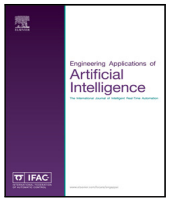
Engineering Applications of Artificial Intelligence 118 (2023) 105702

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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



A fuzzy convolutional attention-based GRU network for human activity recognition

Ghazaleh Khodabandelou *, Huiseok Moon, Yacine Amirat, Samer Mohammed

Laboratoire Images, Signaux et Systèmes Intelligents (LISSI), University Paris-Est Créteil, France

ARTICLE INFO

Dataset link: http://www.lissi.fr/;dataset-for-gait-mode-recognition-on-activities-of-daily-living/

Keywords:
Activity recognition
Fuzzy neural networks
Convolutional GRU neural networks
Lower limb wearable robots

ABSTRACT

Human activity recognition has become a pillar of today intelligent Human–Computer Interfaces as it typically provides more comfortable and ubiquitous interaction. This paper proposes a novel fuzzy-based deep learning-based algorithm to predict future sequences of activities from a given sequence of daily living activities of a subject wearing a lower limb exoskeleton. The engineering application concerns the challenging task of recognizing locomotion activities of the wearer in real-time, which is needed to ensure appropriate control of the robot during daily living activities. Indeed, real-time locomotion activity recognition is very challenging for controlling lower-limb exoskeletons. The model proposes a new adaptive kernel, based on the data features derived from the fuzzy rules on the input sequences to enrich the features of the activity sequences. Then, a CNN is applied to extract local subsequences from the whole sequences to identify local patterns in the convolution window. Finally, an attention-based GRU is incorporated into the model to extract meaningful parts of the time-series sequences. The results show high accuracy in the estimation of the transition between gait modes which is critical to ensure smooth control of the exoskeleton. The performance of the model is evaluated using the dynamic activity data gathered from different subjects. The proposed model outperforms the traditional models used in the literature.

1. Introduction

The prevalence of neurological diseases such as stroke, spinal cord injury, and traumatic brain injury is continuously increasing along with the increase of the elderly population in the world (Feigin et al., 2017). This leads usually to a lack of muscular strengths and endurances, which increases the failure risks during walking. Many mechatronic devices such as active exoskeletons are widely developed in the last decade to assist this dependent population in recovering its functional normal walking (Tucker et al., 2015).

For any type of human–computer interaction, especially human–robot interaction, the most fundamental communication mechanism is dialogue, which includes speech, gestures, semantic and pragmatic knowledge (Merdivan et al., 2019).

Human–AI interactions need some additional measures, especially in the direction of the often legally required traceability in the delicate environment of medical applications (Holzinger and Muller, 2021).

Controlling a lower limb exoskeleton for walking activities, i.e. alternatively between swing and stance phases, is largely constrained by the terrain nature. Thus, the control of the wearable robot will be gait-mode dependent, which means that the exoskeletons should be able to smoothly adapt their assistive strategies with respect to the current gait mode. To this end, real-time detection of the gait mode is essential to provide optimal and personalized control of the wearable device per gait mode. Indeed, a major limitation facing the development of wearable robots for daily living use is the lack of real-time human activity recognition. Various solutions have been proposed in the literature to deal with human activity detection during locomotion activities (Novak and Riener, 2015; Soleimani et al., 2022). The sensor used for gait mode detection can be categorized into two major groups: (1) bio-electrical sensors, e.g. electromyography and encephalography sensors, and (2) mechanical sensors, such as kinetics and kinematics sensors (e.g. Inertial Measurement Units, Encoders, Ground Reaction Forces plates, etc.). Manual switching-based methods, such as pushing a button (Baiden and Ivlev, 2013), voice commands (Farris et al., 2011) or conducting some fingers or arms movements (Hasegawa et al., 2009) are practical solutions for gait mode transition, particularly for patients with complete paraplegia.

These conventional methods are not practical to patients with partial impairments assume that the wearer should be thoroughly trained to operate the wearable robot. Alternatively, automatic recognition methods for detecting the gait modes have been also proposed in the literature (Huang et al., 2008; Yuan et al., 2014; Mannini and Sabatini, 2012). Deep Neural Networks and Fuzzy-logic based methods have been extensively used in the last decade to estimate gait variables that

* Corresponding author.
E-mail address: ghazaleh.khodabandelou@u-pec.fr (G. Khodabandelou).

https://doi.org/10.1016/j.engappai.2022.105702
Received 6 July 2022; Received in revised form 1 November 2022; Accepted 29 November 2022
Available online 12 December 2022
0952-1976/

are essential inputs to controllers used within an assistance-as-needed robotic devices in particular in daily living activities (Du et al., 2021; Parri et al., 2017; Rubio-Solis et al., 2020; Medjahed et al., 2009; Kang et al., 2021; Zhu et al., 2021; Su et al., 2020; Wang, 2021; Kidziński et al., 2020).

The fusion of fuzzy with deep neural networks is an emerging research direction and is receiving increasing attention. In particular, fuzzy systems have a great influence on deep neural models in the aspect of classification, prediction, natural language processing, self-control, and the fusion is applied in various fields, such as, but not limited to medical systems, patient rehabilitation, assistive robotics, movement analysis, automatic control, and machinery industry.

Despite the advantages of the deep neural network and its combination with the fuzzy system, this type of approach has not been widely used or explored in robotics and Human Activity Recognition (HAR) research using sensor data. Real-time recognition of gait-mode activities is a critical performance indicator of any controller used within an assistance-as-needed robotic devices. Conventional methods relying on subject-specific training procedures are efforts-demanding for the subjects and cumbersome as they often require long calibration processes. Automatic gait mode recognition-based approaches would promote the use of exoskeleton technology in more realistic settings where the wearer can naturally and smoothly operate through different terrain settings. Indeed, machine learning–based methods become recently a popular approach for estimating gait features that are used in wearable robot controllers (Shepherd et al., 2022; Kang et al., 2019; Tu et al., 2021; Lee et al., 2021).

As shown in Zheng et al. (2022), fuzzy systems can deal with the inherent uncertainty information widely present in the data, the redundant information block, and the classification of boundary points by membership function and overfitting control by expressing the data in a fuzzy space instead of presenting them with exact numbers. Thus, it improves the accuracy of predictions in deep learning models. The advantage of using fuzzy systems is twofold: (i) it is a robust system that takes into account the incompleteness of the available data, such as sensor data where the data may be noisy or inaccurate; (ii) it allows to obtain enriched features from the initial attributes of the data. A convolutional neural network (CNN) is used to extract local patterns. The advantage of CNNs trained with fuzzy rules is also to eliminate both the limited availability of training data and the need for extensive data preprocessing.

In the proposed approach, the combination of the proposed fuzzy logic with a CNN benefits from the use of fuzzy membership degrees to produce more refined outputs, thus reducing the ambiguities that exist in the nature of human activity recognition. It also allows for high-level feature extraction through convolutional neural representation. In addition, it benefits from its ability to sufficiently improve the detection capability of CNN even with a small amount of data.

To this end, this paper proposes a deep learning algorithm based on combining a CNN with a novel fuzzy system to predict human activity. The model first applies an adaptive kernel on the input sequences to derive the fuzzy rule features. The output of the fuzzy logic is fed to a CNN algorithm. The proposed fuzzy system is Takagi–Sugeno–Kang (TSK) and uses a Gaussian mixture model (GMM) to construct the fuzzy rules by dividing the input space into overlapping clusters. Since the GMM generates orthogonal clusters, the membership functions can be obtained by projecting them onto the input variables in the cluster space.

Then, a CNN is applied on these features to extract the local subsequences from the set of sequences to identify the local patterns in the convolution window. Finally, an attention-based GRU (Gated Recurrent Unit) (Cho et al., 2014) is incorporated into the model to take advantage of the inherent interdependence of these transformed sequences to extract meaningful parts of the short-term sequences. In addition to the main contributions mentioned above, this paper provides a public data set to serve research advances in the robotics community.

The five common daily living activities that are: level walking, stair Up/Down, and ramp Up/Down for a subject wearing a lower limb exoskeleton.

The rest of the paper is organized as follows: Section 2 presents the related work. Section 3 describes the proposed fuzzy convolution attention-based GRU network model. Section 4 presents the data acquisition feature extraction for model training. Section 5 analyzes the experimental results and the evaluation of the gait mode detection process. And Section 6 concludes the paper.

2. Related work

In recent years, many automatic gait modes recognition methods have been proposed in the literature based on conventional and advance machine learning approaches. This section presents a review of these approaches.

2.1. Approaches based on conventional machine leaning methods

Some of the most commonly used methods are decision trees (Tucker et al., 2015) and finite state machines, Young and Ferris (2016) which operate based on a set of rules. In particular, gait phase-based finite-state control approaches have been widely used in lower limb exoskeletons for rehabilitation and/or power-assist purposes (Murray et al., 2014; Au et al., 2008; Wang et al., 2014). To switch between gait modes, Wentink et al. detected the transition from standing to walking using IMU-based thresholds and footswitch signals (Wentink et al., 2014). Li and Hsiao-Wecksler (Li and Hsiao-Wecksler, 2013) similarly detected the switch between level walking and stair ascent/descent for an ankle–foot orthosis by measuring the vertical position. Zhang et al. (2011) used a laser distance sensor mounted on the waist to detect stairs in front of the user, then to switch between level walking and ascent/descent.

In Varol et al. (2009) Varol et al. detected the switch between gait phases such as sitting and quiet standing based on inputs from joint angles and ground contact forces. Signals are segmented into 100-ms windows, from which, means and standard deviations are extracted. Features are reduced using Linear Discriminant Analysis (LDA) and classified using a Gaussian mixture model. Finally, a majority vote scheme is used to generate the control output.

In Jang et al. (2015), an online gait recognition algorithm for hip exoskeleton is proposed. The algorithm provides automatic and prompt recognition using a rule-based inference system based on measurements of the hip joint angle during the stance phase. Authors of Yuan et al. (2014) introduced a fuzzy logic-based terrain identification approach to control a prosthesis. The gait phase is divided, using a simple rule-based approach, into four subphases (early stance, middle stance, late stance, and swing phase). The likelihood of each gait phase is calculated based on the membership values of each feature and the fuzzy rules using the method proposed in Kong and Tomizuka (2009). The terrain (i.e., gait modes) can be identified prior to the swing phase of the prosthesis. The same method is proposed in Huo et al. (2018b) to detect five gait modes (Level walking, Stairs ascent, Stairs descent, ramp ascent, ramp descent). A review of state-of-the-art techniques for abnormal human activity recognition and smart homes was presented in Dhiman and Vishwakarma (2019) and De Silva et al. (2012).

Dutta et al. proposed an approach to identify different gait modes (level walking, ram ascent, ramp descent) in leg prosthesis. A Kalman filter with EMG input is created for each gait mode. The filters' outputs are weighted using a Bayesian inference system according to the probability of each gait mode, then summed to create a 'mixture' of filter outputs (Dutta et al., 2011).

In Long et al. (2016) an online support vector machine (SVM) optimized using particle swarm optimization (PSO) is proposed to estimate different gait modes. The inputs of the SVM are the signals measured through the foot pressure sensors integrated into the insoles of the

wearable shoes and the MEMS-based attitude and heading reference systems attached to the shoes and shanks of the leg segments. The algorithm ensures a smooth and automatic transition between the different gait modes. The PSO is used to ensure optimal performances of the SVM. Novak et al. used IMUs and insoles measurements to detect transitions between standing and walking by means of a decision tree algorithm. Zheng et al. (2014) placed capacitive sensors into the socket between the human limb and the prosthesis to classify different gait modes using quadratic discriminant analysis. In Mannini and Sabatini (2012), a classifier based on a hidden Markov model (HMM) was proposed for gait phase detection and walking/jogging discrimination. The gait events such as foot strike, foot flat, heel off, toe off were detected and walking/jogging activities were discriminated using data acquired from a gyroscope.

These conventional machine learning algorithms require human intervention, such as choosing features and classifiers and making a set of rules or thresholds. The human must check if the result is satisfactory and adjust the algorithm if it is not. In contrast, deep neural network methods make it possible to entrust this work to the machine automatically.

2.2. Approaches based on deep neural networks

In recent years, several deep neural networks have been proposed for activity recognition.

Shin et al. (2020) proposed a method for automatic locomotion mode detection (Level Walking, Stair Ascent, Stair Descent) based on the use of Radial Basis Function Support Vector Machine (RBF-SVM) for hip gait assistive robot. Recently, some methods in the literature proposed combination of fuzzy neural networks with the convolutional neural network. In Cordero-Martínez et al. (2022) a new CNN model is applied using an hierarchical genetic algorithm method to increase the classification accuracy for diabetic retinopathy classification. The proposed approach in Singh et al. (2017) uses a CNN to automatically extract features from ambient sensor environment data, and then it uses an Long Short Term Memory (LSTM) for daily activity recognition of time-series sensor data. A PSO-based approach proposed in Fregoso et al. (2021) uses a heuristic search for the adjustment of CNN hyperparameters and the optimal network architecture.

Some fuzzy-based combined with deep neural networks have been proposed in the literature for gait mode classification. A fuzzy rule-based fuzzified deep convolutional neural network (FDCNN) architecture is proposed in Gomathi et al. (2022) for the classification of smartphone sensor-based human activity recognition. In this approach, a data normalization technique is used by fusing the λ-max method for weight initialization. In Banerjee et al. (2020), the 3D skeleton-based HAR problem is addressed. Using the Choquet integral, fuzzy fusion is used to combine CNN decision scores and generate the final decision.

