



# Enhanced fuzzy finite state machine for human activity modelling and recognition

Gadelhag Mohmed<sup>1</sup> · Ahmad Lotfi<sup>1</sup> · Amir Pourabdollah<sup>1</sup>

Received: 14 October 2019 / Accepted: 26 March 2020 / Published online: 30 April 2020  
© The Author(s) 2020

## Abstract

A challenging key aspect of modelling and recognising human activity is to design a model that can deal with the uncertainty in human behaviour. Several machine learning and deep learning techniques are employed to model the Activity of Daily Living (ADL) representing the human activity. This paper proposes an enhanced Fuzzy Finite State Machine (FFSM) model by combining the classical FFSM with Long Short-Term Memory (LSTM) neural network and Convolutional Neural Network (CNN). The learning capability in the LSTM and CNN allows the system to learn the relationship in the temporal human activity data and to identify the parameters of the rule-based system as building blocks of the FFSM through time steps in the learning mode. The learned parameters are then used for generating the fuzzy rules that govern the transitions between the system's states representing activities. The proposed enhanced FFSMs were tested and evaluated using two different datasets; a real dataset collected by our research group and a public dataset collected from CASAS smart home project. Using LSTM-FFSM, the experimental results achieved 95.7% and 97.6% for the first dataset and the second dataset, respectively. Once CNN-FFSM was applied to both datasets, the obtained results were 94.2% and 99.3%, respectively.

**Keywords** Human activity modelling and recognition · Fuzzy finite state machine · Long short-term memory neural network · Convolutional neural network

## 1 Introduction

Monitoring and recognising human activities within a home environment are investigated in order to support the independent living of older adults in Ambient Intelligence (AmI) environments (Chen et al. 2012; Medina-Quero et al. 2018). Several techniques are used to gather the information that represents the Activity of Daily Living (ADL) from a real environment for the monitored user (Cook et al. 2013; Langensiepen et al. 2014). This information is commonly gathered from the signals that are collected from ambient sensors such as door entry sensors, movement and occupancy

sensors etc., (Hassan et al. 2018). The information can also be extracted based on vision sensors such as cameras that capture images or video streams (Cook et al. 2013), although, for privacy concerns, recent attention has predominantly focused on data collected by ambient sensors, which are more acceptable to users (Aicha et al. 2017). The gathered information is then processed and analysed in a useful format to be used in many different applications, including anomaly detection in daily human activities, energy consumption optimisation, addressing health and safety concerns, leading to an improved level of comfort and quality of life (Langensiepen et al. 2014; Lotfi et al. 2012).

One of the promising techniques to model dynamic processes when data changes over time is the Finite State Machine (FSM) (Mohmed et al. 2018b; Alvarez-Alvarez et al. 2012). The FSM contains several states representing different actions and the mechanism of transitions between them. Many researchers have considered using the FSM to model and represent human activities. Since humans behaviour is not restricted to a single state at any time and there are uncertainties associated with each state, it is reasonable to consider some degree of fuzziness within the FSM,

✉ Gadelhag Mohmed  
gadelhag.mohmed2016@my.ntu.ac.uk

Ahmad Lotfi  
Ahmad.lotfi@ntu.ac.uk

Amir Pourabdollah  
amir.pourabdollah@ntu.ac.uk

<sup>1</sup> School of Science and Technology, Nottingham Trent University, Clifton Campus, Clifton Lane, Nottingham NG11 8NS, UK

thereby creating a more powerful tool to model dynamic processes that may change over time (Langensiepen et al. 2014; Mohmed et al. 2018a; Medina-Quero et al. 2018). The classical version of FSM is enhanced by incorporating fuzzy states/transitions leading to Fuzzy Finite State Machine (FFSM). Both inputs and outputs are treated as fuzzy sets instead of being treated as crisp values. This allows the system to handle and process the input information with a degree of belonging, which often provides more flexibility and human comprehensibility (Unal and Khan 1994). The FFSM is one of the most suitable technique to deal with a large amount of uncertain data gathered from low-level sensory devices in AmI environments. In this case, the system can assign a degree of truth to the occurrence of each activity. The transitions between the system's states in the FFSM are triggered by fuzzy values rather than crisp values used in the classical FSM. This provides a realistic model supported by a fuzzy reasoning mechanism, represented by a degree of truth related to each state transition. Therefore, more than one state can be in an active mode at any time based on the membership values of each state (Langensiepen et al. 2014; Alvarez-Alvarez et al. 2011; Sridhar et al. 2019).

By nature, several activities can be undertaken by a single user at the same time (Cook et al. 2013). For example, people can watch TV while they are eating their meal. In this particular scenario, it is not necessary to know which activity started first. However, it is essential to know the degree of involvement in each activity at that time. That means the existence of simultaneous activities when an activity (e.g., eating) starts while the other activity is already started (e.g., watching TV). A specialised approach is required to recognise these non-sequential behaviours. One of the most promising technique to deal with such uncertainties that associated with human activities is using fuzzy sets. Hence, the classical FSM is integrated with a fuzzy logic system to address these uncertainties.

This paper is an extension of the authors' research work in developing an FFSM used for modelling and recognising human activity. In this work, the FFSM is introduced as a means of defining daily human activities and the transition between the states (here, the activities). There are many unknown parameters in the FFSM which needs to be identified in order to represent a model for the ADL. The aim of the research reported here is to identify the parameters to represent the real activities of a human subject in an AmI environment accurately. The research presented in this paper addresses only the challenges in modelling and recognising a single-occupancy at a real-home environment based on a dataset collected from ambient sensory devices. Another research exploring the domain of modelling and recognising human activities within multi-occupancy environments are currently ongoing with other researchers in our research

group. The research reported in this paper has made the following contributions:

- Integrating the FFSM with Long Short-Term Memory (LSTM) neural networks to enhance the learning capability of the FFSM model for accurately generating the fuzzy rules that govern the transition between the system states. The new model is referred to as a Short-Term Memory-Fuzzy Finite State Machine (LSTM-FFSM).
- Integrating the FFSM with Convolutional Neural Network (CNN) to add the learning ability to model daily human activities based on the numerical and temporal information gathered from the sensory data. The new model is referred to as Convolutional-Fuzzy Finite State Machine (CNN-FFSM).
- Testing and evaluating the proposed models using two different datasets gathered from real home environments representing ADL for a single user.

The rest of this paper is organised as follows: a review of the related literature is provided in Sect. 2, the methodologies are presented in Sect. 3 introducing fuzzy feature representation and Fuzzy Finite State Machine (FFSM). In Sect. 4 two proposed FFSM models namely LSTM-FFSM and CNN-FFSM are explained. In Sect. 5, a human activity recognition case study is detailed, including the experiment using two different datasets. Followed by the obtained results in Sect. 6. The pertinent conclusions are drawn in Sect. 7.

## 2 Related work

Most of the research related to human activity recognition is carried out using statistical techniques including Support Vector Machine (SVM) (Khemchandani and Sharma 2017; Anguita et al. 2012) and Finite State Machine (FSM) (Trinh et al. 2011) are used to find the relationship between the real action (activity) and the temporal data gathered from sensors. The ultimate goal of these techniques is to identify the activity of the user. Several graphical techniques are introduced to recognise and model human activities, for example, Hidden Markov Model (HMM) can represent random variables, actions and temporal variation within the collected data (Chung and Liu 2008; Kong and Fu 2018). Relatively new research in Aicha et al. (2017) and Malasinghe et al. (2019) have presented a new model based on the Markov Modulated Poisson Process (MMPP) which promises to come up with a model to represent multi-visitor recognition with more accuracy. The only issue with this approach is the difficulty in processing a large amount of low-level data such as the data gathered from ambient sensory devices.

In Langensiepen et al. (2014), the authors used FSM for locating and modelling the activity of a single user in

an apartment. The FSM is a powerful technique for modelling dynamic events, that is, the events that change over time, such as human activity. The FSM model is enhanced by being integrated with a fuzzy system, which is used to increase the efficiency of the FSM by proposing the Fuzzy Finite-State Machine (FFSM), where transitions between the states are triggered by the sense of fuzziness instead of using crisp values. This has the advantage of smooth modelling and reasoning with a degree of truth, which proves to be more accurate. Thus, the system can be in more than one state at a time, based on the truth degree for each state (Alvarez-Alvarez et al. 2011; Langensiepen et al. 2014). The main advantage of using the fuzzy state is that it can deal with uncertain data and can be represented in more than one state at the same time as membership degrees.

