IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 11, 2022, accepted May 24, 2022, date of publication June 3, 2022, date of current version June 9, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3180045

Reservoir Computing for Early Stage Alzheimer's Disease Detection

NICKSON MWAMSOJO[1], FREDERIC LEHMANN[2], (Member, IEEE), MOUNIM A. EL-YACOUBI[1], (Member, IEEE), KAMEL MERGHEM[1], YANN FRIGNAC[3], (Member, IEEE), BADR-EDDINE BENKELFAT[1], (Senior Member, IEEE), AND ANNE-SOPHIE RIGAUD[4]

[1]Electronics and Physics Department, SAMOVAR, Telecom Sudparis, IP Paris, 91120 Palaiseau, France
[2]Communications, Images and Information Processing Department, SAMOVAR, Telecom Sudparis, IP Paris, 91120 Palaiseau, France
[3]Huawei Technologies France, 92100 Boulogne-Billancourt, France
[4]AP-HP, Groupe Hospitalier Cochin Paris Centre, Hôpital Broca, Pôle Gérontologie, 75013 Paris, France

Corresponding author: Nickson Mwamsojo (nickson_mwamsojo@telecom-sudparis.eu)

ABSTRACT Artificial Neural Networks (ANNs) have amassed unprecedented success in information processing ranging from image recognition to time series prediction. The success can largely be attributed to the availability of large datasets for training and the increased complexity of the models. Unfortunately, for some applications only a limited amount of samples is available for training. Fewer training samples increases the risk of over-fitting and poor generalization especially in high complexity models. Moreover, complex models with a large number of trainable parameters require more energy to train and optimize compared to simpler ones. In this paper, to the best of our knowledge, we propose the first use of ANNs for Early Stage Alzheimer Disease classification (ES-AD) from the handwriting (HW). We propose using a framework for building Recurrent Neural Networks (RNNs) known as Reservoir Computing (RC), both numerically and experimentally, that simplifies training by optimizing the output layer only. We also propose the Bidirectional Long Term Short Term (BiLSTM) and Convolutional Neural Network (CNN) methods for comparison. For a fairer comparison, we not only consider the accuracies but also the energy costs incurred to obtain the respective accuracies in order to assess the accuracy-efficiency trade-off. Our numerical and experimental results show that RC yields a classification accuracy of 85%, which is 3% worse than that of BiLSTM and 2% better than that of CNN, at a relatively lower training and significantly lower inference costs. We hope that our findings highlight the importance of examining the accuracy-efficiency trade-off of various models in the community in order to reduce the overall impact of ANNs training on the environment.

INDEX TERMS Alzheimer's, artificial neural networks, handwriting, reservoir computing, green AI.

I. INTRODUCTION

Alzheimer's is a brain disease caused by the progressive destruction of nerve cells in various parts of the brain. It falls under dementia, an umbrella term describing the symptoms associated with the decline of individual's ability to think, to learn and to memorize information. In its severe forms, the disease interferes with patients' work and social lives making them incapable of handling everyday tasks. Of the various forms of dementia, Alzheimer's is the most common type and accounts for between $60-80\%$ of all the neurodegenerative diseases [1].

Like all other types of dementia, Alzheimer's is strongly correlated with the ages of the patients and hence it is more prevalent in senior members of the population. With the increase in life expectancy, it should be expected for Alzheimer's disease to be an alarming cause of dependency among the elderly. In the USA and in France, for example, the prevalence of Alzheimer's is as high as 13.8% (year 2021) [1] and 17.8% (year 2015) [2] for people aged above 75 years of age and older, respectively.

Unfortunately, Alzheimer's disease has an insidious onset due to its slow and progressive nature. In early stages, it is asymptomatic, then it passes through a spectra of mild cognitive impairment (MCI) symptoms before evolving into severe forms which could be fatal. This progression makes early detection of the disease difficult while late diagnosis, in turn,

The associate editor coordinating the review of this manuscript and approving it for publication was Dominik Strzalka.

reduces the effectiveness of the treatments for slowing the development of severe symptoms. The need for Early Stage Alzheimer Detection (ES-AD) is therefore a crucial area of research.

Traditional methods for the detection of the disease are based on the recommendations of the report [3] proposing that physicians use a set of various tools to make the diagnosis. These methods are prone to bias and offer poor repeatability of the test. More accurate methods include methods such as the Positron Emission Tomography (PET) of brain amyloid [4] or the examination of the Cerebrospinal Fluid (CSF) after lumbar puncture procedure [5]. These methods offer good sensitivity to Alzheimer, however, they are intrusive and expensive [6].

Neurodegenerative diseases are known to impair the control of fine motor movements [7] of patients. Since writing requires fine motor control, the impact of the diseases manifests itself in the handwriting (HW) of patients. This makes the handwriting kinematic patterns important bio-markers that are discriminatory for certain pathologies. In the literature, we find investigations on Alzheimer's [7]–[14], Parkinson's [16], [17], [25] and Huntington [19], [20] diseases, among many others. We interest ourselves in the ES-AD problem using HW dynamics for two reasons: Firstly, because Alzheimer's is the most prevalent cause of dementia and secondly, because HW analysis is an inexpensive yet effective tool for the task.

The HW acquisition for pathology detection can be either dynamic (online) [21] or paper-based (offline) [22]. The dynamic acquisition means that the HW trajectory is recorded in real time hence the temporal information for every point is available. The offline method records only the positional information of the trajectory. The temporal information allows to capture the kinematics of the entire writing or drawing process which is not the case for the offline paper-based counterpart. As a result, the dynamic HW has richer data while the paper-based approach lacks the subtle temporal patterns that may be discriminatory for the pathology. It is for this reason that we work with the dynamic HW, as it has proven crucial for the task at hand.

In the literature, most works on HW for AD are based on statistical tests of the HW [7]–[9]. Some approaches considered Machine Learning-based classification methods. The authors of [10]–[12] used the Linear Discriminant Analysis method. These works, however, are based on the assumption that there is a unique behavioral pattern associated with Alzheimer's disease, and as a result, the problem was approached by extraction of global kinematic parameters (e.g. average velocity) of the HW for the Alzheimer Detection (AD). The authors of [13] pointed out the flaws of this heavy assumption, as it denied classifiers the rich temporal information that could be useful for detection. Their results indicate the advantage of exploring the full dynamics of the raw data and highlight the limits of working on the global kinematic parameters.

The authors of [14] also used the full kinematics of HW pattern in a temporal clustering based on k-Medoids to uncover clusters using Dynamic Time Warping (DTW) as a dissimilarity measure. For classification, they consider a Bayesian classifier that aggregates the contributions of the resultant clusters to each class and obtained the state-of-the-art classification accuracy of 74%. Inspired by [13], [14] we process the full HW dynamics in our work as well, to benefit fully from information contained in the raw time-series. However, unlike these previous works, and to the best of our knowledge, we propose the first use of ANNs approaches for the ES-AD task using HW dynamics as a time series.

We work with Bidirectional Long Short-Term Memory (BiLSTM) and Reservoir Computing (RC), two methods adapted for processing serial data. BiLSTM is a well known special class of Recurrent Neural Networks (RNNs) and has been considered on a stand-alone basis [23] or in combination with Convolutional Neural Networks (CNNs), for Parkinson's disease detection [24]. Moreover, we implement a stand-alone CNN, a class of neural networks that are often regarded as reference models for classification tasks and have also been applied successfully for Parkinson's disease detection [25]. One potential downside to BiLSTMs and CNNs is the high complexity and the large number of trainable parameters which makes them prone to over-fitting and poor generalization especially when the number of training samples is small. The high complexity also results in higher training costs in terms of energy consumed in the process. On the other hand, RC, another special class of RNNs, significantly simplifies the learning process and reduces considerably the number of trainable parameters by training the output layer only. This hints lesser energy consumption for training RCs compared to BiLSTM, CNN and a potential for lower inference costs as well. Lower inference cost of a model is advantageous in cases where the classification is run repeatedly; whether locally on a portable electronic device or on a distant server in the cloud. Moreover, unlike BiLSTM, RCs' recurrent part can be implemented in analog hardware setups which comes with further energy and speed advantages. We test the performance and efficiency of a hardware RC setup and compare it to BiLSTM, CNN and the equivalent digitally simulated RC in this work. It is worth noting that, despite the simplicity and versatility of the concept, RC has proven successful, both in numerical and hardware experiments, in processing serial tasks such as speech recognition [26]–[28], Chaotic Time Series Prediction [29] and non-linear optical channel equalization [30], [31].

