Artificial Intelligence In Medicine 135 (2023) 102459

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Research paper

Real-time detection of freezing of gait in Parkinson's disease using multi-head convolutional neural networks and a single inertial sensor

Luigi Borzì [a,*], Luis Sigcha [b,c], Daniel Rodríguez-Martín [d,e], Gabriella Olmo [a]

[a] Department of Control and Computer Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Turin, Italy
[b] Instrumentation and Applied Acoustics Research Group (I2A2), Universidad Politecnica de Madrid, Ctra. Valencia, Km 7, 28031 Madrid, Spain
[c] ALGORITMI Research Center, School of Engineering, University of Minho, 4800-058 Guimaraes, Portugal
[d] Sense4Care S.L., Cornellà de Llobregat, 08940 Barcelona, Spain
[e] Technical Research Centre for Dependency Care and Autonomous Living (CETPD), Universitat Politecnica de Catalunya, 08800 Vilanova i la Geltrù, Spain

ARTICLE INFO

Keywords:
Parkinson's disease
Freezing of gait
Activities of daily living
Wearable sensors
Accelerometer
Convolutional neural network
Deep learning

ABSTRACT

Background: Freezing of gait (FOG) is one of the most disabling symptoms of Parkinson's disease (PD), contributing to poor quality of life and increased risk of falls. Wearable sensors represent a valuable means for detecting FOG in the home environment. Moreover, real-time feedback has proven to help reduce the duration of FOG episodes. This work proposes a robust real-time FOG detection algorithm, which is easy to implement in stand-alone devices working in non-supervised conditions.
Method: Data from three different data sets were used in this study, with two employed as independent test sets. Acceleration recordings from 118 PD patients and 21 healthy elderly subjects were collected while they performed simulated daily living activities. A single inertial sensor was attached to the waist of each subject. More than 17 h of valid data and a total number of 1110 FOG episodes were analyzed in this study. The implemented algorithm consisted of a multi-head convolutional neural network, which exploited different spatial resolutions in the analysis of inertial data. The architecture and the model parameters were designed to provide optimal performance while reducing computational complexity and testing time.
Results: The developed algorithm demonstrated good to excellent classification performance, with more than 50% (30%) of FOG episodes predicted on average 3.1 s (1.3 s) before the actual onset in the main (independent) data set. Around 50% of FOG was detected with an average delay of 0.8 s (1.1 s) in the main (independent) data set. Moreover, a specificity above 88% (93%) was obtained when testing the algorithm on the main (independent) test set, while 100% specificity was obtained on healthy elderly subjects.
Conclusion: The algorithm proved robust, with low computational complexity and processing time, thus paving the way to a real-time implementation in a stand-alone device that can be used in non-supervised environments.

1. Introduction

Parkinson's disease (PD) is one of the most common neurodegenerative disorders, affecting more than 1% of individuals over the age of 60 [1]. Cardinal PD motor signs include rigidity, tremor at rest, bradykinesia (i.e. slowness of movement), and postural instability [2]. As a result, a reduction in the quality of life and an increase in the risk of falls are observed in the PD population [3].

At present, the Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDS-UPDRS) [4] represents the standard clinical scale for assessing motor and non-motor impairments in PD. However, the clinical evaluation of PD is commonly performed only during prescheduled medical visits, and this makes it difficult for neurologists to assess short-term variations of the patient's disability level and to plan proper therapy adjustments. Moreover, the scale scores are not always indicative of the patient's perception of their difficulties during activities of daily living (ADLs).

In this context, the combination of wearable sensors and machine learning (ML) techniques has been successfully employed for the monitoring of several motor aspects of PD [5]. Wearable sensors are cheap, lightweight, and unobtrusive, thus representing a feasible solution for objectively evaluating PD motor symptoms both in the laboratory and in the home environment [6–9]. Wearable technology has been employed for assessing bradykinesia [10–12], dyskinesia [13,14], postural

[*] Corresponding author.
E-mail addresses: luigi.borzi@polito.it (L. Borzì), luisfrancisco.sigcha@upm.es (L. Sigcha), daniel.rodriguez-martin@upc.edu (D. Rodríguez-Martín), gabriella.olmo@polito.it (G. Olmo).

https://doi.org/10.1016/j.artmed.2022.102459
Received 5 May 2022; Received in revised form 17 November 2022; Accepted 18 November 2022
Available online 24 November 2022
0933-3657/© 2022 Elsevier B.V. All rights reserved.

stability [15–17], and tremor [18]. Moreover, some studies have implemented ML methods for predicting single MDS-UPDRS items related to motor symptoms [19] or a set of MDS-UPDRS motor scores [20].

Freezing of gait (FOG) represents one of the most troublesome symptoms of PD [3,21], affecting gait in more than 50% of patients [22]. FOG is defined as a "brief, episodic absence or marked reduction of forward progression of the feet despite having the intention to walk" [21]. It manifests in different forms, including shuffling steps, trembling legs, or complete akinesia [23]. The duration of FOG events is variable, with 50% of episodes lasting less than 5 s and 90% less than 20 s [24]. Some situations are known to represent a possible trigger factor of FOG, including turning, gait initiation, managing narrow spaces, and negotiating obstacles [23]. Moreover, cognitive challenges (e.g. dual tasking) [25] and emotional stress (e.g. anxiety) [26] may affect the manifestation of this motor sign. It has been demonstrated that FOG increases the risk of falls and is an early predictor of shortened survival in PD [27].

The clinical assessment of FOG is challenging, due to the sporadic nature of the symptom, the therapy effectiveness, and the patient's attention to gait. Moreover, situations such as medical consultations may inhibit the manifestation of FOG. For these reasons, FOG seldom occurs during a brief medical examination. Consequently, FOG assessment is mainly based on patients' diaries and responses to questionnaires, which are highly subjective and not very reliable sources [28]. The clinical observation of the phenomenon may be improved using triggering factors, such as asking the patient to perform cognitive or motor dual tasks [29,30], or suppressing therapy. However, these approaches are time-consuming and not compatible with everyday clinical practice. Finally, specific spatial–temporal gait parameters (e.g. stride time, step amplitude, and gait variability) degrade progressively as the FOG event is approached, raising the opportunity to recognize typical pre-FOG periods [31]. These are specific movement patterns occurring during gait just before FOG episodes. Prediction and timely detection of FOG may allow the adoption of corrective strategies to prevent FOG, such as the administration of external sensory cueing [32].

From these considerations, it is clear that FOG assessment can be improved using objective data, collected continuously during ADLs [33, 34] using wearable sensors. In the last 15 years, wearable technology has been widely employed for the automatic detection of FOG episodes [7]. The employed sensors include commercial [35] or prototype inertial measurement units (IMUs) [36], smartphones [37,38], and single accelerometers and/or gyroscopes [6,39], sometimes in combination with surface electromyography [40].

This research aims to assess the possibility of implementing a prevention strategy for FOG occurrence. To this end, an algorithm that can detect FOG in real time is proposed. Computational complexity, memory requirement, and testing time were calculated for assessing the possibility of an on-board implementation of the proposed solution.

The rest of this paper is organized as follows. Related work is discussed in Section 2, together with the main contributions of this work. Section 3 describes the data used in this study, pre-processing procedures, classification algorithm, post-processing of the obtained results, and computational complexity analysis. The results are reported in Section 4 and discussed in Section 5. Finally, conclusions are drawn in Section 6, along with future prospects.

2. Related work

Automatic FOG detection based on inertial sensors has received increasing attention over the past two decades. Various sensors, sensor configurations, experimental procedures, and classification algorithms have been proposed, providing incremental improvements in detection performance. Tri-axial accelerometers have been used alone [41,42], in combination with tri-axial gyroscopes [43–45], or as part of multimodal systems that include electromyography [46] or plantar pressure sensors [47]. The number of inertial sensors varies from 1 [38,41,43] to 6 [48], and several sensor locations have been proposed, including the wrist [44], lower back [38], waist [41,43], thigh [42], shank [45,46], or a combination thereof [49].

Experimental procedures include a wide variety of gait tasks. Some studies addressed the timed-up-and-go test [30,46], which includes gait initiation and turns. In other studies, self-designed protocols included walking [45], along with turns and stops to elicit FOG [42,47]. Finally, few studies have included ADL-like activities besides free-like walking tasks [34,43]. Regarding sample size, most of the published works have enrolled a population ranging from 7 to 12 PD patients [30,42,45–47,49], with few studies addressing larger samples of 21 [34,41,43] and 38 [38] patients.

A wide variety of FOG detection algorithms have been proposed. Early approaches were based on the calculation of some indices that characterized FOG and the use of a simple threshold to distinguish FOG from other activities [50,51]. The freeze index [50] represents the first index of FOG described in the literature and is evaluated as the ratio of the power in the freezing band (i.e. 3–8 Hz) to that in the locomotion band (i.e. 0–3 Hz).

Subsequent works aimed to improve classification performance by extracting more features and using ML algorithms such as support vector machine [31,34,38] and k-nearest neighbor [49]. Finally, deep learning (DL) was proposed to detect FOG, outperforming classical ML models [41,43,44]. DL algorithms offer the advantage of automatic feature extraction directly from raw input data, without requiring any feature engineering. This reduces the effort devoted to defining hand-created features and avoids user errors. In addition, DL approaches can automatically identify hidden features, providing a more in-depth representation of the input data. On the other hand, a large amount of data is required to use these methods correctly.

Recent studies have exploited the high potential of DL algorithms to perform FOG detection and prediction tasks. In particular, convolutional neural networks (CNNs) [41,44,45], long short-term memory networks (LSTMs) [43,47], and deep autoencoders [42] have achieved good to excellent performance in FOG detection, with sensitivity ranging from 0.63 [45] to 0.92 [43] and specificity ranging from 0.75 [34] to 0.98 [45].

The main limitations of the cited studies include the small cohort of PD patients, the lack of external data sets, the use of laboratory environments and supervised experiments, the high computational complexity of the designed classification algorithms, and/or the unsuitability for real-time implementations. The present study aims to overcome these limitations by developing a robust and lightweight algorithm intended for real-time FOG detection. The main contributions of this paper are summarized as follows.

Acceleration data from a large number of subjects (118 PD patients and 21 healthy elderly subjects) were included in this study, with more than 1000 FOG episodes recorded. Analyses were performed on three different data sets, with two employed as independent test sets. This allows us to test the generalization ability of the detection algorithm when processing data that include a wide variety of gait and activity patterns.