Computational intelligence techniques are also widely used to recognise and model human activities, as an alternative or in combination with other statistical methods. Neural Networks (NNs) are used to deal with and process pattern recognition based on numerical data that is gathered from sensors in an AmI environment (Benmansour et al. 2017; Subramanian and Suresh 2012). Recurrent Neural Networks (RNNs) are proven to be a powerful tool to solve the difficulties of the temporal relationships of inputs and outputs at different time steps (Medina-Quero et al. 2018; Tran et al. 2018). In Medina-Quero et al. (2018), authors created fuzzy temporal windows for the collated binary data representing the human activities, and then applied them to an ensemble classifier based on LSTM neural networks. The LSTM is used in Jenckel et al. (2018) for annotating historical documents, where authors are used fuzzy ground truth to represent the input data to provide all possible annotations for each input variable, instead of just one. The LSTM neural networks prove their ability to be a good approach to model sequential forms of data such as human activity data (Yulita et al. 2017a). Moreover, the LSTM can save past information by looping it inside its architecture, which reuses the information from the previously learned iteration. The main purpose of using the LSTM is to reduce the risk of vanishing gradients in the sequential temporal data. In some of the recent works in human activity recognition Arifoglu and Bouchachia (2019); Gochoo et al. (2019), researchers focused on employing the CNN with binary datasets to recognise human activities and to detect any abnormal activities in the users' behavioural pattern based on a trained CNN model.

A widely used method for feature representation is the fuzzy feature representation approach. In Deng et al. (2017), the authors proposed a fuzzy computational approach to extract features from one-dimensional input vectors. They employed a deep neural network with the extracted features for classifying the given data. A fuzzy temporal windows approach is proposed in Medina-Quero et al. (2018); Yulita

et al. (2017a) to define temporal-sequence representations to aggregate information from binary sensors for real-time recognition of human activities. The methods in Deng et al. (2017); Medina-Quero et al. (2018) have been successful in capturing features that improved the performance of the classification tasks for human activity recognition.

Based on the literature review conducted for this research, the fuzzy feature representation approach is used to fuzzify the data representing human activities. Two different deep learning techniques, namely LSTM and CNN are integrated with the FFSM for enhancing the learning process of the parameters that are used to generate the fuzzy rules governing the states' transitions.

### 3 Methodologies

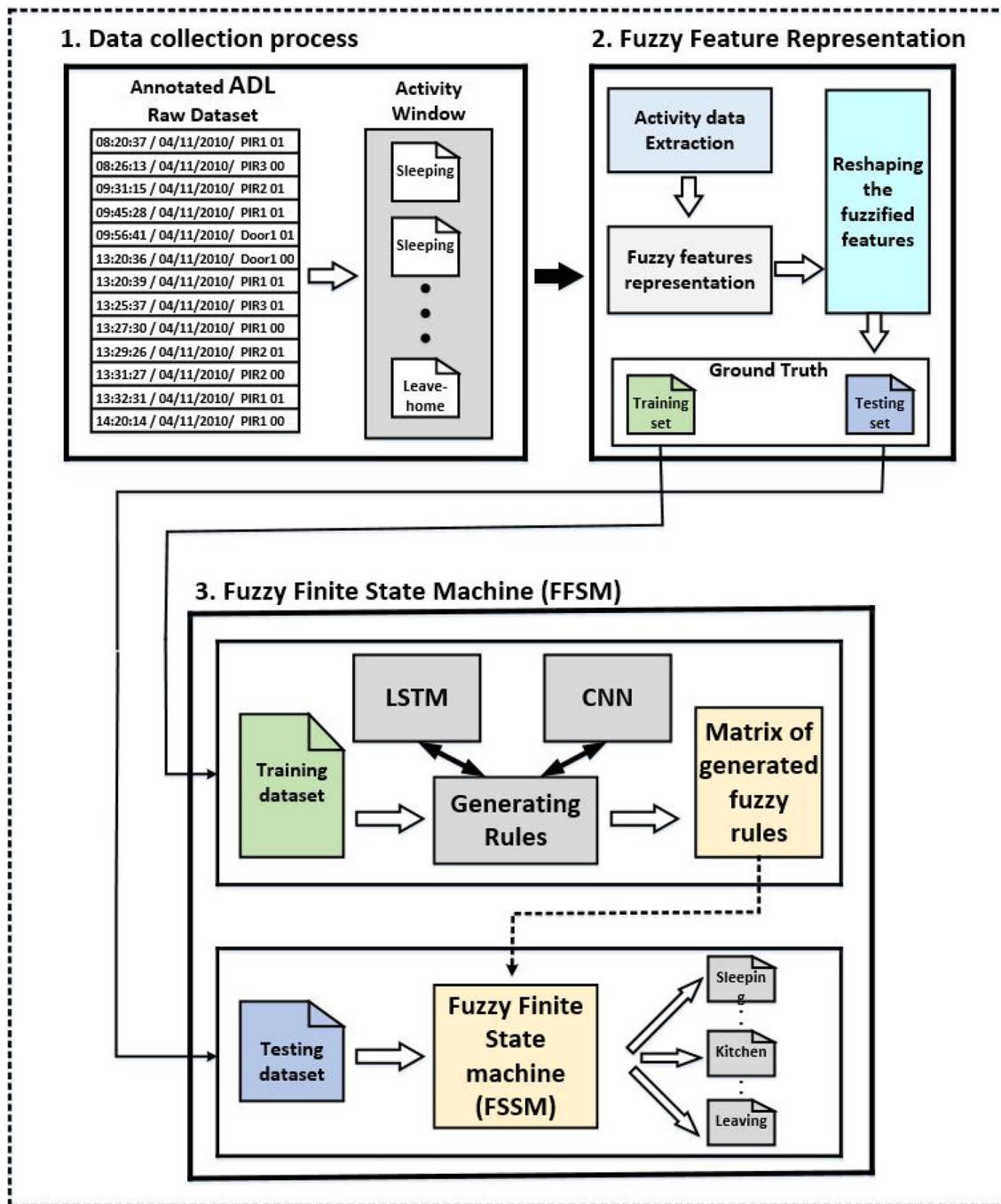
In this section, the Fuzzy Finite State Machine (FFSM) incorporating fuzzy feature representation is introduced initially. Then, two proposed, enhanced FFSM approaches are introduced. Fig. 1 illustrates the schematic diagram of the proposed FFSM. This consists of three different stages; data collection process, fuzzy feature representation, and the fuzzy finite state machine model. In the data collection stage, the data from sensors in an AmI environment representing the ADL is collected. The fuzzy feature representation stage is designed to transform the data into fuzzy features to be used as inputs to the proposed FFSM model. In the third stage, the proposed FFSM model will generate the fuzzy rules employing the capabilities of learning algorithms in LSTM and CNN to representing the states' transitions.

#### 3.1 Fuzzy feature representation

Fuzzy feature representation approach is designed to convert the collected information into their relevant membership degrees (Yulita et al. 2017b). The resulting membership degrees are taken as features to be used as inputs to train the proposed model. Therefore, fuzzy feature representation is applied to determine the number of Membership Functions (MFs) representing the input data as membership degrees. By replacing each value in the input data with their corresponding degree of memberships; thus, each value in the input data is represented as fuzzified values obtained for each MF as follows:

$$X_{uj} = [\mu_{A_{uj}^1}, \mu_{A_{uj}^2}, \dots, \mu_{A_{uj}^M}] \quad j = 1, \dots, P \quad (1)$$

$X_{uj}$  is the fuzzified set of the input variable  $u_j$ .  $P$  is the last value in the input variable  $u_j$ .  $\mu_A$  is the degree of MF associated with each linguistic label. The Fuzzy feature representation process can be summarised as follows:



**Fig. 1** A schematic diagram of the proposed Fuzzy Finite State Machine models

1. Apply the input data to the fuzzifier algorithm consists of  $M$  MFs that are represented with the linguistic labels.
2. Define the degree of fuzziness  $\mu_A$  that corresponds to each MF.
3. Determine the maximum degree of fuzziness for the variable  $u_j$  in each iteration.
4. Create a matrix  $q = r \times z$  to store the degree of MFs for each variable  $u_j$ . Where  $r$  is the total number of activity instances in the input data  $u_j$ , and  $z$  is the number of fuzzified values for the variable  $u_j$ .
5. Update the matrix  $q$  after each iteration with the new fuzzified values that corresponding to the next input value.

The final set of the fuzzified features  $X_{ij} = [\mu_{A_{ij}^1}, \mu_{A_{ij}^2}, \dots, \mu_{A_{ij}^M}]$ , will be used as inputs to train the



proposed model for learning the relations between the inputs and output data, as it is explained in the next sections. The process of fuzzy feature representation is elaborated in Sect. 5 when modelling and recognising human activity datasets are presented.