A feature extraction approach based on direct causality and fuzzy temporal windows is proposed in Hwang et al. (2021) for human activity recognition applied on smart home data. The causality and fuzzy features are then merged to learn the spatio-temporal dependencies of the merged feature, using LSTM and CNN.

A melanoma lesion detection method is proposed in Nida et al. (2019) using a deep region-based convolutional neural network to simplify localization through Fuzzy C-mean clustering. A deep convolutional fuzzy system is proposed for big data classification problems in Hsu et al. (2020) using a Wang–Mendel method for the training of each sub-fuzzy system and hierarchical architecture. In Wang (2019), authors proposed a layer-by-layer iterative process for training a deep convolutional fuzzy system using the Wang–Mendel method for fast training algorithms. In Sadaei et al. (2019), a fuzzy time series combined with convolutional neural networks for short-term load forecasting is proposed for short-term load forecasting. In Zhang et al. (2021) a method based on FNN (Fuzzy Neural Network) is proposed to tackle with the problem of missing data in a large-scale road network. A fuzzy gravity search algorithm is used in Poma et al. (2020) to increase the performance of the CNN by optimizing hyperparameters such as the number of images per block, the filters and the size of the filters in the convolutional layer.

In Du et al. (2021), authors proposed a fuzzy logic based algorithm for locomotion and transition mode recognition based on the uses of inertial sensors mounted on a hip joint exoskeleton. The proposed approach is able to overcome the subject-dependent parameters in data training, avoid thus a training procedure per subject. In Parri et al. (2017), authors proposed an event-based fuzzy-logic approach that is triggered by the measurement of the foot pressure to classify the gait modes for the control of a lower limb wearable robot. Authors of Rubio-Solis et al. (2020), proposed a multilayer interval type-2 fuzzy logic system to recognize walking activities and gait events using inertial wearable sensors. The proposed approach showed excellent performances in terms walking activities recognition even in the presence of sponsor noise.

In Medjahed et al. (2009), authors proposed a fuzzy logic-based approach for recognizing activities in home environment using a set of physiological sensors, microphones, infrared sensors, etc. The fusion of the different sensors data has been performed by using fuzzy logic. In Kang et al. (2021), a convolutional neural network-based gait phase estimator is used under different locomotion mode settings for exoskeleton assistance and has shown its efficiency in accurately predicting the gait phase during multimodal locomotion. Authors of Zhu et al. (2021) proposed a knee/ankle joint trajectory prediction method based on attention-based CNN-LSTM model. The attention mechanism is used to ensure correlation of gait data in time and space avoiding thus data redundancy and ensuring high prediction accuracy. Deep convolutional neural networks was used in Su et al. (2020) to classify five phases in a given gait cycle, based on the use of IMU data. Accuracy was the best in the swing phase and lowest in the terminal stance gait phase. In Wang (2021), a convolutional neural network based approach is proposed for motion recognition intention of patients with lower limb amputation under different terrains when using lower limb prosthesis. Authors of Kidziński et al. (2020) used a deep learning-based method for predicting clinically relevant motion parameters from using a low-cost single-camera video. The motion parameters include walking speed, cadence, knee flexion angle at maximum extension, Gait Deviation Index (GDI), etc.

Despite the interest and benefit of the fuzzy system and its combination with the deep neural network, the robotics community still lacks robust systems from this fusion that take into account the incompleteness, noisiness or incorrectness of sensor data without the need for extensive data preprocessing. A system that generates enriched features and identifies spatio-temporal patterns and can handle the inherent characteristics of HAR data, such as ambiguity and a limited amount of data (number of subjects).

3. Proposed method

This section presents a formal description of the proposed method, which is a fuzzy convolutional attention-based GRU network model for the recognition of human activity using sequential activities. A model trained to capture temporal activity evolution can capture the activity patterns, and consequently, it is more likely to predict accurately and to adjust to different conditions. These requirements motivate the use of a model that takes into account temporality and sequentiality in the activity data.

A block scheme representing the different phases of the model is portrayed in Fig. 1. The detailed overview of the method is displayed in Fig. 2. All parts of the model are described in the next section.

In order to enrich activity sequence features, they are transformed into a new space with 23 features. These new features are not only representatives of the initial ones but also represent much more affluent inherent characteristics of raw activity sequences. To do so, an adaptive

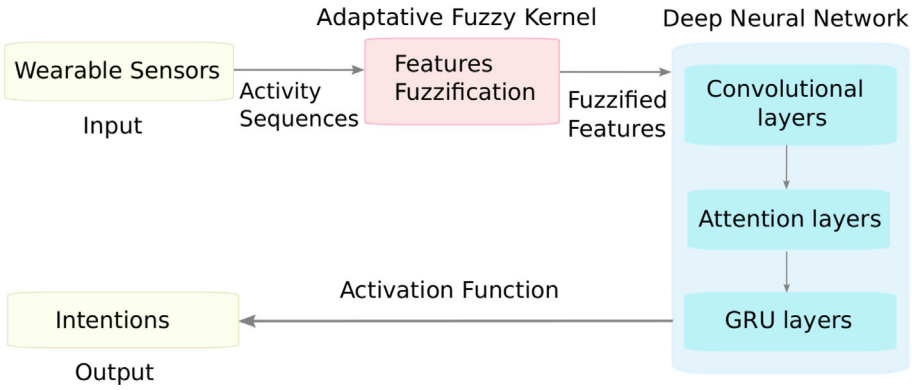


Fig. 1. Block scheme representing the different phases of Fuzzy Convolutional attention-based GRU network model.

kernel-based is applied on data characteristics, which are derived from the fuzzy rules over input sequences. In the proposed model, the prediction of humans' activities based on their activity sequences are leveraged to extract local patterns over these sequences. To do so, a CNN is used for extracting local subsequences from sequences to identify local patterns within the convolution window. The outputs of the CNN are fed into an attention mechanism layers to take advantage of these local subsequences to retrieve the meaningful parts of the short-term sequences (Khodabandelou et al., 2020, 2021, 2019). Then a GRU (Gated Recurrent Unit) (Cho et al., 2014) is used in the model architecture to excerpt the temporal features of the dataset. GRUs architecture is used to cope with vanishing gradient. Finally, all features are combined and fed into a classification layer, *i.e.,* fully connected layer. A model trained to capture temporal activity evolution can capture the activity patterns, and consequently, it is more likely to predict accurately and to adjust to different conditions. These requirements motivate the use of a model that takes into account temporality and sequentiality in the activity data.

The detailed overview of the method is displayed in Fig. 2.

3.1. Fuzzification of human activity inputs

A fuzzy system consists of a collection of fuzzy rules utilizing human knowledge processing ability. Fuzzy rules are composed of different preconditions assessing the association of the input variables to fuzzy sets and they are Takagi–Sugeno–Kang type (Takagi and Sugeno, 1985) consequence. As shown in Juang (2002) facing intricate problems a fuzzy system with TSK-type consequence usually outperforms fuzzy singleton or fuzzy set consequence. As shown in Juang and Lin (1998) and Juang (2002) facing intricate problems a fuzzy system with TSK-type consequence usually outperforms fuzzy singleton or fuzzy set consequence. Let $S = \{(X_1, y_1), (X_2, y_2), \ldots, (X_n, y_n)\}$ be a set of training data, where $X_i \in \mathcal{R}^d$. Each input X_i is a d-dimensional real vector and $y_i \in \{-1, +1\}$ stands for the input labels.

Each rule in a TSK-type fuzzy system is defined as follows:

Rule c : IF x_1 is A_{c1} AND $\cdots$ AND x_d is A_{cd}

$$\text{THEN } y = \beta_{c_0} + \sum_{i}^{d} \beta_{ci} x_i, \qquad c = \{1, \ldots, n_c\} \tag{1}$$

where A_{ci} stands for the fuzzy set of the input variables x_i, β_{ci} is a real number, n_c is the number of fuzzy rules and d is the number of dimensions.

The number of fuzzy rules for each dataset is determined as in Khodabandelou and Ebadzadeh (2019). The first step is to use a Gaussian mixture model algorithm (GMM) clustering algorithm to construct the fuzzy rules by dividing input space into 23 overlapping clusters. In fact, the generated clusters correspond to rules in the rule-base, i.e. new features. The clustering of input space has to be applied on the basis that the points in a given cluster have the same values or belong to the same class of the output variable. The parameters of the Gaussian function for a cluster c in the input space are represented by μ_c and Σ_c, which stand for mean and diagonal covariance matrix of samples, respectively. Thereby, the Gaussian function is represented as follows:

$$g_c(X) = exp\left\{-\gamma(X - \mu_c)^T \Sigma_c^{-1}(X - \mu_c)\right\} \tag{2}$$

where the parameter γ stands for the inverse of the radius of influence of samples selected by the model as support vectors.

The preconditions of the fuzzy rules are determined using the clusters generated in the previous step. Since GMM generates orthogonal clusters, the membership functions can be achieved by projecting them onto the input variables in the cluster space.

Let $I^{(L)}$ and $O^{(L)}$ be the input and output variables of a node in layer L, respectively. In what follows, we describe the basic functions and the signal propagation in each layer of the network:

Layer 1 – Input layer. This layer is the input layer and each node represents an input variable. In this layer, the input values are directly forwarded towards the next layer. The output of this layer is defined as:

$$O_i^{(1)} = I_i^{(1)} = x_i, \tag{3}$$

where x_i is the ith input variables of the network.

Layer 2 – Membership function layer. In this layer, nodes represent membership functions corresponding to a fuzzy set of the input variables. Indeed, the membership functions evaluate the membership degree of input values to that fuzzy set. The membership value of each node is calculated as follows:

$$O_{c_i}^{(2)} = exp\left\{\frac{-(x_i - \mu_{c_i})^2}{\sigma_{c_i}^2}\right\} = M_{c_i}(x_i), \tag{4}$$

where $O_{c_i}^{(2)}$ stands for the output of the ith membership function of the cth cluster (rule).

Layer 3 – Rule layer. In this layer, nodes represent fuzzy logic rules and carry out fuzzy preconditioning of the rules. In fact, the output of this layer represents the firing force of a fuzzy rule. The AND operation is applied to all nodes of this layer as follows:

$$O_c^{(3)} = \prod_{i=1}^{d} I_{c_i}^{(3)} = \prod_{i=1}^{d} O_{c_i}^{(2)} = \prod_{i=1}^{d} M_{c_i}(x_i), \tag{5}$$

When X is the network input vector, $g_c(X)$ is equivalent to $O_c^{(3)}$. Thus, the Eq. (5) can be defined as follows:

$$\begin{aligned} O_c^{(3)} &= \prod_{i=1}^{d} exp\left\{-\frac{(x_i - \mu_{c_i})^2}{\sigma_{c_i}^2}\right\} \\ &= exp\left\{-\sum_{i=1}^{d} \frac{(x_i - \mu_{c_i})^2}{\sigma_{c_i}^2}\right\} \\ &= exp\left\{-(X - \mu_c)^T \Sigma_c^{-1}(X - \mu_c)\right\} = g_c(X), \end{aligned} \tag{6}$$

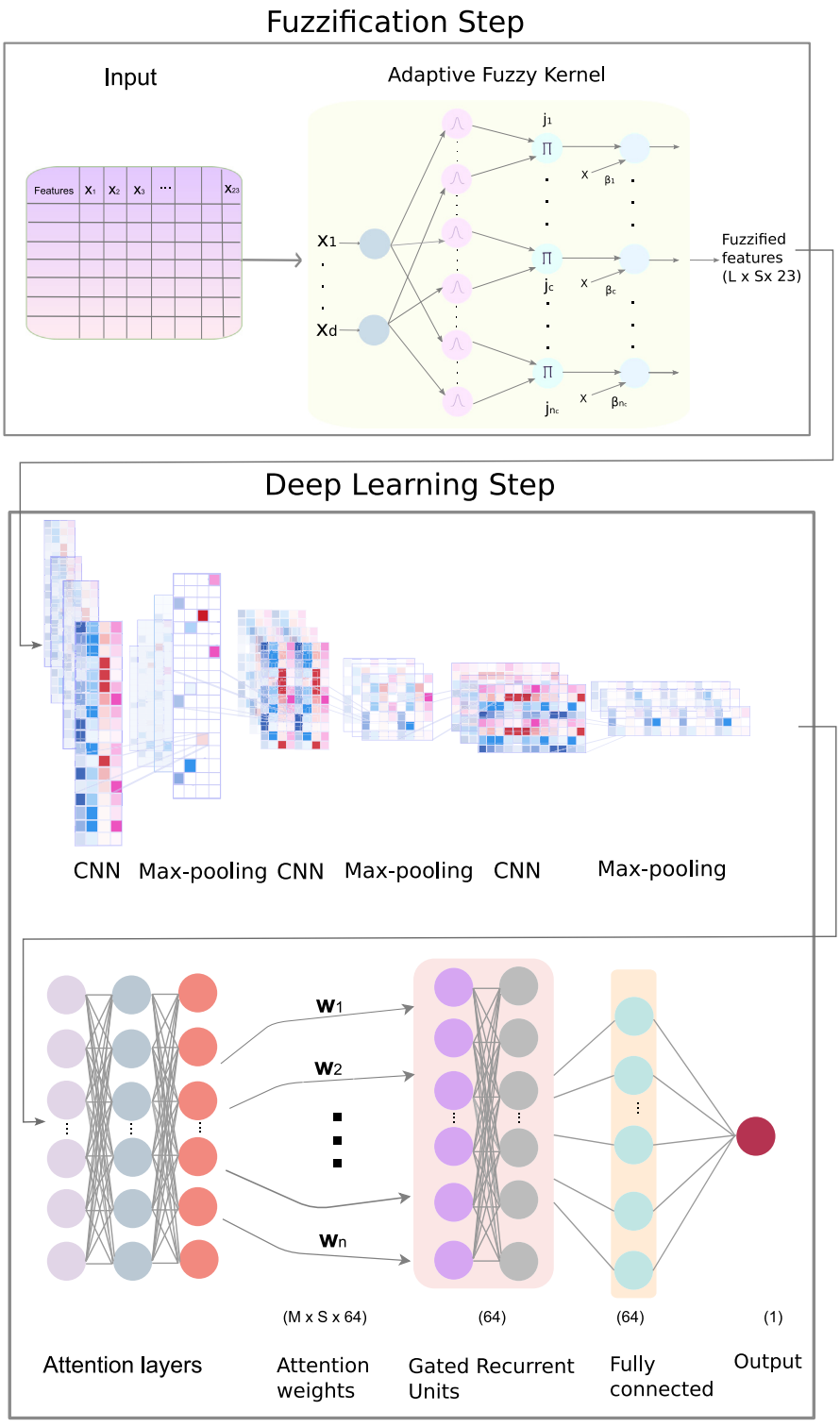


Fig. 2. Overview of Fuzzy Convolutional attention-based GRU network model.

where Σ_c is a diagonal matrix such that each diagonal element of $\Sigma_{c_{ii}} = \sigma_{c_i}^{(2)}$.

