H. Alzheimer's Disease - Why We Need Early Diagnosis. Degenerative Neurological and Neuromuscular Disease. 2019; 9: 123–130. https://doi.org/10.2147/DNND.S228939

[4] Amato LG, Lassi M, Vergani AA, Carpaneto J, Mazzeo S, Moschini V, et al. Digital twins and non-invasive recordings enable early diagnosis of Alzheimer's disease. Alzheimer's Research & Therapy. 2025; 17: 125. https://doi.org/10.1186/s13195-025-01765-z

[5] Grieves M. Intelligent digital twins and the development and management of complex systems. Digital Twin. 2024; 1. https://doi.org/10.12688/digitaltwin.17574.1

[6] Chen Y, Wang H, Zhang D, Zhang L, Tao L. Multi-feature fusion learning for Alzheimer's disease prediction using EEG signals in resting state. Frontiers in Neuroscience. 2023; 17: 1272834. https://doi.org/10.3389/fnins.2023.1272834

[7] Nour M, Senturk U, Polat K. A novel hybrid model in the diagnosis and classification of Alzheimer's disease using EEG signals: Deep ensemble learning (DEL) approach. Biomedical Signal Processing and Control. 2024; 89. https://doi.org/10.1016/j.bspc.2023.105751

[8] Biondi A, Dursun E, Viana PF, Laiou P, Richardson MP. New wearable and portable EEG modalities in epilepsy: The views of hospital-based healthcare professionals. Epilepsy & Behavior: E&B. 2024; 159: 109990. https://doi.org/10.1016/j.yebeh.2024.109990

[9] Terry MA, Alvarez Vazquez E. Efficacy of wearable devices in monitoring and treating Parkinson's disease symptoms: A systematic review and meta analyses. IEEE Access. 2025; 13: 84240–84258. https://doi.org/10.1109/ACCESS.2025.3569056

[10] Kachare PH, Sangle SB, Puri DV, Khubrani MM, Al-Shourbaji I. STEADYNet: Spatiotemporal EEG analysis for dementia detection using convolutional neural network. Cognitive Neurodynamics. 2024; 18: 3195–3208. https://doi.org/10.1007/s11571-024-10153-6

[11] Kachare P, Puri D, Sangle SB, Al-Shourbaji I, Jabbari A, Kirner R, et al. LCADNet: a novel light CNN architecture for EEG-based Alzheimer disease detection. Physical and Engineering Sciences in Medicine. 2024; 47: 1037–1050. https://doi.org/10.1007/s13246-024-01425-w

[12] Wang Y, Mammone N, Petrovsky D, Tzallas AT, Morabito FC, Zhang X. ADformer: A multi-granularity spatial-temporal transformer for EEG-based Alzheimer detection. arXiv. 2024. https://doi.org/10.48550/arXiv.2409.00032 (preprint)

[13] Vo T, Ibrahim AK, Zhuang H, Bang C. Extraction and interpretation of EEG features for diagnosis and severity prediction of Alzheimer's Disease and Frontotemporal dementia using deep learning. Biomedical Signal Processing and Control. 2026; 112. https://doi.org/10.1016/j.bspc.2025.108667

[14] Kang J, Kang H, Seo JH. CNN-based framework for Alzheimer's disease detection from EEG via dynamic mode decomposition. Frontiers in Neuroinformatics. 2025; 19: 1706099. https://doi.org/10.3389/fninf.2025.1706099

[15] Ntetska A, Miltiadous A, Tsipouras MG, Tzimourta KD, Afrantou T, Ionnadis P, et al. A complementary dataset of scalp EEG recordings featuring participants with Alzheimer's disease, Frontotemporal dementia, and healthy controls, obtained from photostimulation EEG. Data. 2025; 10. https://doi.org/10.3390/data10050064

[16] Al-Nuaimi AH, Blūma M, Al-Juboori SS, Eke CS, Jammeh E, Sun L, et al. Robust EEG Based Biomarkers to Detect Alzheimer's Disease. Brain Sciences. 2021; 11: 1026. https://doi.org/10.3390/brainsci11081026

[17] Imani M. Alzheimer's diseases diagnosis using fusion of high informative BiLSTM and CNN features of EEG signal. Biomedical Signal Processing and Control. 2023; 86. https://doi.org/10.1016/j.bspc.2023.105298