### 3.2 Fuzzy finite state machine

Fuzzy Finite State Machine (FFSM) is an extended version of the classical FSM. The FSM can be presented as a model made of two or more states; each state represents one event from a sequence of events in a dynamic process (Mohmed et al. 2018b). Only one single state of this model can be active at a time. The model is moved from one state to another by triggering crisp values. In human activity recognition and modelling, a user may be associated with multiple states. This would require to be quantified with a degree of belonging (degree of fuzziness). Once the fuzziness aspect is added to the state transitions in the classical FSM, the transitions are not triggered based on crisp values, but using fuzzy variables (Langensiepen et al. 2014; Alvarez-Alvarez et al. 2012; Unal and Khan 1994). This implies that the current *activated* state of the model is not necessarily one state, but it could be more than one state at any given time with belonging degrees (Unal and Khan 1994).

In an FFSM, the system’s states are represented as a set of linguistic variables  $S(t) = [s_1(t), s_2(t), \dots, s_i(t), \dots, s_N(t)]$  where  $N$  is the number of states. For a non-sequential system at a time  $t$ , the system’s states are represented as a state vector  $S(t)$ . When the system evolves in time, the next state is represented as a vector  $S(t + 1)$ .

In general, as in Alvarez-Alvarez et al. (2012); Langensiepen et al. (2014), the FFSM is defined as a tuple of parameters  $(S(t), U(t), f, Y(t), g)$ . where;

- **Fuzzy State**,  $S(t) = [s_1(t), s_2(t), \dots, s_i(t), \dots, s_N(t)]$  is presenting a vector identifying the system’s states at time  $t$  and  $N$  is the number of states. Each individual state at time  $t$  is  $s_i(t); i = 1 \dots N$  is a numerical value that is in fact the membership grade (between 0 and 1) given to each linguistic variable  $s_i(t)$  within the set of FFSM’s states  $S(t)$ .
- **Input Vector**,  $U(t) = [u_1(t), u_2(t), \dots, u_i(t), \dots, u_p(t)]$  is the input vector at time  $t$  representing the associated value to the linguistic variables that are generally obtained after a fuzzification process for the input data.  $P$  are the number of input variables. This input data could be a sensors’ data, a combination of different signals, or any other calculation to numerical data. The fuzzification process that is designed based on experts’ view to translating the numerical input values to a set of membership grades given to each linguistic label that defines all the acceptable values in the input vector. The labels that are associ-

ated with the input  $u_j(t)$  is represented as  $A_{u_j} = \{A_{u_j}^1, A_{u_j}^2, \dots, A_{u_j}^M\}$ , where  $M$  is the number of the associated linguistic labels (Alvarez-Alvarez et al. 2011).

- **Transition Function**,  $f$  is the state transition function that is mainly used to calculate the next state vector  $S(t + 1)$ , at each time instant  $t$ . The transition function  $f$  controls the allowed transitions between the defined system’s states. Also, it is implemented as a set of fuzzy rules. There are different ways to define the rules, e.g., using the human expert knowledge (Mohmed et al. 2019) or learning from the numerical input-output data by applying machine learning algorithms such as Artificial Neural Network (ANN) and Genetic Algorithm (GA) (Bombardier and Schmitt 2010; Wang et al. 2012; Wang and Mendel 1992). A combination of these approaches can also be implemented to have one framework containing the rules that are generated by learning from the numerical data and those assigned by the human experts’ knowledge (Wang and Mendel 1992).
- **Output Vector**,  $Y(t) = [y_1(t), y_2(t), \dots, y_k(t), \dots, y_Q(t)]$  is the output vector consisting of crisp values associated to each output at the time  $t$  and  $Q$  is the number of output variables. Values in the output vector  $Y(t)$  are calculated based on the current state of the system  $S(t)$  and the input vector  $U(t)$ .
- **Output Function**,  $g$  is the output function that is used to calculate the value of output vector  $Y(t)$ , at each time instant  $t$ .

The states and outputs of the time-invariant FFSM (Alvarez-Alvarez et al. 2012, 2011) are expressed as:

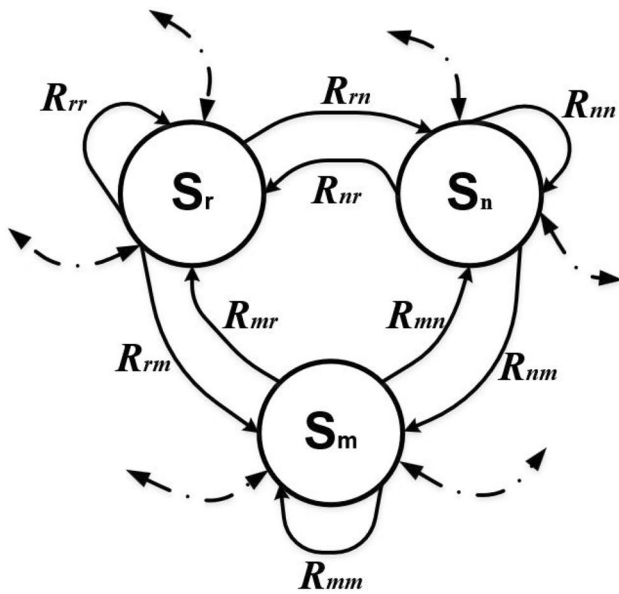
$$S(t + 1) = f(S(t), U(t)) \tag{2}$$

$$Y(t) = g(S(t), U(t)) \tag{3}$$

The states’ transition mechanism between two exemplary states  $s_m$  and  $s_n$  in the FFSM is illustrated in Fig. 2. Considering the complexity of modelling a large scale dataset, it may be impossible to analytically identify the functions  $f$  and  $g$ . This complexity will be even harder when it is used for time-invariant models (Langensiepen et al. 2014; Alvarez-Alvarez et al. 2012). Therefore, a rule  $R_{mn}$  is used to establish the relationship between states  $s_m$  and  $s_n$ . These states’ transitions can be expressed as a general fuzzy rule format (Unal and Khan 1994), as follows:

$$R_{mn}^\lambda : \text{IF } (S(t) \text{ is } s_m) \text{ AND } H_{mn} \text{ THEN } S(t + 1) \text{ is } s_n \quad \lambda = 1, \dots, A$$

where the fuzzy rule has the following parts:



**Fig. 2** States and states' transition diagram of Fuzzy Finite State Machine

- *The Antecedent Part:* is a combination of two terms; the first term,  $(S(t) \text{ is } s_m)$  is used to determine if the state  $s_m$  is an *activated* state in time instant  $t$ . Therefore, the system can change from state  $s_m$  to state  $s_n$  or remains in state  $s_m$ , only if  $m = n$ , the second term of the antecedent part is  $H_{mn}$  which represents all constraints imposed on the input variables that are required to either remain in state  $s_m$  (when,  $m = n$ ) or change to state  $s_n$ , e.g.,  $H_{mn} = (u_1(t) \text{ is } A_{u1}^3) \text{ AND } (u_2(t) \text{ is } A_{u2}^4 \text{ OR } A_{u2}^2)$ .
- *The Consequent Part:*  $(S(t+1) \text{ is } s_n)$  is the **THEN** part of the fuzzy rule, which determines the next value of the state vector  $S(t+1)$  for being in state  $s_n$ . The linguistic variables of the consequent are considered as being singletons, i.e. all elements of the  $S(t)$  vector are zero, except for the  $m^{\text{th}}$  element which is 1 (Alvarez-Alvarez et al. 2012).

For a rule-base consisting of  $A$  rules, the next value of the state vector  $S(t+1)$  is the weighted average utilising the firing degree of each rule (Mohmed et al. 2018b), defined as:

$$S(t+1) = \frac{\sum_{\lambda=1}^A w_{\lambda} \cdot S(t)}{\sum_{\lambda=1}^A w_{\lambda}} \quad \text{if } \sum_{\lambda=1}^A w_{\lambda} \neq 0 \quad (4)$$

$$S(t+1) = S(t) \quad \text{if } \sum_{\lambda=1}^A w_{\lambda} = 0 \quad (5)$$

Readers are referred to (Langensiepen et al. 2014; Alvarez-Alvarez et al. 2012; Ambres and Trivino 2012; Mohmed et al. 2018b) for more details about FFSM. More details

about the transition function elements and how they can be enhanced by integrating them with different learning techniques based on fuzzy rules are explained in the following sections.

## 4 Enhanced fuzzy finite state machine

Defining the relationship between states of a system based on fuzzy rules comes with its shortcomings. An FFSM can represent the system's states and the transition between them, assuming that all parameters are known. In this research, the aim is to enhance the performance of the FFSM by identifying optimum values for parameters of fuzzy rules. In our previous publication (Mohmed et al. 2018b), the FFSM was integrated with standard NNs to learn and adapt the parameters based so the FFSM model could represent the states' transitions and the output of each state accurately.

The work presented in this paper improves the performance of FFSM even further by integrating it with Long Short-Term Memory and Convolutional Neural Networks leading to two new models referred to as Long Short-Term Memory Fuzzy Finite State Machine (LSTM-FFSM) and Convolutional Fuzzy Finite State Machine (CNN-FFSM). The details of both enhanced FFSM is provided in the consequent sections.