The success and progress of ANNs can be largely attributed to the increase in their depth and complexity, with a consequential increase in the cost of training them. Researchers tend to focus solely on reaching or surpassing state-of-the-art accuracies with little attention paid to the energy costs incurred. Oftentimes, an insignificant increase in accuracy comes at a consequential increase in the training costs. This approach is not environmentally friendly and has been coined the term Red-AI [32]. Red-AI not only increases the cost of

running the models but also complicates the implementation of the ANN models on mobile devices such as telephones or tablets [33] which could be useful in certain applications, such as the ES-AD task. The greener approaches on the other hand, termed Green-AI, should incorporate the energy efficiency analysis as an added metric for both model selection and optimization as we do in this paper.

Our contribution is therefore three-fold. First, we propose a relatively simple but powerful concept known as Reservoir Computing (RC) for the ES-AD task in both numerical and hardware implementations. Secondly, we benchmark the performance of our approach with that of Bidirectional Long Short-Term Memory (BiLSTM) k-Medoids and the Convolution Neural Network (CNN) on the full HW dynamics taken as a time-series for the classification of healthy control patients (HC) and patients with the Alzheimer Disease (AD). Finally, we highlight the importance of extending the model analysis and selection to include energy consumption by adding energy efficiency on top of prediction accuracy during the model selection, training and inference [32], [34]. We note that, the proposed work-flow is not specific to the ES-AD task but can be generalized to assess accuracy-efficiency trade-off for any other task.

We structure the rest of this paper as follows: in Section II, we detail the proposed methods, that is, the Reservoir Computing, BiLSTM, CNN and then the technique for models' efficiency comparison. In Section III, we explain the task at hand, the data used and our designed experiments for gauging the classification systems. In Section IV we present and discuss the results obtained and finally in Section V we give our concluding remarks.

II. PROPOSED MODELS

A. RESERVOIR COMPUTING

Reservoir Computing was independently introduced as Echo State Networks (ESN) [35] and Liquid State Machines (LSM) [36]. The two are fundamentally similar, with the LSM closely mimicking the sophistication of the spiking biological neurons while the ESN, the variant of focus in this work, is based on the traditional Recurrent Neural Networks (RNNs).

RC architecture typically consists of three layers, the input layer, the reservoir layer and the output (read-out) layer, as shown in Fig. 1. The input layer receives the input signal $\mathbf{u}(n) \in \mathbb{R}^F$ where n is the discrete time $n = 1, 2, \ldots T$ and F the number of features. Then a random projection of the input signal according to a randomly generated $N \times F$ matrix $\mathbf{W}_{in}$ (1) gives the input signal $\mathbf{s}(n)$ where N is the number of neurons/nodes in the reservoir layer. This matrix is called an input mask and has the role of enriching the dynamics by breaking all the symmetries that may occur in the input before being injected into the reservoir layer [37].

$$\mathbf{s}(n) = \mathbf{W}_{in}\mathbf{u}(n). \tag{1}$$

The signal $\mathbf{s}(n)$ drives the second layer called the reservoir layer. It consists of highly recurrent neurons that carry the nonlinear expansion of the inputs. The neurons will be referred to as non-linear nodes or just nodes in this paper. The number of these nodes N determines the *capacity* of the reservoir [35]. The nodes are interconnected spatially according to a sparse $N \times N$ connectivity matrix $\mathbf{W}_{res}$. Like $\mathbf{W}_{in}$, $\mathbf{W}_{res}$ may in general be a randomly generated matrix from any zero-mean probability distribution. The node states $\mathbf{x}(n) \in \mathbb{R}^N$ evolve in discrete time n according to (2):

$$\mathbf{x}(n) = f_{NL}(\alpha\mathbf{W}_{in}\mathbf{u}(n) + \beta\mathbf{W}_{res}\mathbf{x}(n-1) + \mathbf{b}). \tag{2}$$

The α and β are called the input and feedback scaling factors respectively and are responsible for controlling the resultant scaling of the dynamics of the reservoir layer and $\mathbf{b}$ accounts for the bias term. The function f_{NL} can be any nonlinear activation function, which in practice is typically a hyperbolic tangent or a sigmoid.

The third layer of the RCs is the output layer also known as the readout layer. This layer takes the node states and linearly transforms them to an output $\hat{\mathbf{y}}(n) \in \mathbb{R}^C$. This is accomplished by the multiplication of the recorded node transient states $\mathbf{x}(n)$ by a $C \times N$ matrix $\mathbf{W}_{out}$ where C is the number of classes as in (3):

$$\hat{\mathbf{y}}(n) = \mathbf{W}_{out}\mathbf{x}(n). \tag{3}$$

For RCs, only the readout weights in the $\mathbf{W}_{out}$ matrix are to be optimized through training. This simplifies the learning when compared to traditional methods where every layer is trained by the back-propagation method. The training in RCs aims to minimize the Mean Squared Error (MSE) between the expected and predicted target vectors as given in the equation below:

$$MSE = \frac{1}{T}\sum_{n=1}^{T}\left\|\mathbf{y}(n) - \hat{\mathbf{y}}(n)\right\|_2^2, \tag{4}$$

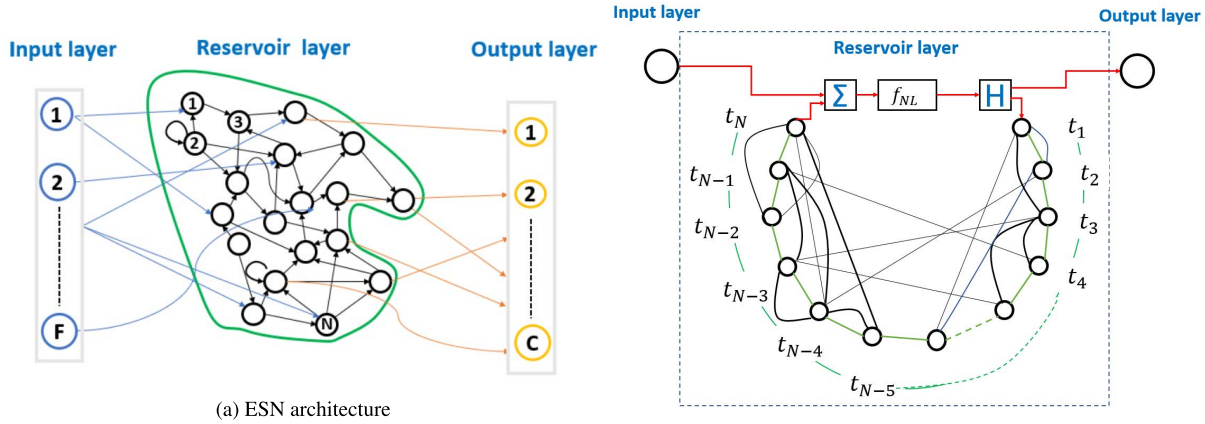
where $\mathbf{y}(n)$ is the expected output vector and $\hat{\mathbf{y}}(n)$ is the predicted output at instant n.