Layer 4 – Consequent layer. In this layer, the TSK-type consequence of the fuzzy rules are generated by the nodes. A linear combination of the input vector $\beta_c^T X$ is applied to cth rule (node), where β_c represents the network parameter to be adjusted by SVM. The output value of the cth node is as follows:

$$O_c^{(4)} = \left(\beta_{c_0} + \beta_c^T X\right) \times exp\left\{-(X-\mu_c)^T \Sigma_c^{-1}(X-\mu_c)\right\}. \tag{7}$$

Layer 5 – Output layer. In this layer, the global output is calculated as the weighted sum of all the inputs with a bias term. The output of this layer is as follows:

$$\begin{aligned} O^{(5)} &= \sum_{c=1}^{n_c} \left(\beta_{c_0} + \beta_c^T X\right) \times exp\left\{-(X-\mu_c)^T \Sigma_c^{-1}(X-\mu_c)\right\} + b \\ &= \sum_{c=1}^{n_c} \left(\beta_{c_0} + \beta_c^T X\right) \times g_c(X) + b \end{aligned} \tag{8}$$

Consequently, the decision function can be written as follows:

$$f(X) = sign\left\{\sum_{c=1}^{n_c}\left(\beta_{c_0} + \beta_c^T X\right) \times g_c(X) + b\right\} \tag{9}$$

where β_c and β_{c_0} represent the weights and b is the offset for cth rule (cluster) of the input vector X. with considering :

$$\beta_{c_0} = \sum_{i=1}^{SV} \alpha_i y_i g_c(X_i), \quad \beta_c = \sum_{i=1}^{SV} \alpha_i y_i g_c(X_i) X_i \tag{10}$$

where α_i are Lagrange multipliers and SV stand for the support vectors. By replacing Eq. (10) in Eq. (9):

$$\begin{aligned} f(X) &= \sum_{c=1}^{n_c}\sum_{i=1}^{SV} \alpha_i y_i g_c(X_i) g_c(X)(1 + X_i^T X) + b \\ &= \sum_{i=1}^{SV} \alpha_i y_i (1 + X_i^T X) \sum_{c=1}^{n_c} g_c(X_i) g_c(X) \end{aligned} \tag{11}$$

Then the proposed adaptive fuzzy kernel is defined as:

$$K(X_i, X) = (1 + X_i^T X) \sum_{c=1}^{n_c} g_c(X_i) g_c(X) \tag{12}$$

where $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}_0^+$, functions $g_c(X_i)$ and $g_c(X)$ determine the membership degree of a sample to cluster c and n_c represents the number of clusters.

The kernel $K(X_i, X)$ is adaptive since the number of Gaussian functions in $g_c(X_i)$ and $g_c(X)$ along with their parameters are automatically determined according to data distribution of the problem. Indeed, in the clustering phase, the data distribution is approximated by a set of Gaussian functions. Algorithm 1 shows how weight parameters β_{c_0} and β_c are obtained.

Algorithm 1: Learning algorithm of weight parameters β_{c_0} and β_c.

Input: X_i, n_c
Output: β_{c_0}, β_c
$\mu_c, \Sigma_c \leftarrow$ *Call GMM algorithm (input:* X_i, n_c*)*
for $\forall X_i$ **do**
  $g_c(X_i) \leftarrow exp(-\gamma(X_i - \mu_c)^T \Sigma_c^{-1}(X_i - \mu_c))$;
  $K(X_i, X_j) \leftarrow (1 + X_i^T X_j) \sum_{c=1}^{n_c} g_c(X_i) g_c(X_j)$
  $\alpha_i \leftarrow$ Call SVM (input: $K(X_i, X_j)$)
  $\beta_{c_0} \leftarrow \sum_{i=1}^{SV} \alpha_i y_i g_c(X_i)$;
  $\beta_c \leftarrow \sum_{i=1}^{SV} \alpha_i y_i g_c(X_i) X_i$;
end

3.2. Local feature extraction

In the proposed method, the recognition of humans activities is leveraged based on their activity sequences to extract local patterns over these sequences. Convolutional neural networks (CNNs) (Krizhevsky et al., 2012) are the powerful tools to extract spatial feature. A CNN structure is used in the proposed architecture to model human activity sequences. The CNN extracts local patches or fragments from sequences to capture the local patterns in an activity sequence. The convolutional layers carry out a series of sliding windows operation (convolution) using kernels to scan motifs all over the sequences.

In order to predict activity of a given subject $s \in S$ $\{s_i^T\}_{i=1}^{l}$ for producing predictions of h future time horizon $\mathcal{T} = (t, t+1, \ldots, t+h)$, we make use of the historical activity sequences data $\{a_i^T\}_{i=1}^{l}$ of these s subject as inputs for n time window size during $T = (t-n, t-n+1, \ldots, t-1)$. The near-term activity sequences matrix is defined as:

$$\mathcal{A} = \begin{bmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \vdots \\ \mathcal{A}_l \end{bmatrix} = \begin{bmatrix} a_1^{(t-n)} & a_1^{(t-n+1)} & \cdots & a_1^{(t-1)} \\ a_2^{(t-n)} & a_2^{(t-n+1)} & \cdots & a_2^{(t-1)} \\ \vdots & \vdots & \cdots & \vdots \\ a_s^{(t-n)} & a_s^{(t-n+1)} & \cdots & a_s^{(t-1)} \end{bmatrix}_{\boldsymbol{a} \in \mathcal{A}} \tag{13}$$

The first layer kernels perform convolution on successive input sequences vectors $\mathcal{P}_c$ of matrix $\mathcal{A}$ to recognize relevant patterns (motifs) and updating themselves during the training phase:

$$\mathcal{P}_c = [a_1^{(t-n+c)}, a_2^{(t-n+c)}, \ldots, a_l^{(t-n+c)}]^T \tag{14}$$

where $c < n - 1$ The next convolutional layers scan low-level features in the previous layers in order to model high-level features. This filtering operation generates an output feature map. The kth feature map resulting from the convolution operation is computed as follows:

$$\mathcal{M}_c^k = ReLU(\mathcal{W}_c^k * \mathcal{P}_c^k) + \mathcal{B}_c^k, \tag{15}$$

where $\mathcal{W}$ denotes the weights, which are updated using a back-propagation step so that the loss decreases and $\mathcal{B}$ denotes the bias. This step is followed by applying a non-linear function, here a Rectified Linear Unit (ReLU). This activation function computes $f_{ReLU}(\mathcal{M}) = max(0, \mathcal{M})$ to incorporate nonlinearity by transforming all negative values to zero.

To reduce the input dimensionality and provide an abstract form of the representation, the max-pooling process is applied with pool size m over the output of $f_{ReLU}(\mathcal{M})$. In order to preserve all the local information. Indeed, max-pooling reduces the input parameters to a lower dimension by taking the maximum values of the range in pooling size. One of the important issues of any learning algorithm is over-fitting. To deal with this issue, a regularization parameter (dropout) is used. In the training step, to avoid overfitting some outputs of the previous layer are randomly discarded and then the remaining information is fed as the inputs to the next layer.

3.3. Attention mechanism

The outputs of the CNN are fed into an attention mechanism layers to take advantage of these local subsequences to retrieve the meaningful parts of the short-term sequences. Attention layers turn the input sequences into a matrix to recognize relevant parts of these latter to select the appropriate class. The attention model uses the convolved activity sequences matrix $\mathcal{M}$ for each subject s to learn a weight matrix $\mathcal{U}$, so-called attention matrix. The matrix $\mathcal{U}$ components represent the relative significance (probability) of the past activity sequences for the future sequences. To learn the attention matrix $\mathcal{U}$, a fully-connected neural network is used with one hidden layer and a *linear* function $\mathcal{L}$ as activation function. The attention model matrix $\mathcal{U}$ is defined as:

$$\mathcal{U} = \zeta_a(\eta(\mathcal{M})) = \mathcal{L}(\mathcal{W}\eta(m) + \mathcal{B}) \tag{16}$$

where η stands for the projection function for the inputs and the hidden neurons, $\mathcal{W}$ represents neurons weights, m stands for the vectorized convolved activity matrix $\mathcal{M}$, and ζ_a is the attention learning network, A weighted matrix $\mathcal{M}_{\mathcal{U}} = \mathcal{M} \odot \mathcal{U}$ is finally computed as the input of the GRU network.

3.4. Temporal feature extraction

A GRU (Gated Recurrent Unit) is used in the model architecture to excerpt the temporal features of the dataset. The GRU networks use gated neurons to seize both the short-term and the long-term memories, which are crucial in activity recognition. As reported in Cho et al. (2014), GRU models obtain comparable results regarding LSTM using a fewer number of neurons. The GRU network inputs for short-term $(t-n)$ and long-term (q) features are $T_q = [g_1^{(t-n+q)}, g_2^{(t-n+q)}, \ldots, g_l^{(t-n+q)}]$ in $\mathcal{G}_{\mathcal{A}}$

where $T = (T_0, T_1, \ldots, T_{n-1})$. In every historical input, the kth temporal layer output is denoted as $H_k = (H_0^k, H_1^k, \ldots, H_{n-1}^k)$. The temporal features are generated using the following equations:

$$\mathcal{Z}_q = \sigma_g(W_z T_q + U_z H_{q_1} + b_z) \tag{17}$$

$$\mathcal{R}_q = \sigma_g(W_r T_q + U_z H_{q_1} + b_r) \tag{18}$$

$$\mathcal{H}_q = (1 - \mathcal{Z}_q) \odot \mathcal{H}_{q_1} + \mathcal{Z}_q \odot \sigma_h(W_h T_q + U_h(\mathcal{R}_q \odot \mathcal{H}_{q-1}) + b_h) \tag{19}$$

where $\sigma(.)$ and $\sigma_h(.)$ stands for the activation functions, and $\odot$ is an element-wise product. The reset and the update gates at q are denoted as $\mathcal{R}$ and $\mathcal{Z}_q$, respectively. In cases where the values of $\mathcal{Z}$ approach zero and those of $\mathcal{R}$ approach one, the network is inclined to learn long-term and inversely to learn short-term memories. Two GRUs layers are used in the proposed architecture.

3.5. Final fully connected layer

Finally, all features are combined and fed into a classification layer (i.e., fully connected layer). A fully connected layer is used to make the final decision by matching sequences of activities $g^{(n)}$ with predicted sequences of activity classes $y^{(n)}$. A cross-entropy (log loss) is used to evaluate the performance of the multi-class classifier. It is calculated as follows:

$$\mathcal{M} = -1/N \sum_{i=1}^{N} [y_i^{n,c} log(\hat{y}_i^{n,c}) + (1 - y_i^{n,c}) \times log(1 - \hat{y}_i^{n,c})] \tag{20}$$

where $y_i^{n,c}$ stands for the true classes and $\hat{y}_i^{n,c}$ are the predicted classes c for the input activity sequences $g_i^{n,c}$. A gradient descent $\mathcal{M}$ minimizes the learning error using a back-propagation process by updating all learning parameters of the model Θ, in the opposite direction:

$$\nabla\Theta = -\gamma \frac{\delta\mathcal{M}}{\delta\Theta} \tag{21}$$

where γ stands for the learning rate.

4. Dataset description

4.1. Experiment protocol

Ten healthy subjects (gender: 9 males, 1 female; age: 28 ± 4 years; height: 1.74 ± 0.05 m; weight: 66.8 ± 10 kg) have participated in this study. Nine subject's data are used for the learning process and the tenth subject data are used for the evaluation process. The experiments are conducted by each subject wearing the exoskeleton and performing five different gait modes: level walking (LW), ramp up (RU), ramp down (RD), stair up (SU), and stair down (SD) as shown in Fig. 4. During the data acquisition, the Angelegs are in a passive mode, the segment lengths of the robot (shank and thigh) are adjusted to fit the subject's height and to avoid misalignment between human and robot joints (see Fig. 5).