### 4.1 Long short-term memory-fuzzy finite state machine

To improve the learning capability of the FFSM, an integration of Long Short-Term Memory and Fuzzy Finite State Machine is proposed. A brief explanation about the LSTM is provided first, and then the enhanced LSTM-FFSM is introduced.

The LSTM is a particular kind of RNNs designed to solve vanishing and gradients problems in the standard RNNs (Yulita et al. 2017a; Jenckel et al. 2018). The LSTM is a powerful tool for learning the sequential tasks that are represented as temporal data. It can also remember previous information for long periods. These characteristics make LSTM especially useful for temporal data classification problems (Medina-Quero et al. 2018). The LSTM cell consists of three gating mechanisms to provide the ability to remove or add information to the memory cell. These three gates are used to regulate the impact of the input through an input gate, the previous cell state through a forget gate and an output through the output gate. The essential gate in the LSTM cell is the forget gate as it decides if the information is going to be remembered or be forgotten from the previous states.

The LSTM-FFSM is an enhancement version of FFSM, allowing the system to learn the temporal relations in the

data by storing the information through the time-sequential steps. The learned relations are then used to formulate the fuzzy rules that control the transitions between the system’s states and identify the current activated states at any given time  $t$ . In this approach, the experts are also allowed to introduce their knowledge over the whole system by defining the following aspects:

- Defining the system states.
- Specifying the general structure of the fuzzy rules that represent the state transitions.
- Specifying the number of linguistic labels that are associated with each input variable.

In a typical FFSM, a rule to identify the transition between state  $m$  and state  $n$  is presented as  $R_{mn}^\lambda$  in Sect. 3.2. This demonstrates the relation between the system’s current state  $S(t)$  and the input variables that are represented as  $H_{mn}$  to identify the next state  $S(t + 1)$ . Each input variable involved in the term  $H_{mn}$  is fuzzified in order to convert the numerical data into their relevant membership degrees as it is explained in Sect. 3.1. These membership degrees for each input  $X_{uj}$  are represented as  $X_{uj} = [\mu_{A_{uj}^1}, \mu_{A_{uj}^2}, \dots, \mu_{A_{uj}^M}]$ , where  $M$  is the number of associated linguistic labels that represented as membership degrees  $\mu_{A_{uj}^M}$ . At this point, LSTM is employed to learn the temporal relations in the data by storing the previous information in a time-sequential manner. Thus the term  $H_{mn}$  will be represented as:

$$H_{mn} = X_{uj}(t) + Z_t \tag{6}$$

where  $Z$  is current output of the LSTM at time  $t$  and  $X_{uj}(t) = [\mu_{A_{uj}^1}, \mu_{A_{uj}^2}, \dots, \mu_{A_{uj}^M}] M \neq 0$  is the fuzzified input at time  $t$ .  $M$  is the number of membership degrees  $\mu_A$  that are associated with the input  $uj$ . Based on the explanation introduced in the proceeding sections, LSTM-FFSM is proposed to generate the fuzzy rules representing the transition based on learning the relations in the sequential temporal data. Therefore, the term  $H_{mn}$  is computed based on the proceeding explanation, and then the obtained parameters are used to demonstrate the fuzzy rule  $R_{mn}^\lambda$  that governs the transition between state  $m$  and state  $n$ .

### 4.2 Convolutional-fuzzy finite state machine

The integration of CNN with FFSM is also proposed for enhancing the learning capabilities of the FFSM. This is achieved by selecting the most effective features to learn the relationship between the inputs and outputs data. In this section, a brief explanation about CNN is provided first, and then the enhanced Convolutional-Fuzzy Finite State Machine (CNN-FFSM) is produced (Flagel et al. 2018).

Generally, the input data to a CNN is a matrix  $c$  in dimensions of  $h \times w \times d$ , where  $h$ ,  $w$  and  $d$  are the height, width and the number of channels in the input matrix  $c$  (Arifoglu and Bouchachia 2019; Gochoo et al. 2019). When the input window has only one class, the number of channel  $d$  is 1.

The common use of CNN architecture has two conventional layers or more and one fully-connected layer. Each convolutional layer contains multiple feature filters to optimise the values during the training phase. Each convolutional layer is followed by a max-pooling layer that has a window in a certain size to ensure the outputs from each conventional layer are smaller than the inputs. Rectified Linear Unit (ReLU) is added after each convolutional layer that operates as an activation function. The used fully-connected layer in this architecture is a traditional Multi-Layer Perceptron (MLP) that operates a softmax activation function for the output layer. By using the softmax activation function for the output layer, the CNN classifier model will be able to classify the input features into various classes based on the learned relations during the training stage.

In case of expecting highly complex input data, the CNN architecture can contain more than one pair of the convolutional and max-pooling layers with different sizes of border filters to process such data (Arifoglu and Bouchachia 2019). Also, the top convolutional layer is followed by one or more fully-connected layers for the final classification purpose. During the training phase, the standard forward and backward propagation algorithms are used to estimate the values of the CNN parameters. The selected features are mapped by the convolutional operator (Arifoglu and Bouchachia 2019) as follows:

$$V_t = \frac{1}{1 + \exp(d_\eta + \sum_i \kappa_{\eta i} \vartheta x_i)} \tag{7}$$

where  $\vartheta$  denotes the convolutional operator,  $\kappa_{\eta i}$  is the convolutional filter for the  $i$ -th input,  $V_t$  is the generated  $\eta$ -th output feature map which is achieved by selecting the most effective features over the non-overlapping pooling regions from the input data  $x_i$  and  $d_\eta$  denotes the bias.

Based on the provided explanation about the CNN in this section and the FFSM in the previous section, the CNN-FFSM may be considered as an enhancement version of FFSM by employing CNN that is allowing the system to select the most effective features from the input dataset and then learn the temporal relations from the selected features by storing the information through the time-sequential steps. The learned relations are used for formulating the fuzzy rules that are used to control the system’s state transitions and identify the current activated states at any given time  $t$ . In this approach, the experts are also allowed to introduce their knowledge over the whole system. Defining the

system's states, the general structure of the fuzzy rules, and the number of associated linguistic labels where each input variable are the aspects that are specified by experts.

As mentioned earlier, the rule  $R_{mn}^\lambda$  used to control the transition between state  $m$  and  $n$  involves the relation between the system's current state  $S(t)$  and the input variables that are represented as  $H_{mn}$ . The final value obtained from this calculation is used to identify the next state  $S(t+1)$ . As each input variable involved in the term  $H_{mn}$  is fuzzified to convert the numerical data into their relevant membership degrees  $\mu_{uj}^M$ . At this stage, CNN is employed to learn the relations in the inputs (features) and outputs (labels) data by selecting and mapping the most effective features. Therefore, the term  $H_{mn}$  will be represented as:

$$H_{mn} = X_{uj}(t) + V_t \quad (8)$$

where  $V_t$  is the generated  $\eta$ -th output feature map using CNN at time  $t$  and  $X_{uj}(t) = [\mu_{A_{uj}^1}, \mu_{A_{uj}^2}, \dots, \mu_{A_{uj}^M}]$   $M \neq 0$  is the fuzzified input dataset at time  $t$ .  $M$  is the number of membership degrees  $\mu_A$  that are associated with the input  $uj$ . Based on that, CNN is used as state-of-the-art to select the most effective features from the input dataset and learn the relations between the inputs (selected features) and the outputs (labels). The learned parameters are used essentially in this work to generate the fuzzy rules  $R_{mn}^\lambda$  representing the transition between the system's states in the proposed CNN-FFSM.

The proposed LSTM-FFSM and CNN-FFSM are employed to learn the unknown parameters for generating the fuzzy rules representing the transition between the FFSM's states. The next section introduces experiments with the proposed approaches, which integrates the learning abilities of the LSTM and CNN by selecting and mapping the

most effective features in the temporal dataset representing daily human activities.

## 5 Experimental setup

To evaluate the performance of the proposed approach, experimental works are conducted where Activity of Daily Living (ADL) for a single user is used for modelling and recognising the user's activities. The experimental setup is presented below and all results are presented in the next section.