We compute the optimal $\mathbf{W}_{out}$ using ridge regression at the output layer to compute the optimal $\mathbf{W}_{out}^{opt}$. From (4), using the generalized Moore–Penrose pseudo-inverse, (5) is obtained where $\mathbf{Y}$ and $\mathbf{X}$ are large matrices from the column-wise concatenation of $\mathbf{y}(n)$ and $\mathbf{x}(n)$ for all n respectively. $\mathbf{I}$ is an identity matrix and λ is a regression parameter aimed at combating over-fitting. We set $\lambda = 10^{-8}$ manually after testing different values.

$$\mathbf{W}_{out}^{opt} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I})^{-1}. \tag{5}$$

1) DIGITAL RESERVOIR COMPUTER

We implement a numerical simulation of an architecture similar to the one introduced by the authors of [38]. They proposed an approach where the spatial nonlinear nodes are replaced by one node coupled to a delayed feedback. The previous spatial nodes are virtually distributed along a delay line using Time Division Multiplex (TDM). This is shown in Fig. 1b, placed side by side with Fig. 1a to highlight the similarity between the original ESN architecture and the



(a) ESN architecture

(b) NDN architecture

FIGURE 1. **(a) A spatiotemporal architecture showing the input, reservoir and output layers. The reservoir nodes are randomly distributed in space. (b) The NDN architecture where a single non-linear node is coupled to a delay line along which the virtual nodes are distributed. Nodes correspond to the TDM intervals i.e. t_1, t_2 up to t_N.**

equivalent Non-linear Delay Nodes (NDN) architecture with their nodes in spatial and temporal fashion, respectively. The NDN approach rendered feasible the experimental implementations of RCs leading to the opto-electronic [26]–[28], [39], [40] and the optical implementations [41], [42] respectively. The non-linear function proposed in the experiments of [26] is the Mach-Zehnder Modulator transfer function, the sin^2, particularizing (2) to:

$$\mathbf{x}(n) = sin^2(\alpha \mathbf{W}_{in}\mathbf{u}(n) + \beta \mathbf{W}_{res}\mathbf{x}(n-1) + \phi). \tag{6}$$

The bias parameter ϕ now controls the operating regime which in physical implementations corresponds to the Mach-Zehnder Modulator (MZM) bias voltage. Its value determines whether the dynamics of the reservoir fall on the highly linear or highly non-linear regime of the MZM sinusoidal transfer function. For an in-depth understanding of the NDN architectures, we refer the reader to [26]–[28], [39]–[42]. In this work, we implement the NDN type of RCs previously described both as numerical simulation on a CPU and then proceed to implement an equivalent hardware setup in the lab.

In practice, it is generally recommended to make $\mathbf{W}_{in}$ and $\mathbf{W}_{res}$ sparse. The sparsity has two advantages: it speeds up matrix computations when sparse matrix representations are used especially in large reservoirs [43], and it encourages the emergence of sub-networks with richer localized dynamics which may improve performance [35]. We, therefore, generate $\mathbf{W}_{in}$ with its elements sampled from the probability distribution below:

$$P\left(\mathbf{W}_{in}(i,j) = w_{i,j}\right) = \begin{cases} \frac{1}{5}, & \text{if } w_{i,j} = -1, 1. \\ \frac{3}{5}, & \text{if } w_{i,j} = 0. \end{cases} \tag{7}$$

The weights of the matrix $\mathbf{W}_{res}$ in our digital RC simulation, rather than being completely random, are generated with inspiration from our subsequent hardware implementation. In the hardware RC, $\mathbf{W}_{res}$ elements are governed by the coefficients of a first order low-pass filter $h(t) = e^{-t/T_R}$ of a response time $T_R \approx 240ns$ sampled at intervals equal to node duration θ. The ratio $\frac{T_R}{\theta} = 4.6$ implies that 5 neighboring nodes are strongly coupled and are weakly coupled to the rest. We emulate this when setting up $\mathbf{W}_{res}$ in our digital RC simulations, using coefficients sampled from $h_i = h(i\theta)$ where $\theta = 52.18ns$, and placing them in $\mathbf{W}_{res}$ while respecting the causality condition as follows:

$$\mathbf{W}_{res} = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & . & . & . & 0 \\ h_1 & h_0 & 0 & 0 & 0 & . & . & . & 0 \\ h_2 & h_1 & h_0 & 0 & 0 & . & . & . & 0 \\ h_3 & h_2 & h_1 & h_0 & 0 & . & . & . & 0 \\ h_4 & h_3 & h_2 & h_1 & h_0 & . & . & . & 0 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ & . & . & . & h_4 & h_3 & h_2 & h_1 & h_0 \end{bmatrix} \tag{8}$$

The rest of simulation parameters, such as the size of the reservoir (N), the input (α) and the feedback (β) factors are determined through an exhaustive search optimization procedure to determine their optimal values provided in Section IV.

2) HARDWARE RESERVOIR COMPUTER

Compared to a classical Von-Neumann based architecture for the aforementioned digital RC, an optoelectronic physical implementation has a potential of large savings in terms of processing time [27]. We, therefore, extend the analysis of the simulated digital RC in Section II-A1 by implementing an equivalent optoelectronic hardware counterpart. The setup we use in this work is a physical implementation of the scheme shown in Fig. 2. This setup was provided to us by Laurent Larger one of the authors of [26] from FEMTO-ST in Besançon, France. It consists of several components:

- A distributed feedback (DFB) laser diode emitting light at wavelength $1550nm$.
- A Digital-to-Analog Converter (DAC) that carries the masking operation $\mathbf{W}_{in}\mathbf{u}(n)$, then encodes this discrete information into a continuous electric signal $u_I(t)$ using a sample and hold procedure.

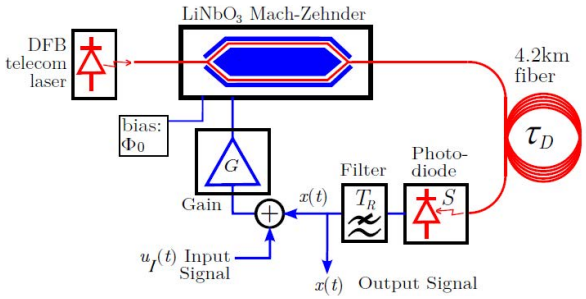


FIGURE 2. Optoelectronic setup from FEMTO-ST. (Reprinted with permission from [26] © The Optical Society.)

- A Mach-Zehnder Modulator (MZM) modulated by the signal from the DAC with a sinusoidal transfer function encoding the electric signal into the phase of the laser light.
- A 4.2 kms fiber spool acting as a delay line (of $\tau_D \approx 20.87\mu s$) through which the modulated light propagates.
- A photodiode that converts the optical intensity variations into electric variations.
- An electronic feedback circuit acting as a low-pass filter with response time T_R whose exact coefficients are not known. The circuit also enables the summation of the input $u_I(t)$ and the feedback signal $x(t - \tau_D)$.

This system can be described by the scalar differential equation below:

$$T_R \frac{dx(t)}{dt} + x(t) = \Omega \sin^2(\alpha u_I(t - \tau_D) + \beta x(t - \tau_D) + \phi), \tag{9}$$

where ϕ is the MZM bias and Ω is the nonlinearity gain. Like the simulated counterpart, with the delay-spacing between consecutive nodes of $\theta \approx 52.18\ ns$ [26], this setup allows the creation of up to $N = 400$ along the delay line. The filter response time $T_R = 4.6\theta$ allows for local coupling between the virtual nodes emulating the sparse connections of reservoir nodes. Since the exact filter weights are unknown, the coefficients of $\mathbf{W}_{res}$ for the hardware are unknown. Finally, part of photo-detected $x(t)$ is recorded and used for the offline post-processing of the output layer.

B. BIDIRECTIONAL LONG-SHORT TERM MEMORY

Long Short-Term Memory (LSTM) neural networks are a special class of RNNs with special gates that can store, read and reject information from both recent and previous outputs further back in time (short and long memories) [44]. BiLSTM is a variant of LSTM that exploits, simultaneously, the past and future (hence bidirectional in time) for computation which makes it well suited for certain tasks. Fig. 3 shows the principle of operation with the classifier having the serial information ($\mathbf{u}(n)$) processed in both past-future and future-past directions to give the output ($\mathbf{d}(n)$) for processing in the subsequent dense layers. BiLSTM is extensively studied in the literature (we direct the reader to [45]–[47]). We use Python's Tensorflow to implement the BiLSTM models for this paper.

