IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 20, 2022, accepted March 4, 2022, date of publication March 8, 2022, date of current version March 17, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3157726

Activities Recognition, Anomaly Detection and Next Activity Prediction Based on Neural Networks in Smart Homes

KHALED A. ALAGHBARI[1], **MOHAMAD HANIF MD. SAAD**[1,2], **AINI HUSSAIN**[3], **AND MUHAMMAD RAISUL ALAM**[1,4]

[1]Institute of IR4.0, Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor 43600, Malaysia
[2]Department of Mechanical and Manufacturing Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor 43600, Malaysia
[3]Department of Electrical, Electronic and System, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor 43600, Malaysia
[4]Department of Computer Science, Department of Occupational Science & Occupational Therapy, University of Toronto, Toronto, ON M5S 1A1, Canada

Corresponding author: Mohamad Hanif Md. Saad (hanifsaad@ukm.edu.my)

This work was supported by the Universiti Kebangsaan Malaysia under Grant DIP-2020-021.

ABSTRACT In this paper, we propose a unified deep learning model for monitoring elderly in execution of daily life activities such as eating, sleeping or taking medication. The proposed approach consists of three stages which are activity recognition, anomaly detection and next activity prediction. Such a system can provide useful information for the elderly, caregivers and medical teams to identify activities and generate preventive and corrective measures. In literature, these stages are discussed separately, however, in our approach, we make use of each stage to progress into the next stage. At first, activity recognition based on different extracted features is performed using a deep neural network (DNN), then an overcomplete-deep autoencoder (OCD-AE) is employed to separate the normal from anomalous activities. Finally, a cleaned sequence of consecutive activities is constructed and used by a long short-term memory (LSTM) algorithm to predict the next activity. Since the last two stages depend on the activity recognition stage, we propose to increase its accuracy by exploiting different extracted features. The performance of the proposed unified approach has been evaluated on real smart home datasets to demonstrate its ability to recognize activities, detect anomalies and predict the next activity.

INDEX TERMS Smart homes, activity recognition, anomaly detection, sequence prediction, deep neural network, autoencoder, LSTM.

I. INTRODUCTION

The care of elderly people who are unable to effectively develop activities of daily living (ADL) requires a lot of attention and dedication, as both their lifestyle and health are affected. The spread of dementia-related problems in older adults aged 60 years or above is one of the world's major public health challenges [1]. For instance, an elderly person suffering from Alzheimer's disease may forget to have their medicine, lunch or wake up in the middle of the night. As a result, secondary issues may arise that affect mental, physical, and mobility capabilities [2], [3]. Nowadays, there is a growing need for society to take care of their health while incorporating the use of technology to develop assisted living technology (ALT) such as human activity recognition (HAR) and anomalous detection. The ALT enables monitoring of people's quality of life, and as time goes on, new features and functions emerge in this domain, relying on a diverse set of hardware and software components. The daily home activities that involve basic functions like preparing meals, walking, sleeping, showering, etc. can be used to evaluate the well-being of elderly people. The goals of ALT are: i) the development of predictive models that allow for the classification of normal and abnormal behaviour in individuals [4], and ii) the provision of tools for caregivers and medical teams to identify activities and generate preventive and corrective measures [5], [6].

Sensor-based ambient systems in smart homes can be used to recognise various behaviours and complex activities [7] by monitoring the interaction between objects and inhabitants.

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

This process can be done using machine learning (ML) techniques which can effectively classify ADLs performed by people. Existing activity classification methods employ a variety of ML techniques to recognise activities, including the Hidden Markov Model (HMM) [8], Conditional Random Fields (CRF) [9], Random Forest (RF) [10], Support Vector Machine (SVM) [11], Naive Bayes (NB) [12], Decision Tree (DT) [10] and K-nearest neighbour (KNN) [13]. Recently, increased attention has been given to deep learning (DL) techniques such as deep neural network (DNN), convolutional neural network (CNN), autoencoders (AE) and recurrent neural networks (RNN) in several fields because they have great impacts in terms of flexibility and performance. The accuracy of the ML and DL techniques to classify activities is an important measure that depends on the classifier parameters and on extracted features from a pre-segmented dataset. Number of active sensors, start time, end time and duration of activity are example of extracted features. The extracted features are critical for the classifiers to learn useful representations and capture spatial information and local dependency of granular-level patterns [14].

Another useful technique is the prediction of the next sensor event based on the sequence of events [15], [16], which usually used to improve the operation of automation functions such as adjusting the temperature sufficient time prior to the person waking up, informing the resident if the predicted activity has not performed yet or recognizing changes in person's habits [17], [18].

Most of the previous studies, consider activity recognition (AR) [9], [12], anomalous detection (AD) [19], and next sequence prediction [17], [20] separately, or AR and AD are discussed together only [2], [14], [21]. A comprehensive tool that provides features and functionality such as activity recognition, anomalous detection and next activity prediction will be useful to asses behaviour change, detect early, meaningful, cognitive change in order to prevent or delay the impact of dementia.

In this paper, we propose an elderly monitoring system that consists of three stages which are activity recognition, anomalous detection and next activity prediction. At first, we propose to increase the accuracy of activity recognition using different extracted features from pre-labelled activity instances, train and test the ability of the DNN model to classify a given activity, and secondly, a proposed overcomplete autoencoder (OCD-AE) is used to identify anomalous instances within each activity class. An activity instance is considered anomalous if its execution time is unusually long or if it contains an unusually large number of sub-events. After activity classification and anomaly extraction, a clean sequence of consecutive activities is formed. Finally, the sequence is trained using long short-term memory (LSTM) algorithm and its model is used to predict the next activity. A unified monitoring system with such features can provide useful information for elderly people who suffer from dementia, and for medical team to analysis elderly health and take preventive action once needed. A dataset from CASAS project called Aruba [22] and Cairo are selected for this study since its labelled activities are good example to represent the daily life pattern of elderly residents in smart homes. The usage of the proposed unified model can be extended and implemented for different smart home applications such as energy-saving or security monitoring and threat detection [23].

The rest of the paper is organised as follows. Section 2 provides an overview of related works. Section 3 presents the details of the proposed approach together with the models used. Section 4 describes the dataset, extracted features, and experimental results for activity recognition, anomaly detection and next activity prediction followed by a discussion. Finally, Section 5 concludes the paper.

II. RELATED WORKS

A. ACTIVITY RECOGNITION

Several datasets such as CASAS [22], [24] and Kasteren [25] are provided for the public to test the ability of ML and DL classifiers to recognise human activities in smart homes. The Aruba dataset [22] that was collected as a part of the CASAS project suffers from class imbalance problem which reduces the performance of the classifiers to recognise some of the activities (i.e., "Resperate" activity). Features are important factors that play significant roles in the performance of activity recognition techniques. Several studies have discussed feature extraction from different datasets. In [10], many features are extracted such as start and end time of activity, duration of activity, sensors states and location of activities. Then different ML classifiers are used such as RF, DT and Naïve Bayes for activity recognition task, best accuracy is achieved by RF with 75.82%. In [14], last-fired sensor (i.e., if there are multiple sensors, then the sensor that lastly changes its state is represented by 1 and other sensor states are set to 0) is considered as feature and trained by combination of CNN-2D and LSTM to achieve accuracy of 89.72%. In [11], last-state sensor (i.e., each sensor is represented by its final state ON/OFF), mutual information of sensors and its extension method are exploited to attain accuracy of 87.71%. In [12], graph-based features are extracted by representing motion sensors in a graph and resident's movements as edges in the graph, accuracy of 93.41% is obtained by using SVM classifier. In [13], number of times the sensors are activated during activity and feature selection based on principle component analysis (PCA) are used. In [21], three features namely duration of activity, number of active sensor events within each activity and total number of each activity performed per day are used to train probabilistic neural network (PNN) to achieve accuracy of 90%. In our paper, we proposed to add sensor states feature in addition to the three features used in [21] to increase the activity recognition task. Increasing the accuracy of activity recognition stage has a great impact on reducing error-propagation to anomaly detection and next activity prediction stages.