4.2. Wearable robot

Fig. 3 represents a statistical analysis of the data as a function of the number of samples per subject/class (top) to discern patterns and trends. As can be seen, the level walking class has more samples than the other classes. The samples are almost equally distributed for the other classes. The box plots show the distribution of the dataset and the skewness by displaying the quartiles and means by class (bottom). The quartiles and mean of the walking class are higher than those of the other classes. The dataset is provided here.[1]

[1] http://www.lissi.fr/;dataset-for-gait-mode-recognition-on-activities-of-daily-living/.

4.2.1. Lower limb exoskeleton

Angelegs is a full lower limb wearable robot designed to assist elderly and post-stroke patients to recover their locomotion functions (see Fig. 4). The weight of the whole exoskeleton is about 12 Kg. The main parts of the Angelegs are the exoskeletal frames, hardware control system, sensor system, and power unit. Ten degrees of freedom (DoFs) constitute the exoskeletal frames that are attached using straps to the wearer's waist and legs. Each limb has three DoFs, one at the hip joint level, one at the knee joint level, and one at the ankle joint. The shank's and thigh's lengths are adjustable for different wearer's heights (165–190 cm). The hip and knee joints of each limb are actuated in the sagittal plane (flexion/extension movements) using two SEAs Actuators module. All the remaining DoFs are passive. To record the accelerations, velocities, and orientations of the subject's feet while wearing the exoskeleton, two IMus (MTw Awinda, Xsens, Netherland) on both shoes (see Fig. 4(A)) are used. The IMU consists of a 3-D accelerometer, a 3-D gyroscope, and a 3-D magnetometer. An extended Kalman filter is already built-in the IMUs to compute the 3D orientation estimation with no drift for both static and dynamic movements (Paulich et al., 2018). The data are acquired from two IMUs using a 100 Hz sampling frequency.

4.3. Features extraction

The feature extraction process is needed to calculate representative kinematic features from raw data such as acceleration, quaternions, and gyroscope data. To process the features, reference and non-reference foot are defined based on the detection of swing and stance foot respectively. During the double stance phase, the prior stance foot is designed as the reference foot. The processed features consist of the absolute horizontal position ($\check{p}_h$), vertical position ($\check{p}_v$), absolute horizontal velocity ($\check{v}_h$), and vertical velocity ($\check{v}_v$) of the non-reference foot.

The choice of these features is justified from the fact that the pathological gait data acquired from patients with different characteristics such as genders, ages, body height and body weight showed a direct link between the kinematics and the gait modes (Lencioni et al., 2019). Moreover, It is observed that the relative vertical and horizontal position and velocity of the foot represent the main features of the locomotor tasks (Huo et al., 2018b). Thus, these kinematic features are considered in this study rather than raw data. The processed features can be calculated as follows:

$$\begin{aligned} \check{p}_h &= \sqrt{(r_{nf,x} - r_{rf,x})^2 + (r_{nf,y} - r_{rf,y})^2} \\ \check{p}_v &= r_{nf,z} - r_{rf,z} \\ \check{v}_h &= \sqrt{(v_{nf,x})^2 + (v_{nf,y})^2} \\ \check{v}_v &= v_{nf,z} \end{aligned} \tag{22}$$

where, $r_{rf,(x,y,z)}$, $r_{nf,(x,y,z)}$ and $v_{rf,(x,y,z)}$ represent reference and non-reference foot position and reference foot velocity, which corresponds to the estimates in x, y and z axes, respectively.

4.3.1. Reference foot detection

The gait cycle can be divided into two main phases: the stance phase and the swing phase. The stance phase starts at the initial contact (IC) of the heel and ends just after toe-off (TO) event. During the swing phase, the swing leg rises the foot to ensure sufficient clearance of the foot and anticipates the foot position for the IC event of the next step. Therefore, The relative motions of the swing foot are considered to estimate the gait modes. To detect foot state (i.e., reference/non-reference foot), the swing phase detector algorithm was used in this study. This algorithm has the advantages to be based only on the use of an IMU to detect the swing phase from the stance phase with a relatively high accuracy. For

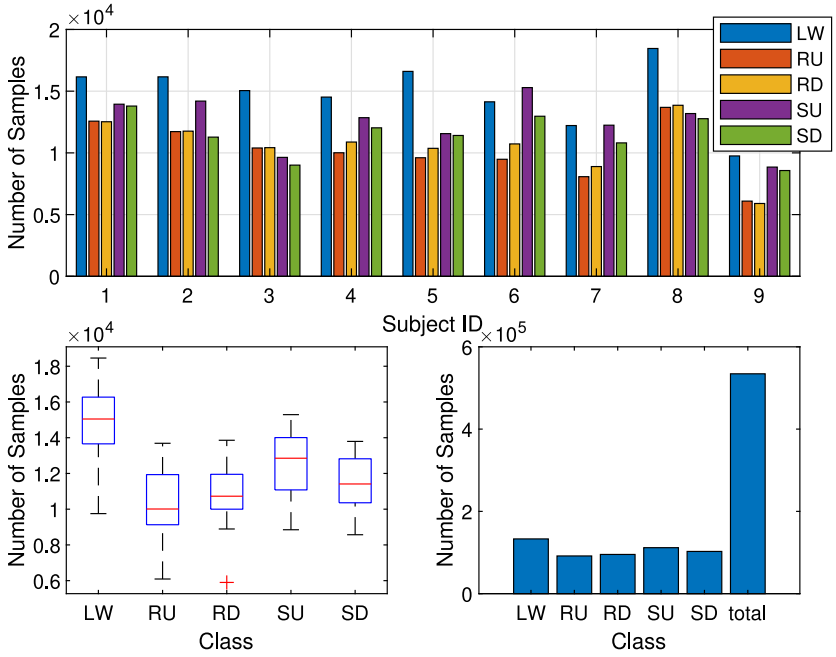


Fig. 3. Training dataset: Statistical analysis of the data as a function of the number of samples per subject/class (top), box plots show the distribution of the dataset and the skewness by displaying the quartiles and means by class (bottom).

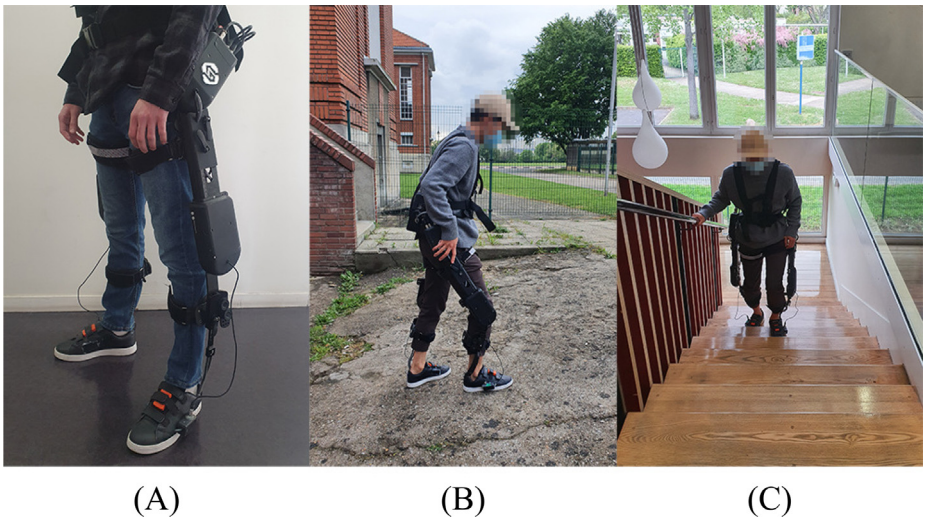


Fig. 4. Environments for different gait modes, (A): 19 m hallway for level walking (LW); (B) 7° ramp-way for ramp locomotion (RU/RD); (C): 10 *steps* stairs for stair climbing (SU/SD).

more details the reader can refer to Huo et al. (2018a). The reference foot position $r_{rf,\,k} \in [r_{rf,x}, r_{rf,y}, r_{rf,z}]$ can be defined as:

$$r_{rf,\,k} = \begin{cases} r_{r,\,k}, & \text{for } (D_{r,\,k} = 0) \bigwedge (D_{l,\,k} = 1), \\ r_{l,\,k}, & \text{for } (D_{r,\,k} = 1) \bigwedge (D_{l,\,k} = 0), \\ r_{rf,\,k-1}, & \text{Otherwise}, \end{cases} \tag{23}$$

In (23), D_r and D_l show the foot states as binary values (i.e., stance phase defined as 0, and swing phase defined as 1.) which are right foot and left foot, respectively. The reference state $D_{nf,\,k} \in \{0, 1\}$ is defined as the binary value reference foot state at time k as:

$$D_{nf,\,k} = \begin{cases} 0, & \text{for } (D_{r,\,k} = 0) \bigwedge (D_{l,\,k} = 1), \\ 1, & \text{for } (D_{r,\,k} = 1) \bigwedge (D_{l,\,k} = 0), \\ D_{nf,\,k-1}, & \text{Otherwise}, \end{cases} \tag{24}$$

where, $D_{nf,\,k} = 0$ and $D_{nf,\,k} = 1$ refer to which is the non-reference left and right foot, respectively. Therefore, the non-reference foot position $r_{nf,\,k} \in [r_{nf,x}, r_{nf,y}, r_{nf,z}]$ and velocity $v_{nf,\,k} \in [v_{nf,x}, v_{nf,y}, v_{nf,z}]$ can be defined as:

$$\begin{aligned} r_{nf,\,k} &= r_{r,\,k} + (1 - D_{nf,\,k})(r_{l,\,k} - r_{r,\,k}) \\ v_{nf,\,k} &= v_{r,\,k} + (1 - D_{nf,\,k})(v_{l,\,k} - v_{r,\,k}) \end{aligned} \tag{25}$$

4.3.2. Extended Kalman filter

The method of Pedestrian Dead-Reckoning (PDR) is implemented in an extended Kalman-based framework (Foxlin, 2005). This method ensures correction of the errors resulting from an Inertial Navigation System by estimating the bias of the acceleration and integral error when calculating the velocity and position though integration. To calculate the features, the EKF process loop is used, for each iteration, as follows:

1. Compensation of the bias from the raw acceleration data and calculation of the net acceleration in global frame by removing gravitational acceleration component.

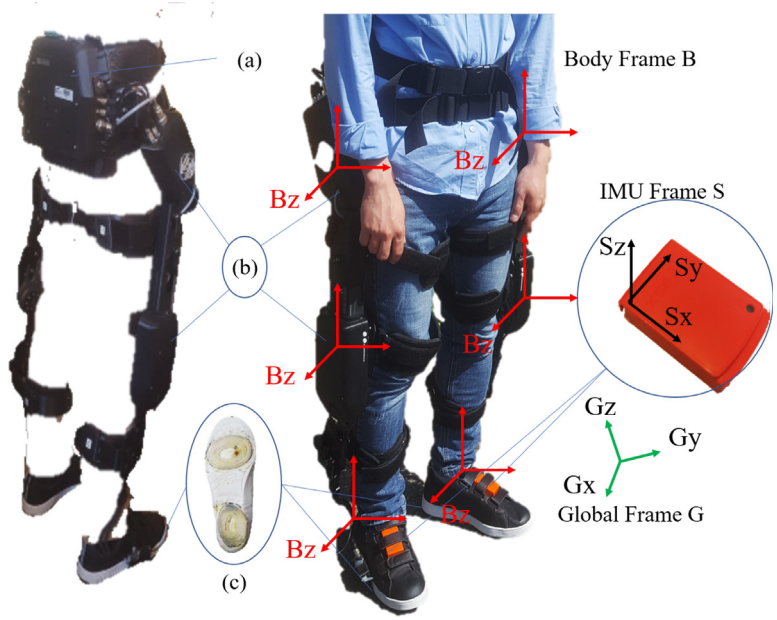


Fig. 5. Lower limb exoskeleton: Angelegs. (a) Controller and power unit. (b) SEA hip and knee actuators. (c) Air Pressure GRF. (In standing position the body frame B, is designed as follows: origin, the center of rotation; X, forward in the sagittal plane; Y, up; Z, right in frontal plane; the global frame G, is defined as: X, local magnetic north direction; Y: local magnetic west direction; Z, vertical direction).