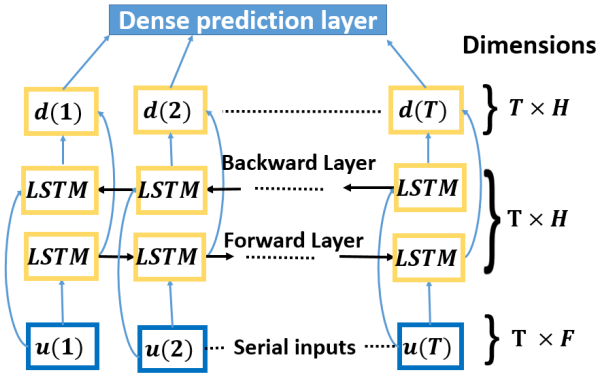


FIGURE 3. Block diagram illustrating the BiLSTM working principle. The serial inputs features are introduced to the hidden layers both in past-future (right arrows) and future-past (left arrows) directions.

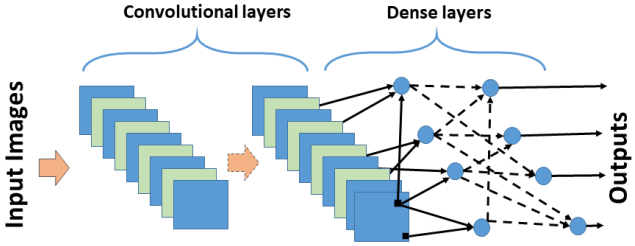


FIGURE 4. Block diagram illustrating the CNN working principle.

C. CONVOLUTIONAL NEURAL NETWORKS

CNNs are a special class of ANN notorious for image processing. They consist of stacked layers performing convolution operations with a number of filters automatically deduced from the data during training. The layers extract various discriminatory features from images. These features are then sent to the dense layer(s) for neural processing and classification as shown in Fig. 4. In this work, for comparison, we implement a CNN architecture similar to the one in [25] that proved successful for the detection of Parkinson's disease from HW features.

D. MODEL ENERGY EFFICIENCY

In line with our discussion on the environmental impact of AI in the Introduction, we gauge the efficiency of our models by examining the cost incurred during the optimization, training, and inference to attain the reported performances. Various metrics are proposed in the literature for model's cost estimation such as the time it takes to run the algorithm [48]–[50], the energy consumed [49], [50], and the count of Floating Point Operations (FPOs) [50], [51]. A framework for estimating the energy consumed by machine learning algorithms was introduced in [52]. This framework, named the *experiment-impact-tracker*, tracks the energy consumption of the target algorithm and then estimates the mass of carbon dioxide gas or equivalent green house gasses (CO_{2eq}) emissions. It does so by making use of the models from [53] that take into account the nature of the power lines in the selected geographical location for more accurate estimation of the CO_{2eq} emissions. We use the proposed framework

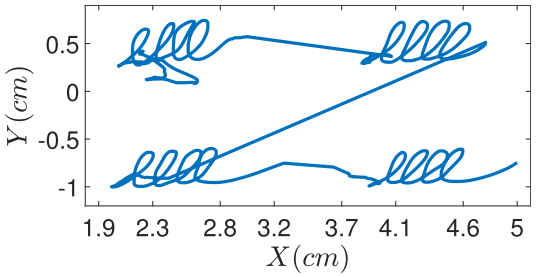


FIGURE 5. Patient cursive-ℓ data entry.

to obtain the energy costs of hyperparameter optimization and model selection. Additionally, to estimate the relative cost of training and inference for the RC, CNN, BiLSTM and k-Medoids we will compare the number of trainable parameters and the FPOs carried on the processors using a tool called PAPI [54]. For the hardware implementation, we use the duration of the experiment and the sum of powers consumed by various components of the setup to estimate the energy consumption when running the experiment.

III. DATASET AND METHODOLOGY

A. DATASET

The HW data used in our work is the cursive-ℓ data where participants were asked to write four sets of four ℓ letters, i.e., $\ell\ell\ell\ell$ on a tablet to form a pattern shown in Fig. 5. The data were collected at Broca Hospital in Paris from 54 participants, evenly split into 27 healthy control patients (HCs) and 27 with Alzheimer's Disease patients (AD), and aged between 68 and 86. They were collected using a WACOM Intuos Pro Large Tablet at a sampling rate of 125Hz. The tablet recorded the x-coordinate (X), y-coordinate (Y), pen-pressure (P), pen-azimuth (Az) and altitude of the pen on its on air trajectory (Al). The tablet allowed for the dynamic acquisition by also recording the corresponding time-stamps throughout the acquisition process.

B. DATA PREPROCESSING

The dataset is small, with only 54 participants. In RC, like for all other machine learning methods, the need for enough training data is crucial to ensure proper generalization and reduce over-fitting during the optimization process. The data was augmented by segmenting the four $\ell\ell\ell\ell$ letters into individual loops in order to use each loop as a separate data sequence. The segmentation process is done in three steps: (i) computation of the velocity of the pen in the y-direction to obtain V_y as presented in the Fig. 6, (ii) filtering by a low-pass filter V_y where the cut-off frequency is set to its fundamental frequency F_0 to obtain a smoothed version shown in Fig. 7, (iii) the individual loops are obtained by selecting sets of two consecutive points where the smoothed-out $V_y = 0$. These points correspond to the top and bottom points of the loop where V_y changes its sign. These points were also used to segment the rest of the features such that we obtain 16 loops for each patient making the total number of loops 866 (two patients made one extra loop each) [14].

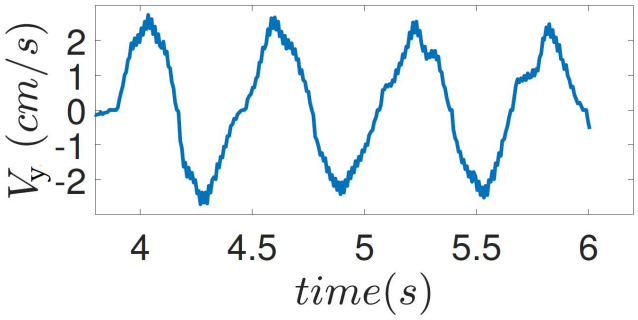


FIGURE 6. Raw y-velocity (V_y) before filtering.

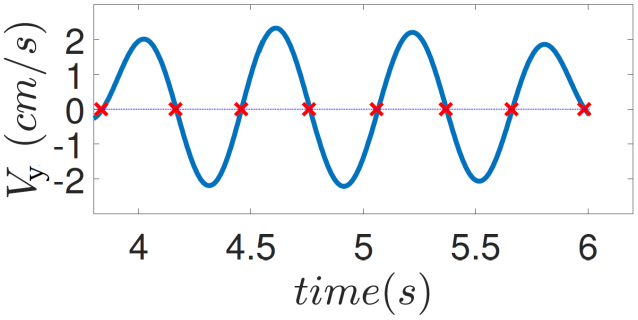


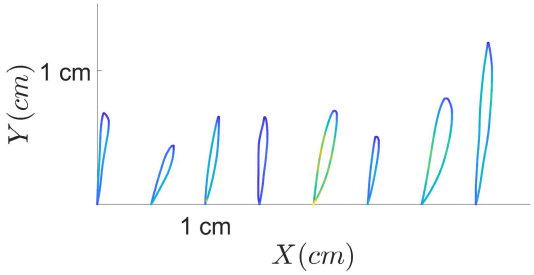
FIGURE 7. Smoothed V_y after low-pass filtering. The red crosses show points used for segmentation.