Data were collected under unsupervised conditions, during ADL-like activities. The availability of labeled activities provided insight into the context in which false-positive events were detected.

The detection algorithm can work in real-time. It uses a single time window, with no pre-processing of the entire signal and no information about past or future data. Moreover, computational complexity, testing time, memory requirements, and detection latency were carefully estimated.

Extensive post-processing procedures were performed to provide a comprehensive analysis of the effectiveness in detecting FOG and discarding other activities.

The effect of the activity threshold on the performance of the detection system and the percentage of discarded windows was evaluated. An effective tool to exclude data from the classification process would help reduce computational load and increase battery life.

3. Material and methods

3.1. Study design

It is well known that a large amount of data is needed to train ML and especially DL algorithms and implement a robust classification model. In addition, PD patients have high inter-subject variability in terms of gait patterns, symptom severity, and FOG manifestations. Therefore, data sets including a large number of patients are needed to validate the model. In addition, a large and varied number of FOG episodes, possibly from multiple subjects, is needed to avoid overfitting. In this work, three data sets comprising a total number of 118 PD patients were used to train, validate, and test the performance of the model (Section 3.2).

To implement a real-time algorithm, minimal pre-processing is required to make the calculation as fast as possible. In this work, only the removal of the mean value was performed before classification (Section 3.3). In addition, it is necessary to reduce memory usage and thus use a minimum amount of data for the testing phase. In particular, by limiting the analysis to only the current time window of the data being tested and avoiding the processing of previous windows, significant memory savings are achieved. As for signal segmentation, short time window length and slide are preferred to reduce latency in FOG detection. However, too short time windows may not be able to capture signal features that can distinguish FOG from other activities. In this study, a 2-s window with a 0.5-s advance was used (Section 3.3).

As for the classification algorithm, it must be light and fast to reduce memory usage and speed up calculation. High sensitivity is needed to capture as many FOG episodes as possible, but also high specificity to avoid too many false positives. Finally, latency in FOG detection is a key issue for a real-time algorithm, so it must be tightly controlled. DL models are a suitable choice, as they can capture hidden features and exploit a large amount of available data. In this work, a CNN with a small number of layers was designed (Section 3.4).

3.2. Data

This section describes the three data sets used in this study. Detailed information about the main database is provided in Section 3.2.1. Only binary class labels (i.e. FOG, not FOG) are available for this data set. Given a large number of recorded FOG episodes, it was used for training, validation, and testing of the classification model. The additional data sets are described in Sections 3.2.2 and 3.2.3. Both databases have the advantage of including class labels for different tasks. The former includes a large amount of gait data and some episodes of FOG. In the second, although no FOG episodes were recorded during the data collection procedures, several ADLs were performed during the experiments. Therefore, the latter data set was included to test the robustness of the FOG detection algorithm to false positives, which is of paramount importance for a real-life-oriented detection tool. Both data sets were used in this study as independent test sets. A summary of the characteristics of the three data sets is given in Table 1, along with the list of labeled activities.

3.2.1. REMPARK dataset

The dataset [34] includes data from 21 PD patients. Inclusion criteria were a clinical diagnosis of PD with motor symptoms, a Hoehn and Yahr (H&Y) stage greater than 2 in the OFF state of therapy, the absence of dementia or visual impairment that prevented participants from performing required tasks, and a FOG questionnaire score (FOG-Q) greater than 6. Subjects who required walking assistance (e.g. walking stick, crutch) were included in the study. The experiments were conducted in the patients' homes. Data were recorded both under (ON) and not under (OFF) dopaminergic therapy. In detail, the sample included 18 males and 3 females, with an age of 69.3 ± 9.7, disease duration of 9 ± 4.8, H&Y score of 3.1 ± 0.4, FOG-Q of 15.8 ± 4.1, Mini-Mental State Examination (MMSE) of 27.8 ± 1.9, and MDS-UPDRS part-III total of 16.2 ± 9.7 in ON and 36.3 ± 14.4 in OFF. The tasks performed included walking tasks (e.g. showing the house, stand up and go test, and walking outdoors) and some tasks designed for false-positive analysis (e.g. brushing teeth, painting/drawing/deleting on a sheet of paper, and cleaning windows). Acceleration data were recorded with an IMU attached on the left side of the waist (Fig. 1) through an elastic band and stored locally on the device. The sensor range was set to ± 6 g and the sampling rate to 200 Hz, with data subsequently resampled to 40 Hz. During the experiments, 9.1 h of inertial data were recorded, including 93 min of FOG.

Table 1
Characteristics of the datasets used in this study.

Dataset	REMPARK	6MWT	ADL
Subjects (% male)	21 (86%)	38 (75%)	59 (63%)
Signal duration	9.1 h	2.4 h	5.9 h
FOG duration	93 min	5.3 min	0
# FOG episodes	1058	52	0
Labeled activities	FOG Non-FOG	FOG Gait Stance	Gait Stance Sit Sit-to-stand Stand-to-sit toe tapping (UPDRS item 3.7) Leg agility (UPDRS item 3.8) Retropulsion test (UPDRS item 3.12)

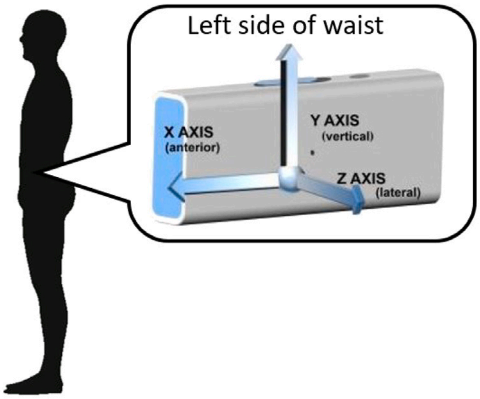

Fig. 1. Sensor position and axes orientation in the REMPARK data set. The front, vertical, and lateral (left) direction corresponds to the *x*-axis, *y*-axis, and *z*-axis of the sensor reference system, respectively.

3.2.2. 6MWT dataset

The dataset [30,33] includes data from 38 patients with PD and 21 control subjects. The inclusion criteria for the sample of PD patients were a clinical diagnosis of PD with motor symptoms (with or without a history of FOG events) and no major comorbidities or visual/cognitive impairments that prevented them from performing the required tasks. Subjects who required assistance in ambulation were included in the study. The experiments were conducted during pre-scheduled outpatient visits, and all PD participants were in a daily ON state, i.e. they had taken their usual dose of medication and a variable amount of time had elapsed since then. The sample included 28 males and 10 females, with an age of 70.7 ± 8.2, disease duration of 9 ± 4.8, and H&Y score of 2.5 ± 0.8. Inclusion criteria for controls were the absence of clinically evident signs of parkinsonism, severe visual impairment, dementia, and other significant neurological disorders. Subjects who required walking assistance were included in the study. The control sample included 7 males and 14 females, with an age of 85.6 ± 7.2. Being enrolled in a nursing home, the age of the control subjects was

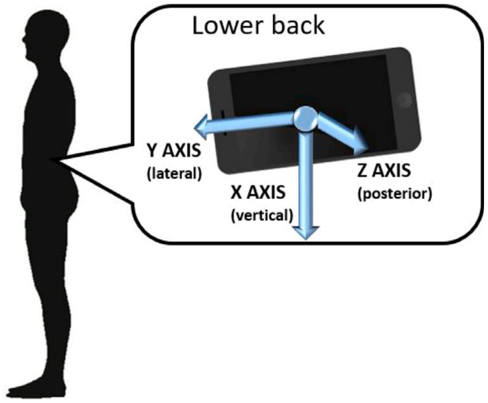

Fig. 2. Sensor position and axes orientation in the independent data sets (6MWT, ADL). Vertical, lateral (left), and posterior direction corresponds to the x-axis, y-axis, and z-axis of the sensor reference system, respectively.

significantly higher than that of the PD patients, and this could provide challenging gait patterns in terms of gait pattern and turning speed. Participants were asked to perform the 6-min walking test (6MWT), which consists of walking back and forth along a 10-m corridor for 6 min at their preferred pace. Data from a 3-axis accelerometer and 3-axis gyroscope were recorded with a smartphone mounted on the lower back via an elastic band (Fig. 2). A range of ± 2 g and 2000 dps was used for the accelerometer and gyroscope, respectively, and a sampling rate of 200 Hz was selected. Inertial data were stored locally in the smartphone. During the experiments, 2.4 h of inertial data were recorded from PD patients, including 97.6 min of gait, 17.4 min of stance, and 5.3 min of FOG. Additional 1.4 h of data were recorded from control subjects, including 72 min of gait and 4 min of stance.

3.2.3. ADL dataset

The dataset [17] includes data from 59 PD patients. Inclusion criteria were a clinical diagnosis of PD with motor symptoms, no major comorbidities, or visual/cognitive impairments that prevented them from performing required tasks. Subjects who required walking assistance were included in the study. All participants were in daily ON condition. The sample included 37 males and 22 females, with an age of 69.2 ± 10.2 years, disease duration of 6.7 ± 5.3 years, and H&Y score of 2.14 ± 0.8. The same sensor configuration as in the 6MWT data set was used. Experiments were conducted during pre-scheduled medical examinations, and participants were asked by physicians to perform various tasks, including walking freely, turning with different angular amplitudes, standing up, sitting down, standing for several seconds, and other tasks required for MDS-UPDRS assessment. These tasks, performed under semi-supervised conditions, are quite representative of activities performed in the home environment. A total of 5.9 h of inertial data were recorded during the experiments, including 32.8 min of walking, 40.2 min of stance, and 13.5 min of postural transitions (i.e. stand up, sit down), while the remaining tasks included items related to the MDS-UPDRS assessment and unlabeled activities.

3.3. Pre-processing

Raw data from the main data set (REMPARK) were segmented using sliding windows of a fixed length of 2 s. Since the data set is inherently unbalanced due to the different proportions of FOG and non-FOG instances, a differential segmentation process was performed to generate the training and validation sets. The segmentation procedure, shown in Fig. 3, consists of using different overlaps for FOG and non-FOG data. Specifically, overlaps of 50% (1-s advance) and 87.5% (0.25-s advance) were used for non-FOG and FOG data, respectively. Windows that included only non-FOG data were labeled as non-FOG, windows that included at least 50% FOG were labeled as FOG, and the remaining windows were discarded.