2. Integration of the net acceleration value to calculate velocity and position.
3. Detection of the foot state by using the phase detector (i.e., stance/swing phases).
4. Estimation of the state error based on the state vector.
5. Compensation of the error in velocity and position, and calculation of the features from reference foot to non-reference foot

To compensate bias from the raw acceleration, the bias term δa of the acceleration is subtracted and net acceleration can be calculated by subtracting gravity component.

$$\check{a}_k = {}^G\boldsymbol{R}_{S,k}(a_{imu,k} - \delta a_k) - g \in \mathbb{R}^{3\times1} \tag{26}$$

where, ${}^G\boldsymbol{R}_{S,k}$ is the rotation matrix from global frame to sensor frame measured at time k, a_{imu} is the acceleration value measured in sensor frame, and g is the gravitational acceleration vector (i.e., $g = [0\ 0\ -9.8]^T$).

$$\begin{aligned} v_k &= \check{v}_{k-1} + (\check{a}_k + \check{a}_{k-1})\frac{dt}{2} \\ r_k &= \check{r}_{k-1} + (v_k + \check{v}_{k-1})\frac{dt}{2} \end{aligned} \tag{27}$$

The state vector is utilized to cope with the error through state vector at discretized time k for extended Kalman filter (EKF) as:

$$\delta x_{k|k} = [\delta r_k^T, \delta v_k^T, \delta a_k^T]^T \in \mathbb{R}^{9\times1} \tag{28}$$

where, δr and δv are the estimated three axis elements error of the position and velocity, respectively. The estimated error state vector is updated at each iteration and initialized to zero when the foot is in stationary state (i.e., stance phase). Following the state vector in (28), the state space equation of the model at prediction step is:

$$\delta x_{k+1|k} = F_k \delta x_{k|k} + \omega \tag{29}$$

where, ω represents the process noise with the covariance matrix $Q = E(\omega, \omega^T)$ and F_k is the discrete-time dynamics system state transient matrix as:

$$F_k = \begin{pmatrix} I_{3\times3} & dtI_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & dt\,{}^G\boldsymbol{R}_S^k \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} \end{pmatrix} \in \mathbb{R}^{9\times9} \tag{30}$$

where, $I_{3\times3}$ and $0_{3\times3}$ are defined as identity and zero matrices, respectively. Also, in (30), dt represents the sampling time and ${}^G\boldsymbol{R}_S^k$ is represents the rotation matrix from global frame to sensor frame at time k. The measurement model is:

$$y_{k+1} = H\delta x_{k+1|k} + v \tag{31}$$

where, H represents the measurement matrix and v represents the measurement error with the covariance matrix $R = E(v, v^T)$, which is assumed as white noise with normal probability distribution. Therefore, the estimation state can be expressed as:

$$\delta x_{k+1|k+1} = \delta x_{k+1|k} + K(z_k - H\delta x_{k+1|k}) \tag{32}$$

Since the velocity during the stance phase is defined as zero, then z_k is defined as the error measurement state as: $z_k = v_k - [0, 0, 0]$ during the stance phase. And the Kalman gain K_k at time k is calculated as:

$$K_k = P_{k+1|k}H^T(HP_{k+1|k}H^T + R)^{-1} \tag{33}$$

where, $P_{k+1|k}$ is the predicted error covariance matrix at time k+1 based on measurement data at time k as:

$$P_{k+1|k} = F_k P_{k|k} F_k^T + Q \tag{34}$$

The estimated covariance matrix $P_{k|k}$ can be update using the Josheph form equation as:

$$\begin{aligned} P_{k+1|k+1} =& (I_{9\times9} - K_k(-H))P_{k+1|k}(I_{9\times9} - K_k(-H))^T \\ &+ K_k R K_k^T \end{aligned} \tag{35}$$

Therefore, the modified foot velocity and position can be calculated as:

$$\begin{aligned} \check{v}_k &= v_k - \delta v_k \\ \check{r}_k &= r_k - \delta r_k \end{aligned} \tag{36}$$

For the experiments, the measurement matrix is defined as: $H = [0_{3\times3}, I_{3\times3}, 0_{3\times3}]$, the measurement noise covariance matrices for both IMUs are set as: $R = \mathrm{diag}(0.001I_{3\times3}, 0.1I_{3\times3}, 0.001I_{3\times3}) \in \mathbb{R}^{9\times9}$, for both foot, system covariance matrices are set as: $Q = \mathrm{diag}(0_{3\times3}, 0.0001I_{3\times3}, 0_{3\times3}) \in \mathbb{R}^{9\times9}$, and both initial error covariance matrices are set as: $P = \mathrm{diag}(0_{3\times3}, 0.0001I_{3\times3}, 0_{3\times3}) \in \mathbb{R}^{9\times9}$ by trial and error methods. the calculated features is shown in Fig. 6.

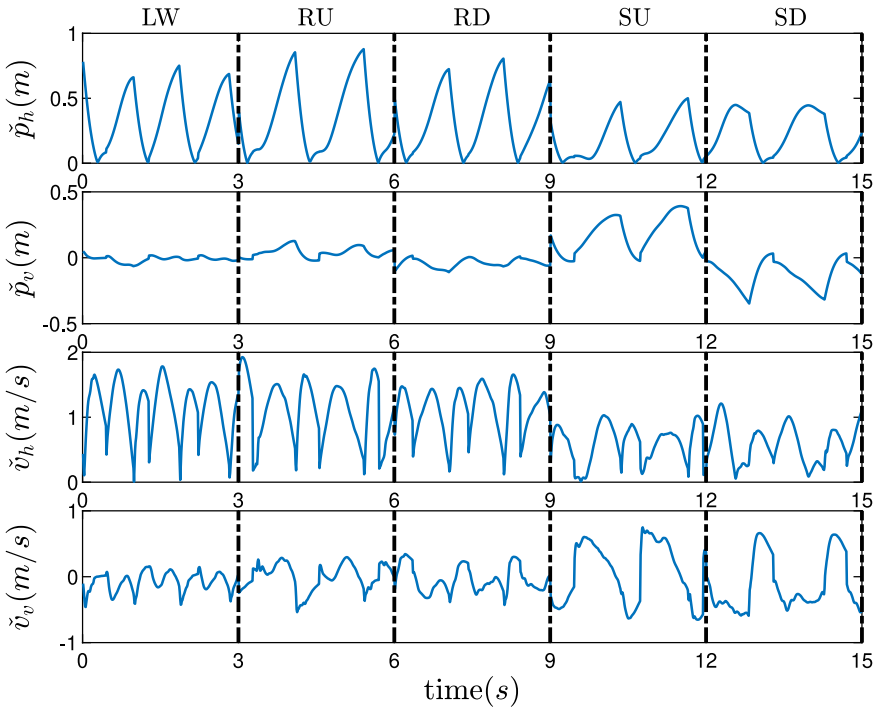


Fig. 6. Description of the refined features according to the different gait modes. Each time transient features was cropped only three seconds during the movement and concatenated each others.

Table 1
Hyper-parameters of methods in Table.

Method	Parameter	Values
Fuzzy-CNN-GRU-Att	Input layer dimension	(371 341,)[a], (300,)[b], (24,)[c]
	Number of convolutional blocks	3
	Number of Maxpooling layers	3
	Kernel size of Conv. layers in convolutional blocks	(3, 3)
	Sliding stride of Conv. layers in the blocks	(1, 1)
	Number of filters in Conv. layers in the blocks	64
	Number of attention blocks	3
	Number of GRU blocks	2
	Dimensionality of the GRU output space	64
	Number of neurons in GRU Dense output layer	(64)
	Dropout	0.1
	Number of neurons in final Dense output layer	(1)
CNN-GRU-Att	Input layer dimension	(371 341,)[a], (300,)[b], (24,)[c]
	Number of attention blocks	3
	Number of GRU blocks	2
	Dimensionality of the GRU output space	64
	Number of neurons in GRU Dense output layer	(64)
	Dropout	0.1
	Number of neurons in final Dense output layer	(1)
LSTM	Input Layer dimension	(371 341,)[a], (300,)[b], (24,)[c]
	Dimensionality of the LSTM output space	64
	Number of LSTM blocks	4
	Dropout	0.1
	Number of neurons in Dense output layer	(1)

[a]Number of samples in training set (100 or 300).
[b]Number of previous sequence steps taken into account (look back).
[c]This value represents all the features.

5. Experimental results

5.1. Evaluation of the gait mode detection

The proposed model is trained, validated, and tested respectively on 371 341, 105 781, and 110 753 samples of activity sequences representing 9 subjects. Data of an additional subject with 48 696 samples are used as the evaluation set. The architecture of the proposed model (Fuzzy-CNN-GRU-Att.) is reported in Table 1.

Fig. 7 portrays the trends of loss and accuracy values over the training dataset. The accuracy values increase, while the loss values progressively decrease over the epochs. The whole framework converges after about 50 epochs.

The performance of the proposed models is evaluated over test and evaluation sets using the following metrics:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{37}$$

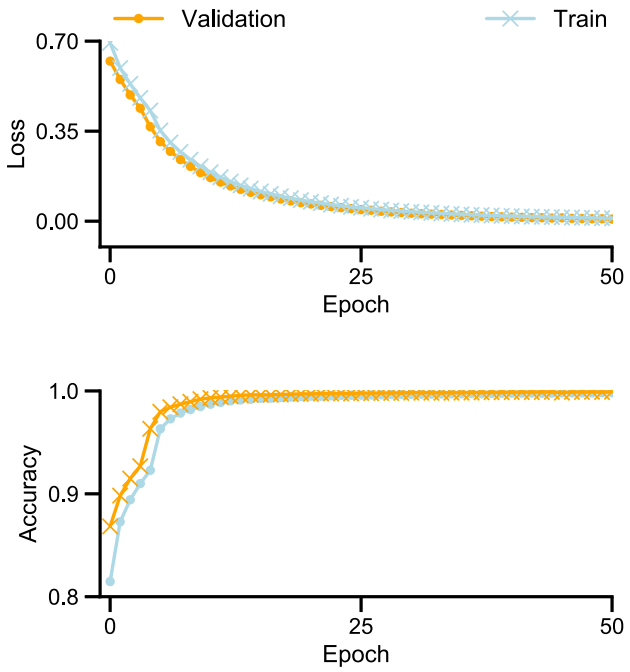


Fig. 7. The accuracy and loss progress throughout epochs for train and validation sets (model look back = 300).

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{38}$$

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{39}$$

where TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively.

5.2. Evaluating the effect of previous activity sequences on activity classes predictions

The activity sequences with different look-back lengths, i.e., 100 and 300 are fed as input into the model to evaluate the previous activity sequences effect on model prediction. Fig. 8 portrays the confusion matrices related to two different look-backs model performance. The model performance slightly changes from the time step (look back) 100 towards 300. This shows the previous sequences impact on the next ones. The same observation is perceptible from Fig. 9. All metric values increase with the length of the time steps.

The improved accuracy scores of the proposed model endorse the effectiveness of the model. The same trend is observable for log loss values. However, increasing the time step sequence does not always imply high performance in prediction. This statement is verified using time steps of 400 and 500. Moreover, a higher look back implies higher computational resource requirements. These results suggest the use of a model with a look back 300 since it is a fair compromise between the reliability of the model and the computational resources (time/cost), the look back 300 is considered in the rest of this work as the reference model. This choice is also motivated by the performance of the model.

For the model with the look back 300, the true positive rate as a function of the false positive rate represented by ROC (Receiver Operating Characteristic) and the area under the ROC curve (AUC) curves for all classes in Fig. 11. The precision, recall, and F1-score for all classes are reported in Table 2. Class 3, 4, and 5 represent the best prediction scores and class 2 shows the lowest scores. Note that the ROC and AUC curves have limited value when used with unbalanced data. As mentioned in Carrington et al. (2020), these curves describe

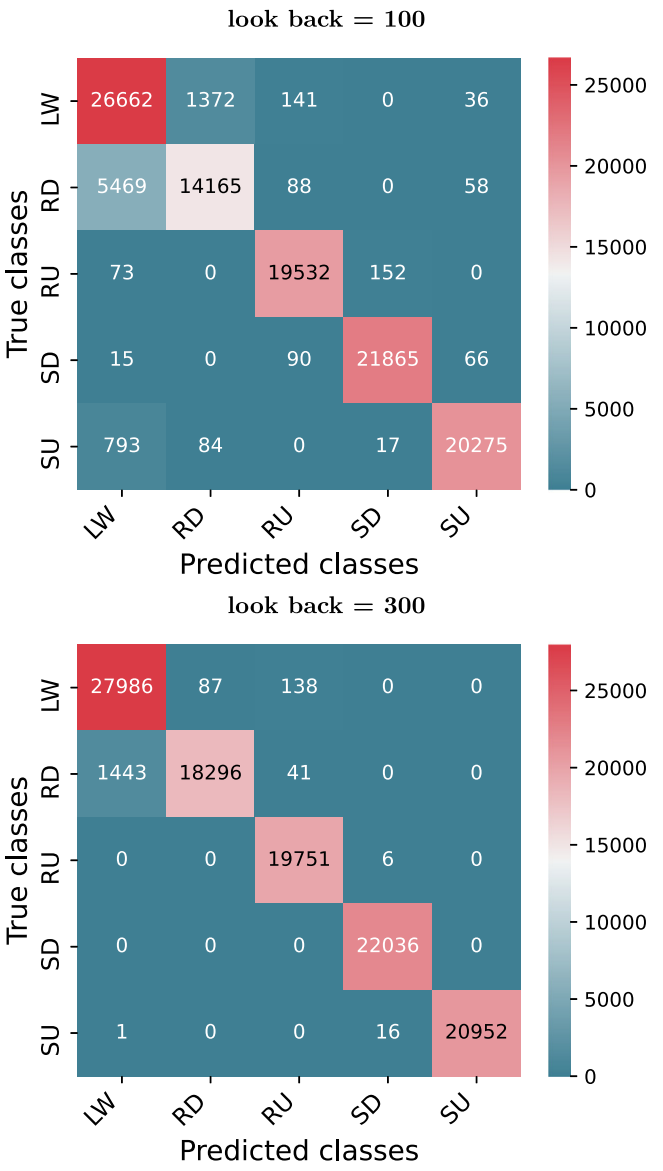


Fig. 8. Test dataset: Confusion matrices for the proposed model with look backs 100 and 300.

Table 2
Test dataset: Precision, recall, F1-score and accuracy obtained for 5 classes (look-back = 300).