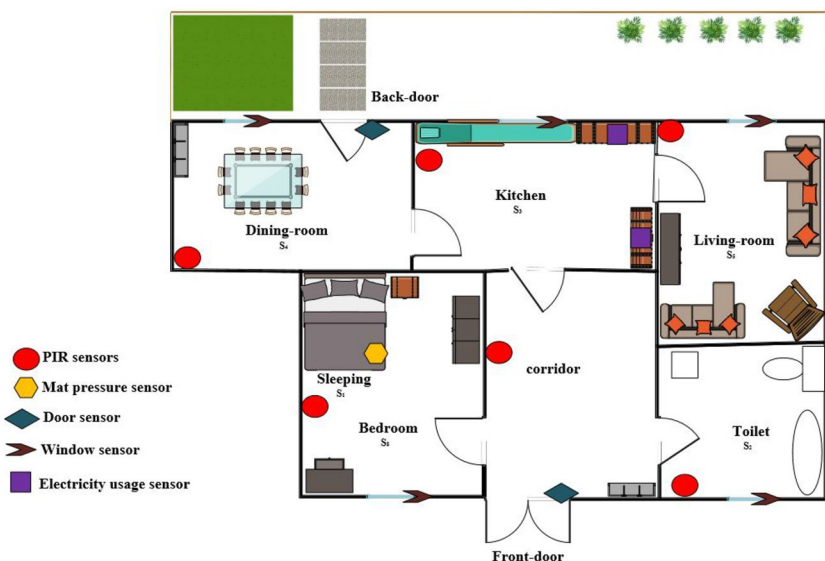
### 5.1 Datasets

Two datasets referred to as Dataset A and Dataset B are used to evaluate the proposed approaches for human activity modelling and recognition. Details of these datasets are provided below.

**Dataset A:** The dataset was collected by our research group from a real home environment representing the ADL of a single user. The dataset was collected at the Smart Home facilities within Nottingham Trent University. A floor plan of the house is shown in Fig. 3. A list of the used sensors for collecting this dataset is listed in Table 1. There are seven activities, which are *Sleeping, Toilet, Kitchen, Dining-room, Living-room, Garden, and Leaving*.

**Dataset B:** This is a publicly available dataset known as Aruba representing ADL for a single user was collected using the Centre for Advanced Studies in Adaptive System (CASAS) at Washington State University (Cook 2010). They used motion, door, and temperature sensors. However, as this work focuses on the ADL, the temperature sensors are excluded, and the other 34 sensors (3 door sensors and 31

**Fig. 3** Floor plan layout and location of the installed sensors used for data collection in dataset A





**Table 1** List of sensors used for collecting the dataset A to measure different conditions and activities

Sensor	Purpose of use
Passive Infrared (PIR)	Detecting the movement.
Door on/off switches	Detecting when doors are opened and closed.
Mat pressure sensor	Measuring bed and sofa occupancy.
Electricity consumption plugs	Measuring electricity consumption.
* Indoor temperature sensor	Measuring ambient temperature.
* Outdoor temperature sensor	Measuring outdoor temperature.
* humidity sensor	Measuring ambient humidity.
* Light intensity sensor	Measuring ambient light intensity.

\*The unused sensors in this study

motion sensors) are used. A single elderly woman lived in the Aruba testbed, and she had received regular visits from her children and grandchildren during the data collection period. The final dataset is saved as a list of sensor-ID, time-stamp, and sensor status.

In this dataset, there are 11 activities performed by the women who was living in the apartment and the data were collected over a period of 224 days. These activities are *Sleeping* (401 instances), *Meal Preparation* (1606 instances), *Relaxing* (2910 instances), *Bed-to-toilet* (157 instances), *Leaving home* (431 instances), *Entering home* (431 instances), *Housekeeping* (33 instances), *Eating* (257 instances), *Washing dishes* (65 instances), *Work* (171 instances) and *Resperate* (6 instances). The *Resperate* activity is excluded from the dataset as it has only 6 instances.

### 5.2 Feature fuzzification

Extracting the numerical information from acquired raw sensor data is crucial to any learning system as raw data does not provide adequate information that can be used as inputs to the model. The collected data was gathered from low-level ambient sensory devices; it will be saved to a database as time-stamped binary data. This gathered raw data would be represented and interpreted using the ontology data representation approach in Wongpatikaseree et al.

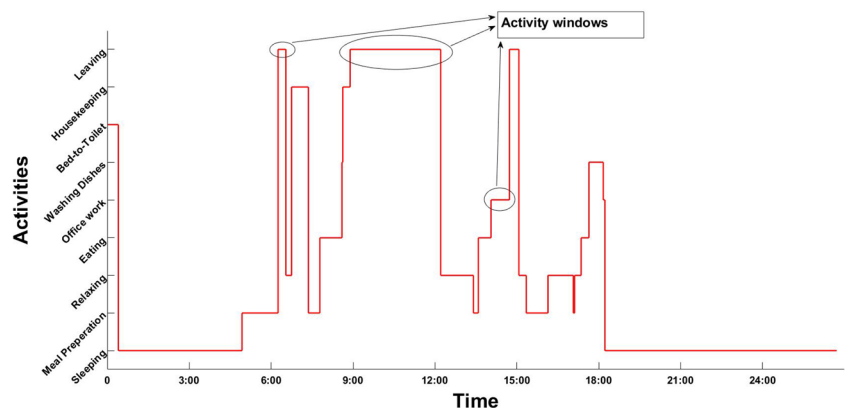
(2012) to convert it into an occupancy data for chunking the activity-windows as it is shown in Fig. 4. To fuzzify the activity data for each activity-window start time, end time, duration, an activity count and activity sequential order are extracted. Therefore, the activity data are extracted for each activity window and represented as a matrix where rows are the length of the activity window and columns are the number of recorded information from the sensors in the window.

The extracted information from each activity-window is mapped into a numerical activity data representing the input variables  $U(t) = [u_1(t), \dots, u_j(t), \dots, u_p(t)]$ . An overall framework of the used approach for representing the activity data as fuzzy features is illustrated in Fig. 5. Each value in the input variable  $u_j$  is represented with the relevant membership values to each fuzzy set. The final set of the fuzzified features  $X_{uj} = [\mu A_{uj}^1, \mu A_{uj}^2, \dots, \mu A_{uj}^M]$ , will be used as inputs to train the proposed models for modelling and recognising the activities, as it is explained in the next sections.

### 5.3 System definition

The collected datasets represent 7 and 11 different activities in dataset A and dataset B, respectively. Each activity is represented as one state in the FFSSM model. These states are defined based on the experts' knowledge. This is easily

**Fig. 4** An illustration of activity windows for 1-day activities



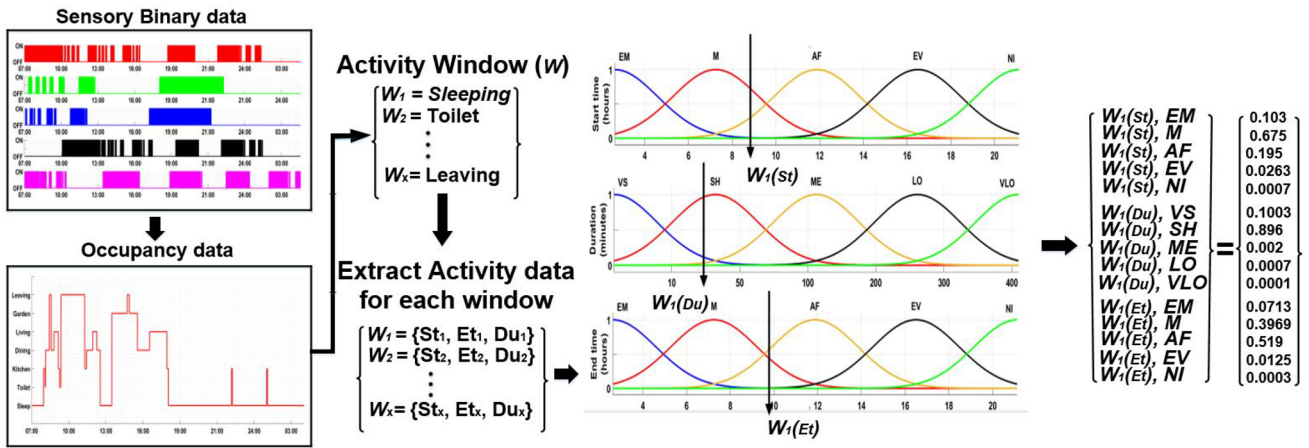
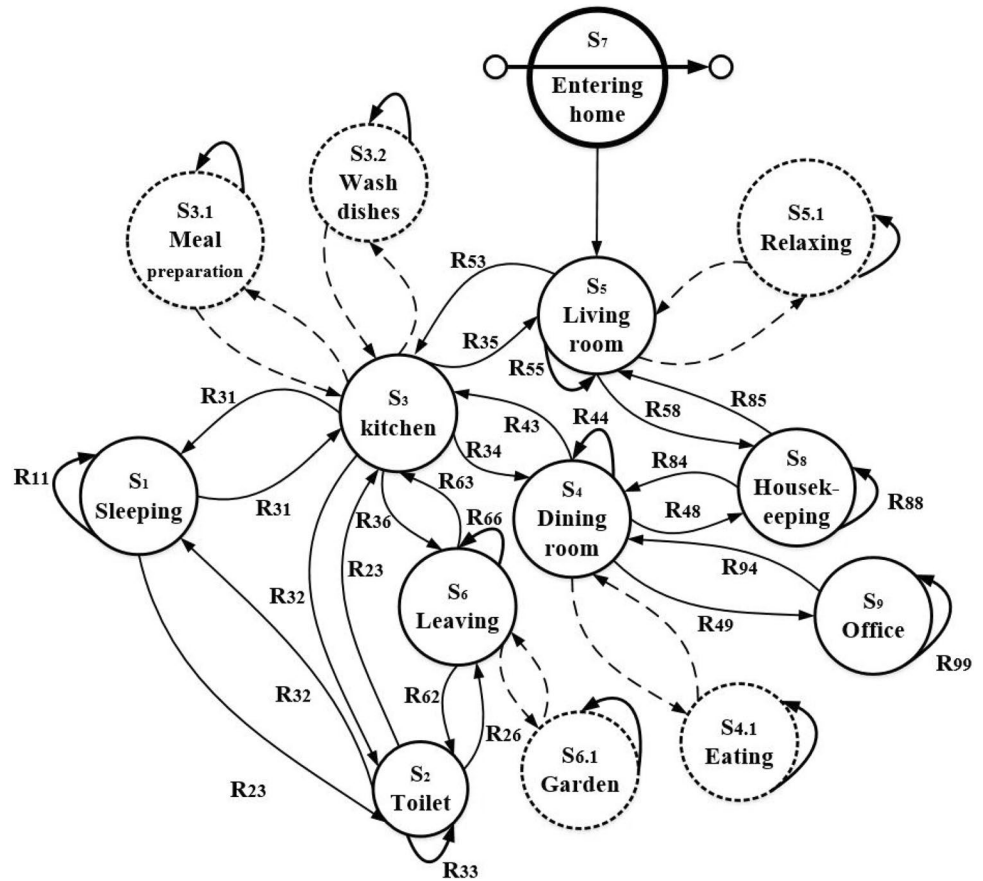