B. ANOMALY DETECTION

Anomaly detection is the process of detecting unusual behavior in occupancy's normal lifestyle. In [19], convLSTM autoencoder is used for anomaly detection, instances with high reconstruction error are assumed to be anomalies because they cannot be reconstructed like normal instances. This is because the autoencoder has been used to train the normal activities only, hence it will be able to do the construction for the normal activities, but, the anomalous activities are not trained, so the reconstruction error is expected to be higher. However, the activities in the dataset have been modified to reflect abnormal behavior related to dementia such as activity repetition, disturbance in sleep and confusion, for instance, activities such as having multiple lunches or forgetting to have dinner and prepare it in irrelevant time are synthesized in the dataset. Hence, about 150 activities such as "Eating", "Bed_to_toilet", and "Sleeping" are added in [19]. In [14], a method is proposed to artificially produce abnormal activities to reflect typical behaviors of elderly people suffering from dementia. The highest sensitivity is achieved by HMM, while the highest specificity is achieved by LSTM. The technique of adding synthesized data is done because there exists no publicly available dataset on abnormal behaviors for people suffering from dementia. In [21], three levels of anomaly detection are proposed based on boxplot outlier analysis such as instances of activities with irregular number of subevents, unusual durations of activities, and irregular frequency of activities performed in a day. H2O autoencoder is used as a binary classifier within each class. Anomalies detection based on statistical information of activities can be a promising solution since it learns what is normal and abnormal from generating ground truth using the training data. The ground truth is the information that is known to be real or true, provided by direct observation. Since it is difficult to obtain real observed data, boxplot analysis (such as median, maximum, minimum and outliers) can be used to have approximate information about activities such duration, number of active sensors and so on. This technique has been widely used in previous works such as [21]. In our paper, we have adopted this technique to detect anomalous instances since it doesn't require any modification to the dataset.

C. SEQUENCE PREDICTION

Sequence prediction provides the ability to know in advance the next activity, which can be a helpful aid for adults with cognitive impairment or dementia [17]. In recent years, a number of algorithms for sequence prediction have been investigated. These algorithms typically train a model to predict the next sensor event based on a sequence of symbols, where the symbol represents the sensor states, for instance, sensor 1 and sensor 2 are presented by capital letter A and B in ON state and small letters a and b in OFF states. Active LeZi (ALZ) predicts the next symbol in a sequence using Markov Models [26]. ALZ algorithm is an improved version of the LZ78 algorithm that uses a sliding window technique [16], [27]. ALZ involves a variable-length window of previously observed events. The Sequence Prediction via Enhanced Episode Discovery (SPEED) algorithm, inspired by ALZ, predicts the next sensor event based on the frequency of observed patterns [15]. In [17], a recurrent neural network (RNN) with long short-term memory (LSTM) is used as a sequence prediction method, where the LSTM is configured as a text generation network. In this paper, we applied a different approach from the above papers, for example instead of taking a sequence of sensor events to apply sequence prediction algorithms, we used a sequence of activities that has been identified by an activity recognition classifier and categorized as a normal activity by an anomaly detector. The activities in a sequence are then converted to capital letters and used by LZ78 and ALZ algorithms, while for LSTM it is converted into one-hot encoding.

III. UNIFIED DEEP LEARNING MODEL

Let $A = \{A_1, \ldots A_k, \ldots A_K\}$ be a set of K activity classes and $I_k = \{I_{1k}, \ldots, I_{jk}, \ldots . I_{Jk}\}$ be a set of J activity instances of A_k in training data set observed by S sensors deployed at different locations in a smart home. Each instance of an activity I_{jk} is represented by a set of R features, $F_{jk} = \left\{f_{jk}^r\right\}_{r=1}^R$. The features represent the duration spent to complete an activity, the number of sensors that are activated during an activity, the number of times an activity is performed by day, and the states (e. g., ON/OFF) of all sensors during an activity. Our proposed unified model consists of three steps as shown in Fig. 1. In the first step, it extracts useful features from a given pre-segmented or labeled activities dataset. The extracted features should provide enough information about the classes so that a classifier can give high prediction accuracy. These features are then divided into train and test features. The second step is to train our models for activity recognition and anomaly detection. We propose to use a deep neural network (DNN) for activity recognition and an overcomplete-deep autoencoder (OCD-AE) for anomaly detection. The third step is to do testing process. Once the DNN model is built, it is used for testing new features to classify the activity. Finally, the AE model uses the activity class and the input features to determine if an anomalous instance is detected. The correctly identified normal activities are sent to the sequence construction block to form a series of consecutive sequences. LSTM algorithm is then trained and employed to predict the next activity.

A. ACTIVITY RECOGNITION

We use a deep neural network (DNN) for activity recognition and compared its performance with other popular classifiers such as support vector machine (SVM), K-nearest neighbor (KNN), decision tree (DT), random forest (RF), and Naïve Bayes (NB). The training of the classifiers is performed on a training set containing different features of activity instances for each activity class. As shown in Fig. 2, the DNN has three main layers namely: input layer which has the feature set F_{jk}

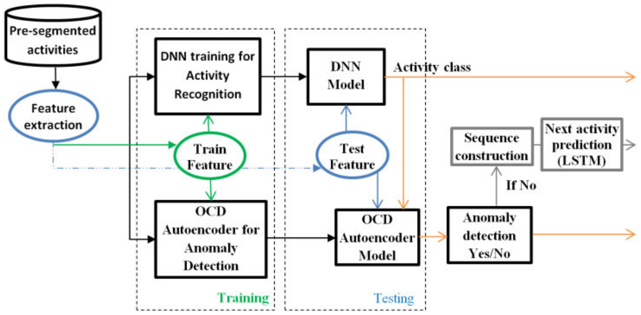


FIGURE 1. Block diagram of the proposed unified deep learning model consisting of activity recognition, anomaly detection, and next activity prediction algorithms.

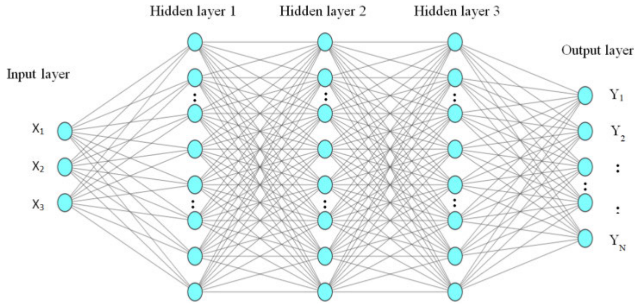


FIGURE 2. Architecture of deep neural network (DNN) used for training of activity recognition system.

of the training samples. The second layer is the hidden layer which consists of many hidden layers with Relu activation functions for the first layers and Softmax activation function for the last layer. The third layer is the output layer which has the same number of neurons as the activity classes. The trained DNN classifier can be used later to recognize new activity events.