Class	Precision	Recall	F1-score	Support
1	0.95	0.99	0.97	28 211
2	1.00	0.92	0.96	19 780
3	0.99	1.00	1.00	19 757
4	1.00	1.00	1.00	22 036
5	1.00	1.00	1.00	20 969
Accuracy			0.98	110 753

how an adjustable threshold results in changes in false positives and false negatives. However, only a portion of the ROC curve and AUC are informative when used with unbalanced data. This problem is addressed in Carrington et al. (2020) by applying a new partial match AUC and a new partial c-statistic for the ROC.

Fig. 10 depicts ROC curves for all classes for the model with the look back 100. The precision, recall, and F1-score for all classes are reported in Table 3. Class 3 and 4 represent the best prediction scores and class 2 shows the lowest scores.

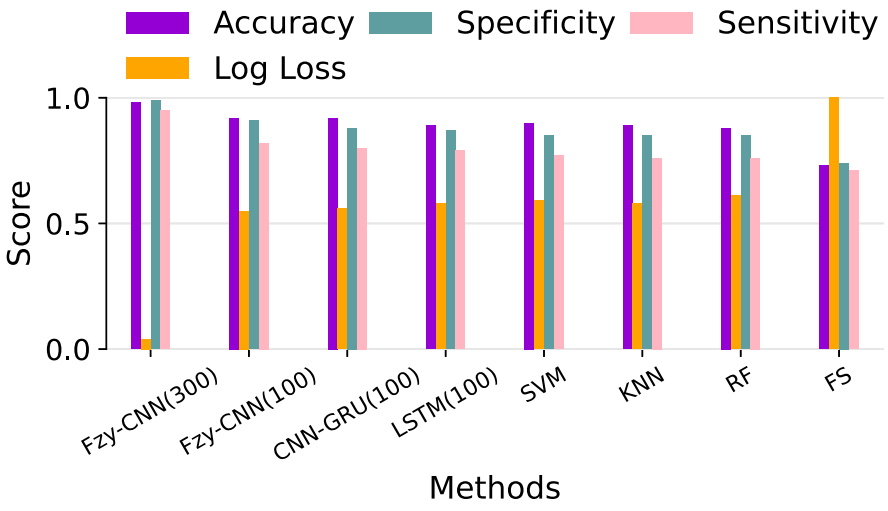


Fig. 9. Test dataset: Comparison of different models for activity recognition.

Table 3
Test dataset: Precision, recall, F1-score and accuracy obtained for 5 classes (look back = 100).

Class	Precision	Recall	F1-score	Support
1	0.81	0.95	0.87	28 211
2	0.91	0.72	0.80	19 780
3	0.98	0.99	0.99	19 757
4	0.99	0.99	0.99	22 036
5	0.99	0.96	0.97	21 169
Accuracy			0.92	110 753

Table 4
Comparative performance scores of test and evaluation data for the proposed approach and baseline methods (the best performances are shown in bold).

Method	Accuracy		Log loss		Specificity		Sensitivity	
	Test	Eval.	Test	Eval.	Test	Eval.	Test	Eval.
F-C-G-A[a] (300)	0.98	**0.99**	0.04	**0.10**	0.99	**0.98**	0.95	**1**
F-C-G-A[a] (100)	0.93	**0.95**	0.55	**0.23**	0.91	**0.94**	0.82	**0.92**
C-G-A[b] (100)	0.92	0.94	0.56	0.31	0.88	0.93	0.980	0.89
LSTM (100)	0.89	0.91	0.58	0.31	0.87	0.90	0.79	0.83
SVM	0.90	0.92	0.59	0.33	0.85	0.89	0.77	0.80
KNN	0.89	0.91	0.58	0.32	0.85	0.89	0.76	0.79
Random forest	0.88	0.89	0.61	0.37	0.85	0.89	0.76	0.79
Feature selection	0.73	0.75	1.2	0.92	0.74	0.79	0.71	0.79

[a]Fuzzy-CNN-GRU-Attention.
[b]CNN-GRU-Attention.

Table 5
Comparison of overall performance scores for proposed and other methods (the best performances are shown in bold).

Approach	Accuracy	F1-score
Fuzzy-CNN-GRU-Attention	**0.9978**	**0.9991**
FDCNN (Gomathi et al., 2022)	0.9789	0.9795
Fuzzy-CNN-3D-HAR (Banerjee et al., 2020)	0.9791	Not reported
FTW-LSTM-CNN (Hwang et al., 2021)	0.744	Not reported

5.2.1. Evaluating the proposed model with regard to baselines

To compare the proposed model with the baseline methods, the following deep neural network architectures are used: fuzzy convolutional attention-based GRU (Fuzzy-CNN-GRU-Att), convolutional attention-based GRU (CNN-GRU-Att), and long short term memory (LSTM). The architecture of the baseline methods are given in Table 1. The observation here is that the model using convolutional and attention layers with the GRU network allows detecting inherent temporal and local properties within activity sequences as it outperforms both LSTM and CNN-GRU-Att models. Consequently, the proposed model is suitable for unveiling temporal and local motifs along with inter-classes dependencies, which is the case of the analyzed daily living activities.

The classical baseline methods such as Support Vector Machine (SVM) (Suykens and Vandewalle, 1999), k Nearest Neighbors (KNN) (Goldberger et al., 2005), Random Forest (RF) (Breiman, 2001) and Feature Selection (FS) (Guyon and Elisseeff, 2003) are applied on the test and evaluation sets. The results are reported in Table 4. Fig. 9 shows that the proposed model Fuzzy-CNN-GRU-Attention (F-C-G-A) with both look-backs 100 and 300, largely outperforms all the baseline methods including deep neural network models on both tests and evaluation sets.

As mentioned earlier, although the model with time step 300 performs better than the model with time step 100, the latter is considered for the conducted study. Among traditional baseline methods, SVM and KNN show the best performances and FS represents the lowest results on the test and evaluation sets for all parameters.

The main conclusion of these results is that baseline methods can only recognize linearity in the data while the stochastic and non-linear nature of the sequences will be neglected, as in the case of the transition from one activity to another. This is because baseline methods neglect the temporal and complex aspects of activity sequences.

5.2.2. Evaluating the proposed model with regard to other methods

As mentioned in the introduction, despite the advantages of the deep neural network and its combination with the fuzzy system, this type of approach has not been widely used or explored in robotics and HAR research using sensor data. However, the performance of the proposed approach is compared to some similar approaches/case studies in terms of accuracy and F1-score. The comparative results are given in Table 5.

The fuzzified deep convolutional neural network (FDCNN) architecture based on fuzzy rules, proposed in Gomathi et al. (2022), is applied to smartphone sensor data for human activity recognition. It reports 0.9789 for accuracy and 0.9795 for F1-score. In this approach, fuzzy logic is only used for simple data normalization by merging the *lambda*-max method to initialize the weights. The interest and benefit of the fuzzy system, such as taking into account the incompleteness, noisiness or incorrectness of the available data without the need for extensive data preprocessing. is not put into practice in this approach. The proposed initialization weight method could be replaced by another method without much change in the results.

While we only use the sensor data in our approach, the fuzzy fusion approach proposed in Banerjee et al. (2020) uses the skeleton images to combine the CNN decision scores and generate the final decision.

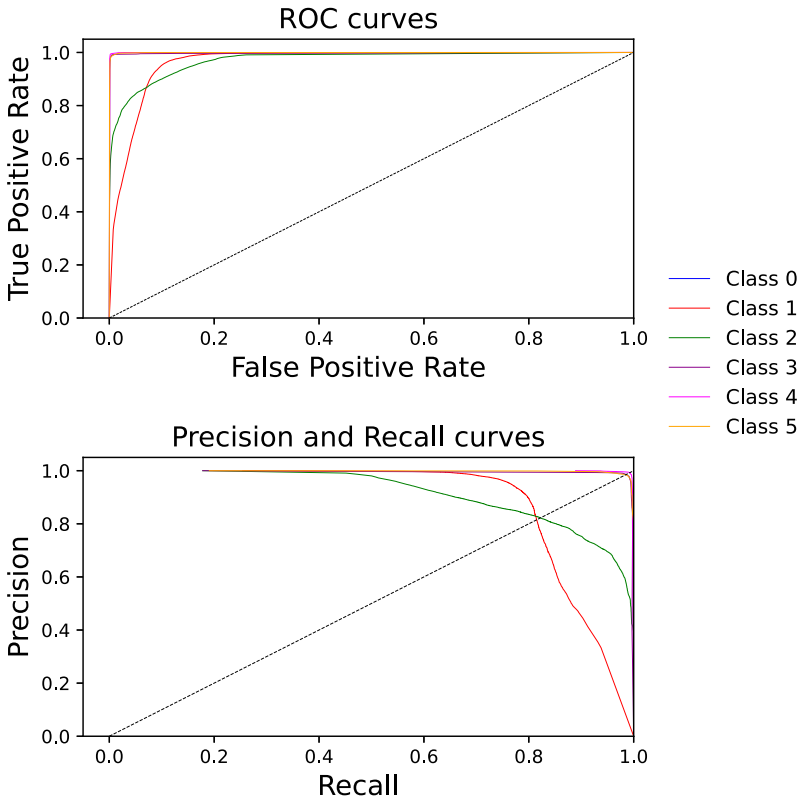


Fig. 10. Test dataset: The proposed model with time step 100 for 5 classes (top) ROC curves, (bottom) precision/recall curves.

Moreover, in the paper in Banerjee et al. (2020), the fuzzy system is used in the ultimate layers to generate the final decision while in our approach, it is used in the early layers to generate enriched features. The highest accuracy performance reported in Banerjee et al. (2020) is 0.9791. The method proposed in Hwang et al. (2021) is based on direct causality and fuzzy temporal windows, the merged feature are then fed to LSTM and CNN. This approach uses a large amount of historical data, which is not always available. In addition, the proposed approach requires an extensive data preprocessing, which is not a viable method for real-world data. The highest reported accuracy is 0.744.

All of the above approaches do not account for incompleteness, noisiness or incorrectness of sensor data without the need for extensive data preprocessing. They do not allow for the generation of enriched features, which may result in inferior performance in the recognition of spatio-temporal patterns. They have difficulties in handling the inherent characteristics of HAR data, such as ambiguity and limited amount of data. Our proposed approach outperforms the accuracy score of all these approaches in dealing with these problems.

5.3. Assistive torque generation

To provide physical support to the wearer's legs by the exoskeleton, the applied assistive strategy varies based on the estimated gait mode. Four assistive strategies are used in this study: partial gravity compensation (GC), virtual spring (VS), virtual damper (VD) and zero impedance (ZI) . According to the detected gait modes and phases, the assistive control strategy is selected. During the swing phases, in order to reduce the effect of the actuators' frictions (Oh et al., 2015), zero impedance (ZI) control is applied regardless of the gait mode. To reduce the muscular power During the stance phases, gravity compensation (GC) is applied for all the considered gait modes. Furthermore, during the stance phase, the subjects are assisted with different strategies according to the gait modes. For instance, the knee joint stiffness is enhanced using a virtual spring algorithm for weight support during level walking and ramp ascent modes, Unluhisarcikli et al. (2011). Meanwhile, both the knee and hip joints are assisted using a virtual spring algorithm for the same purpose in the case of stair ascent mode. The passivity of the joints during the initial stance of level walking and stair/ramp descent modes are enhanced using the virtual damper algorithm (Murray et al., 2014).

5.3.1. Assistive strategies

The following methods are applied for the assistive torque generation.

1. **Gravity compensation:** The partial gravity compensation is used in order to reduce the human torques of the stance leg, thus the ankle joint is taken as pivot. Assuming a five link model of the human body in the sagittal plane (i.e., left shank and thigh, torso, as well as right shank and thigh). The required torque for the hip and knee joints to support the body's gravity is expressed as follows (Craig, 2009):

$$\tau_G = \eta \begin{bmatrix} r_4 \times f_4 + r_{g,4} \times G_4 + r_{g,4} \times G_3 \\ r_2 \times f_3 + r_{g,2} \times G_2 \end{bmatrix} \tag{40}$$

where η represents the assistance ratio, f_i the force acting on the joint i by the (i + 1)th link (i.e., segment), and G_i the gravitational force of link i. r_i and $r_{g,i}$ denote the locations of the joint and the center of gravity, respectively.

2. **Virtual damper:** Based on the virtual damper, the assistive joint torques are calculated by:

$$\tau_d = \begin{bmatrix} \beta_{mk} * (\dot{\theta}_k - \dot{\theta}_{ek}) \\ \beta_{mh} * (\dot{\theta}_h - \dot{\theta}_{eh}) \end{bmatrix} \tag{41}$$

where β_{mh} and β_{mk} represent the stiffness parameters of the hip and knee joint, respectively, for the gait mode $\dot{\theta}_h$ and $\dot{\theta}_k$ denote the measured hip and knee joint angular velocity, while $\dot{\theta}_{eh}$ and $\dot{\theta}_{ek}$ represent the desired angular velocities of the hip and knee joints, respectively.