A single loop, therefore, has the following set of time-series features associated to it: x-position (X), y-postition (Y), x-velocity (V_x), y-velocity (V_y), pen-pressure (P), pen-azimuth (Az), pen-altitude (Al) and a corresponding label (AD or HC). Fig. 8 shows 16 loops (8 from HC and 8 from AD patients) that were extracted through this segmentation with a color-map showing the corresponding velocities at each point. For the CNN, the image inputs are obtained by converting these time-series to the Gramian Angular Field, a polar coordinate representation of the time-series [18].

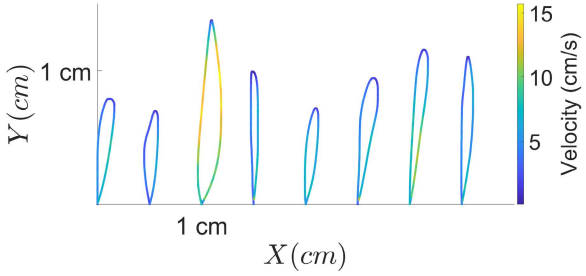
C. CLASSIFICATION SETUP

The ES-AD task is essentially a classification task with labels being either AD (1) for Alzheimer's patients or HC (0) for healthy control patients. We start by optimizing the hyperparameters of the models using the grid search algorithm over the RC parameters such as the reservoir size N, α, and β (see (6)) with one important addition: the feature combination index (k) referring to one of the feature combinations indicated on the **Parameters** column of Table. 4 was also set as a parameter to be optimized.

Since our data is small and we wish to minimize potential over-fitting while we tune the hyperparameters and select the final model, we adopted the Nested-Cross Validation (NCV) technique [55], [56]. With this technique, we have two loops, the inner loop for hyperparameter optimization using $\frac{7}{9}$ (42 subjects) and $\frac{1}{9}$ (6 subjects) of the data for training and validation respectively. We use the remaining $\frac{1}{9}$ (6 subjects) of the data for testing and final model selection in the outer loop. The NCV, unlike the traditional Cross Validation (CV), guarantees that the $\frac{1}{9}$ (6 subjects) held out for test are never



(a) Healthy control patients loops



(b) Alzheimer patients loops

FIGURE 8. (a) and (b) show loops extracted from HC and AD patients respectively. The color gradient corresponds to the velocity of the pen along its loop trajectory.

used for optimization of the hyperparameters in the inner loop. For the 7-1-1 split, we used the stratified sampling method to ensure class homogeneity and we used the classification accuracy as the metric for gauging the models' performances. The accuracy was expressed as the percentage of the correctly classified patients after voting from all their individual $\ell-$ loop classification scores using the maximum voting approach. The reported performance metrics are averages obtained on the test set from the $\frac{7}{9}$-$\frac{1}{9}$-$\frac{1}{9}$ splits turnovers. We use the nested cross-validation for both the RC, CNN and BiLSTM models with an addition of the energy tracker as described in Section II-D for cost estimation. The results on accuracy and energy costs incurred are presented in the Section IV.

IV. RESULTS AND DISCUSSIONS

We run the nested cross-validation technique for the hyperparameters tuning and model selection as described in Section III-C for the digital RC and BiLSTM approaches. Tables 1 and 2 show the characteristics of the resultant models for RC and BiLSTM respectively. The optimal size of the reservoir was found to be $N = 300$. The digital RC model summarized in Table. 1 corresponds to the use of only the velocity features (V_x and V_y), hence $F = 2$. We note that $F = 3$ is another possibility when we run the nested cross-validation. The $T = 100$ here shows the already predefined number of discrete time-steps obtained after the preprocessing of the dataset making the input a $F \times T$ matrix. The optimal scaling factors for the input and feedback for the digital RC models are found to be $\alpha = 0.6$ and $\beta = 0.1$ respectively, and ϕ manually set to 0 rads since it only impacts the non-linearity of the system. For

TABLE 1. Resultant digital RC model architecture.

Layer type	Output shape	Trainable Parameter
Input	(N = 300 , T = 100)	0
Reservoir	(N = 300, T = 100)	0
Output	(C = 2, N = 300)	600

TABLE 2. Resultant BiLSTM model architecture.

Layer type	Output shape	Trainable parameters
Bidirectional	(T = 100, H = 32)	2432
Flatten (Flatten)	(3200)	0
Dense (Dense)	(D = 128)	409728
Dense	(1)	129

TABLE 3. The CNN model architecture.

Layer type	Output shape	Trainable parameters
conv2d (Conv2D)	(62, 62, 32)	608
MaxPooling2D	(31, 31, 32)	0
conv2d (Conv2D)	(27, 27, 128)	102528
MaxPooling2D	(13, 13, 128)	0
conv2d (Conv2D)	(11, 11, 64)	73792
Flatten (Flatten)	(7744)	0
Dense (Dense)	(128)	991360
Dropout	(128)	0
Dense	(1)	129

the BiLSTM, using the Adam optimizer to minimize binary cross-entropy loss function, the best resultant BiLSTM model is obtained with a learning rate of 10^{-3} with a sigmoid function. We obtain the size of the recurrent BiLSTM layer to be $H = 32$ and the dense layer responsible for final prediction has the size of $D = 128$. Table. 3 shows a CNN architecture not deduced through the model selection procedure used for RC and BiLSTM as described above but adapted from the model that already proved successful for Parkinson's disease detection in [25]. Notice the **Trainable parameters**' column showing the number of variables to be optimized through training for each layer.

A. COMPARISON ON PREDICTION ACCURACY

The hyperparameter tuning and model selection experiments results are listed in Table. 4. Comparing the best accuracies for each of the input feature combination, we find digital RC attained the best accuracy of 85% using the velocity features alone. This observation is in agreement with the results obtained for the k-Medoids method in [14], and we observed that it remains true for all other classification models under

TABLE 4. Feature combination results for the digital RC.

k	Parameters			Accuracy (%)
0	X	Y	-	83
1	X	Y	P	83
2	X	Y	Az	81
3	X	Y	Al	81
4	V_x	V_y	-	85
5	V_x	V_y	P	81
6	V_x	V_y	Az	83
7	V_x	V_y	Al	79
8	All parameters			77

TABLE 5. Comparison of accuracies for all models under consideration for the ES-AD task.

Method	Acc (%)	Sens (%)	Spec (%)
BiLSTM	88	96	79
Digital RC	85	88	83
CNN	83	75	91
k-Medoids [14]	74	75.6	72.2

consideration (i.e. CNN and BiLSTM). We also observed that the addition of more parameters to those of velocity adds noise and degrades digital RC performance. It is well-known that irrelevant features can hurt the overall accuracy [15], which can explain why using all features ($k = 8$) actually leads to the lowest accuracy in our experiments. We compare the best results of digital RC to those of other methods in Table. 5 where the performance metrics are defined as follows:

- Accuracy (**Acc**): The percentage of correctly classified individuals
- Sensitivity (**Sens**): The percentage of correctly classified Alzheimer patients
- Specificity (**Spec**): The percentage of correctly classified healthy control (HC) individuals.

The accuracy obtained using the BiLSTM counterpart is at 88% which is an improvement over the digital RC model. Notice that the recurrent methods, i.e., the digital RC and BiLSTM surpass both CNN and k-Medoids algorithms in accuracy. On that note, before we conclude that the BiLSTM is indeed the best model here, we will analyze the cost that comes with the extra gain (3% gain w.r.t RC) in accuracy of BiLSTM in the subsequent sections.

B. COMPARISON ON EFFICIENCY

Mobile devices such as smart-phones and tablets are not only expedient to collect the raw features as described in Sec. III-A, but they can at the same time be used for a fast and personalized diagnostic delivery. However, since mobile devices are limited by their battery discharge cycle, measuring the energy efficiency of the classification algorithms we consider is of paramount importance. With that in mind, we complement the results on accuracy described

TABLE 6. Energy consumption for hyperparameter tuning and model selection for the RC and BiLSTM.