B. ANOMALY DETECTION

The recognized activities are analyzed to detect anomalies, which are unusual and unexpected deviations from standard patterns. Anomalies in smart homes deviate from the normal trend in terms of an unusual number of events and an unusual duration. For the detection of anomalies, we use learning algorithm based on autoencoder. Autoencoder (AE) is a special type of artificial neural network that consists of input, encoder, feature (bottleneck layer), and decoder layers as shown in Fig. 3. The encoder aims at mapping the input data into N-dimensional representation feature, the function of the decoder is to do the reverse mapping, where the input and output data of the network should be identical. Reconstruction error (RE) is the difference between the input and output of the AE. The AE is usually trained to minimize this error to effectively reproduce the input from the features representation. The AE is a semi-supervised technique that only used to train part of the data (i.e., normal activity). Hence, the trained activity (i.e., normal activity) is expected to have smaller RE, whereas the RE for non-trained activity (i.e., anomalous activity) is expected to have higher RE since the autoencoder has never encountered the non-trained activity before. There are two types of AEs that based on the number of neurons in the encoder layer which are undercomplete and overcomplete AEs. The encoder layer dimensionality of an undercomplete AE is smaller than the input layer. In this type the AE learns to compress high dimensional input data into lower dimensional in order to capture the most important features, this technique is known as dimensionality reduction. On the other hand, the overcomplete AE has more neurons in the encoder layer than the input. This type of network architecture gives the possibility of learning a greater number of features; however, it may lead to learning the identity function of original input and become useless. Sparse AE [28] is a technique proposed in the literature to prevent AE from learning identity function by employing sparsity in the encoder and decoder layers, in other words, only a small fraction of neurons are allowed to be active during the training stage. An alternative approach is to use multiple hidden layers within the encoder and the decoder which we propose to use for anomaly detection within each activity class. We trained an overcomplete-deep autoencoder (OCD-AE) with normal instances only, and a threshold value is chosen based on reconstruction error for each class to detect anomalous instances. The reconstruction error (RE) is the difference between the original data and its N-dimensional reconstruction output, which is used as an anomaly score to detect anomalies [29]. The performance of the proposed OCD-AE is compared with two popular classifiers such as SVM and K-means.

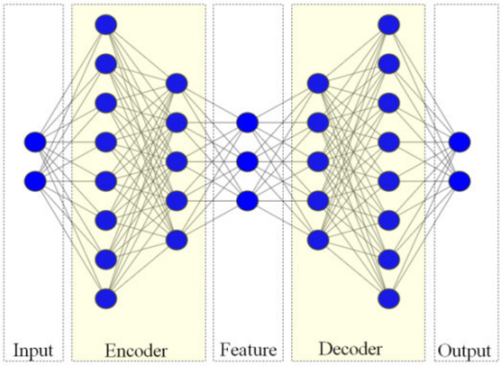


FIGURE 3. Structure of the proposed overcomplete-deep autoencoder for anomaly detection.

C. SEQUENCE PREDICTION

A sequence of activities can be constructed after performing activity recognition and discarding the anomalous instance. This stage can help the elderly to recall what activity to do next. If there is abnormal activity, it may affect the prediction. In other words, the presence of abnormal activities can be treated as noise that negatively affects the next sequence prediction and hence it should be removed. The cleaned sequence can be then trained and tested to predict the next activity. LSTM [30] is a type of recurrent neural network (RNN) designed to be better in storing and accessing internal memory than the standard RNN. We used the LSTM

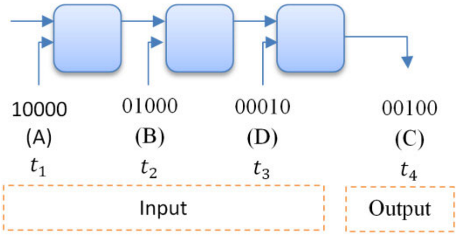


FIGURE 4. LSTM network configuration used for next activity prediction.

network as a text generation network where the number of inputs is equal to the memory length and the output is the predicted next activity in the sequence as shown in Fig. 4. The sequence of activities are first converted into letters (symbols), then both input and output of LSTM network are converted into one-hot encoded form, where each symbol in the sequence is represented by a vector of bits (input raw data) with a length equal to the number of symbols in a sequence. For example if we have sequence = [A, B, C, D, E], where the number of symbols is 5, each symbol will be converted into bits of length 5 such as [10000, 01000, 00100, 00010, 00001], where 1 is placed at specific bit and zeros elsewhere for each symbol. This conversion is widely used in machine learning techniques and LSTM for categorical data, but it is computationally expensive especially when there is a large number of activities. We compared the performance of the LSTM algorithm with two popular sequence prediction algorithms such as ALZ [26] and LZ78 [27][16].

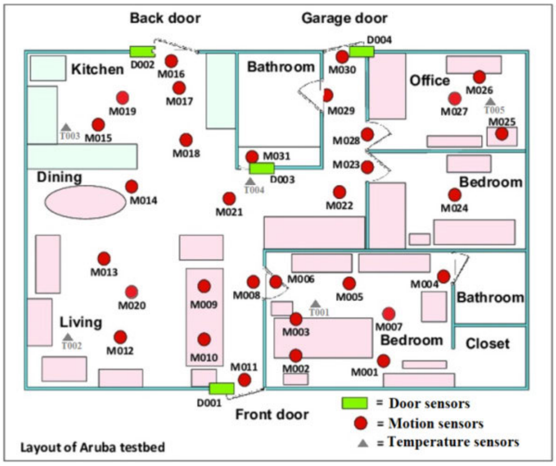
IV. RESULTS AND DISCUSSION

We evaluate our proposed monitoring system on a CASAS smart home datasets called Aruba and Cairo. In this section, we first present an overview of the datasets, followed by feature analysis and extraction. Then we evaluate the performance of different classifiers to recognize activities performed in smart homes, and separate anomalous activities from normal activity. The activities are then organized as consecutive series of events. A sequence prediction technique is employed to predict the next activity. The classification models are implemented in Python 3 using Keras and Sklearn open source libraries.

A. DATASETS

We use Aruba and Cairo dataset to evaluate the performance of our proposed algorithms. The Aruba dataset consists of data from a total of 39 sensors, out of which 31 are used for motion sensors (M001-M031), 3 door closure sensors (D001, D002, and D004) and five temperature sensors (T001-T005). The activities of an elderly woman living in a house with layout as shown in Fig.5a are recorded for 220 consecutive days (~7 months). There are 11 activities annotated within the dataset which consists of 1,719533 raw sensor data as shown in Fig. 5b.

The number of times the activity appears in the dataset is given in Table 1a. As shown, the dataset is imbalanced, as some activities occur more frequently than others. The table also shows the activities performed on the first and second days as an example. Some activities like sleeping begins at one day and completes in the next day, so in our analysis we assumed this activity belongs to the starting day only.



(a)

Date	Time	Sensor		Activity
		ID	State	
2010-11-04	00:03:50.209589	M003	1	Sleeping begin
2010-11-04	00:03:57.399391	M003	0	
2010-11-04	00:15:08.984841	T002	21.5	
2010-11-04	00:30:19.185547	T003	21	
2010-11-04	00:30:19.385336	T004	21	
…				
2010-11-04	05:40:40.482626	M003	0	
2010-11-04	05:40:40.844463	M003	1	
2010-11-04	05:40:42.452746	M007	1	
2010-11-04	05:40:43.642664	M003	0	Sleeping end
2010-11-04	05:40:44.223548	M003	1	
2010-11-04	05:40:45.939846	M005	1	
2010-11-04	05:40:46.310862	M003	0	
2010-11-04	05:40:51.303739	M004	1	Bed_to_Toilet begin
2010-11-04	05:40:52.342105	M005	0	
2010-11-04	05:40:57.176409	M007	0	
…				

(b)

FIGURE 5. (a) Aruba layout (b) dataset.