Fig. 5 Overall framework of the used fuzzy feature representation approach

Fig. 6 A state diagram of human activity's based on an experimental datasets



represented using the proposed state diagram illustrated in Fig. 6. These states in datasets A and B are defined as follows:

### 5.3.1 States representing activities in dataset A

States representing the dataset A are as follows:

- $s_1$  : The sleeping state represents the sleeping activity, either night sleeping or daytime napping.

- $s_2$  : The toilet state represents the time when the user is using the toilet.
- $s_3$  : The kitchen state represents when the user is using the kitchen for preparing food or cleaning, e.g., dish-washing.
- $s_4$  : The dining state, which usually comes after the kitchen state, represents the time when the user is in the dining room to eat the prepared meal.
- $s_5$  : The living room state represents the time spent in the living room for either relaxing or watching TV.
- $s_6$  : The leaving home state represents when the user is leaving home through the front door.
- $s_{6,1}$  : The garden state. This state is part of the leaving state  $S_6$ , which represents the time when the user is leaving to go to the garden through the back door.

### 5.3.2 States representing activities in dataset B

States representing the dataset B are as follows:

- $s_1$  : Sleeping states to represent the sleeping activities.
- $s_2$  : Bed-to-Toilet state to represent the times of using the toilet within in the sleeping time.
- $s_{3,1}$  : Meal preparation state to represent the event of preparing food. This state is the first part of the kitchen state.
- $s_{3,2}$  : Washing dishes state to represent the event of washing dishes in the kitchen area. This state is the second part of the kitchen state.
- $s_{4,1}$  : Eating state to represent the time when the user at the dining room. This state is usually activated after the Meal preparation state.
- $s_{5,1}$  : Relaxing state to represent the time spent in the living room.
- $s_6$  : Leaving state to represent the time when the user leaves the house. As the house has three different doors, this state will be activated when any of these doors are used.
- $s_7$  : Entering home state to represent the time when the user comes back home.
- $s_8$  : Housekeeping state to represent cleaning work, e.g., Hoovering the carpet.
- $s_9$  : Office-work state to represent the event of doing some homework in the office room.

Once the system's states are created, the activity data is extracted as numerical values for each activity-window. Five different numerical values are used in this experiment, representing the start time  $u_1$ , end time  $u_2$ , activity duration  $u_3$ , activity count  $u_4$  and activity sequential order  $u_5$ . These extracted numerical values are then fuzzified using Gaussian MFs. Five different MF degrees are used to convert each value in the start  $u_1$  time, end time  $u_2$  and duration  $u_3$  variables into their relevant membership degrees as it is

illustrated in Fig. 5. Therefore, every single value from these variable is represented with the relevant number of belonging degrees to each MF. The linguistic labels associated with each input are represented as MFs. These MFs are described as follows:

$$X_U(t) = \begin{cases} X_{u1}(t) \rightarrow \{EM_{u1}, M_{u1}, AF_{u1}, EV_{u1}, NI_{u1}\} \\ X_{u2}(t) \rightarrow \{EM_{u2}, M_{u2}, AF_{u2}, EV_{u2}, NI_{u2}\} \\ X_{u3}(t) \rightarrow \{VS_{u3}, SH_{u3}, ME_{u3}, LO_{u3}, VLO_{u3}\} \end{cases} \quad (9)$$

where  $X_U$  is the input vector of fuzzified variables  $\{X_{u1}, X_{u2}, X_{u3}\}$  to the system at time  $t$ . The linguistic labels that are associated with the MFs are explained as:

- The MFs representing activity start time for the input variable  $u_1$  are represented as  $\{EM_{u1}, M_{u1}, AF_{u1}, EV_{u1}, NI_{u1}\}$ . Where  $EM, M, AF, EV$  and  $NI$  are MF labels corresponding to Early Morning, Morning, Afternoon, Evening and Night respectively.
- The MFs representing activity end time for the input variable  $u_2$  are represented as  $\{EM_{u2}, M_{u2}, AF_{u2}, EV_{u2}, NI_{u2}\}$ . Where  $EM, M, AF, EV$  and  $NI$  are the MF labels corresponding to Early Morning, Morning, Afternoon, Evening and Night respectively.
- The MFs representing activity duration for the input variable  $u_3$  are represented as  $\{VS_{u3}, SH_{u3}, ME_{u3}, LO_{u3}, VLO_{u3}\}$ . Where  $VS, SH, ME, LO$  and  $VLO$  are MF labels corresponding to Very Short, Short, Medium, Long and Very long respectively.

The other two variables representing the activity data (activity count  $u_4$  and activity sequential order  $u_5$ ) will not be fuzzified with the other activity data. They will be normalised and then added to the fuzzy represented features before the entire set of input data  $X_U(t)$  is fed into the proposed models.

A set of fuzzy rules is required to control the transition between the system's states. In the standard FFSM, these rules are defined based on the experts' knowledge only. In this contribution, as the generated data is temporal data representing sequential order events, LSTM and CNN are employed to learn the relations in the data through the time steps. The learned relations are used to generate fuzzy rules in the system. The final output for this model,  $Y(t)$ , is represented as the degree of belonging to each state in the system.

## 6 Experimental results

The results obtained from the conducted experiments are presented here. As humans behave with some unpredictability and uncertainty in their environment, datasets representing the human activities are usually imbalanced, where some

activities appear more dominant than the other activities. In that case, if the dominant activities are identified with a high degree of accuracy, the performance over the whole system will be high even if the other activities are not well identified. Therefore, each activity will be evaluated separately, and then the performance over the whole system will

Output Activities	Sleeping	14 14.9%	0 0.0%	0 0.0%	0 0.0%	1 1.1%	0 0.0%	0 0.0%	93.3% 6.7%
	Toilet	0 0.0%	26 27.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Kitchen	0 0.0%	0 0.0%	12 12.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Dining	0 0.0%	0 0.0%	0 0.0%	20 21.3%	0 0.0%	1 1.1%	0 0.0%	95.2% 4.8%
	Living	0 0.0%	0 0.0%	1 1.1%	0 0.0%	12 12.8%	1 1.1%	0 0.0%	85.7% 14.3%
	Leaving	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 3.2%	0 0.0%	100% 0.0%
	Garden	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 3.2%	100% 0.0%
		100% 0.0%	100% 0.0%	92.3% 7.7%	100% 0.0%	92.3% 7.7%	60.0% 40.0%	100% 0.0%	95.7% 4.3%
		Sleeping	Toilet	Kitchen	Dining	Living	Leaving	Garden	
		Target Activities							

(a)