The result of the segmentation process is shown in Fig. 4 (right), in terms of the proportion of FOG and non-FOG instances. As evident from Fig. 4 (left), the use of a fixed-length overlap generates an unbalanced training set, with 75% non-FOG and 25% FOG instances. In contrast, the implemented differential segmentation procedure provided a balanced distribution of non-FOG and FOG data, 52% and 48%, respectively. As for test set generation, segmentation was performed using a fixed overlap of 75% (0.5-s advance), which resembles the real working condition of the FOG detection system (every 0.5 s the algorithm processes the data of the previous 2 s). After segmentation, the removal of the mean value was performed on each window separately to allow the classification model to work properly with centered data.

As for the independent data sets (6MWT, ADL), the axis orientation and sampling rate were different from those of the main data set. Therefore, the following procedures were performed to obtain uniform data. First, the data were resampled at 40 Hz, under-sampling the original data collected at 200 Hz. Then, the axis order was adjusted to match that of the main database.

After resampling and reshaping the data, segmentation was performed using 2-s sliding windows with 75% overlap (0.5-s advance), as done for the main data set test set. Finally, the mean value was removed from each window separately.

To prepare the data for subsequent optimization and testing procedures, the entire REMPARK data set was initially divided into training, validation, and test sets, including 12, 4, and 5 patients, respectively (about 60% for training, 20% for validation, and the remaining 20% for testing). The subsets were generated so that patients in the training and test sets had similar characteristics in terms of age, duration of symptoms, H&Y, MMSE, and MDS-UPDRS-III, while PD patients with more severe FOG were assigned to the test set. This is a conservative situation, which is useful for testing the generalization ability of the algorithm when data from patients with severe walking problems are input into the classification model.

Optimization of the classification model architecture, parameters, and learning process settings was performed using the training and validation sets. Then, the resulting optimized model was tested on the test set and further tested on additional independent data sets.

3.4. Classification algorithm

CNNs can learn a high level of abstraction and features from large data sets by applying convolution operations to the input data. In fact, CNNs exploit three important ideas: sparse interactions, parameter sharing, and equivariant representations [52]. CNNs are capable of automatically extracting features from images and signals and have achieved state-of-the-art results in image classification, speech recognition, and text analysis. When applied to time series classification such as human activity recognition, CNNs have some advantages over other models, including local dependence and scale invariance [53].

The layers included in the proposed CNN architecture are listed and described below.

Convolutional layer (1D-CNN). Given a one-dimensional (1D) signal of length m, convolutional layers perform the convolution between the signal and a number n_f of filters of size f, sliding with a stride s. The generated output has dimensions $\left(\frac{m-f}{s}+1,\ n_f\right)$. Both the f weights and the bias term of each filter are learned during the training stage. In this study, n_f and f were tuned in the range 4–32 and 3–39 respectively, while s was set to 1.

ReLU activation function. Rectified linear unit (ReLU) represents the most common activation function in CNNs, increasing non-linearity and speeding up the computation. It is defined in the $[0, +\infty]$ interval, and computes the output as $a = max(0, z)$, where z is the input value.

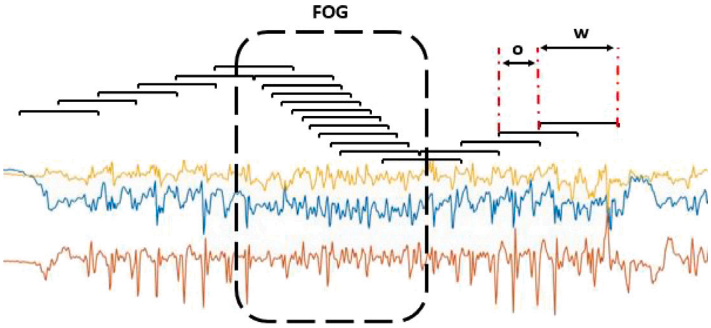

Fig. 3. Differential segmentation process used for training and validation set generation. Window size (w) and overlap (o) are different during FOG and other activities.

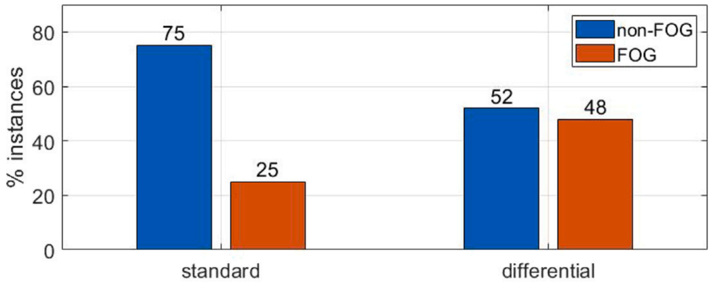

Fig. 4. Proportion of FOG using standard segmentation (left) and class balancing obtained using differential segmentation (right).

Pooling layer. It provides a reduction in the size of the representation generated by the convolutional layer. The main advantages include speeding up the computation and summarizing the presence of features in patches of the feature map. The pooling layer applies a filter of size p with a stride s to the input data of dimensions (d, n_f), generating an output of dimensions $(\frac{d-p+1}{s}, n_f)$. While max-pooling outputs the maximum value of the f values, average-pooling computes the mean of these values. In this study, both methods were implemented, and the one providing the best performance was finally selected. Moreover, p and s were tuned in the range 2–3.

Flatten layer. It consists in unrolling the multidimensional matrix of dimensions (d, n_f) into a 1D vector of size $(1, d \cdot n_f)$. It provides a mechanism to adapt the outputs of a CNN layer to dense layers. Each unit of the layer represents a neuron, which is then connected to every neuron of the subsequent dense layer.

Dense layer. It represents the fully connected layer of the network, in which each neuron is connected to every neuron of the preceding flattened layer. Given a number of neurons d in the preceding layer and n_n in the dense layer, the number of parameters required for the computation is $d \cdot n_n + n_n$, where $d \cdot n_n$ accounts for the neuron weights and n_n for the neuron bias. Both the n_n weights and the bias term of each filter are learned during the learning stage of the algorithm. In this work, the number of dense layers was tuned in the range 1–3 and n_n in 16–128.

Softmax layer. It represents the classification layer, in which the final continuous output of each neuron $\sigma(z_i)$ is computed from the input vector z, as shown in Eq. (1). Given a number of classes k, the normalization term $\sum_{j=1}^{k} e^{z_k}$ ensures that all the output values of the function will sum up to 1, thus representing a valid probability distribution.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \tag{1}$$

Moreover, the following regularization methods were used to avoid over-fitting and improve the generalization capability of the classification algorithm.

Dropout. Dropout regularization consists in randomly removing a given percentage of units (1 - dropout rate), thus training the CNN with a smaller number of neurons. It reduces over-fitting and increases the generalization capability of the classification model. In this work, dropout was used after each convolutional layer, with the dropout rate tuned in the range of 0.2–0.8.

Regularization. Similarly to dropout, regularization aims to reduce the variance, hence the over-fitting. In this study, $L2$ regularization was used in the softmax layer, updating the general cost function by adding an additional term $\frac{\lambda}{2m} \cdot \sum \|\omega\|^2$, where λ is the regularization term, m the input dimension and ω represents the weights vector. The effect of $L2$ is the reduction of the connection weights. In this work, λ was tuned in the range 0.1–0.001.

The simplest CNN architecture includes a convolutional layer followed by a pooling layer, successive flatten and dense layers, and finally a softmax classification layer. From this configuration, additional convolutional, pooling, and dense layers were iteratively added to the architecture. In addition, different CNN heads were used to achieve different spatial resolutions in the analysis of input signals in order to capture useful features from the local to the global level. For each configuration, a grid search procedure was used to optimize hyperparameters (e.g. number of filters and kernel size for convolutional layers; type, size, and stride for pooling layers; number of neurons for dense layers). Finally, the dropout rate and regularization parameters were adjusted to optimize training-validation performance. The area under the receiver operating characteristic curve (AUC) and memory requirements were monitored during the validation phase in order to identify the architecture that provides the best performance without significantly increasing the computational load. Specifically, an increase in AUC of at least 1 percent was required if computational complexity was increased, while no threshold was set if computational complexity was reduced.

To enable the learning process of the classification model, some preliminary settings were adjusted, as reported below.

Learning rate. The learning rate α is one of the most important parameters in the learning process, as it controls the size of the learning step at each iteration as it moves towards the minima of the loss function. The larger α, the faster the update of the model weights and the learning process. However, an excessive α can lead to divergence of the solution. On the other hand, a very small α avoids divergence, but slows down the calculation and can lead to convergence to local minima. In this study, α was adjusted in the range of 0.1–0.0001, observing the training-validation loss learning curve to ensure proper model training and solution convergence.

Batch size. Training the model with mini-batches represents a compromise between the gradient descent (GD) batch algorithm and the stochastic GD. In the former approach, all data are passed to the network at once, while in the latter only one element at a time is passed to the network during the learning process. The first approach makes use of the vectorization of the input data, but it is very slow, as all input data is needed to perform a learning step. The second approach accelerates the movement towards the minimum loss function but loses the speed provided by vectorization. The GD mini-batch consists of dividing the input data into $\frac{m}{b_s}$ batches, where m is the size of the input data and b_s is the size of the mini-batch. Training is then performed using a single batch in each learning phase. In this study, the mini-batch size was tuned in the range 64–1024.

Number of epochs. The number of epochs defines the maximum number of iterations the model undergoes before the training process stops. A small number of iterations can lead to poor performance (under-fitting), while too many iterations can cause over-fitting of the model. In this study, the maximum number of iterations was tuned in the range of 20–250.

Early stopping. To avoid over-fitting and reduce unnecessary calculations during training, early stop conditions were defined in the learning process. Specifically, training was stopped when a decrease of at least 0.001 in validation loss was not observed for at least 5 iterations.