Cairo dataset consists of 27 motion sensors (M001-M027) and 5 temperature sensors (T001-T005). It has 13 activities as listed in Table 1b with shortcuts used in our paper. The dataset is collected for two adult couples (R1 and R2) and a dog living in the smart home. The couple's children also visited the house at least once. The data consist of 726534 raw sensors reading for 57 days distributed for the 13 activities as described in Table 1b.

B. FEATURE EXTRACTION

From the dataset, we extract four features for activities recognition. The first feature is the "**duration**" which defines the total time spent to complete an activity. The second feature is the "**sensor count**" which represents the number of times each sensor remains active during an activity. The third feature is the "**activities per day**" which describes the number

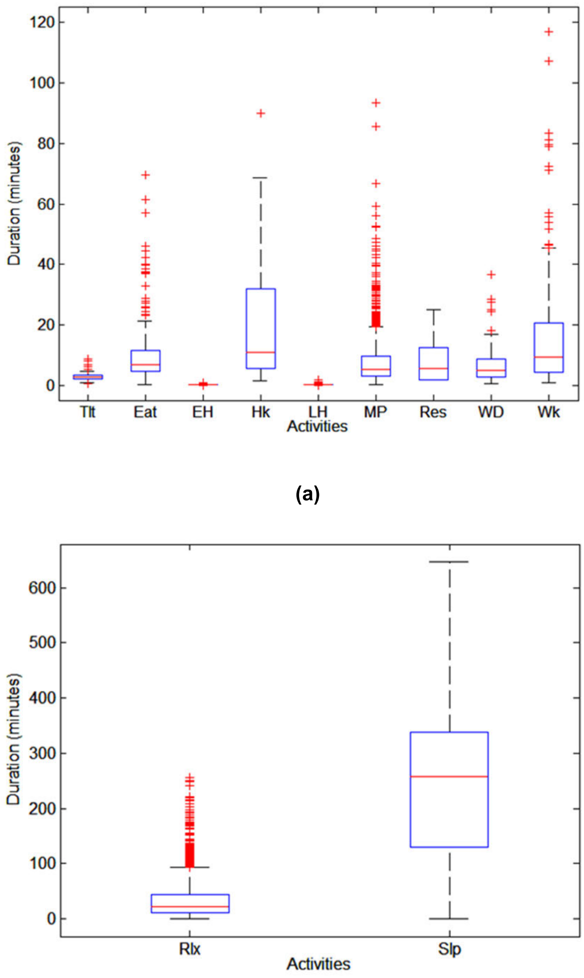
TABLE 1. The annotated activities in (a) Aruba (b) Cairo datasets.

(a)

No.	Shortcuts	Activity	Performed in day 1	Performed in day 2	Performed in 220 days
1	Tlt	Bed_to_Toilet	1	1	157
2	Eat	Eating	4	1	257
3	EH	Enter_Home	2	6	431
4	Hk	Housekeeping	2	1	33
5	LH	Leave_Home	2	6	431
6	MP	Meal_Preparation	9	3	1606
7	Rlx	Relax	5	2	2919
8	Res	Resperate	0	0	6
9	Slp	Sleeping	2	3	401
10	WD	Wash_Dishes	2	1	65
11	Wk	Work	2	7	171
		Total	31	31	6477

(b)

No.	Shortcuts	Activity	Performed in 57 days
1	BTlt	Bed_to_Toilet	30
2	BrkF	Breakfast	48
3	Dinn	Dinner	42
4	Laun	Laundry	10
5	LHom	Leave_Home	69
6	Lund	Lunch	37
7	NiWa	Night_wandering	67
8	R1Sl	R1_sleep	50
9	R1Wa	R1_wake	53
10	R1Wo	R1_work_in_office	46
11	R2Sl	R2_sleep	52
12	R2Me	R2_take_medicine	44
13	R2Wa	R2_wake	52
		Total	600



(a)

(b)

FIGURE 6. The duration in minutes for the 11 activities in Aruba dataset.

of times each activity is performed in a single day. The fourth feature is the set of **"sensor states"** which describes the states of sensors (ON/OFF) during an activity.

Fig. 6 shows boxplot for the duration of activities performed by an elderly woman for 220 days, we plotted Relax (Rlx) and Sleeping (Slp) activities in separate figure because they takes longer duration than others. For illustration purpose, we omitted in Fig. 6 one outlier from the Sleeping activity and one outlier from Meal Preparation activity because they take longer duration than the remaining instances (i.e., 1141 min and 326 min respectively). The box covers the interquartile (IQ or Q_2) interval, where 50% of the data is found. The lower and higher sides of the box are the Q_1 and Q_3 quartiles. The horizontal red line that split the box in two is the median. The whisker is represented by the two lines outside the box, extended from the lowest to highest points that represent the minimum and maximum respectively. Data points that are outside this interval are marked with a red colour plus '+' on the graph and considered outliers. The distributions for most of the activities are skewed since the median is not in the middle of the box, and contains many outliers except for Sleeping (Slp), Resperate (Res) and Housekeeping (HK) activities. Also, the data is imbalanced since the whisker length on the lower side is shorter with no outliers and longer on the upper side with many outliers as in Relax (Rlx), Eat and Meal Preparation (MP) activities.

Fig. 7 shows the number of times each sensor remains active during the performance of an activity. We take into consideration all the motion, doors and temperature sensors as [21]. We can observe that some activities such as Housekeeping (HK), Resperate (Res) and Work (Wk) have no outliers whereas others such as Relax (Rlx) and Sleep (Slp) have many outliers. Moreover, Enter Home (EH) and Leave Home (LH) activities have almost the same boxplots and outliers, which create confusion for the classifier to distinguish between them.

In Fig. 8a, we plotted a boxplot for the third feature, which describes the number of times each activity is performed per day. By summing up all the activities performed per day, we plotted the histogram as in Fig. 8b. The fitting curve on the histogram shows that an average of 29 activities is performed per day with a standard deviation of 7 days. The information provided by the figure determines the daily routine of an elderly resident, moreover, it provides information on whether the elderly is able to independently execute daily activities or need an intervention [21], for instance, a day is considered as anomalous if the number of activities executed in that day deviates from the normal pattern.