Output Activities	Sleeping	34 13.9%	0 0.0%	0 0.0%	0 0.0%	1 0.4%	0 0.0%	0 0.0%	97.1% 2.9%
	Toilet	0 0.0%	66 27.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Kitchen	0 0.0%	0 0.0%	30 12.3%	1 0.4%	0 0.0%	0 0.0%	2 0.8%	90.9% 9.1%
	Dining	0 0.0%	0 0.0%	0 0.0%	53 21.7%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Living	0 0.0%	0 0.0%	2 0.8%	0 0.0%	35 14.3%	2 0.8%	0 0.0%	89.7% 10.3%
	Leaving	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	10 4.1%	0 0.0%	100% 0.0%
	Garden	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	8 3.3%	100% 0.0%
		100% 0.0%	100% 0.0%	93.8% 6.3%	98.1% 1.9%	97.2% 2.8%	83.3% 16.7%	80.0% 20.0%	97.6% 2.4%
		Sleeping	Toilet	Kitchen	Dining	Living	Leaving	Garden	
		Target Activities							

(b)

Fig. 7 Confusion matrix for ADL modelling and recognition results using dataset A; a using LSTM-FFSM model, b using CNN-FFSM model

Output Activities	Sleeping	40 8.9%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.2%	97.6% 2.4%
	Bed-to-Toilet	0 0.0%	13 2.9%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.2%	0 0.0%	0 0.0%	92.9% 7.1%
	Meal Preparation	0 0.0%	0 0.0%	139 31.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	79.3% 20.7%
	Washing Dishes	0 0.0%	0 0.0%	0 0.0%	5 1.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	83.3% 16.7%
	Eating	0 0.0%	0 0.0%	0 0.0%	0 0.0%	5 1.1%	0 0.0%	1 0.2%	0 0.0%	0 0.0%	100% 0.0%
	Relaxing	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	145 32.4%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Leaving-home	0 0.0%	0 0.0%	0 0.0%	0 0.0%	22 4.9%	0 0.0%	53 11.9%	0 0.0%	0 0.0%	70.7% 29.3%
	Entering-home	0 0.0%	0 0.0%	0 0.0%	1 0.2%	0 0.0%	0 0.0%	0 0.0%	6 1.3%	0 0.0%	85.7% 14.3%
	Housekeeping	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.2%	0 0.0%	100% 0.0%
	Office-work	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	14 3.1%	100% 0.0%
	100% 0.0%	100% 0.0%	86.4% 13.6%	83.3% 16.7%	81.5% 18.5%	100% 0.0%	96.4% 3.6%	100% 0.0%	93.3% 6.7%	100% 0.0%	94.2% 5.8%
	Sleeping	Bed-to-Toilet	Meal Preparation	Washing Dishes	Eating	Relaxing	Leaving-home	Entering-home	Housekeeping	Office-work	
	Target Activities										

(a)

Output Activities	Sleeping	39 8.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Bed-to-Toilet	0 0.0%	13 2.9%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Meal Preparation	0 0.0%	0 0.0%	139 31.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Washing Dishes	0 0.0%	0 0.0%	0 0.0%	6 1.3%	0 0.0%	0 0.0%	0 0.0%	2 0.4%	0 0.0%	100% 0.0%
	Eating	0 0.0%	0 0.0%	0 0.0%	0 0.0%	27 6.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Relaxing	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	145 32.4%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Leaving-home	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	55 12.3%	0 0.0%	0 0.0%	100% 0.0%
	Entering-home	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 0.9%	0 0.0%	100% 0.0%
	Housekeeping	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.2%	100% 0.0%
	Office-work	1 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	15 3.4%
	97.5% 2.5%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	66.7% 33.3%	100% 0.0%	100% 0.0%	99.3% 0.7%
	Sleeping	Bed-to-Toilet	Meal Preparation	Washing Dishes	Eating	Relaxing	Leaving-home	Entering-home	Housekeeping	Office-work	
	Target Activities										

(b)

Fig. 8 Confusion matrix for ADL modelling and recognition results using dataset B; a using LSTM-FFSM model; b using CNN-FFSM model

be calculated. Both proposed models, LSTM-FFSM and CNN-FFSM, are tested and evaluated using the two earlier mentioned datasets A and B.

The confusion matrix in Figs. 7 and 8 shows the recall (known as sensitivity) and precision scores obtained using the proposed LSTM-FFSM and CNN-FFSM models for each activity. As well as the accuracy over the models. Fig. 7 shows the obtained results when the proposed



two models are applied to dataset A. The results obtained by applying LSTM-FFSM and CNN-FFSM based on dataset A are illustrated in Fig. 7a, b respectively.

The results based on the application of the proposed LSTM-FFSM and CNN-FFSM models to dataset B are shown in Fig. 8a, b, respectively. As it can be seen from these figures, CNN-FFSM model is more efficient when it is applied to a larger dataset containing confusion activities such as those activities that could occur at the same place (e.g., *Meal Preparation activity* and *Washing Dishes activity*), both of them undertaking at the kitchen. In Fig. 8b, nine out of ten activities, including in dataset B, are recognised with 100% scores of precision.

The information given in the confusion matrix is explained as follows:

- The rows represent the output activities, and columns represent the target activities. The activities in dataset A are named as *Sleeping, Toilet, Kitchen, Dining, Living, Leaving home, and Garden*. The activities in dataset B are named as *Sleeping, Bed-to-Toilet, Meal preparation, Washing dishes, Eating, Relaxing, Leaving-home, Entering-home, House-keeping, and Office-work*.
- The diagonal cells from the upper left to the lower right illustrate the activities that are correctly recognised.
- The off-diagonal cells present the incorrectly modelled and recognised activities.
- The precision for each activity is presented in the last column in the right.
- The recall for each activity is presented in the last row at the bottom.
- The accuracy over the whole model is illustrated in the bottom-right cell.

To evaluate and emphasise the proposed models, the results obtained in the experiments employing the LSTM-FFSM and CNN-FFSM models are compared with the results obtained using six different existing approaches for modelling and recognising human activities such as LSTM, SVM and NNs with both datasets A and B. The comparison between the performance of the proposed LSTM-FFSM and CNN-FFSM with the other existing methods has been made for the accuracy over the whole models in Table 2. The overall performance for the proposed models based on dataset A is 95.7% and 94.2% obtained by applying LSTM-FFSM and CNN-FFSM models, respectively. 97.6% and 99.3% are the obtained results based on dataset B using LSTM-FFSM and CNN-FFSM models, respectively.

The expressions that are used to calculate accuracy, precision and recall for each activity are given below:

**Table 2** The overall accuracy of the proposed LSTM-FFSM and CNN-FFSM comared with the existing approaches based on datasets A and B

Methods	Dataset A	Dataset B
CNN-FFSM	94.2%	<b>99.3%</b>
LSTM-FFSM	<b>95.7%</b>	97.6%
N-FFSM	95.2%	94.5%
LSTM	80.6%	78.1%
Bi-LSTM	83.7%	81.4%
NNs	78.7%	76.8%
FFSM	46.6%	41.3%
SVM	62.8%	60.9%

$$Recall = \frac{1}{C} \sum_{i=1}^C \frac{TP_i}{TP_i + FN_i} \tag{10}$$

$$Precision = \frac{1}{C} \sum_{i=1}^C \frac{TP_i}{TP_i + FP_i} \tag{11}$$

$$Accuracy = \frac{1}{N} \sum_{i=1}^C TP_i \tag{12}$$

where  $TP_i$ ,  $TN_i$ ,  $FN_i$  and  $FP_i$  are the number of true positives, true negatives, false negatives and false positives of  $i^{th}$  activity respectively.  $N$  is the number of the values  $TP_i + TN_i + FP_i + FN_i$  for  $i^{th}$  activity.  $C$  is the activity of which its recall, precision and accuracy are calculated.

Considering the interpretability point of view, the most commonly used approaches for modelling and recognising human activities are the approaches based on mathematical, e.g., NNs and SVM. These models are well known as black-box approaches because of the complexity of understanding their underlying calculations and concepts. This complexity will be more challenging when a large number of input and output variables are expected. As well as designing a model only based on the linguistic information assigned by human experts is not enough for a successful and robust human activities modelling and recognition model. Therefore, the advantages of integrating the experts' knowledge with the learning capabilities in LSTM and CNN can be integrated into the proposed LSTM-FFSM and CNN-FFSM models for generating a successful and robust model that can be used for modelling and recognising human activities.

The obtained results are compared with some previous works, such as the research presented in (Rashidi et al. 2010). They have used a dataset that was collected using ambient sensory devices to discover 5 different activities; *Telephone use, Hand washing, Meal preparation, Eating, and Cleaning*. A new mining method, called Discontinuous

Varied-Order Sequential Miner (DVSM), is used in this research with the collected dataset to find frequent patterns that may be discontinuous and might have variability in the ordering of this activities. The achieved results based on the DVSM is 77.3%.