Optimizer. Adaptive moment estimation (Adam) [54] was used in this study for the optimization of weights and biases of the neural network. It combines both the Momentum and the RMSprop GD algorithms, thus it is very effective and commonly used in deep neural network architectures. First, Momentum GD computes gradients $d\omega$ at each iteration for the t_{th} mini-batch. Then, it computes $V_{dw} = \beta_1 V_{d\omega} + (1 - \beta_1) d\omega$, with $\beta_1 = 0.9$. Finally, weights are updated with $\omega = \omega - \alpha V_{d\omega}$. RMSprop works in a similar way, computing $S_{dw} = \beta_2 S_{d\omega} + (1 - \beta) d\omega^2$, and finally updating weights as in Eq. (2), where ϵ is 10^{-8} and β_2 is set to 0.999.

$$\omega = \omega - \alpha \frac{d\omega}{\sqrt{S_{d\omega} + \epsilon}} \tag{2}$$

Adam first computes $V'_{d\omega} = \frac{V_{d\omega}}{1-\beta_1^t}$ and $S'_{d\omega} = \frac{S_{d\omega}}{1-\beta_2^t}$, and finally updates weights ω as in Eq. (3), where α is the learning rate.

$$\omega = \omega - \alpha \frac{V_{d\omega'}}{\sqrt{S'_{d\omega} + \epsilon}} \tag{3}$$

Loss function. For the classification task, the categorical cross-entropy loss function was used in this study. It is defined in Eq. (4), where N is the total number of samples, y_i is the ith class label, and $\bar{y}_i$ is the ith prediction.

$$E = -\frac{1}{N} \sum_{i=1}^{N} y_i \cdot log(\bar{y}_i) + (1 - y_i) \cdot log(1 - \bar{y}_i) \tag{4}$$

Table 2
Range of values and steps used for the optimization of the model architecture, model training, and regularization parameters. Some parameters were tuned using the automatic grid-search optimization procedure (top), while others were manually adjusted (bottom).

Parameter	Range	Step
# filters	4–32	4
Filter size	3–39	3
Pool size	2–3	1
Pool stride	2–3	1
# neurons	[16, 32, 64, 128]	–
# dense layers	1–3	1
# convolutional heads	1–3	1
Pool type	[average, max]	–
Dropout rate	0.2–0.8	0.1
Regularization term	[0.001, 0.01, 0.1]	–
Learning rate	[0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1]	–
Batch size	[64, 128, 256, 512, 1024]	–
# training epochs	20–250	10

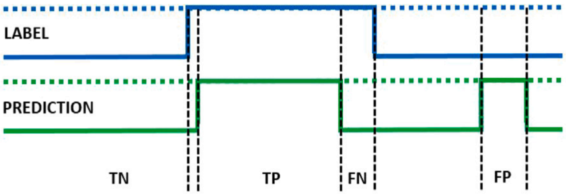

Fig. 5. Definition of true (TP) and false positives (FP) and true (TN) and false negatives (FN). In the first (second) segment, both prediction and label indicate non-FOG (FOG), thus it represents TN (TP). In the third segment, the label indicates FOG while the prediction is non-FOG, thus it represents an FN. Finally, in the last segment, the prediction indicates FOG while the label is non-FOG, thus it is an FP.

The range of values and step size used for parameter optimization are shown in Table 2. While some parameters were optimized using the grid search tuning process (e.g. number of filters, filter size, pool size, pool step size, and number of neurons in dense layers), others were manually adjusted to ensure a proper training process and increase performance. The parameter range and respective step size were selected considering both studies focusing on human activity recognition tasks [55,56] and works in the literature developing FOG detection algorithms [41,43]. However, in some cases, the range was limited to control for model complexity. In particular, the upper limit for the number of filters, the number of dense layers, the number of neurons in dense layers, and the number of convolutional heads was limited to 32, 3, 128, and 3, respectively.

3.5. Performance evaluation metrics

A comprehensive performance evaluation procedure was implemented to evaluate the classification results of the proposed model. True positives (TP) are defined by the windows of FOG correctly identified by the model. False positives (FP) represent windows of data corresponding to activities other than FOG identified by the model as FOG. False negatives (FN) correspond to real FOG windows not recognized by the algorithm. Finally, true negatives (TN) represent correctly classified non-FOG instances. Fig. 5 schematically describes these metrics.

Sensitivity/recall (Eq. (5)) assesses how many FOG windows are recognized by the model. Specificity (Eq. (6)) measures how efficiently non-FOG samples are discarded. Accuracy (Eq. (7)) is an overall performance evaluation metric that provides the percentage of correct classification. The geometric mean of sensitivity and specificity (Eq. (8)) is useful for evaluating situations in which one measure is much smaller than the other. The F-score (Eq. (9)) is calculated as the harmonic mean between sensitivity and precision, with precision calculated as in Eq. (10). In the case of unbalanced data sets, F-score is preferred to accuracy as the global correct classification metric. In addition, the receiver operating characteristic (ROC) curve was calculated for each data set. Finally, the AUC and equal error rate (EER) were computed. The former measures the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The second corresponds to the error observed at the point on the ROC curve where sensitivity equals specificity.

$$sensitivity = \frac{TP}{TP+FN} \tag{5}$$

$$specificity = \frac{TN}{TN+FP} \tag{6}$$

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

$$geometric-mean = \sqrt{sensitivity \cdot specificity} \tag{8}$$

$$F-score = \frac{2 \cdot sensitivity \cdot precision}{sensitivity + precision} \tag{9}$$

$$precision = \frac{TP}{TP+FP} \tag{10}$$

3.6. Post-processing

To provide further details on the prediction performance of the classification algorithm, some post-processing procedures were performed using predictions and class labels. As for the REMPARK data set, only a binary class label was available, equal to 1 or 0 in the case of FOG or non-FOG instances, respectively. As for the other data sets, several class labels were available, including gait and stance for both the 6MWT and ADL datasets, with the latter also including labels for sitting, standing, and some MDS-UPDRS-related tasks.

The following measures were calculated for all data sets. First, the percentage of FOG episodes detected was calculated as the number of actual FOG episodes in which at least one data window was classified as FOG. The percentage of FOG windows detected within each episode is a complementary measure to the percentage of FOG episodes detected. It was calculated as the percentage of true FOG windows classified as FOG by the algorithm in each episode (Fig. 6B). As for false FOG epiosdes, their number, duration (Fig. 6D), and distance from true FOG (Fig. 6C) were computed for the data sets in which FOG data were available. Finally, the FOG detection latency represents the temporal resolution in the detection of FOG episodes. It was computed as the difference between the onset of the actual FOG episode and the onset of the detected FOG episode (Fig. 6A). This measure is expressed in seconds and can be negative or positive, depending on the prediction or delay in the detection of the FOG episode. Regarding the data sets for which the activity label was provided, further analysis was performed to evaluate FPs. Specifically, the number of false FOG episodes detected was counted for each activity. This is important to assess which activities are most misclassified by the algorithm.

3.7. Computational complexity

The test time was computed for different input data sizes. It was calculated separately for the pre-processing stages (i.e. reshaping of data and removal of the mean) and for the classification stage. In addition, to test the feasibility of implementing the algorithm in a stand-alone device, memory requirements for storing the input data and classification model parameters were computed. For further analysis of computational complexity, the calculation of floating-point operations (FLOPs) was performed on the CNN model.

Finally, to provide an estimate of battery consumption, the computational complexity and processing time of the proposed algorithm were compared with those of the detection model currently incorporated in the commercial STAT-ON monitoring device [57].

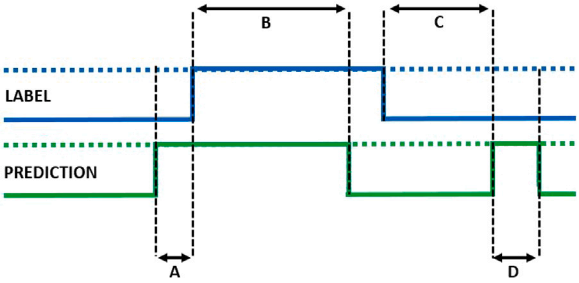

Fig. 6. Schematic of the measures computed for post-processing analysis. A: prediction time; B: amount of FOG detected in the episode; C: distance between false FOG episode and the nearest real FOG episode; D: duration of false FOG episode.

3.8. Activity threshold

The FOG detection algorithm is designed to process and classify each individual time window. However, this is not the most energy-efficient solution for real-life data analysis. Instead, a simple threshold-based method could be used to distinguish between periods of activity and inactivity. Thus, the designed algorithm is run only during periods of activity. In this way, a percentage of FPs (recorded in this study during inactivity periods) can be avoided. In addition, the power consumption of the processing tool would be significantly reduced because the inactivity data is processed using a simple threshold approach instead of a DL algorithm. For this purpose, the magnitude M of the 3D acceleration signal for each window j was calculated according to Eq. (11), where α_x, α_y, and α_z represent the acceleration signal along each axis and the sum is performed for each sample i of each window of length w. Then, both the performance and the percentage of discarded windows were calculated on the validation set of the REMPARK data set. The threshold was selected so as to discard the data windows without degrading the performance of the algorithm. Finally, the effect of the magnitude threshold was evaluated separately on the REMPARK test set and the 6MWT and ADL data sets.

$$M_j = \sqrt{\sum_{i=1}^{w} \left(\alpha_{x_i}^2 + \alpha_{y_i}^2 + \alpha_{z_i}^2\right)} \tag{11}$$

The experiments were performed on a computer with a 2.3 GHz processor, 8 GB RAM and 4 GB GPU. Pre-processing and post-processing were performed with MATLAB (version R2020a), while training, optimization, and testing of the classification model were performed in Python (version 3.6), using the Keras (version 2.4), keras-flops (version 0.1.2) and TensorFlow (version 2.3) libraries.

4. Results

4.1. CNN architecture and parameters

The optimization process aimed at finding the best CNN architecture, model parameters, and learning settings led to the following results. A batch size of 256, a learning rate of 0.001 and a maximum number of iterations of 120 epochs were selected for the training, optimization, and testing procedures. The final CNN architecture that provided the best results is schematized in Fig. 7. It consists of a three-headed CNN block connected to dense classification layers.