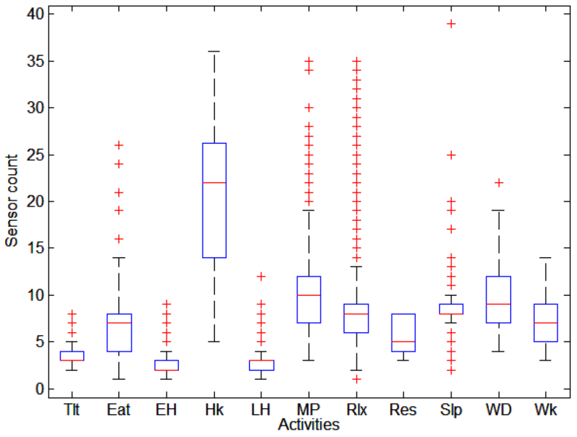
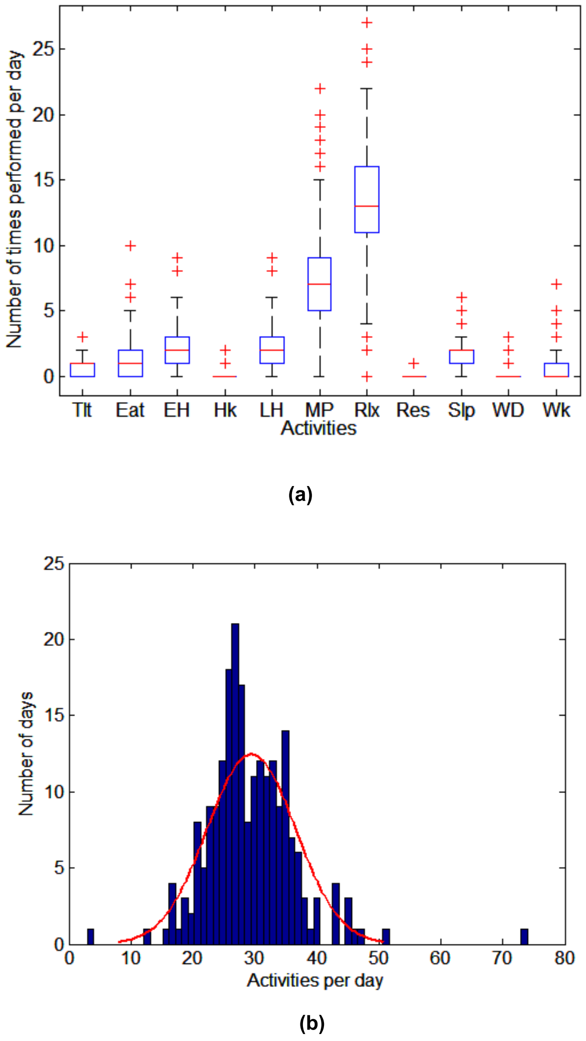


FIGURE 7. The number of times each sensor remains active during an activity performance in Aruba dataset.



(a)

(b)

FIGURE 8. (a) number of times each activity is performed per day (b) histogram of activities per day for Aruba dataset.

We also extracted the three features presented earlier for the Cairo dataset as shown in Fig. 9. The duration boxplot (Fig. 9a) shows the activities have different execution times and the highest outliers are found on Night Wandering activity. The boxplot of the sensor count (Fig. 9b) shows the Cairo dataset is more balancing than Aruba dataset, and it has fewer outliers in terms of sensor count. Fig. 9c shows that most of the activities are performed once a day. The histogram plot (Fig. 9d) shows that on average 10 activities are performed per day with a standard deviation of 3 days.

TABLE 2. Example of the extracted features from Aruba dataset.

Duration	Sensor Count	Activity Per day	Sensor State			Activity/ Class
			S1	S2	…	
2.6495	3	1	OFF	OFF	…	Tlt
2.6783	3	1	OFF	OFF	…	Tlt
…						
2.1131	10	7	OFF	OFF	…	MP
2.5382	6	7	OFF	OFF	…	MP
…						
8.6331	6	12	OFF	OFF	…	Rlx
11.4304	8	10	OFF	OFF	…	Rlx

C. ACTIVITY RECOGNITION ANALYSIS

The extracted feature based on the label activities has 6477 instances, Table. 2 shows an example of the extracted features that will be used for activities recognition.

We used the following performance metrics to evaluate a classifier [31]:

$$\text{Precision (specificity)} = \text{TP}/(\text{TP} + \text{FP}) \tag{1}$$

$$\text{Recall (sensitivity)} = \text{TP}/(\text{TP} + \text{FN}) \tag{2}$$

Precision and recall are used to measure how well the classifiers perform on an imbalance dataset. The balance between the precision (P) and the recall (R) scores is described by F1-score as [32]:

$$\text{F1-score} = (2^{*}\text{Precision}^{*}\text{Recall})/(\text{Precision} + \text{Recall}) \tag{3}$$

On the other hand, the accuracy metric is given by:

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{4}$$

which represents the percentage of correctly classified activities, where TP is true positive, TN is true negative, FP is false positive and FN is false negative. Matthew's correlation coefficient (MCC) is a coefficient that ranges between [−1, 1]. A coefficient of 1 indicates a perfect classifier, while 0 means a totally random prediction. A coefficient of −1 indicates a negative correlation or total disagreement between the prediction and actual value. The MCC is calculated as [33]:

$$MCC = (\text{TP} * \text{TN} - \text{FP} * \text{FN}) \,/\, \sqrt{(\text{TP+FP}) * (\text{TP+FN}) * (\text{TN+FP}) * (\text{TN+FN})} \tag{5}$$

We used a deep neural network with 3 input layers, three hidden layers with 128 neurons each and output layer with 11 neurons. Relu activation function is used for the first and second hidden layers, where softmax function is used for the last layer. Adam optimizer is used with a learning rate of 0.001. The training of the network is performed with 50 epochs consisting of 128 batches. The Aruba data which