Method	CO_{2eq} (kg)	Energy (kWh)	Duration (hours)
BiLSTM	0.521	9.312	44
Digital RC	0.065	1.156	21

TABLE 7. Estimates of FPO counts necessary for training.

Method	Trainable Parameters	Billion Floating Point Operations
CNN	1168417	1313.09
k-Medoids	750016	83.98
BiLSTM	412289	18.93
Digital RC	600	11.92

in Section IV-A by reporting the costs incurred to obtain them. To this end, we use the *experiment-impact-tracker* framework [52] to estimate the energy consumed in the nested-cross validation method for both digital RC and BiLSTM, i.e. the two best methods in terms of accuracy in Table. 5 (however, we will also elaborate on the complexity of the CNN and k-Medoids in the sequel). Table. 6 summarizes the electric energy in *kWh* consumed and the corresponding mass in kilograms of the CO_2 released by our optimization experiments carried out in France. The energy consumed by the BiLSTM approach is 9.312 kWh which is more than 8 times the 1.156 kWh consumed by the digital RC approach during 44 and 21 hours, respectively.

We further estimate the relative cost of training each model using the optimal parameters, obtained in Section IV-A by counting the number of trainable internal model variables and the number of Floating Point Operations (FPOs) carried on the CPU during the training. For RC, the trainable parameters are the weights in the readout matrix (5), while for BiLSTM and CNN it is the number of biases and weights of each neuron in the network. We obtain the results presented in Table. 7 when we train each of the models to convergence on the same hardware. Training the digital RC necessitates 63% of the number of FPOs required by the BiLSTM hinting a lower energy consumed in a single training of digital RC. The CNN consumed the most as a result of its increased depth and complexity compared to BiLSTM whereas for the k-Medoids, the most computationally expensive part is the computation of the distances between the loops with the DTW while clustering.

We also compare the number of FPOs for inference of the trained models for a single patient. The number of FPOs counted cover those from loading the saved model, loading the data and the whole process of inference to obtain a single prediction. Table. 8 shows that, even in inference, the digital RC and BiLSTM will consume the least with the RC requiring only 15.7% the number of FPOs for BiLSTM. The implication here being that RC requires fewer computations and lower energy consumption by the CPU for the predictions.

TABLE 8. Estimates of FPO counts necessary for inference.

Method	Million Floating Point Operations
CNN	79.8
k-Medoids	54
BiLSTM	8.64
Digital RC	1.36

TABLE 9. Estimated end-to-end energy consumption in the reservoir layer.

-	**Energy** (kWh)	**Acc** (%)	**Sens** (%)	**Spec** (%)
Digital RC	1.147×10^{-4}	85	88	83
Hardware RC	8.3×10^{-5}	83	92	74

The reason for higher gain in inference (15.7% w.r.t BiLSTM) compared to training (63% w.r.t BiLSTM) is the fact that training the RC necessitates multiple projections in the reservoir (see (6)), the inversion and multiplication of large matrices (5). These calculations account for most of the energy cost in training and their absence in inference drastically reduces the number of FPOs. This makes digital RC an even better candidate when implementing the models on mobile devices running on batteries (i.e the very tablets on which the patterns are drawn). Also, RCs will incur lower costs of running tests on a distant server in the cloud or edge computing.

C. HARDWARE PERFORMANCE

In Section IV-B we have given the FPOs count necessary on the processor for training the digital RC. Further analysis shows that 49% of these operations are spent in carrying the projection, that is in computing the reservoir states as described in (6). The rest of the operations are for the actual training by the ridge regression described by (5). We can save these FPOs by carrying the projection on the optoelectronic setup described in II-A2. In this case, only the regression part is carried by a processor. For a fair comparison, the hardware RC uses the same feature combination as the digital RC (i.e. the one indexed by k = 4 in Table. 4), both during the training and the inference phase. We estimate the energies consumed by the processor and the optoelectronic setup for the projection and compare them in Table. 9. The power consumed by the setup was computed by summing the energies consumed by the setup's constituent components. Although the estimated values for the hardware setup are rough estimates, they hint a lower energy cost incurred by the setup. There is a marginal reduction in accuracy by 2% that can be explained by either the noisier nature of the hardware experiments and/or the difficulties in fine-tuning the physical components.

V. CONCLUSION

In this paper, we have proposed the use of Artificial Neural Networks (ANNs) for Early-Stage Alzheimer Detection from the handwritten (HW) temporal data by studying the trade-off between accuracy and efficiency (number of parameters, number of FPOs and energy consumed). We found that BiLSTM and Reservoir Computing are the best approaches for the task, compared to alternative methods using k-Medoids or CNNs. Both methods showed an improvement in accuracy compared to state-of-the-art with the digital RC yielding an accuracy of 85% whilst that of BiLSTM is 88%, that is, an increase in 3% in accuracy for the BiLSTM. However, further analysis has shown that, with the digital RC, we incur significantly lower costs in optimization (8 times less energy), training (only 63% of FPOs) and inference (only 15.7% of FPOs) when compared to BiLSTM. The lower energy requirements for optimization and training makes RC the more efficient and the more environmentally friendly approach, especially when the small performance penalty is tolerable. Moreover, the digital RC's lower inference energy cost makes it ideal to run on the same mobile devices used to record the HW pattern running on battery power for a longer period of time in between recharges compared to running the more costy BiLSTM method. If even slightly lower performances can be tolerated in favor of lower energy cost, the hardware RC implementations can provide a good route to even greener solutions by reducing the computation load on power-hungry electronic processors. Future works will consider improved hardware RC architectures with even lower energy consumptions and better classification accuracies.

REFERENCES

[1] Alzheimer's Association, "2021 Alzheimer's disease facts and figures," *Alzheimer's Dementia, J. Alzheimer's Assoc.*, vol. 17, no. 3, pp. 327–406, 2021.

[2] S. Anthony, C. Pradier, R. Chevrier, J. Festraëts, K. Tifratene, and P. Robert, "The French National Alzheimer database: A fast growing database for researchers and clinicians," *Dementia Geriatric Cognit. Disorders*, vol. 38, nos. 5–6, pp. 271–820, 2014.

[3] G. McKhann, D. Drachman, M. Folstein, R. Katzman, D. Price, and E. M. Stadlan, "Clinical diagnosis of Alzheimer's disease: Report of the NINCDS-ADRDA work group under the auspices of department of health and human services task force on Alzheimer's disease," *Neurology*, vol. 34, no. 7, pp. 939–944, 1984.

[4] K. A. Johnson, S. Minoshima, N. I. Bohnen, K. J. Donohoe, N. L. Foster, P. Herscovitch, J. Karlawish, C. Rowe, M. Carrillo, D. D. M. Hartley, S. Hedrick, V. Pappas, and W. H. Thies, "Appropriate use criteria for amyloid pet: A report of the amyloid imaging task force, the society of nuclear medicine and molecular imaging, and the Alzheimer's association," *J. Nucl. Med., Soc. Nucl. Med.*, vol. 54, pp. 1–16, Jan. 2013.

[5] L. M. Shaw, J. Arias, K. Blennow, D. Galasko, J. L. Molinuevo, S. Salloway, S. Schindler, M. C. Carrillo, J. A. Hendrix, A. Ross, J. Illes, C. Ramus, and S. Fifer, "Appropriate use criteria for lumbar puncture and cerebrospinal fluid testing in the diagnosis of Alzheimer's disease," *Alzheimer's Dementia*, vol. 14, no. 11, pp. 1505–1521, Nov. 2018.

[6] M. T. Angelillo, F. Balducci, D. Impedovo, G. Pirlo, and G. Vessio, "Attentional pattern classification for automatic dementia detection," *IEEE Access*, vol. 7, pp. 57706–57716, 2019.

[7] J. H. Yan, S. Rountree, P. Massman, R. S. Doody, and H. Li, "Alzheimer's disease and mild cognitive impairment deteriorate fine movement control," *J. Psychiatric Res.*, vol. 42, no. 14, pp. 1203–1212, Oct. 2008.