Each head consists of two convolutional layers and two max-pooling layers. Each of these heads simultaneously processes the input (80 timesteps × 3 channels) using kernels of different sizes. The outputs of the CNN heads are flattened and concatenated to compose a vector feeding a single dense layer (16 units and a dropout rate of 0.5) and a final output layer with two outputs corresponding to the probability of FOG or no-FOG, respectively. The convolutional layers have 16 filters and ReLU activations each, with different kernel sizes in each CNN

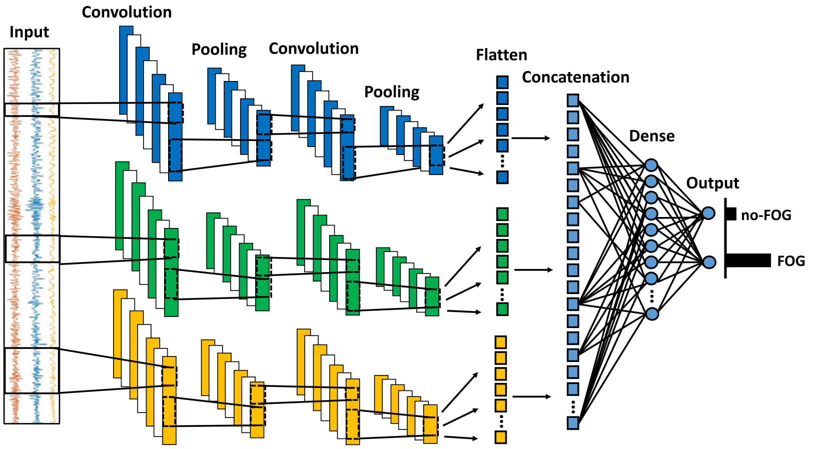

Fig. 7. Architecture of the optimized multi-head convolutional neural network model.

Table 3
Multi-head convolutional neural network layers, features, and parameters. n : number of filters; k : kernel size; d : dropout rate; p : pool size.

	Layer	Features	Shape	# param.
	Input	–	(80,3)	0
Head 1	Conv	n = 16, k = 6	(75,16)	304
	Dropout	d = 0.5	(75,16)	0
	Pool	p = 3, s = 3	(25,16)	0
	Conv	n = 16, k = 3	(23,16)	784
	Dropout	d = 0.5	(23,16)	0
	Pool	p = 3, s = 3	(7,16)	0
	Flatten	–	112	0
Head 2	Conv	n = 16, k = 12	(69,16)	592
	Dropout	d = 0.5	(69,16)	0
	Pool	p = 3, s = 3	(23,16)	0
	Conv	n = 16, k = 6	(18,16)	1552
	Dropout	d = 0.5	(18,16)	0
	Pool	p = 3, s = 3	(6,16)	0
	Flatten	–	96	0
Head 3	Conv	n = 16, k = 18	(63,16)	880
	Dropout	d = 0.5	(63,16)	0
	Pool	p = 3, s = 3	(21,16)	0
	Conv	n = 16, k = 9	(13,16)	2320
	Dropout	d = 0.5	(13,16)	0
	Pool	p = 3, s = 3	(4,16)	0
	Flatten	–	64	0
	Merge	–	272	0
	Dense	–	16	4368
	Dropout	d = 0.5	16	0
	Softmax	–	2	34

head. Specifically, kernel sizes of 6 and 3 were selected in the two convolutional layers of the first head, 12 and 6 in the second head, and 18 and 9 in the third head. The stride was set to 1 in all convolutional layers without padding to gradually reduce the size of the input signal. In addition, a pool size of 3 and a dropout rate of 0.5 were used for all convolutional layers, and the regularization term was set to 0.1 in the softmax layer.

The layers and parameters of the model are given in Table 3, along with the characteristics and output shape of each layer. The implemented model includes a total number of 10 834 trainable parameters, 6432 of which come from the convolutional layers and 4402 from the densely connected layers.

4.2. Performance

The training, validation, and test sets of the REMPARK data set were arranged as in Table 4. As can be seen, participants' age, duration of symptoms, disease progression (as measured by H&Y score) and motor impairment (as measured by MDS-UPDRS-III total score) are similar among the subsets. Participants included in the test set have a slightly higher FOG-Q score than those included in the training set, and the difference is more evident when compared with the validation set.

Table 4
Demographic and clinical features of PD patients include in training, validation, and test set. H&Y : Hoehn and Yahr score; MMSE : mini-mental state examination; FOG-Q : freezing of gait questionnaire; UPDRS : unified Parkinson's disease rating scale; ON : under dopaminergic therapy; OFF : not under dopaminergic therapy.

Set (# subjects)	Training (12)	Validation (4)	Test (5)
Patients ID	1–6,8,11,18–21	9,10,12,14	7,13,15–17
Age (years)	69.5	66.5	72.2
Symptoms duration (years)	11.0	15.5	12.6
H&Y	3.1	2.8	3.2
MMSE	27.9	27.5	27.4
FOG-Q	15.1	11.8	18.6
UPDRS-III ON	18.5	13.0	16.0
UPDRS-III OFF	36.3	37.5	33.6

Table 5
Performance of the implemented classification model on training, validation, and test set of the main dataset. EER : equal error rate; AUC : area under the curve.

Set	Train	Validation	Test
Sensitivity	0.884	0.879	0.877
Specificity	0.885	0.880	0.883
Geometric mean	0.885	0.880	0.880
F-score	0.886	0.838	0.830
EER (%)	11.5	11.9	11.9
AUC	0.955	0.947	0.946

Table 5 reports the performance of the model on the training, validation, and test set of the main data set. The performance of the model is similar across the three sets, with negligible reduction when moving from the training to the test set. This demonstrates the high generalization ability of the classification algorithm, which provides good results even on the test set, which includes data from completely unknown patients.

Regarding the true FOG episodes detected in the REMPARK data set, the algorithm correctly identified 100% of the episodes, with an average percentage of 84.8% of FOG detected in each episode. Specifically, the average percentage of FOG detected in each episode was 76.4%, 87.5%, and 90.2% in FOG episodes of 0–5 s, 5–10 s, and > 10 s duration, respectively (Table 6). Similar results were obtained for the 6MWT data set when considering episodes of more than 5 s duration, while a reduction in the detection rate was observed for FOG episodes of less than 5 s duration.

The duration of FPs was found to be 2.7 ± 1.5 s, with 37.7%, 61.1%, and 83.0% of false episodes less than 5 s, 10 s, and 20 s distant from the

Table 6
Percentage of FOG episodes detected.

Dataset	FOG episodes detection rate (%)		
	0–5 s	5–10 s	>10 s
REMPARK	100	100	100
6MWT	87	100	100

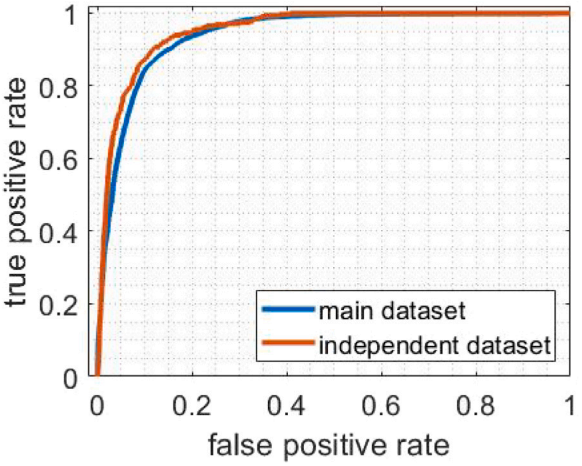

Fig. 8. Receiver operating characteristics of the classification model tested on the main and the external dataset.

nearest real FOG, respectively. This suggests that the false FOG episodes are relatively short and distributed close to the real FOG.

Regarding the resolution in FOG detection, 52.3% of FOG episodes were predicted before the actual onset, with an average advance of 3.1 s (SD = 2.6 s, min = 0.5 s, max = 11 s), while 47.7% of FOG episodes were detected with an average delay of 0.8 s (SD = 0.6 s, min = 0.5 s, max = 3 s).

Testing the classification algorithm on the 6MWT external data set yielded a sensitivity of 0.796, specificity of 0.933, geometric mean of 0.862, accuracy of 0.929, and AUC of 0.953. Fig. 8 shows the ROC curve of the implemented classification model, tested on the main (REMPARK) and external (6MWT) data set.

The ROC curves are similar, with slightly better performance in the external data set. This may be due to the different composition of the two data sets. Specifically, while the REMPARK data set includes data from patients ON and OFF therapy performing free walking activities, in the 6MWT data set the participants were in the daily ON condition and performed a simple 6MWT. However, even considering these differences, no reduction in performance is observed when testing the classification algorithm on a different data set, with data collected from different patients, under different conditions, and using a different sensor setting.

As for the true FOG episodes detected, 91.2% of the episodes were correctly identified by the algorithm, with an average percentage of 68.7% of FOG detected in each episode. More specifically, 87%, 100%, and 100% of the episodes of duration 0–5 s, 5–10 s, and > 10 s were identified by the system on the 6MWT data set, with an average percentage of 67.8%, 68.8%, and 74.5% of FOG detected in each episode (Table 6).

The duration of false FOG episodes was found to be 2.5 ± 1.1 s, with 14.6%, 23.7%, and 33.9% of false episodes being less than 5 s, 10 s, and 20 s distant from the closest true FOG episode, respectively.

With regard to temporal resolution in FOG detection, 32.5% of FOG episodes were predicted before actual onset, with an average advance of 1.3 s (SD = 0.8 s, min = 0.5 s, max = 3.5 s), while 50% of FOG episodes were detected with an average delay of 1.1 s (SD = 0.7 s, min = 0.5 s, max = 3 s). The remaining 17.5% of FOG episodes were detected with a delay of more than 3 s. Analyzing the activities corresponding to the false FOG episodes, 6.7% of gait and 1.4% of stance were classified as FOG. Testing the model on control subjects from the 6MWT data set, a specificity of 1 was obtained, demonstrating an excellent performance in rejecting false positives from elderly subjects without PD.

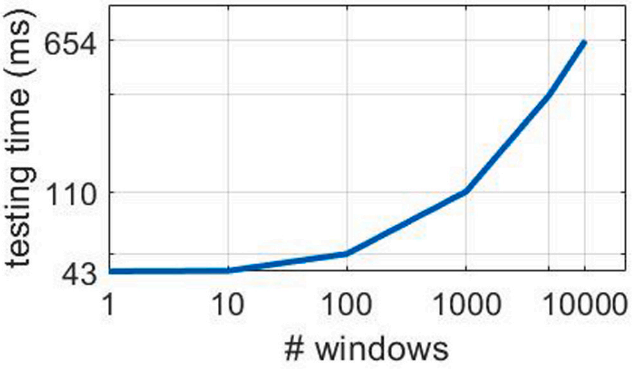

Fig. 9. Testing time for different input dimensions.