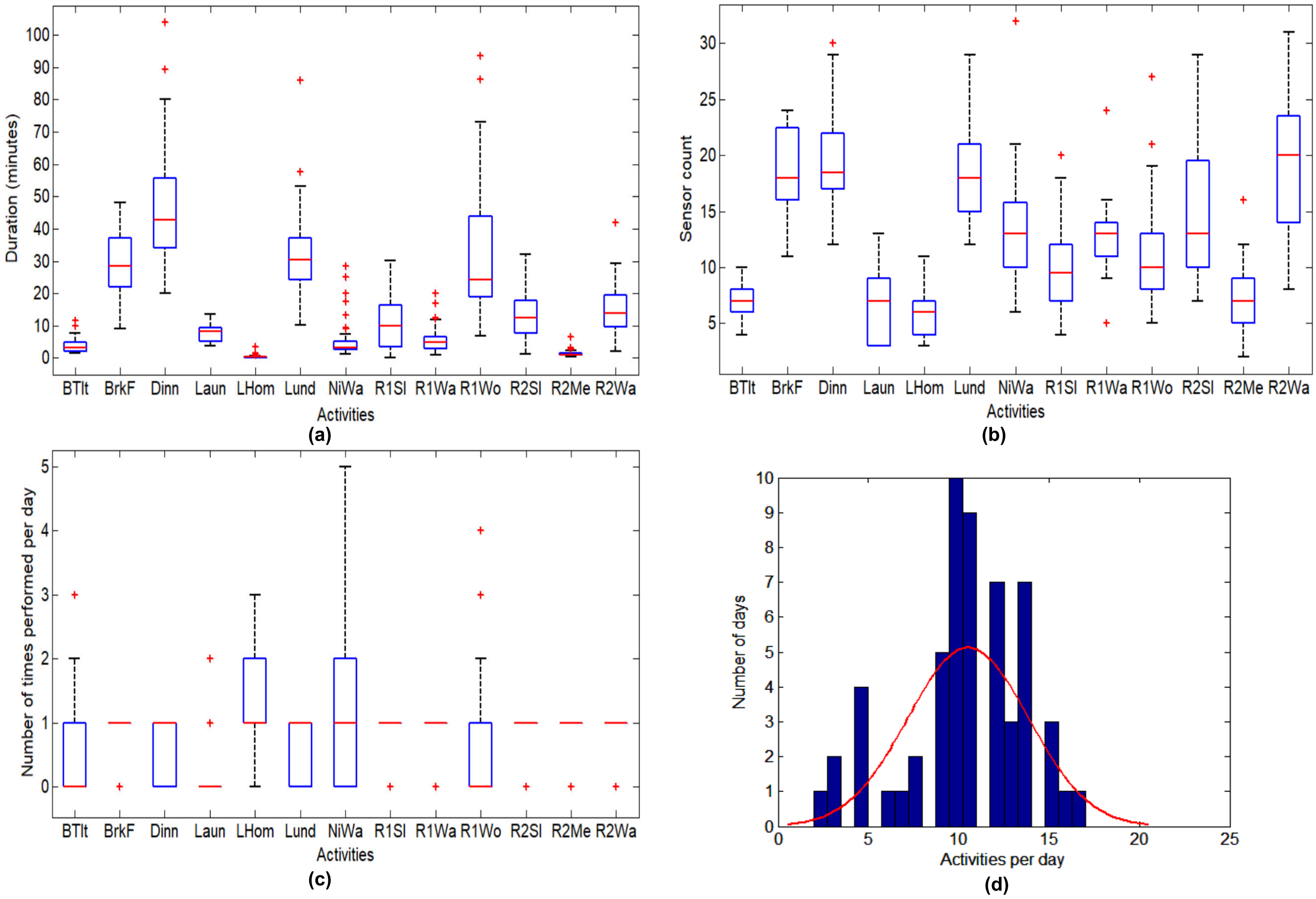


FIGURE 9. Extracted features from Cairo dataset, (a) duration (b) sensor count (c) activity performed per day (d) histogram of activities per day.

includes 6477 instances is divided into 70% for training and 30% for testing. The accuracy and loss learning curves for the trained DNN are shown in Fig. 10.

The confusion matrix and metrics performance of DNN are shown in Fig. 11. From the confusion matrix of the DNN classifier, it can be noted that most of FPs and FNs of leave home (LH) activity is recognized as enter home (EH) activity and vice versa. The two activities are mixed up because the entry and exit of the house use the same main door, and the same sensors are used to capture their events, thus, the input features (duration, sensor counts and activity per days) are high likely to be similar. The 'meal preparation' activity and 'relax' activity are also mixed up even though they involved different sensors; this is because the used input features have common values between the two activities. Therefore, other input features are needed by the classifier to distinguish between these activities.

We also compared the DNN with other different classifiers for activity recognition such as decision tree, random forest, k-nearest neighbor (KNN), support vector machine (SVM), Naïve Bayes and deep neural network (DNN). Table 3a shows the metrics performance of those classifiers using the 3 input features described in Table 2. The best performance is achieved by DNN followed by decision tree classifier.

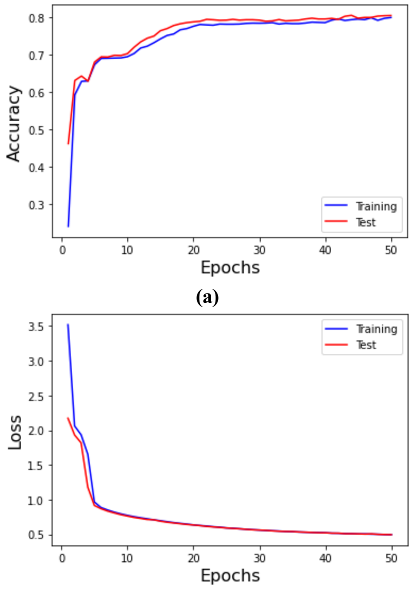
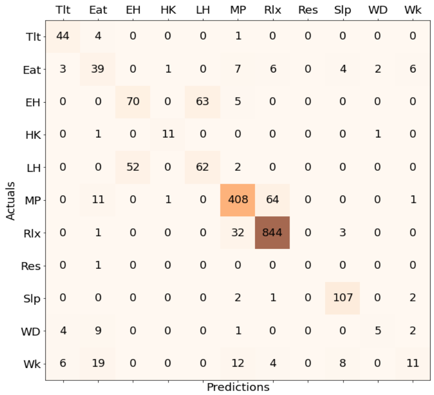


FIGURE 10. (a) Accuracy and (b) Loss performance of the DNN for the activity recognition in Aruba dataset.

The DNN has shown good performance with limit resource of input features.



Actuals \ Predictions	Tlt	Eat	EH	HK	LH	MP	Rlx	Res	Slp	WD	Wk
Tlt	44	4	0	0	0	1	0	0	0	0	0
Eat	3	39	0	1	0	7	6	0	4	2	6
EH	0	0	70	0	63	5	0	0	0	0	0
HK	0	1	0	11	0	0	0	0	0	1	0
LH	0	0	52	0	62	2	0	0	0	0	0
MP	0	11	0	1	0	408	64	0	0	0	1
Rlx	0	1	0	0	0	32	844	0	3	0	0
Res	0	1	0	0	0	0	0	0	0	0	0
Slp	0	0	0	0	0	2	1	0	107	0	2
WD	4	9	0	0	0	1	0	0	0	5	2
Wk	6	19	0	0	0	12	4	0	8	0	11

(a)

Act	TP	FP	TN	FN	Acc.	Prec.	Rec.	F1 sc.	MCC
Tlt	44	13	1881	5	0.99	0.77	0.90	0.83	0.83
Eat	39	46	1829	29	0.96	0.46	0.57	0.51	0.49
EH	70	52	1753	68	0.94	0.57	0.51	0.54	0.50
HK	11	2	1928	2	100	0.85	0.85	0.85	0.85
LH	62	63	1764	54	0.94	0.50	0.53	0.51	0.48
MP	408	62	1396	77	0.93	0.87	0.84	0.85	0.81
Rlx	844	75	988	36	0.94	0.92	0.96	0.94	0.89
Res	0	0	1942	1	100	0.00	0.00	0.00	-
Slp	107	15	1816	5	100	0.88	0.96	0.91	0.91
WD	5	3	1919	16	0.99	0.62	0.24	0.34	0.38
Wk	11	11	1872	49	0.97	0.50	0.18	0.27	0.29
All	**1601**	**-**	**-**	**342**	**0.97**	**0.63**	**0.59**	**0.60**	**0.64**

Overall accuracy = (1601)/(1601+342) = **0.82**

(b)

FIGURE 11. (a) Confusion matrix and (b) Performance evaluation metrics using DNN for Aruba dataset.

The overall performance of the classifiers using the three input features is not adequate; therefore in Table 3b we evaluated the performance of the classifiers using the 39 sensors states, where active sensors during the activity are set to 1, otherwise set to 0. We can observe the metrics performance has been improved for all the classifiers, for example the precision and F1-score of decision tree classifier has been improved from 61% to 77% and from 60% to 78% respectively. The decision tree classifier has shown comparable performance to DNN.

To further improve the classification task, we combined the three input features with the binary sensor states and evaluated the performance of the classifiers as shown in Table 3c. The table shows further improvement in all the metrics performance. The best classification has been achieved by linear SVM, where the F1-score has been improved from 51% in Table 3a to 71% in Table 3b to 90% in Table 3c. It is worth mentioning that, the three classifiers DNN, Decision tree and SVM (linear) have shown competitive performance in the three scenarios.

Table 4 provides a summary of the features that are extracted from the Aruba dataset, the classifiers used and metrics performance that has been achieved in other literature paper in comparison to best result obtained by our works.