[8] A. Schröter, R. Mergl, K. Bürger, H. Hampel, H.-J. Möller, and U. Hegerl, "Kinematic analysis of handwriting movements in patients with Alzheimer's disease, mild cognitive impairment, depression and healthy subjects," *Dementia Geriatric Cogn. Disorders*, vol. 15, no. 3, pp. 132–142, 2003.

[9] N.-Y. Yu and S.-H. Chang, "Kinematic analyses of graphomotor functions in individuals with Alzheimer's disease and amnestic mild cognitive impairment," *J. Med. Biol. Eng.*, vol. 36, no. 3, pp. 334–343, Jun. 2016.
[10] J. Garre-Olmo, M. Faúndez-Zanuy, K. López-de-Ipiña, L. Calvó-Perxas, and O. Turró-Garriga, "Kinematic and pressure features of handwriting and drawing: Preliminary results between patients with mild cognitive impairment, Alzheimer disease and healthy controls," *Current Alzheimer Res.*, vol. 14, no. 9, pp. 960–968, Aug. 2017.
[11] P. Werner, S. Rosenblum, G. Bar-On, J. Heinik, and A. Korczyn, "Handwriting process variables discriminating mild Alzheimer's disease and mild cognitive impairment," *J. Gerontol., Psychol. Sci.*, vol. 61, no. 4, pp. 228–236, 2006.
[12] J. Kawa, A. Bednorz, P. Stępień, J. Derejczyk, and M. Bugdol, "Spatial and dynamical handwriting analysis in mild cognitive impairment," *Comput. Biol. Med.*, vol. 82, pp. 21–28, Mar. 2017.
[13] M. A. El-Yacoubi, S. Garcia-Salicetti, C. Kahindo, A.-S. Rigaud, and V. Cristancho-Lacroix, "From aging to early-stage Alzheimer's: Uncovering handwriting multimodal behaviors by semi-supervised learning and sequential representation learning," *Pattern Recognit.*, vol. 86, pp. 112–133, Feb. 2019.
[14] C. Kahindo, M. A. El-Yacoubi, S. Garcia-Salicetti, A.-S. Rigaud, and V. Cristancho-Lacroix, "Characterizing early-stage Alzheimer through spatiotemporal dynamics of handwriting," *IEEE Signal Process. Lett.*, vol. 25, no. 8, pp. 1136–1140, Aug. 2018.
[15] P. Drotár, J. Mekyska, I. Rektorová, L. Masarová, Z. Smékal, and M. Faundez-Zanuy, "Decision support framework for Parkinson's disease based on novel handwriting markers," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 3, pp. 508–516, May 2015.
[16] H.-L. Teulings and G. E. Stelmach, "Control of stroke size, peak acceleration, and stroke duration in parkinsonian handwriting," *Hum. Movement Sci.*, vol. 10, nos. 2–3, pp. 315–334, May 1991.
[17] C. Taleb, M. Khachab, C. Mokbel, and L. Likforman-Sulem, "Visual representation of online handwriting time series for deep learning Parkinson's disease detection," in *Proc. Int. Conf. Document Anal. Recognit. Workshops (ICDARW)*, Sep. 2019, pp. 25–30.
[18] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," 2015, *arXiv:1506.00327v1[cs.LG]*
[19] M. J. Slavin, J. G. Phillips, J. L. Bradshaw, K. A. Hall, and I. Presnell, "Consistency of handwriting movements in dementia of the Alzheimer's type: A comparison with Huntington's and Parkinson's diseases," *J. Int. Neuropsychol. Soc.*, vol. 5, no. 1, pp. 20–25, Jan. 1999.
[20] M. J. U. Novak and S. J. Tabrizi, "Huntington's disease: Clinical presentation and treatment," in *Pathophysiology, Pharmacology and Biochemistry of Dyskinesia*, 1st ed. Toronto, ON, Canada: Academic, 2011, pp. 297–323.
[21] D. Impedovo and G. Pirlo, "Dynamic handwriting analysis for the assessment of neurodegenerative diseases: A pattern recognition perspective," *IEEE Rev. Biomed. Eng.*, vol. 12, pp. 209–220, 2019.
[22] N. Zhi, B. K. Jaeger, A. Gouldstone, R. Sipahi, and S. Frank, "Toward monitoring Parkinson's through analysis of static handwriting samples: A quantitative analytical framework," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 2, pp. 488–495, Mar. 2017.
[23] R. Damaševičius, O. Abayomi-Alli, R. Maskeliūnas, and A. Abayomi-Alli, "BiLSTM with data augmentation using interpolation methods to improve early detection of Parkinson disease," in *Proc. Federated Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, Sofia, Bulgaria, Sep. 2020, pp. 80–371.
[24] C. Taleb, L. Likforman-Sulem, and C. Mokbel, "Improving deep learning Parkinson's disease detection through data augmentation training," in *Proc. MedPRAI*, Istanbul, Turkey, 2019, pp. 79–93.
[25] C. R. Pereira, D. R. Pereira, G. H. Rosa, V. H. C. Albuquerque, S. A. T. Weber, C. Hook, and J. P. Papa, "Handwritten dynamics assessment through convolutional neural networks: An application to Parkinson's disease identification," *Artif. Intell. Med.*, vol. 87, pp. 67–77, May 2018.
[26] L. Larger, M. Soriano, D. Brunner, L. Appeltant, J. Gutiérrez, L. Pesquera, C. Mirasso, and I. Fischer, "Photonic information processing beyond Turing: An optoelectronic implementation of reservoir computing," *Opt. Exp.*, vol. 20, no. 3, pp. 3241–3249, Jan. 2012.
[27] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udaltsov, Y. K. Chembo, and M. Jacquot, "High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification," *Phys. Rev. X*, vol. 7, Feb. 2017, Art. no. 011015.
[28] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, and S. Massar, "Optoelectronic reservoir computing," *Sci. Rep.*, vol. 2, no. 1, pp. 1–6, Dec. 2012.
[29] Y.-C. Bo, P. Wang, and X. Zhang, "An asynchronously deep reservoir computing for predicting chaotic time series," *Appl. Soft Comput.*, vol. 95, Oct. 2020, Art. no. 106530.
[30] F. Da Ros, S. M. Ranzini, H. Bülow, and D. Zibar, "Reservoir-computing based equalization with optical pre-processing for short-reach optical transmission," *IEEE J. Sel. Topics Quantum Electron.*, vol. 26, no. 5, pp. 1–12, Sep. 2020.
[31] S. M. Ranzini, R. Dischler, F. Da Ros, H. Bülow, and D. Zibar, "Experimental investigation of optoelectronic receiver with reservoir computing in short reach optical fiber communications," *J. Lightw. Technol.*, vol. 39, no. 8, pp. 2460–2467, Apr. 15, 2021.
[32] R. Schwartz, J. Dodge, N. Smith, and O. Etzioni, "Green AI," *Commun. ACM*, vol. 63, no. 12, pp. 54–63, Dec. 2020.
[33] T.-J. Yang, Y.-H. Chen, J. Emer, and V. Sze, "A method to estimate the energy consumption of deep neural networks," in *Proc. 51st Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Oct. 2017, pp. 1916–1920.
[34] D. Patterson, J. Gonzalez, Q. Le, C. Liang, L.-M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, "Carbon emissions and large neural network training," Apr. 2021, *arXiv:2104.10350*.
[35] H. Jaeger, "The 'echo state' approach to analysing and training recurrent neural networks," German Nat. Res. Inst. Comput. Sci., Bonn, Germany, GMD-Rep. 148, Jan. 2001.
[36] W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural Comput.*, vol. 14, no. 11, pp. 2531–2560, Dec. 2002.
[37] F. Duport, A. Smerieri, A. Akrout, M. Haelterman, and S. Massar, "Fully analogue photonic reservoir computer," *Sci. Rep.*, vol. 6, no. 1, pp. 1–2, Mar. 2016.
[38] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer, "Information processing using a single dynamical node as complex system," *Nature Commun.*, vol. 2, no. 1, pp. 1–6, Sep. 2011.
[39] J. B. Héeroux, H. Numata, N. Kanazawa, and D. Nakano, "Optoelectronic reservoir computing with VCSEL," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Rio de Janeiro, Brazil, Jul. 2018, pp. 1–6.
[40] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, "Experimental demonstration of reservoir computing on a silicon photonics chip," *Nature Commun.*, vol. 5, no. 1, pp. 1–6, Mar. 2014.
[41] F. D.-L. Coarer, M. Sciamanna, A. Katumba, M. Freiberger, J. Dambre, P. Bienstman, and D. Rontani, "All-optical reservoir computing on a photonic chip using silicon-based ring resonators," *IEEE J. Sel. Topics Quantum Electron.*, vol. 24, no. 6, pp. 1–8, Nov. 2018.
[42] D.-Z. Yue, Y.-S. Hou, Z.-M. Wu, C.-X. Hu, Z.-Z. Xiao, and G.-Q. Xia, "Experimental investigation of an optical reservoir computing system based on two parallel time-delay reservoirs," *IEEE Photon. J.*, vol. 13, no. 3, pp. 1–11, Jun. 2021.
[43] G. Montavon, G. B. Orr, and K. R. Müller, "A practical guide to applying echo state networks," in *Neural Networks: Tricks of the Trade*, 2nd ed. Berlin, Germany: Springer, 2012, pp. 559–686.
[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Dec. 1997.
[45] P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri, "Exploiting the past and the future in protein secondary structure prediction," *Bioinformatics*, vol. 15, no. 11, pp. 937–946, Nov. 1999.
[46] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Montreal, QC, Canada, Jul. 2005, pp. 2047–2052.
[47] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[48] Y. Jeon and J. Kim, "Constructing fast network through deconstruction of convolution," in *Proc. NIPS*, Montreal, QC, Canada, vol. 31, 2018.
[49] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," Apr. 2017, *arXiv:1605.07678*.
[50] M. Assran, J. Romoff, N. Ballas, J. Pineau, and M. Rabbat, "Gossip-based actor-learner architectures for deep reinforcement learning," 2019, *arXiv:1906.04585*.
[51] M. Assran, J. Romoff, N. Ballas, J. Pineau, and M. Rabbat, "Gossip-based actor-learner architectures for deep reinforcement learning," Apr. 2017, *arXiv:1906.04585*.
[52] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau, "Towards the systematic reporting of the energy and carbon footprints of machine learning," Jan. 2020, *arXiv:2002.05651*.