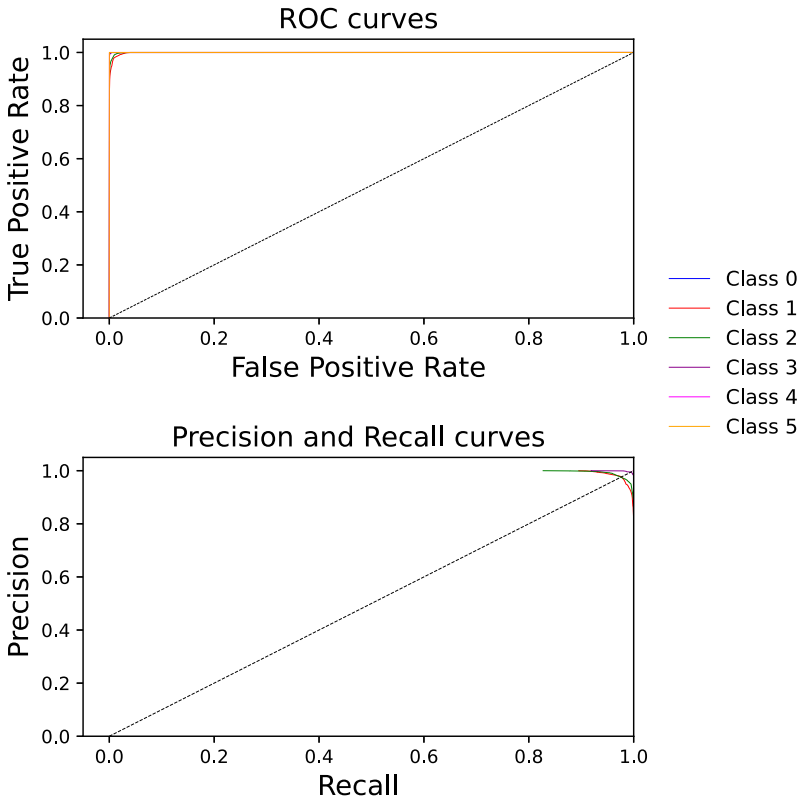


Fig. 11. Test dataset: The proposed model with time step 300 for 5 classes (top) ROC curves, (bottom) precision/recall curves.

3. **Virtual spring:** Based on the virtual spring, the assistive joint torques are calculated by:

$$\tau_s = \begin{bmatrix} \alpha_{mk} * (\theta_k - \theta_{ek}) \\ \alpha_{mh} * (\theta_h - \theta_{eh}) \end{bmatrix} \tag{42}$$

where α_{mh} and α_{mk} represent the stiffness parameters of the hip and knee joint, respectively, for the gait mode θ_h and θ_k denote the measured hip and knee joint angles, while θ_{eh} and θ_{ek} represent the desired angular positions of the hip and knee joints, respectively.

4. **Zero impedance:** When the desired assistive torque is zero, the resistive torque should also be zero. Therefore, in such case, zero-impedance control should be realized by setting $\tau_i = 0$.

5.3.2. Gait phases detection

To implement the aforementioned control strategies according to the detected gait mode, the walking motion is divided into four sub-phases, early stance $S1$, middle stance $S2$, late stance $S3$, and swing phase $S4$. The detection of the sub-phases is achieved by evaluating the GRF of one foot and the forward position of the other foot . The analysis of the features characteristic for each gait phase is as follows:

1. **Early stance** $S1$: During the early stance phase the GRF is relatively large and the forward position is small.
2. **Middle stance** $S1$: During the middle stance phase the GRF is relatively large and the forward position of the two foot is relatively close, so it is almost equal to zero.
3. **Late stance** $S1$: During the late stance phase the GRF is relatively large and the forward position is relatively large.
4. **Swing phase** $S1$: During the swing phase the GRF is small. The forward position of the two foot is not taken into consideration.

The likelihood of each gait phase is calculated based on the membership values of each feature and the fuzzy rules shown in Table 7 using the method proposed in Kong and Tomizuka (2009).

5.3.3. Torque generation

To make the wearer comfortable and stable, sudden change of the assistive torque according to the gait mode and phase must be changed smoothly. Thus, the torque generation method should provide continuous torque:

$$\tau = \begin{bmatrix} \tau_h \\ \tau_k \end{bmatrix} = U_s \begin{bmatrix} M_h \tau_c^T \\ M_k \tau_c^T \end{bmatrix} \tag{43}$$

where $\tau_c = \begin{bmatrix} \tau_G & \tau_{VS} & \tau_{VD} & \tau_{ZI} \end{bmatrix} \in \mathbb{R}^{2\times}$, $U_s = \begin{bmatrix} u_{s1} & u_{s2} & u_{s3} & u_{s4} \end{bmatrix}$ is the gait phases likelihood. $M_h, M_k \in \mathbb{R}^{4\times4}$ denotes the likelihood of the hip and knee joint assistive strategies, respectively. M_h, M_k can be expressed as follows:

$$M_h = \begin{bmatrix} 1 & 0 & u_{sd} + u_{rd} & 0 \\ 1 & u_{sa} & 0 & 0 \\ 1 & u_{sa} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{44}$$

and

$$M_k = \begin{bmatrix} 1 & 0 & u_{sd} + u_{rd} + u_{lw} & 0 \\ 1 & u_{lw} + u_{sa} + u_{ra} & 0 & 0 \\ 1 & u_{sa} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{45}$$

5.4. Model evaluation

The proposed model is applied to an evaluation dataset (subject 10) to further assess the generalizability of the model. The test subject is asked to perform different gait modes (e.g., different step lengths and different step speeds). The subject is also asked to walk at relatively low speeds during experiments since the goal is to mimic the walking of dependent people. Fig. 12 shows the values of accuracy, specificity, and sensitivity metrics for the proposed model with both look back 300 and 100, which outperforms all reference methods, including deep

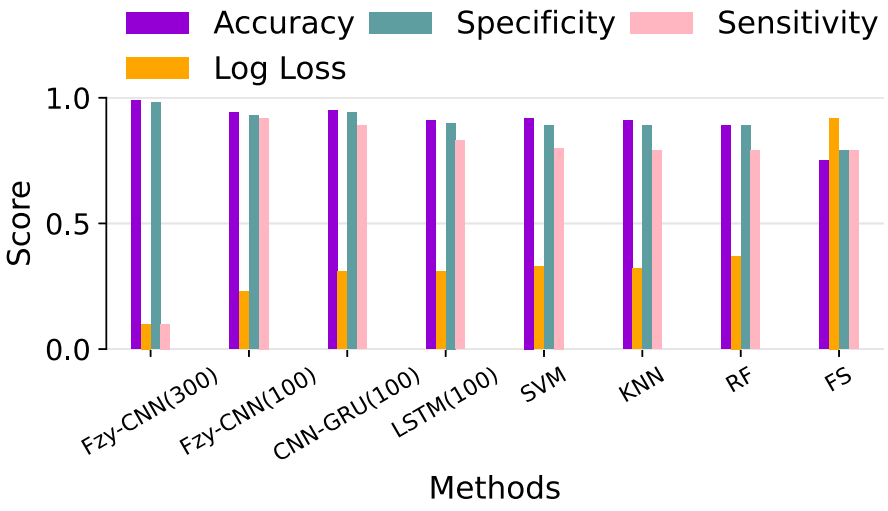


Fig. 12. Evaluation dataset: Comparison of different models for activity recognition.

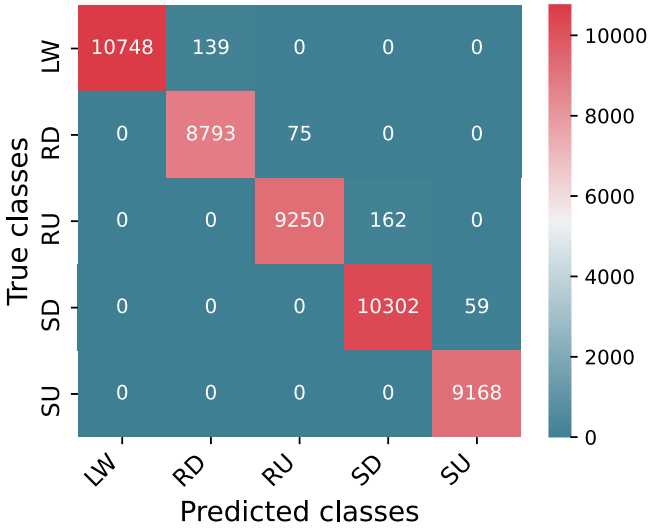


Fig. 13. Evaluation dataset: Confusion matrices: giving into the model the previous n sequences (look back = 300).

Table 6
Evaluation dataset: Precision, recall, F1-score and accuracy obtained for 5 classes (look back = 300).

Class	Precision	Recall	F1-score	Support
Class 1	1.00	0.99	0.99	10 887
Class 2	0.98	0.99	0.99	8868
Class 3	0.99	0.98	0.99	9412
Class 4	0.98	0.99	0.99	10 361
Class 5	0.99	1.00	1.00	9168
Accuracy			0.99	48 696

Table 7
Fuzzy rules for gait phases detection.

Sub-phases	GRF	Foot position
$S1$	Large	Small
$S2$	Large	Zero
$S3$	Large	Large
$S4$	Small	Not considered

neural network models on the evaluation set (see Table 4 for numerical results). Precision, recall, and F1-scores for all classes are presented in Table 6 where almost all classes represent a score close to 1. Fig. 13 represents the confusion matrix for the evaluation dataset. As these results show, almost all samples are correctly classified into the corresponding class. These results demonstrate once again the generalizability of the proposed model.

6. Conclusion and future works

This paper proposes a new approach for activity recognition using historical activity sequences as model input. The proposed approach is a fusion of a deep learning algorithm with a novel fuzzy system for predicting human activity.

The combination of the proposed fuzzy logic with a CNN benefits from the use of fuzzy membership degrees that take into account the incompleteness of available data, such as sensor data to produce more refined outputs, thus reducing the ambiguities that exist in the nature of human activity recognition. In addition, the proposed fuzzy system provides enriched features from the initial attributes of the data. The high-level features are extracted by a convolutional neural representation. The advantage of fuzzy rule-trained CNNs is also to eliminate both the limited availability of training data and the need for extensive data preprocessing.

The proposed model yields several properties: (i) it uses an adaptive kernel-based on data characteristics which is derived from the fuzzy rules over input sequences; (ii) it extracts local subsequences from sequences to identify local patterns in activities using a CNN; (iii) using an attention mechanism, it exploits the underlying interdependence of activity sequences to extract the significant parts of short-term sequences and a GRU model to extract the complex and non-linear aspects of time series data on time sequences. These new features are not only representatives of the initial ones but also represent much more affluent inherent characteristics of raw activity sequences.

The proposed approach also reveals the advantages of using Fuzzy, CNN, GRU and the attention mechanism together over these models used separately. In addition to the main contributions mentioned above, this paper provides a public data set to serve research advances in the robotics community. The proposed approach outperforms the accuracy score of approaches in the literature by addressing problems such as incompleteness, noise, or inaccuracy of sensor data without the need for extensive data preprocessing. It solves the difficulties of conventional approaches in dealing with the inherent characteristics of HAR data, such as ambiguity and limited amount of data. Accurate detection of gait modes, such as level walking, stair ascent/descent, and ramp ascent/descent is essential for a wearable robot to provide

appropriate assistance. The proposed model shows great potential in detecting these gait modes for different kinematic and kinetic characteristics of the lower-limbs in real-time to enable implementation of appropriate, natural, and smooth assistive torques. In future work, the proposed model will be evaluated in a real environment trial.

Extra reasoning modules could be appended to the system in order to provide the capability of fault correction in addition to the anticipation. Reasoning and ontology-based analysis are applicable owing to our various provided labels. Sentiment analysis could be valuable for improving human–robot interaction and increasing human engagement. Using the Kinect's outputs of a skeleton dataset, augmenting vision-based analysis could be practicable as well with the intention of enhancement in the anticipation of activities or faults.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Ghazaleh Khodabandelou reports was provided by University Paris-Est Créteil Laboratory of Images, Signals and Intelligent Systems.

Data availability

http://www.lissi.fr/;dataset-for-gait-mode-recognition-on-activities-of-daily-living/

References

Au, S., Berniker, M., Herr, H., 2008. Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits. Neural Netw. 21 (4), 654–666.

Baiden, D., Ivlev, O., 2013. Human-robot-interaction control for orthoses with pneumatic soft-actuators—concept and initial trails. In: 2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR). IEEE, pp. 1–6.

Banerjee, A., Singh, P.K., Sarkar, R., 2020. Fuzzy integral-based CNN classifier fusion for 3D skeleton action recognition. IEEE Trans. Circuits Syst. Video Technol. 31 (6), 2206–2216.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Carrington, A.M., Fieguth, P.W., Qazi, H., Holzinger, A., Chen, H.H., Mayr, F., Manuel, D.G., 2020. A new concordant partial AUC and partial c statistic for imbalanced data in the evaluation of machine learning algorithms. BMC Med. Inform. Decis. Mak. 20 (1), 1–12.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation, ArXiv.

Cordero-Martínez, R., Sánchez, D., Melin, P., 2022. Hierarchical genetic optimization of convolutional neural models for diabetic retinopathy classification. Int. J. Hybrid Intell. Syst. (Preprint), 1–13.

Craig, J.J., 2009. Introduction to Robotics: Mechanics and Control, Vol. 3/E. Pearson Education India.

De Silva, L.C., Morikawa, C., Petra, I.M., 2012. State of the art of smart homes. Eng. Appl. Artif. Intell. 25 (7), 1313–1321.

Dhiman, C., Vishwakarma, D.K., 2019. A review of state-of-the-art techniques for abnormal human activity recognition. Eng. Appl. Artif. Intell. 77, 21–45.

Du, G., Zeng, J., Gong, C., Zheng, E., 2021. Locomotion mode recognition with inertial signals for hip joint exoskeleton. Appl. Bionics Biomech. 2021.