For Cairo dataset, Table 5 shows comparison of the classifiers using 35 inputs which are duration, sensor count, activity per day and the states of 32 sensors during an activity. The best performance is achieved by the proposed DNN. In [12], accuracy of 70.10% is achieved using motion sensor ON/OFF, 72.73% using count of motion sensors, and 76.57% using graphic-based method. Our achieved result for Cairo dataset is very closed to the graphic-based method used in [12].

TABLE 3. Comparison of different classifiers using a) 3 inputs only (duration + sensor count +activity per day) b) 39 inputs (All sensor states during activities) c) 42 inputs (duration + sensor count +activity per day + All sensor states during activities), for Aruba dataset.

(a)

Classifier/Metric	Accuracy	Precision	Recall	F1-score	MCC
Decision Tree	**0.82**	0.61	**0.60**	0.60	0.63
Random Forest	**0.82**	0.56	0.55	0.55	0.57
KNN (N = 7)	**0.82**	0.54	0.52	0.52	0.55
SVM (RBF)	0.75	0.30	0.34	0.30	0.53
SVM (linear)	0.80	0.54	0.56	0.51	0.63
Naïve Bayes	0.74	0.51	0.56	0.47	0.46
Proposed DNN	**0.82**	**0.63**	0.59	**0.60**	**0.64**

(b)

Classifier/Metric	Accuracy	Precision	Recall	F1-score	MCC
Decision Tree	0.92	**0.77**	**0.80**	**0.78**	0.77
Random Forest	0.91	0.73	0.67	0.69	0.69
KNN (N = 7)	**0.92**	0.71	0.65	0.67	0.81
SVM (RBF)	**0.92**	0.74	0.67	0.69	0.84
SVM (linear)	**0.92**	0.74	0.70	0.71	**0.86**
Naïve Bayes	0.62	0.65	0.76	0.56	0.59
Proposed DNN	0.91	0.72	0.70	0.71	0.70

(c)

Classifier/Metric	Accuracy	Precision	Recall	F1-score	MCC
Decision Tree	**0.93**	0.82	0.88	0.82	0.83
Random Forest	**0.93**	0.88	0.84	0.86	0.85
KNN (N = 7)	0.89	0.74	0.65	0.68	0.74
SVM (RBF)	0.80	0.39	0.37	0.34	0.62
SVM (linear)	**0.93**	**0.91**	**0.90**	**0.90**	**0.90**
Naïve Bayes	0.80	0.71	0.84	0.70	0.72
Proposed DNN	**0.93**	0.88	0.88	0.88	0.87

D. ANOMALY DETECTION

After performing activity recognition, it is important to check if the given activity is normal or abnormal. However, the dataset does not provide ground truth to know the anomaly detection. We can generate approximate ground truth based on boxplot analysis by defining two conditions of anomaly events. The first condition is the deviation of duration of activity from the whisker (Fig. 6), and the second condition is based on the deviation of number of sub-events for activity from whisker (Fig. 7), where the whiskers are $Q_3 + 1.5(Q_3 - Q_1)$ and $Q_1 - 1.5(Q_3 - Q_1)$ and Q_s is the s^{th} quartile. If a certain event satisfies the two conditions, it is considered an anomaly. Table 6 shows the number of normal and anomaly events for each activity of the Aruba and Cairo datasets. We can observe the number of normal behavior is greater than the anomalous per activity, activity like 'Resperate' has no anomaly event recorded and activity 'Housekeeping' has only one anomaly. In the Cairo dataset, the number of anomaly instances is very small compared to Aruba instances.

TABLE 4. Summary of results, features and methods used in the literature papers for activity recognition based on Aruba dataset.

Ref. Year	Accuracy %	Precision %	Recall %	F-score [0-1]	Features	Method/Classifier	Other
[21] 2021	90	74	80	0.74	– Duration of activity – Count of the sensor events within activity – Count of activity times performed per day	– Probabilistic Neural Network (PNN)	70%:30% Training-to-test
[10] 2020	75.82	-	-	-	– Start time of activity – End time of activity – Duration of activity – Sensors states – location	– Random Forest (RF)* – Decision Tree (DT) – Extra-tree (ExT) – Naive Bayes (NB)	Ten folds cross-validation
[14] 2019	89.72	51.2	50.55	50.87	– Last-fired sensor	– CNN-1D – CNN-2D – CNN-2D+LSTM*	139 days for training and 70 days for testing
[11] 2015	87.23	-	-	0.63	– Time of the first sensor events – Time of the last sensor events – Duration of the window Ei – Count of the different sensor events within the window	– Support vector machine (SVM)	Cross-validation
	87.71	-	-	0.68	– Mutual information (MI) of sensors – Extension method of mutual information of sensors – Last-state Sensor method	– Support vector machine (SVM)	
[12] 2014	93.31	-	-	-	– Sensor states – Count of the sensor events within activity	– Naive Bayes (NB), – Hidden Markov Model (HMM) – Conditional Random Fields (CRF) – Support vector machine (SVM)*	Ten folds cross-validation
	93.41	-	-	-	– Graphical feature		
[13] 2014	91.40	79.65	76.46	0.75	– Number of times the sensors are activated during activity – Feature selection based on Principle component analysis (PCA) is used.	– Clustering based Classification (AR-CbC)* – ET-KNN – KNN – PNN	Three folds ross-validation
Our work	93	91	90	0.9	– Duration of activity – Count of the sensor events within an activity – Count of activity times performed per day – Sensor states	– Decision Tree – Random Forest – KNN – SVM* – Naïve Bayes – Deep neural network	70%:30% Training-to-test

TABLE 5. Comparison of different classifiers using 35 inputs (duration + sensor count +activity per day + All sensor states during activities) for Cairo dataset.

Classifier/Metric	Accuracy	Precision	Recall	F1-score	MCC
Decision Tree	0.69	0.67	0.69	0.66	0.65
Random Forest	0.71	0.69	0.69	0.68	0.66
KNN (N = 7)	0.60	0.60	0.55	0.53	0.52
SVM (RBF)	0.50	0.34	0.42	0.36	0.50
SVM (linear)	0.74	**0.74**	0.70	0.70	0.69
Naïve Bayes	0.66	0.64	0.62	0.58	0.58
Proposed DNN	**0.76**	**0.74**	**0.73**	**0.73**	**0.71**

Within each activity class, the anomaly detection approach identifies activity instances with abnormal duration and number of events. The Aruba datasets are used with a training-to-test ratio of 70:30 for each activity class. The anomalies correctly identified by the anomaly detection approach are represented by true positives (TPs).

Since we have only two input features (duration and sensor count), we propose to use OCD-AE with hidden layers consist of 32, 14, 7 and 14 neurons each. Activity regularizer of 10^{-7} and reduction in hidden layers neurons are used to prevent AE from learning identity function and improve the ability to capture useful features. The performance evaluation of the proposed autoencoder and other popular classifiers such as SVM and K-means for anomaly detection in the Aruba dataset are shown in Table 7. We also include the results obtained by the H2O autoencoder used in [21] for comparison purpose. It is noted that the proposed OCD-AE correctly identifies the maximum number of TPs (anomalies) in 10 out of 11 activities with accuracy of more than 90%. The K-means classifier shows a comparable performance with accuracy of more than 90% for 8 activities, in which 4 of them achieved higher accuracy than the proposed autoencoder.

TABLE 6. Number of normal and anomaly activities in (a) Aruba (b) Cairo dataset.