A specificity of 0.977 was obtained by testing the model on the external ADL data set, which does not include true FOG episodes. The analysis of the activities corresponding to the false FOG episodes resulted in 4.7% of gait, 0.9% of stance, 3.4% of postural transitions (e.g. standing up and sitting down), 21% of the pull test, and 6.8% of the foot tapping task being classified as FOG, while the rest of the FPs were recorded during unlabeled activities. However, it is worth noting that some activities (e.g. pull test and foot tapping) are only performed during the MDS-UPDRS assessment and do not represent common ADLs.

4.3. Computational complexity

Fig. 9 shows the testing time required by the model for different input data dimensions. In particular, 43 ms are required to test a single window, which represents the actual working condition for real-time applications. Furthermore, 110 s and 655 ms are required to classify 1000 (8.4 min of data) and 10 000 windows (1.4 h of data), respectively. Considering a FOG detection system receiving raw acceleration data from a single inertial sensor, the time required for the pre-processing steps must be taken into account.

Indeed, before moving on to the classification phase, it is necessary to perform the removal of the mean and properly reshape the resulting data for input to the classification model. However, the pre-processing time was negligible compared to the classification time, with 0.07 ms required for removing the mean and 0.003 ms for reshaping the data. Finally, the total memory required by the model was found to be 54.94 kB, with a single data window (i.e. 80 samples; Fs = 40 Hz) accounting for 44.10 kB, while the CNN parameters required only 10.84 kB. The proposed model presents a total of 0.399 M FLOPs to perform a prediction on a single (2-s long) window. This result is comparable with the evaluation of related DL methods such as that proposed in [43] (0.337 M FLOPs), and significantly lower than those proposed in [44] (3.14 M FLOPs) and [41] (4.76 M FLOPs).

The battery used in the STAT-ON device is a lithium-polymer battery with a capacity of 1200 mAh and an autonomy of 7 days when working continuously for 8 h [57]. The sensor has an average consumption of 4.1 ± 4.2 mA under current conditions, taking into account that the Bluetooth process consumes the most. Under normal conditions, the Bluetooth system does not activate and the consumption drops to about 3.7 mA. Currently, the total time spent by the microprocessor for calculation between samples is 0.279 ms (25 ms available between samples) and the processor needs 9.7 ms per window to calculate the

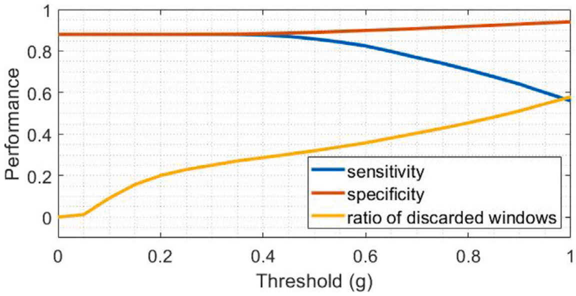

Fig. 10. The effect of the activity threshold tuning on the performance of the detection algorithm and on the ratio of discarded windows.

Table 7
The effect of different activity thresholds on the performance of the detection algorithm and on the ratio of discarded windows.

Activity threshold	Sensitivity	Specificity	Rejection ratio
0	0.879	0.880	0
0.4	0.877	0.884	0.287
0.7	0.769	0.910	0.405

outcome of the window plus 12.47 ms to write the information to the flash memory. Since the current algorithm needs 0.399 M FLOPS per window (2 s) and the processor executes about 210 Dhrystones Mega instructions per second (MIPS), the time needed in the current processor would be about 1.9 ms more in a 2-s period. Therefore, the change in battery life is practically insignificant and corresponds to a 0.095% increase in the extra running time the battery needs.

4.4. Activity threshold

Fig. 10 shows the performance of the developed FOG detection algorithm, in terms of sensitivity and specificity, and the ratio of windows discarded by increasing the threshold on the magnitude vector. The analyses were performed on the validation set of the REMPARK data set.

As can be observed, for threshold values below 0.4, performance does not change significantly (−0.2% sensitivity and +0.4% specificity), while the rejection ratio increases from 0 to 0.287. The further increase of the threshold value up to 1 g leads to a progressive improvement in specificity (from 0.884 to 0.941) and rejection ratio (from 0.287 to 0.580), with a clear reduction in sensitivity (from 0.877 to 0.560). Table 7 shows the performance of the algorithm and the rejection ratio in the absence of the activity threshold and for values of 0.4 and 0.7. Increasing the threshold from 0.4 to 0.7 results in a −10.8% reduction in sensitivity and a +2.6% increase in specificity, and +11.8% increase in the rate of discarded windows.

The effect of the activity threshold on the REMPARK test set and the two independent data sets is shown in Table 8. In general, the effect of the activity threshold is a gradual reduction in sensitivity and a slight improvement in specificity, as expected. This effect is more evident in the REMPARK data set than in the external data sets. Specifically, setting the threshold value to 0.4 g results in a −3.1% reduction in sensitivity and a +2.5% increase in specificity in the REMPARK data set, while performance is unaffected in the two external data sets. On the other hand, 27% and 20% of the data were discarded prior to classification in the REMPARK and 6MWT data sets respectively, whereas up to 63.2% of the windows were discarded in the ADL data set.

5. Discussion

In this study, a robust, lightweight, real-time freezing of gait detection algorithm is proposed. The main data set, comprising more than one thousand FOG episodes, was used to optimize, validate, and test the developed multi-head convolutional neural network. Performance was stable between the different sets, demonstrating the good generalization capability of the detection algorithm. When the model was tested on the external 6MWT data set, a reduction in sensitivity and an improvement in specificity was observed. This may be due to the different clinical characteristics and therapeutic conditions of the patients in the two data sets, with the main corpus including subjects with impaired gait and more severe FOG manifestations. Furthermore, the results should be interpreted considering that the main and independent corpus data were recorded using a different device placed in a different location (on the left side of the waist and lower back, respectively). By testing the model on the external ADL data set, comprising 59 PD patients performing different activities, a very high specificity was obtained, with few false positives recorded during common ADLs. Furthermore, the false FOG episodes were short and located close to the actual FOG episodes, suggesting a degraded walking pattern before and/or after FOG. Finally, unit specificity was achieved when testing the algorithm on elderly control subjects, demonstrating a high ability to reject false positives from elderly subjects without PD.

Regarding the FOG detection rate, 52.3% (32.5%) of FOG episodes were predicted 3.1 s (1.3 s) before their actual onset and 47.7% (50.0%) detected after 0.8 s (1.1 s), in the main (independent) test set. In a previous study [30], we focused on the detection and prediction of FOG using wearable sensors and machine learning techniques. The results suggested that although a robust classifier can be developed for FOG detection, a clear reduction in performance was observed when a FOG prediction system was implemented. A similar reduction in performance was observed in [47]. Furthermore, in both studies, data were recorded in the laboratory during predefined walking tasks. Data recorded in home environments under unsupervised conditions pose an even greater challenge for FOG prediction. However, in this study we demonstrated that the implementation of an algorithm for early FOG detection offers the opportunity to predict up to 50% of FOG episodes before their actual occurrence.

As far as computational complexity and test time are concerned, the minimal pre-processing, together with a memory requirement of less than 55 kB and a test time of 43 ms, make the algorithm suitable for real-time implementation in a stand-alone device.

Finally, the analysis performed on the activity threshold showed that it is possible to reduce the computational load of the detection algorithm without a significant decrease in performance.

In Table 9, the data, methods, and results of the proposed study are compared with those of recent research works using wearable inertial sensors and DL methods for FOG detection. As can be seen, most of the studies used inertial sensors on the back and ankles. In a cohort of 7 PD patients wearing a total of 6 inertial sensors, these positions were found to provide the best results [48]. The same study found that these locations represented the best sensor positions according to a sample of 16 PD patients.

The present study included the largest population of PD patients and is the only study using independent data sets to validate the detection

Table 8
The effect of the activity threshold on the main test set and on the two independent datasets.

Dataset	Performance	Activity threshold		
		0	0.4	0.7
REMPARK	Sensitivity	0.877	0.846	0.801
	Specificity	0.883	0.908	0.911
	Rejection ratio	0	0.270	0.376
6MWT	Sensitivity	0.796	0.796	0.744
	Specificity	0.933	0.934	0.938
	Rejection ratio	0	0.201	0.227
ADL	Specificity	0.977	0.978	0.981
	Rejection ratio	0	0.632	0.692

Table 9
Comparison between the present study and recent works. ADL : activities of daily living; acc : accelerometer; FFT : fast Fourier transform; gyro : gyroscope; CNN : convolutional neural network; LSTM : long short-term memory; GM : geometric mean; LOSO : leave-one-subject-out.

Study	Data	Sensor	Input	Algorithm	Performance
[41]	21 PD 1058 FOG episodes ADLs	acc waist	FFT 3.2s window	CNN-LSTM 5 layers	Sensitivity 0.871 Specificity 0.871 GM 0.871 (LOSO)
[42]	10 PD 237 FOG episodes walking, turning	acc thigh	Raw data 2s window	CNN Autoencoder 8 layers	Sensitivity 0.909 Specificity 0.670 GM 0.780 (LOSO)
[43]	21 PD 1058 FOG episodes ADLs	acc+gyro waist	FFT 3.2s window	CNN 6 layers	Sensitivity 0.919 Specificity 0.895 GM 0.907 (hold-out)
[44]	11 PD 184 FOG episodes walking, turning, stops	acc+gyro wrist	Raw data 3 s window	CNN 3 layers	Sensitivity 0.830 Specificity 0.880 GM 0.855 (LOSO)
[45]	7 PD 154 FOG episodes walking, turning	acc+gyro ankles	Raw data 2s window	CNN + MLP 6 layers	Sensitivity 0.630 Specificity 0.986 GM 0.788 (LOSO)
Proposed	118 PD 1110 FOG episodes ADLs	acc lower back	Raw data 2s window	CNN 3 layers	Sensitivity 0.877 Specificity 0.883 GM 0.880 (hold-out)

algorithm. In [42,45], 2-s windows were obtained from the raw inertial recordings and this is an advantage for real-time implementation of classification models. However, the performance obtained is lower (geometric mean 0.780 and 0.788, respectively) despite the use of complex architectures (8 and 6 layers, respectively). Furthermore, in [45], data were recorded by two inertial sensors (including an accelerometer and a gyroscope) on both ankles. Finally, [44] used a simple architecture (i.e. a 3-layer CNN) and obtained a geometric mean of 0.855. However, accelerometer and gyroscope data were used for the analysis and 3-s windows were used for segmentation.