In a recent publication for recognition of interleaved human activities, researchers have proposed a new human activity model containing three phases, namely Preprocessing, Discovery Method for Varying Patterns (DMVP) and Predictive Modelling (Raeiszadeh et al. 2019). The first phase is used to convert the collected raw sensor data into event sequences, which are then fed to DMVP in the second phase to discover frequent activities that naturally happen during the normal daily routine, and then a classification model is applied to predict the activities in the third phase. The achieved results from this approach is 87.94% once it is evaluated with the CASAS dataset.

## 7 Conclusion

The work presented in this paper has proposed two new methods for improving modelling and recognising human activities using data gathered from low-level sensory devices. Considering the results obtained from the conducted experiments, it can be concluded that the LSTM-FFSM and CNN-FFSM models exhibit a high score for accuracy, recall, and precision when its performance is tested for each activity separately. Also, the overall activity recognition performance, when it is over the whole system, demonstrates the effectiveness of the proposed approaches. The CNN-FFSM model shows more robust and reliable performance once applied to a larger dataset (e.g., dataset B) representing ten activities over 240 days. In particular, when this dataset contains some activities that could be happening at the same place (such as *Washing Dishes activity* and *Meal Preparation activity* in the kitchen). In real-life scenarios, it is hard to know which activity is the current activity based on the data collected from the PIR sensory devices. Thus using a fuzzy feature representation approach with CNN-FFSM will be used to deal with such cases as it can detect the changes in the fuzzy feature patterns. The essential feature of the proposed approach is that it integrates the available expert's knowledge with the learned information from the deep learning techniques. The LSTM-FFSM has shown better performance for a simple scenario once it is applied to a short period dataset (e.g., dataset A). The CNN-FFSM achieved more accurate results to detect the ADL activities for a longer period dataset (e.g., dataset B). Besides, it can be seen how the proposed models can follow the proper sequence of states with the correct state activation degree.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Aicha AN, Englebienne G, Kröse B (2017) Unsupervised visit detection in smart homes. *Pervasive Mobile Comput* 34:157–167
- Alvarez-Alvarez A, Trivino G, Cordon O (2011) Body posture recognition by means of a genetic fuzzy finite state machine. In: IEEE 5th International Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS), pp 60–65
- Alvarez-Alvarez A, Trivino G, Cordon O (2012) Human gait modeling using a genetic fuzzy finite state machine. *IEEE Trans Fuzzy Syst* 20:205–223
- Ambres O, Trivino G (2012) Gait quality monitoring using an arbitrarily oriented smartphone. In: *International Workshop on Ambient Assisted Living*, Springer. pp 224–231
- Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL (2012) Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In: *International workshop on ambient assisted living*, Springer, New York, pp 216–223
- Arifoglu D, Bouchachia A (2019) Detection of abnormal behaviour for dementia sufferers using convolutional neural networks. *Artif Intell Med* 94:88–95
- Benmansour A, Bouchachia A, Feham M (2017) Modeling interaction in multi-resident activities. *Neurocomputing* 230:133–142
- Bombardier V, Schmitt E (2010) Fuzzy rule classifier: capability for generalization in wood color recognition. *Eng Appl Artif Intell* 23:978–988
- Chen L, Hoey J, Nugent CD, Cook DJ, Yu Z (2012) Sensor-based activity recognition. *IEEE Trans Syst Man Cybern Part C (Applications and Reviews)* 42:790–808
- Chung PC, Liu CD (2008) A daily behavior enabled hidden markov model for human behavior understanding. *Pattern Recognit* 41:1572–1580
- Cook DJ (2010) Learning setting-generalized activity models for smart spaces. *IEEE Intell Syst* 2010:1
- Cook DJ, Crandall AS, Thomas BL, Krishnan NC (2013) Casas: a smart home in a box. *Computer* 46:62–69
- Deng Y, Ren Z, Kong Y, Bao F, Dai Q (2017) A hierarchical fused fuzzy deep neural network for data classification. *IEEE Trans Fuzzy Syst* 25:1006–1012
- Flagel L, Brandvain Y, Schrider DR (2018) The unreasonable effectiveness of convolutional neural networks in population genetic inference. *Mol Biol Evol* 36:220–238
- Gochoo M, Tan TH, Liu SH, Jean FR, Alnajjar FS, Huang SC (2019) Unobtrusive activity recognition of elderly people living alone using anonymous binary sensors and dcnn. *IEEE J Biomed Health Inform* 23:693–702
- Hassan MM, Uddin MZ, Mohamed A, Almogren A (2018) A robust human activity recognition system using smartphone sensors and deep learning. *Future Gen Comput Syst* 81:307–313

- Jenckel M, Parkala SS, Bukhari SS, Dengel A (2018) Impact of training lstm-rnn with fuzzy ground truth. In: ICPRAM, IEEE. pp 388–393
- Khemchandani R, Sharma S (2017) Robust parametric twin support vector machine and its application in human activity recognition. In: Proceedings of International Conference on Computer Vision and Image Processing, Springer, New York. pp 193–203
- Kong Y, Fu Y (2018) Human action recognition and prediction: a survey. arXiv preprint [arXiv:1806.11230](https://arxiv.org/abs/1806.11230)
- Langensiepen C, Lotfi A, Puteh S (2014) Activities recognition and worker profiling in the intelligent office environment using a fuzzy finite state machine. In: IEEE international conference on fuzzy systems (FUZZ-IEEE), pp. 873–880
- Lotfi A, Langensiepen C, Mahmoud SM, Akhlaghinia MJ (2012) Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour. *J Ambient Intell Humaniz Comput* 3:205–218
- Malasinghe LP, Ramzan N, Dahal K (2019) Remote patient monitoring: a comprehensive study. *J Ambient Intell Humaniz Comput* 10:57–76
- Medina-Quero J, Zhang S, Nugent C, Espinilla M (2018) Ensemble classifier of long short-term memory with fuzzy temporal windows on binary sensors for activity recognition. *Expert Syst Appl* 114:441–453
- Mohmed G, Lotfi A, Langensiepen C, Pourabdollah A (2018a) Clustering-based fuzzy finite state machine for human activity recognition. In: UK Workshop on Computational Intelligence, Springer, New York, pp 264–275
- Mohmed G, Lotfi A, Pourabdollah A (2018b) Human activities recognition based on neuro-fuzzy finite state machine. *Technologies* 6:110
- Mohmed G, Lotfi A, Pourabdollah A (2019) Long short-term memory fuzzy finite state machine for human activity modelling. In: Proceedings of the 12th ACM International Conference on Pervasive Technologies Related to Assistive Environments, Association for Computing Machinery, New York, NY, USA. p 561–567
- Raeiszadeh M, Tahayori H, Visconti A (2019) Discovering varying patterns of normal and interleaved ads in smart homes. *Applied Intelligence* 1–14
- Rashidi P, Cook DJ, Holder LB, Schmitter-Edgecombe M (2010) Discovering activities to recognize and track in a smart environment. *IEEE Trans Knowl Data Eng* 23:527–539
- Sridhar K, Baskar S, Shakeel PM, Dhulipala VS (2019) Developing brain abnormality recognize system using multi-objective pattern producing neural network. *J Ambient Intell Humaniz Comput* 10:3287–3295
- Subramanian K, Suresh S (2012) Human action recognition using meta-cognitive neuro-fuzzy inference system. *Int J Neural Syst* 22:1250028
- Tran N, Nguyen T, Nguyen BM, Nguyen G (2018) A multivariate fuzzy time series resource forecast model for clouds using lstm and data correlation analysis. *Procedia Comput Sci* 126:636–645
- Trinh H, Fan Q, Jiyan P, Gabbur P, Miyazawa S, Pankanti S (2011) Detecting human activities in retail surveillance using hierarchical finite state machine. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 1337–1340
- Unal FA, Khan E (1994) A fuzzy finite state machine implementation based on a neural fuzzy system. In: Proceedings of the Third IEEE World Congress on Computational Intelligence, pp 1749–1754
- Wang LX, Mendel JM (1992) Generating fuzzy rules by learning from examples. *IEEE Trans Syst Man Cybern* 22:1414–1427
- Wang Z, Jiang M, Hu Y, Li H (2012) An incremental learning method based on probabilistic neural networks and adjustable fuzzy clustering for human activity recognition by using wearable sensors. *IEEE Trans Inf Technol Biomed* 16:691–699
- Wongpatikaseree K, Ikeda M, Buranarach M, Supnithi T, Lim AO, Tan Y (2012) Activity recognition using context-aware infrastructure ontology in smart home domain. In: 2012 Seventh International Conference on Knowledge, Information and Creativity Support Systems, IEEE, pp 50–57
- Yulita IN, Fanany MI, Arymurthy AM (2017a) Fuzzy clustering and bidirectional long short-term memory for sleep stages classification. In: International Conference on Soft Computing, Intelligent System and Information Technology (ICSIT), pp 11–16
- Yulita IN, Fanany MI, Arymurthy AM (2017b) Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification. *Procedia Comput Sci* 116:530–538

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.