[18] Ingolfsson TM, Hersche M, Wang X, Kobayashi N, Cavigelli L, Benini L. EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces. , 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 2958–2965). IEEE. 2020. https://doi.org/10.1109/SMC42975.2020.9283028

[19] Zhao R, Wang Y, Cheng X, Zhu W, Niu H, Cheng J, et al. A mutli-scale spatial-temporal convolutional neural network with contrastive learning for motor imagery EEG classification. Medicine in Novel Technology and Devices. 2023; 17. https://doi.org/10.1016/j.medntd.2023.100215

[20] Russell JS, Norvig P. Artificial Intelligence: A Modern Approach. 4th edn. Prentice Hall: New Jersey. 2020.

[21] Miltiadous A, Tzimourta K, Afrantou T, Ionnadis P, Grigoriadis

N, Tsalikakis DG, et al. A dataset of scalp EEG recordings of Alzheimer's disease, frontotemporal dementia and healthy subjects from routine EEG. Data. 2023; 8. https://doi.org/10.3390/data8060095

[22] Hu J, Shen L, Albanie S, Sun G, Wu E. Squeeze-and-Excitation Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2020; 42: 2011–2023. https://doi.org/10.1109/TPAMI.2019.2913372

[23] Lawhern VJ, Solon AJ, Waytowich NR, Gordon SM, Hung CP, Lance BJ. EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces. Journal of Neural Engineering. 2018; 15: 056013. https://doi.org/10.1088/1741-2552/aace8c

[24] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Long Beach, CA, USA. Curran Associates Inc. 2017. Available at: https://papers.nips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (Accessed: 17 June 2026).

[25] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation. 1997; 9: 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[26] scikit-learn. RobustScaler. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html (Accessed: 17 June 2026).

[27] Mikheev I, Steiner H, Martynova O. Detecting cognitive traits and occupational proficiency using EEG and statistical inference. Scientific Reports. 2024; 14: 5605. https://doi.org/10.1038/s41598-024-55163-w

[28] TensorFlow Model Optimization. Available at: https://www.tensorflow.org/model_optimization/guide (Accessed: 7 March 2026).

[29] Reddy TB. Linden's Handbook of Batteries. 4th edn. McGraw-Hill: London. 2011.

[30] Meghdadi AH, Stevanović Karić M, McConnell M, Rupp G, Richard C, Hamilton J, et al. Resting state EEG biomarkers of cognitive decline associated with Alzheimer's disease and mild cognitive impairment. PloS One. 2021; 16: e0244180. https://doi.org/10.1371/journal.pone.0244180

[31] Schirrmeister RT, Springenberg JT, Fiederer LDJ, Glasstetter M, Eggensperger K, Tangermann M, et al. Deep learning with convolutional neural networks for EEG decoding and visualization. Human Brain Mapping. 2017; 38: 5391–5420. https://doi.org/10.1002/hbm.23730

[32] Puri DV, Kachare PH, Sangle SB, Kirner R, Jabbari A, Al-Shourbaji I. LEADNet: Detection of Alzheimer's disease using spatiotemporal EEG analysis and low-complexity CNN. IEEE Access. 2024; 12: 113888–113897. https://doi.org/10.1109/ACCESS.2024.3435768

[33] Akor PA, Enemali G, Muhammad U, Singh RR, Larijani H. An Attention-Residual Convolutional Network for Real-Time Seizure Classification on Edge Devices. Sensors (Basel, Switzerland). 2025; 25: 6855. https://doi.org/10.3390/s25226855

[34] Hizem M, Bousbia L, Ben Dhiab Y, Aoueileyine MOE, Boullegue R. Reliable ECG Anomaly Detection on Edge Devices for Internet of Medical Things Applications. Sensors (Basel, Switzerland). 2025; 25: 2496. https://doi.org/10.3390/s25082496

[35] Krasteva V, Stoyanov T, Jekova I. Implementing deep neural networks on ARM-based microcontrollers: Application for ventricular fibrillation detection. Applied Sciences. 2025; 15. https://doi.org/10.3390/app15041965

[36] Nguyen HT, Mai ND, Lee BG, Chung WY. Behind-the-ear EEG-based wearable driver drowsiness detection system using embedded tiny neural networks. IEEE Sensors Journal. 2023; 23: 23875–23892. https://doi.org/10.1109/JSEN.2023.3307766