In [41,43], the same data set as in the main corpus of this study was used for the analysis. The developed DL models demonstrated very good performance, achieving a geometric mean of 0.871 and 0.907, respectively. However, both of these studies used a larger window, calculated the fast Fourier transform (FFT) of the inertial signals, and used the FFT of four windows as input for the classification algorithm. Finally, complex DL architectures (five and six layers, respectively) were designed.

In addition to the studies using wearable inertial sensors reported in Table 9, in [47] plantar pressure data of 11 PD patients were recorded, 7 of which experienced a total number of 362 FOG events. A two-layer LSTM was implemented and optimized, which provided a sensitivity of 0.821 and specificity of 0.895 in LOSO validation, with 95% of FOG episodes detected correctly.

In summary, the proposed lightweight CNN architecture achieved state-of-the-art performance in FOG detection, slightly superior to [41] and second only to [43]. However, the present work has some limitations.

First, the REMPARK data set included most of the recorded FOG episodes, but the activity label was not available. On the other hand, the ADL data set included several annotations of the activities performed, but no FOG episodes were recorded. A large data set including a large number of patients and FOG episodes, as well as careful annotation of the most informative activities (e.g. gait, standing, sitting, and postural transitions) is needed.

Furthermore, raw input data were used in this study to limit the computational load and reduce processing time. However, this may not be the best performing solution. New implementations of time–frequency transforms can improve performance without significantly increasing the computational load.

Finally, although the computational complexity, memory requirements, and test time have been calculated in the present work, the developed detection algorithm has not yet been incorporated into a stand-alone device for real-time use in home environments.

6. Conclusion and future work

In this study, a lightweight and minimally invasive yet efficient solution for real-time FOG detection was proposed using a single inertial sensor. The experiments included a large number of PD patients and three different data sets with different experimental procedures and sensor settings. The implemented FOG detection algorithm demonstrated good to excellent performance, with a high detection rate, high specificity obtained during ADLs, low detection latency, good prediction capability, low memory requirements, and very short test times. The developed multi-head convolutional neural network showed a superior predictive capability compared to single-head CNN approaches, even considering the use of only one inertial sensor. The use of different spatial resolutions seems to be effective in detecting the local and global characteristics of FOG signals, thus enabling improved prediction performance without increasing the complexity of the classification model. The test procedure performed on the two external data sets provided useful insights into both the generalization capability of the detection algorithm and the context of the false positives recorded. The results suggested that the developed model is robust in rejecting common ADLs (e.g. gait, stance, and postural transitions). Furthermore, the activity threshold method developed in this study reduced the computational load without significantly impacting system performance. Therefore, the designed solution could be implemented in a stand-alone device and used to provide real-time feedback to trigger some sort of cueing system. However, the developed algorithm has not been tested in a stand-alone device for unsupervised real-time monitoring. Battery consumption should be carefully evaluated in future studies and possible countermeasures could be considered to further reduce the computational load of the detection algorithm.

CRediT authorship contribution statement

Luigi Borzì: Conceptualization, Methodology, Software, Investigation, Visualization, Validation, Writing – original draft. **Luis Sigcha:**

Conceptualization, Formal analysis, Software, Writing – original draft. **Daniel Rodríguez-Martín:** Resources, Supervision, Writing – review & editing. **Gabriella Olmo:** Project administration, Resources, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Data from the main corpus of this study has been obtained from the FP7 REMPARK project ICT–287677. The authors would like to thank the Technical Research Centre for Dependency Care and Autonomous Living (CETpD) for sharing the data used in this study. The authors wish to thank Professor Joan Cabestany, Universitat Politecnica de Catalunya, Spain for useful feedback and discussions.

References

[1] Samii A, Nutt J, Ransom B. Parkinson's disease. Lancet 2004;363(9423):1783–93. http://dx.doi.org/10.1016/S0140-6736(04)16305-8.

[2] Magrinelli F, Picelli A, Tocco P, Federico A, et al. Pathophysiology of motor dysfunction in parkinson's disease as the rationale for drug treatment and rehabilitation. Parkinson's Dis 2016;2016. http://dx.doi.org/10.1155/2016/9832839.

[3] Jankovic J. Parkinson's disease: Clinical features and diagnosis. J Neurol Neurosurg Psychiatry 2008;79(4):368–76. http://dx.doi.org/10.1136/jnnp.2007.131045.

[4] Goetz CG, Poewe W, Rascol O, Sampaio C, et al. Movement disorder society task force report on the hoehn and yahr staging scale: Status and recommendations. Movem Disord 2004;19(9):1020–8. http://dx.doi.org/10.1002/mds.20213.

[5] Tauţan A-M, Ionescu B, Santarnecchi E. Artificial intelligence in neurodegenerative diseases: A review of available tools with a focus on machine learning techniques. Artif Intell Med 2021;117:102081. http://dx.doi.org/10.1016/j.artmed.2021.102081.

[6] Pardoel S, Kofman J, Nantel J, Lemaire ED. Wearable-sensor-based detection and prediction of freezing of gait in parkinson's disease: A review. Sensors (Switzerland) 2019;19(23). http://dx.doi.org/10.3390/s19235141.

[7] Irrera F, Cabestany J, Suppa A. New advanced wireless technologies for objective monitoring of motor symptoms in Parkinson's disease. Front Neurol 2018;9:216. http://dx.doi.org/10.3389/fneur.2018.00216.

[8] Virginia Anikwe C, Friday Nweke H, Chukwu Ikegwu A, Adolphus Egwuonwu C, Uchenna Onu F, Rita Alo U, Wah Teh Y. Mobile and wearable sensors for data-driven health monitoring system: State-of-the-art and future prospect. Expert Syst Appl 2022;202:117362. http://dx.doi.org/10.1016/j.eswa.2022.117362.

[9] Goncalves HR, Rodrigues A, Santos CP. Gait monitoring system for patients with Parkinson's disease. Expert Syst Appl 2021;185:115653. http://dx.doi.org/10.1016/j.eswa.2021.115653.

[10] Borzì L, Varrecchia M, Sibille S, Olmo G, Artusi CA, Fabbri M, Rizzone MG, Romagnolo A, Zibetti M, Lopiano L. Smartphone-based estimation of item 3.8 of the MDS-UPDRS-III for assessing leg agility in people with Parkinson's disease. IEEE Open J. Eng. Med. Biol. 2020;1:140–7. http://dx.doi.org/10.1109/ojemb.2020.2993463.

[11] Daneault JF, Lee SI, Golabchi FN, Patel S, Shih LC, Paganoni S, Bonato P. Estimating Bradykinesia in Parkinson's Disease with a minimum number of wearable sensors. In: Proceedings - 2017 IEEE 2nd international conference on connected health: applications, systems and engineering technologies, CHASE 2017. 2017, p. 264–5. http://dx.doi.org/10.1109/CHASE.2017.94.

[12] Dai H, Cai G, Lin Z, Wang Z, Ye Q. Validation of inertial sensing-based wearable device for tremor and bradykinesia quantification. IEEE J Biomed Health Inf 2021;997–1005. http://dx.doi.org/10.1109/JBHI.2020.3009319.

[13] Rodríguez-Molinero A, Pérez-López C, Samà A, Rodríguez-Martín D, et al. Estimating dyskinesia severity in parkinson's disease by using a waist-worn sensor: concurrent validity study. Sci Rep 2019;9(1):1–7. http://dx.doi.org/10.1038/s41598-019-49798-3.

[14] Hssayeni M, Jimenez-Shahed J, Burack M, Ghoraani B. Dyskinesia severity estimation in patients with parkinson's disease using wearable sensors and a deep LSTM network. In: 42th annual international conference of the IEEE engineering in medicine & biology society (EMBC). 2020, p. 6001–4. http://dx.doi.org/10.1109/EMBC44109.2020.9176847.

[15] Borzì L, Fornara S, Amato F, Olmo G, Artusi CA, Lopiano L. Smartphone-based evaluation of postural stability in Parkinson's disease patients during quiet stance. Electronics (Switzerland) 2020;9(6):1–14. http://dx.doi.org/10.3390/electronics9060919.

[16] Hasegawa N, Maas KC, Shah VV, Carlson-Kuhta P, Nutt JG, Horak FB, Asaka T, Mancini M. Functional limits of stability and standing balance in people with Parkinson's disease with and without freezing of gait using wearable sensors. Gait Posture 2021;87:123–9. http://dx.doi.org/10.1016/j.gaitpost.2021.04.023.

[17] Borzì L, Olmo G, Artusi C, Fabbri M, Rizzone M, Romagnolo A, Zibetti M, Lopiano L. A new index to assess turning quality and postural stability in patients with parkinson's disease. Biomed Signal Process Control 2020;62:1–7. http://dx.doi.org/10.1016/j.bspc.2020.102059.

[18] Sigcha L, Ignacio P, Nélson C, Susana C, Miguel G, Arezes P, López JM, De Arcas G. Automatic resting tremor assessment in Parkinson's Disease using smartwatches and multitask convolutional neural networks. Sensors 2021;1:291. http://dx.doi.org/10.3390/s21010291.

[19] Rodríguez-Molinero A, Samà A, Pérez-López C, D. R-M, et al. Analysis of correlation between an accelerometer-based algorithm for detecting parkinsonian gait and UPDRS subscales. Front Neurol 2017;8:431. http://dx.doi.org/10.3389/fneur.2017.00431.

[20] Borzì L, Mazzetta I, Zampogna A, Suppa A, Irrera F, Olmo G. Predicting axial impairment in Parkinson's Disease through a single inertial sensor. Sensors 2022;22:412. http://dx.doi.org/10.3390/s22020412.

[21] Nutt JG, Bloem BR, Giladi N, Hallett M, Horak FB, Nieuwboer A. Freezing of gait: Moving forward on a mysterious clinical phenomenon. Lancet Neurol 2011;10(8):734–44. http://dx.doi.org/10.1016/S1474-4422(11)70143-0.

[22] Weiss A, Herman T, Giladi N, Hausdorff JM. New evidence for gait abnormalities among parkinson's disease patients who suffer from freezing of gait: insights using a body-fixed sensor worn for 3 days. J Neural Transm 2015;122(3):403–10. http://dx.doi.org/10.1007/s00702-014-1279-y.