[53] K. Ricke, L. Drouet, K. Caldeira, and M. Tavoni, "Country-level social cost of carbon," *Nature Climate Change*, vol. 8, no. 10, pp. 895–900, Sep. 2018.
[54] D. Terpstra, H. Jagode, H. You, and J. Dongarra, "Collecting performance data with PAPI-C," in *Tools for High Performance Computing*. Berlin, Germany: Springer, 2010, pp. 157–173.
[55] J. Wainer and G. Cawley, "Nested cross-validation when selecting classifiers is overzealous for most practical applications," *Expert Syst. With Appl., Int. J.*, vol. 182, Nov. 2021, Art. no. 115222.
[56] M. Stone, "Cross-validatory choice and assessment of statistical predictions (with discussion)," *J. Roy. Stat. Soc. B, Methodol.*, vol. 38, no. 2, pp. 111–147, 1976.

NICKSON MWAMSOJO received the B.E. degree in telecommunications and information technologies from the L'Institut National des Télécommunications et des Technologies de l'Information et de la Communication, Oran, Algeria, in 2018, and the M.S. degree in optical networks and photonic systems from Paris Saclay University, Paris, France, in 2019. He is currently pursuing the Ph.D. degree in neuromorphic photonic systems for information processing and transport at IP Paris, Palaiseau, France.

His research interests include the development of optical systems for information processing, artificial intelligence, machine learning, and reservoir computing.

FREDERIC LEHMANN (Member, IEEE) received the E.E. and M.S.E.E. degrees in microelectronics and signal processing from the Ecole Nationale Superieure d'Electronique et de Radioelectricite de Grenoble, Grenoble, France, in 1998, and the Ph.D. degree in electrical engineering from the National Polytechnical Institute, Grenoble, in 2002. From 1999 to 2002, he was a Research Engineer with ST Microelectronics. From 2003 to 2004, he was a Postdoctoral Researcher with the Laboratory for Analysis and Architecture of Systems, Centre national de la recherche scientifique, Toulouse, France. He is currently a Professor with the Télécom SudParis, Institut Polytechnique de Paris, Palaiseau, France. His research interests include communication theory, nonlinear signal processing, and statistical image processing.

MOUNIM A. EL-YACOUBI (Member, IEEE) received the Ph.D. degree from the University of Rennes, France, in 1996. He was with the Service de Recherche Technique de la Poste (SRTP), France, from 1992 to 1996. He was a Visiting Scientist with the Centre for Pattern Recognition and Machine Intelligence (CENPARMI), Montreal, Canada, and an Associate Professor with the Catholic University of Parana, Curitiba, Brazil, from 1998 to 2000. From 2001 to 2008, he was a Senior Software Engineer at Parascript, Boulder, Colorado. At SRTP and Parascript, he has developed handwriting recognition software for real-life automatic mail sorting, bank check reading, and form processing. Since June 2008, he has been a Professor at Telecom SudParis, Institut Polytechnique de Paris. His research interests include artificial intelligence, data science, machine learning, modeling human user data, especially behavioral signals like handwriting, voice, gesture and activity recognition, biometrics, e-health, smart agriculture, and smart cities.

KAMEL MERGHEM received the M.S. and Ph.D. degrees from the Université des Sciences et Technologies de Lille, in 2000 and 2011, respectively. In 2000, he joined the CNRS Laboratory for Photonics and Nanostructures. He is involved in the research on innovative photonics components for optical fiber telecommunication and data centers applications. From 2017 to 2019, he worked as a Research and Development Project Manager for 3SP Technologies in the development of new optoelectronic components for optical communication systems and LiDAR applications. He is currently an Assistant Professor with the Telecom SudParis, Institut Polytechnique de Paris. He has authored or coauthored more than 100 publications in peer-reviewed journals and international conferences. His research interests include optical frequency combs for optical transmission and photonic reservoir computing for optical signal processing.

YANN FRIGNAC (Member, IEEE) was born in Clermont-Ferrand, in 1975. He received the Engineering degree from the Ecole Nationale Supérieure de Physique, in 1999, and the Ph.D. degree in electronic and communications on fiber optics transmission system optimization from the Ecole Nationale Supérieure des Télécommunications, Paris, France, and Alcatel CIT. In 2006, was with the "Institut National des Télécommunication"s (now TSP) as an Associate Professor. In 2017, he was a Full Professor with the Electronics and Physics Department. He is currently working with Huawei Technologies France. He is the author or coauthor of about 60 papers and ten patents in the domain of WDM fiber optic transmission systems and recently has also investigated photonic circuits for neuromorphic processing.

BADR-EDDINE BENKELFAT (Senior Member, IEEE) received the master's and Ph.D. degrees in optics and signal processing from the University of Franche-Comté, Besançon, France, in 1981 and 1984, respectively. In 1990, he joined the Institut Mines-Telecom—Telecom SudParis (formerly called INT) as an Associate Professor and founded the Optics and Photonics Group. He is currently a Professor with the Electronics and Physics Department. His research interests include optoelectronic systems for data processing and optical devices for high-speed optical fiber communications.

ANNE-SOPHIE RIGAUD received the M.D. and Ph.D. degrees in neurosciences. She is currently a Geriatrician and a Psychiatrist. She is also a Professor in geriatrics with the University Paris 5. She heads the Geriatric Department, Broca Hospital and the Research Unit *EA*4468, including LUSAGE Laboratory. She has expertise in dementia and gerontechnology.