Dutta, A., Koerding, K., Perreault, E., Hargrove, L., 2011. Sensor-fault tolerant control of a powered lower limb prosthesis by mixing mode-specific adaptive Kalman filters. In: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, pp. 3696–3699.

Farris, R.J., Quintero, H.A., Goldfarb, M., 2011. Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals. IEEE Trans. Neural Syst. Rehabil. Eng. 19 (6), 652–659.

Feigin, V.L., Norrving, B., Mensah, G.A., 2017. Global burden of stroke. Circ. Res. 120 (3), 439–448.

Foxlin, E., 2005. Pedestrian tracking with shoe-mounted inertial sensors. IEEE Comput. Graph. Appl. 25 (6), 38–46.

Fregoso, J., Gonzalez, C.I., Martinez, G.E., 2021. Optimization of convolutional neural networks architectures using pso for sign language recognition. Axioms 10 (3), 139.

Goldberger, J., Hinton, G.E., Roweis, S.T., Salakhutdinov, R.R., 2005. Neighbourhood components analysis. In: Advances in Neural Information Processing Systems. pp. 513–520.

Gomathi, V., Kalaiselvi, S., et al., 2022. Sensor-based human activity recognition using fuzzified deep CNN architecture with λmax method. Sens. Rev..

Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. J. Mach. Learn. Res. 3 (Mar), 1157–1182.

Hasegawa, Y., Jang, J., Sankai, Y., 2009. Cooperative walk control of paraplegia patient and assistive system. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 4481–4486.

Holzinger, A., Muller, H., 2021. Toward human–AI interfaces to support explainability and causability in medical AI. Computer 54 (10), 78–86.

Hsu, M.-J., Chien, Y.-H., Wang, W.-Y., Hsu, C.-C., 2020. A convolutional fuzzy neural network architecture for object classification with small training database. Int. J. Fuzzy Syst. 22 (1), 1–10.

Huang, H., Kuiken, T.A., Lipschutz, R.D., et al., 2008. A strategy for identifying locomotion modes using surface electromyography. IEEE Trans. Biomed. Eng. 56 (1), 65–73.

Huo, W., Arnez-Paniagua, V., Ghedira, M., Amirat, Y., Gracies, J.-M., Mohammed, S., 2018a. Adaptive FES assistance using a novel gait phase detection approach. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 1–9.

Huo, W., Mohammed, S., Amirat, Y., Kong, K., 2018b. Fast gait mode detection and assistive torque control of an exoskeletal robotic orthosis for walking assistance. IEEE Trans. Robot. 34 (4), 1035–1052.

Hwang, Y.M., Park, S., Lee, H.O., Ko, S.-K., Lee, B.-T., 2021. Deep learning for human activity recognition based on causality feature extraction. IEEE Access 9, 112257–112275.

Jang, J., Kim, K., Lee, J., Lim, B., Shim, Y., 2015. Online gait task recognition algorithm for hip exoskeleton. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 5327–5332.

Juang, C.-F., 2002. A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms. IEEE Trans. Fuzzy Syst. 10 (2), 155–170.

Juang, C.-F., Lin, C.-T., 1998. An online self-constructing neural fuzzy inference network and its applications. IEEE Trans. Fuzzy Syst. 6 (1), 12–32.

Kang, I., Kunapuli, P., Young, A.J., 2019. Real-time neural network-based gait phase estimation using a robotic hip exoskeleton. IEEE Trans. Med. Robot. Bionics 2 (1), 28–37.

Kang, I., Molinaro, D.D., Duggal, S., Chen, Y., Kunapuli, P., Young, A.J., 2021. Real-time gait phase estimation for robotic hip exoskeleton control during multimodal locomotion. IEEE Robot. Autom. Lett. 6 (2), 3491–3497.

Khodabandelou, G., Ebadzadeh, M.M., 2019. Fuzzy neural network with support vector-based learning for classification and regression. Soft Comput. 23 (23), 12153–12168.

Khodabandelou, G., Jung, P.-G., Amirat, Y., Mohammed, S., 2020. Attention-based gated recurrent unit for gesture recognition. IEEE Trans. Autom. Sci. Eng..

Khodabandelou, G., Katranji, M., Kraiem, S., Kheriji, W., HadjSelem, F., 2019. Attention-based gated recurrent unit for links traffic speed forecasting. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC). IEEE, pp. 2577–2583.

Khodabandelou, G., Kheriji, W., Selem, F.H., 2021. Link traffic speed forecasting using convolutional attention-based gated recurrent unit. Appl. Intell. 51 (4), 2331–2352.

Kidziński, Ł., Yang, B., Hicks, J.L., Rajagopal, A., Delp, S.L., Schwartz, M.H., 2020. Deep neural networks enable quantitative movement analysis using single-camera videos. Nature Commun. 11 (1), 1–10.

Kong, K., Tomizuka, M., 2009. A gait monitoring system based on air pressure sensors embedded in a shoe. IEEE/ASME Trans. Mechatronics 14 (3), 358–370.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 1097–1105.

Lee, D., Kang, I., Molinaro, D.D., Yu, A., Young, A.J., 2021. Real-time user-independent slope prediction using deep learning for modulation of robotic knee exoskeleton assistance. IEEE Robot. Autom. Lett. 6 (2), 3995–4000.

Lencioni, T., Carpinella, I., Rabuffetti, M., Marzegan, A., Ferrarin, M., 2019. Human kinematic, kinetic and EMG data during different walking and stair ascending and descending tasks. Sci. Data 6 (1), 1–10.

Li, Y.D., Hsiao-Wecksler, E.T., 2013. Gait mode recognition and control for a portable-powered ankle-foot orthosis. In: 2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR). IEEE, pp. 1–8.

Long, Y., Du, Z.-J., Wang, W.-D., Zhao, G.-Y., Xu, G.-Q., He, L., Mao, X.-W., Dong, W., 2016. PSO-SVM-based online locomotion mode identification for rehabilitation robotic exoskeletons. Sensors 16 (9), 1408.

Mannini, A., Sabatini, A.M., 2012. Gait phase detection and discrimination between walking–jogging activities using hidden Markov models applied to foot motion data from a gyroscope. Gait Posture 36 (4), 657–661.

Medjahed, H., Istrate, D., Boudy, J., Dorizzi, B., 2009. Human activities of daily living recognition using fuzzy logic for elderly home monitoring. In: 2009 IEEE International Conference on Fuzzy Systems. IEEE, pp. 2001–2006.

Merdivan, E., Singh, D., Hanke, S., Holzinger, A., 2019. Dialogue systems for intelligent human computer interactions. Electron. Notes Theor. Comput. Sci. 343, 57–71.

Murray, S.A., Ha, K.H., Hartigan, C., Goldfarb, M., 2014. An assistive control approach for a lower-limb exoskeleton to facilitate recovery of walking following stroke. IEEE Trans. Neural Syst. Rehabil. Eng. 23 (3), 441–449.

Nida, N., Irtaza, A., Javed, A., Yousaf, M.H., Mahmood, M.T., 2019. Melanoma lesion detection and segmentation using deep region based convolutional neural network and fuzzy C-means clustering. Int. J. Med. Inform. 124, 37–48.

Novak, D., Riener, R., 2015. A survey of sensor fusion methods in wearable robotics. Robot. Auton. Syst. 73, 155–170.

Oh, S., Baek, E., Song, S.-k., Mohammed, S., Jeon, D., Kong, K., 2015. A generalized control framework of assistive controllers and its application to lower limb exoskeletons. Robot. Auton. Syst. 73, 68–77.

Parri, A., Yuan, K., Marconi, D., Yan, T., Crea, S., Munih, M., Lova, R.M., Vitiello, N., Wang, Q., 2017. Real-time hybrid locomotion mode recognition for lower limb wearable robots. IEEE/ASME Trans. Mechatronics 22 (6), 2480–2491.

Paulich, M., Schepers, M., Rudigkeit, N., Bellusci, G., 2018. Xsens MTw Awinda: Miniature wireless inertial-magnetic motion tracker for highly accurate 3D kinematic applications. Xsens: Enschede Neth. 1–9.

Poma, Y., Melin, P., González, C.I., Martínez, G.E., 2020. Optimization of convolutional neural networks using the fuzzy gravitational search algorithm. J. Autom. Mob. Robot. Intell. Syst. 109–120.

Rubio-Solis, A., Panoutsos, G., Beltran-Perez, C., Martinez-Hernandez, U., 2020. A multilayer interval type-2 fuzzy extreme learning machine for the recognition of walking activities and gait events using wearable sensors. Neurocomputing 389, 42–55.

Sadaei, H.J., e Silva, P.C.d.L., Guimarães, F.G., Lee, M.H., 2019. Short-term load forecasting by using a combined method of convolutional neural networks and fuzzy time series. Energy 175, 365–377.

Shepherd, M.K., Molinaro, D.D., Sawicki, G.S., Young, A.J., 2022. Deep learning enables exoboot control to augment variable-speed walking. IEEE Robot. Autom. Lett. 7 (2), 3571–3577.

Shin, D.B., Lee, S.C., Hwang, S.H., Baek, I.H., No, J.K., Hwang, S.W., Han, C.S., 2020. Development of the algorithm of locomotion modes decision based on RBF-SVM for hip gait assist robot. J. Korean Soc. Precis. Eng. 37 (3), 187–194.

Singh, D., Merdivan, E., Hanke, S., Kropf, J., Geist, M., Holzinger, A., 2017. Convolutional and recurrent neural networks for activity recognition in smart environment. In: Towards Integrative Machine Learning and Knowledge Extraction. Springer, pp. 194–205.

Soleimani, E., Khodabandelou, G., Chibani, A., Amirat, Y., 2022. Generic semi-supervised adversarial subject translation for sensor-based activity recognition. Neurocomputing.

Su, B., Smith, C., Gutierrez Farewik, E., 2020. Gait phase recognition using deep convolutional neural network with inertial measurement units. Biosensors 10 (9), 109.

Suykens, J.A., Vandewalle, J., 1999. Least squares support vector machine classifiers. Neural Process. Lett. 9 (3), 293–300.

Takagi, T., Sugeno, M., 1985. Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Syst. Man Cybern. (1), 116–132.

Tu, X., Li, M., Liu, M., Si, J., Huang, H.H., 2021. A data-driven reinforcement learning solution framework for optimal and adaptive personalization of a hip exoskeleton. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 10610–10616.

Tucker, M.R., Olivier, J., Pagel, A., Bleuler, H., Bouri, M., Lambercy, O., del R Millán, J., Riener, R., Vallery, H., Gassert, R., 2015. Control strategies for active lower extremity prosthetics and orthotics: a review. J. Neuroeng. Rehabil. 12 (1), 1.

Unluhisarcikli, O., Pietrusinski, M., Weinberg, B., Bonato, P., Mavroidis, C., 2011. Design and control of a robotic lower extremity exoskeleton for gait rehabilitation. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 4893–4898.

Varol, H.A., Sup, F., Goldfarb, M., 2009. Multiclass real-time intent recognition of a powered lower limb prosthesis. IEEE Trans. Biomed. Eng. 57 (3), 542–551.

Wang, L.-X., 2019. Fast training algorithms for deep convolutional fuzzy systems with application to stock index prediction. IEEE Trans. Fuzzy Syst. 28 (7), 1301–1314.

Wang, Q., 2021. Research on the improved CNN deep learning method for motion intention recognition of dynamic lower limb prosthesis. J. Healthc. Eng. 2021.

Wang, S., Wang, L., Meijneke, C., Van Asseldonk, E., Hoellinger, T., Cheron, G., Ivanenko, Y., La Scaleia, V., Sylos-Labini, F., Molinari, M., et al., 2014. Design and control of the MINDWALKER exoskeleton. IEEE Trans. Neural Syst. Rehabil. Eng. 23 (2), 277–286.

Wentink, E., Schut, V., Prinsen, E., Rietman, J.S., Veltink, P.H., 2014. Detection of the onset of gait initiation using kinematic sensors and EMG in transfemoral amputees. Gait Posture 39 (1), 391–396.

Young, A.J., Ferris, D.P., 2016. State of the art and future directions for lower limb robotic exoskeletons. IEEE Trans. Neural Syst. Rehabil. Eng. 25 (2), 171–182.

Yuan, K., Wang, Q., Wang, L., 2014. Fuzzy-logic-based terrain identification with multisensor fusion for transtibial amputees. IEEE/ASME Trans. Mechatronics 20 (2), 618–630.

Zhang, F., Fang, Z., Liu, M., Huang, H., 2011. Preliminary design of a terrain recognition system. In: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, pp. 5452–5455.

Zhang, T., Zhang, D.-g., Yan, H.-r., Qiu, J.-n., Gao, J.-x., 2021. A new method of data missing estimation with FNN-based tensor heterogeneous ensemble learning for internet of vehicle. Neurocomputing 420, 98–110.

Zheng, E., Wang, L., Wei, K., Wang, Q., 2014. A noncontact capacitive sensing system for recognizing locomotion modes of transtibial amputees. IEEE Trans. Biomed. Eng. 61 (12), 2911–2920.

Zheng, Y., Xu, Z., Wang, X., 2022. The fusion of deep learning and fuzzy systems: A state-of-the-art survey. IEEE Trans. Fuzzy Syst. 30 (8), 2783–2799. http://dx.doi.org/10.1109/TFUZZ.2021.3062899.

Zhu, C., Liu, Q., Meng, W., Ai, Q., Xie, S.Q., 2021. An attention-based CNN-LSTM model with limb synergy for joint angles prediction. In: 2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM). IEEE, pp. 747–752.