[23] Schaafsma JD, Balash Y, Gurevich T, Bartels AL, Hausdorff JM, Giladi N. Characterization of freezing of gait subtypes and the response of each to levodopa in Parkinson's disease. Eur J Neurol 2003;10(4):391–8. http://dx.doi.org/10.1046/j.1468-1331.2003.00611.x.

[24] Mazilu S, Blanke U, Roggen D, Tröster G, Gazit E, Hausdorff JM. Engineers meet clinicians: Augmenting parkinson's disease patients to gather information for gait rehabilitation. In: Proceedings of the 4th augmented human international conference. Association for Computing Machinery; 2013, p. 124–7. http://dx.doi.org/10.1145/2459236.2459257.

[25] Zhao H, Wang R, Qi D, Xie J, Cao J, Liao W-H. Wearable gait monitoring for diagnosis of neurodegenerative diseases. Measurement 2022;202:111839. http://dx.doi.org/10.1016/j.measurement.2022.111839.

[26] Giladi N, Hausdorff JM. The role of mental function in the pathogenesis of freezing of gait in Parkinson's disease. J Neurol Sci 2006;248(1–2):173–6. http://dx.doi.org/10.1016/j.jns.2006.05.015.

[27] Bäckström D, Granåsen G, Domellöf ME, Linder J, Mo SJ, Riklund K, Zetterberg H, Blennow K, Forsgren L. Early predictors of mortality in parkinsonism and Parkinson disease A population-based study. Neurology 2018;91(22):E2045–56. http://dx.doi.org/10.1212/WNL.0000000000006576.

[28] Hulzinga F, Nieuwboer A, Dijkstra B, Mancini M, Strouwen C, Bloem B, Ginis P. Parkinson's disease. Movem Disord Clin Pract 2020;7:199–205. http://dx.doi.org/10.1002/mdc3.12893.

[29] Barthel C, Mallia E, Debû B, Bloem BR, Ferraye MU. The practalities of assessing freezing of gait. J Parkinson's Dis 2016;6(4):667–74. http://dx.doi.org/10.3233/JPD-160927.

[30] Borzì L, Mazzetta I, Zampogna A, Suppa A, Olmo G, Irrera F. Prediction of freezing of gait in Parkinson's Disease using wearables and machine learning. Sensors (Basel) 2021;21(2). http://dx.doi.org/10.3390/s21020614.

[31] Guo Y, Huang D, Zhang W, Wang L, Li Y, Olmo G, Wang Q, Meng F, Chan P. High-accuracy wearable detection of freezing of gait in Parkinson's disease based on pseudo-multimodal features. Comput Biol Med 2022;146:105629. http://dx.doi.org/10.1016/j.compbiomed.2022.105629.

[32] Ginis P, Nackaerts E, Nieuwboer A, Heremans E. Cueing for people with Parkinson's disease with freezing of gait: A narrative review of the state-of-the-art and novel perspectives. Ann Phys Rehabil. Med. 2018;61(6):407–13. http://dx.doi.org/10.1016/j.rehab.2017.08.002.

[33] Borzì L, Varrecchia M, Olmo G, Artusi C, Fabbri M, Rizzone M, Romagnolo A, Zibetti M, Lopiano L. Home monitoring of motor fluctuations in Parkinson's disease patients. J Reliab Intell Environ 2019;5:145–62. http://dx.doi.org/10.1007/s40860-019-00086-x.

[34] Rodríguez-Martín D, Samà A, Pérez-López C, Català A, Moreno Arostegui J, et al. Home detection of freezing of gait using support vector machines through a single waist-worn triaxial accelerometer. PLoS One 2017;12(2). http://dx.doi.org/10.1371/journal.pone.0171764.

[35] Tahafchi P, Molina R, Roper JA, Sowalsky K, et al. Freezing-of-Gait detection using temporal, spatial, and physiological features with a support-vector-machine classifier. In: Proceedings of the annual international conference of the IEEE engineering in medicine and biology society, EMBS, Vol. 352. IEEE; 2017, p. 2867–70. http://dx.doi.org/10.1109/EMBC.2017.8037455.

[36] Sigcha L, Borzì L, Pavón I, Costa N, Costa S, Arezes P, López JM, De Arcas G. Improvement of performance in freezing of gait detection in Parkinson's Disease using transformer networks and a single waist-worn triaxial accelerometer. Eng Appl Artif Intell 2022;116:105482. http://dx.doi.org/10.1016/j.engappai.2022.105482.

[37] Zoetewei D, Herman T, Brozgol M, Ginis P, Thumm PC, Ceulemans E, Decaluwé E, Palmerini L, Ferrari A, Nieuwboer A, Hausdorff JM. Protocol for the DeFOG trial: A randomized controlled trial on the effects of smartphone-based, on-demand cueing for freezing of gait in Parkinson's disease. Contemp Clin Trials Commun 2021;24:100817. http://dx.doi.org/10.1016/j.conctc.2021.100817.

[38] Borzì L, Olmo G, Artusi C, Lopiano L. Detection of freezing of gait in people with Parkinson's Disease using smartphones. In: 2020 IEEE 44th annual computers, software, and applications conference (COMPSAC). 2020, p. 625–35.. http://dx.doi.org/10.1109/COMPSAC48688.2020.0-186.

[39] Samà A, Rodríguez-Martín D, Pérez-López C, Català A, Alcaine S, Mestre B, Prats A, Crespo MC, Bayés À. Determining the optimal features in freezing of gait detection through a single waist accelerometer in home environments. Pattern Recognit Lett 2018;105:135–43. http://dx.doi.org/10.1016/j.patrec.2017.05.009.

[40] Mazzetta I, Zampogna A, Suppa A, Gumiero A, Pessione M, Irrera F. Wearable sensors system for an improved analysis of freezing of gait in Parkinson's Disease using electromyography and inertial signals. Sensors 2019;19(4):948. http://dx.doi.org/10.3390/s19040948.

[41] Sigcha L, Costa N, Pavón I, Costa S, Arezes P, López J, De Arcas G. Deep learning approaches for detecting freezing of gait in Parkinson's Disease patients through on-body acceleration sensors. Sensors 2020;20(7). http://dx.doi.org/10.3390/s20071895.

[42] Noor M, Nazir A, Wahab M, Ling J. Detection of freezing of gait using unsupervised convolutional denoising autoencoder. IEEE Access 2021;9(11):115700–9. http://dx.doi.org/10.1109/ACCESS.2021.3104975.

[43] Camps J, Sama A, Martin M, Rodriguez-Martin D, Perez-Lopez C, Arostegui J, Cabestany J, Catala A, Alcaine S, Mestre B, et al. Deep learning for freezing of gait detection in Parkinson's disease patients in their homes using a waist-worn inertial measurement unit. Knowl-Based Syst 2018;139:119–31. http://dx.doi.org/10.5555/3163587.3163748.

[44] Bikias T, Iakovakis D, Hadjidimitriou S, Charisis V, Hadjileontiadis LJ. DeepFoG: An IMU-based detection of freezing of gait episodes in Parkinson's Disease patients via deep learning. Front Robot AI 2021;8:537384. http://dx.doi.org/10.3389/frobt.2021.537384.

[45] Naghavi N, Wade E. Towards real-time prediction of freezing of gait in patients with Parkinsons Disease: A novel deep one-class classifier. IEEE J Biomed Health Inform 2021. http://dx.doi.org/10.1109/JBHI.2021.3103071.

[46] Mazzetta I, Zampogna A, Suppa A, Gumiero A, Pessione M, Irrera F. Wearable sensors system for an improved analysis of freezing of gait in Parkinson's disease using electromyography and inertial signals. Sensors 2019;19(4):948. http://dx.doi.org/10.3390/s19040948.

[47] Shalin G, Pardoel S, Lemaire E, Nantel J, Kofman J. Prediction and detection of freezing of gait in Parkinson's disease from plantar pressure data using long short-term memory neural-networks. J NeuroEng Rehabil 2021;18:167. http://dx.doi.org/10.1186/s12984-021-00958-5.

[48] O'Day J, Lee M, Seagers K, Hoffman S, Jih-Schiff A, Kidziński L, et al. Assessing inertial measurement unit locations for freezing of gait detection and patient preference. J NeuroEng Rehabil 2022;19(20). http://dx.doi.org/10.1186/s12984-022-00992-x.

[49] Demrozi F, Bacchin R, Tamburin S, Cristani M, Pravadelli G. Toward a wearable system for predicting freezing of gait in people affected by Parkinson's Disease. IEEE J Biomed Health Inform 2020;24(9):2444–51. http://dx.doi.org/10.1109/JBHI.2019.2952618.

[50] Moore S, MacDougall H, W.G. O. Ambulatory monitoring of freezing of gait in Parkinson's disease. J Neurosci Methods 2008;167(2):340–8. http://dx.doi.org/10.1016/j.jneumeth.2007.08.023.

[51] Bächlin M, Hausdorff J, Roggen D, Giladi N, Plotnik M, Tröster G. Online detection of freezing of gait in Parkinson's Disease patients: A performance characterization. In: Proceedings of the fourth international conference on body area networks. ICST; 2009, http://dx.doi.org/10.4108/ICST.BODYNETS2009.5852.

[52] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44. http://dx.doi.org/10.1038/nature14539.

[53] Jindong W, Yiqiang C, Shuji H, Xiaohui P, Lisha H. Deep learning for sensor-based activity recognition: A survey. Pattern Recognit Lett 2019;119:3–11. http://dx.doi.org/10.1016/j.patrec.2018.02.010.

[54] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, http://dx.doi.org/10.48550/arXiv.1412.6980, arXiv, arXiv:1412.6980.

[55] Zebin T, Scully PJ, Peek N, Casson AJ, Ozanyan KB. Design and implementation of a convolutional neural network on an edge computing smartphone for human activity recognition. IEEE Access 2019;7:133509–20. http://dx.doi.org/10.1109/ACCESS.2019.2941836.

[56] Moya Rueda F, Grzeszick R, Fink G, Feldhorst S, Ten Hompel M. Convolutional neural networks for human activity recognition using body-worn sensors. Informatics 2018;5:26. http://dx.doi.org/10.3390/informatics5020026.

[57] Rodríguez-Martín D, Cabestany J, Pérez-López C, Pie M, Calvet J, Samà A, et al. A new paradigm in Parkinson's Disease evaluation with wearable medical devices: A review of STAT-ON. Front Neurol 2022;2(13):912343. http://dx.doi.org/10.3389/fneur.2022.912343.