(a)

No.	Shortcuts	Activity	Performed in 220 days	Normal	Anomaly
1	Tlt	Bed_to_Toilet	157	121	36
2	Eat	Eating	257	230	27
3	EH	Enter_Home	431	366	65
4	Hk	Housekeeping	33	32	1
5	LH	Leave_Home	431	362	69
6	MP	Meal_Preparation	1606	1461	145
7	Rlx	Relax	2919	2601	318
8	Res	Resperate	6	6	0
9	Slp	Sleeping	401	362	39
10	WD	Wash_Dishes	65	58	7
11	Wk	Work	171	155	16
		Total	6477	5754	723

(b)

No.	Shortcuts	Activity	Performed In 57 days	Normal	Anomaly
1	BTlt	Bed_to_Toilet	30	28	2
2	BrkF	Breakfast	48	48	0
3	Dinn	Dinner	42	39	3
4	Laun	Laundry	10	10	0
5	LHom	Leave_Home	69	65	4
6	Lund	Lunch	37	35	2
7	NiWa	Night_wandering	67	58	9
8	R1Sl	R1_sleep	50	49	1
9	R1Wa	R1_wake	53	47	6
10	R1Wo	R1_work_in_office	46	42	4
11	R2Sl	R2_sleep	52	52	0
12	R2Me	R2_take_medicine	44	39	5
13	R2Wa	R2_wake	52	51	1
		Total	600	563	37

TABLE 7. Anomaly detection comparison of different techniques within each class of Aruba dataset.

No.	Activity (No.test)	Classifier	TP	FP	TN	FN	Accuracy	Precision	Recall	F1-score	MCC
1	(47)	[21]	8	2	37	0	0.95	0.80	1.00	0.88	-
	Tlt (48)	SVM (linear)	13	1	33	1	0.96	0.95	0.95	0.95	0.90
		K-means (N=7)	11	0	34	3	0.94	0.96	0.89	0.92	0.85
		Proposed OCD-AE	13	0	35	0	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
2	(77)	[21]	11	0	66	0	**1.00**	**1.00**	**1.00**	**1.00**	-
	Eat (78)	SVM (linear)	7	5	66	0	0.94	0.79	0.96	0.85	0.74
		K-means (N=7)	4	0	71	3	0.96	0.98	0.79	0.85	0.74
		Proposed OCD-AE	11	2	65	0	0.97	0.92	0.99	0.95	0.91
3	(129)	[21]	14	1	111	3	0.96	0.93	0.82	0.87	-
	EH (130)	SVM (linear)	15	8	106	1	0.93	0.82	0.93	0.86	0.75
		K-means (N=7)	15	0	114	1	**0.99**	**1.00**	**0.97**	**0.98**	**0.96**
		Proposed OCD-AE	14	0	107	9	0.93	0.96	0.80	0.86	0.75
4	(9)	[21]	0	0	8	1	0.88	-	0	0	-
	Hk (10)	SVM (linear)	0	0	10	0	1.00	1.00	1.00	1.00	1.00
		K-means (N=7)	0	0	10	0	1.00	1.00	1.00	1.00	1.00
		Proposed OCD-AE	1	0	9	0	1.00	1.00	1.00	1.00	1.00
5	(129)	[21]	17	1	106	5	0.95	0.94	0.77	0.85	-
	LH (130)	SVM (linear)	14	116	0	0	0.11	0.05	0.50	0.10	-
		K-means (N=7)	9	0	116	5	**0.96**	**0.98**	**0.82**	**0.88**	**0.79**
		Proposed OCD-AE	10	0	106	14	0.89	0.94	0.71	0.76	0.61
6	(481)	[21]	40	7	428	6	0.97	0.85	0.87	0.86	-
	MP (482)	SVM (linear)	47	23	412	0	0.95	0.84	0.97	0.89	0.80
		K-means (N=7)	42	0	435	5	**0.99**	**0.99**	0.95	**0.97**	**0.94**
		Proposed OCD-AE	50	12	419	1	0.97	0.90	**0.98**	0.93	0.88
7	(875)	[21]	64	1	762	48	0.94	0.98	0.57	0.72	-
	Rlx (876)	SVM (linear)	89	62	721	4	0.92	0.79	0.94	0.84	0.72
		K-means (N=7)	87	2	781	6	**0.99**	**0.98**	**0.97**	**0.98**	**0.95**
		Proposed OCD-AE	79	2	779	16	0.98	**0.98**	0.91	0.94	0.89
8	(1)	[21]	0	0	1	0	1	-	-	-	-
	Res (2)	SVM (linear)	-	-	-	-	-	-	-	-	-
		K-means (N=7)	-	-	-	-	-	-	-	-	-
		Proposed OCD-AE	-	-	2	-	1	1	1	1	-
9	(120)	[21]	5	0	108	7	0.94	1	0.41	0.58	-
	Slp (121)	SVM (linear)	15	30	74	2	0.74	0.65	0.80	0.65	0.43
		K-means (N=7)	3	4	100	14	0.85	0.65	0.57	0.58	0.21
		Proposed OCD-AE	6	3	111	1	**0.97**	**0.83**	**0.92**	**0.87**	**0.74**
10	(19)	[21]	0	0	17	2	0.89	-	0	0	-
	WD (20)	SVM (linear)	3	0	16	1	0.95	0.97	0.88	0.91	0.84
		K-means (N=7)	0	0	16	4	0.80	0.40	0.50	0.44	-
		Proposed OCD-AE	2	0	18	0	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
11	(51)	[21]	5	0	45	1	0.98	1.00	0.83	0.90	-
	Wk (52)	SVM (linear)	3	0	49	0	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
		K-means (N=7)	3	1	48	0	0.98	0.88	0.99	0.92	0.86
		Proposed OCD-AE	5	1	44	2	0.94	0.89	0.85	0.87	0.74

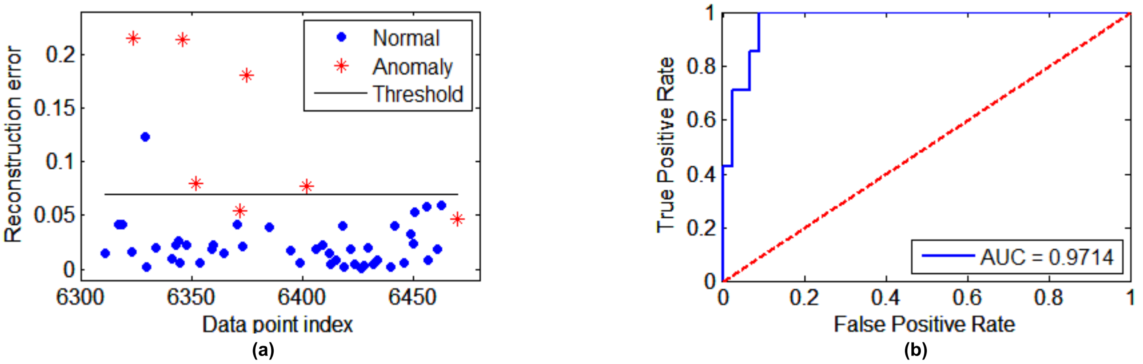


FIGURE 12. (a) Autoencoder reconstruction error (b) Receiver operating characteristic (ROC) for activity 'work' in Aruba dataset.

Fig. 12a shows example of AE reconstruction error that is used to classify normal and anomaly event for the activity 'Work'. The threshold value is chosen to be 0.07 to decide the anomaly events. Five events are TPs classified correctly as anomaly and 44 events are TNs classified correctly as normal events. However, two anomaly events are FPs classified wrongly as normal and 1 event is FN classified as anomaly event. The drawback of autoencoder is the requirement to manually select the threshold value based on the reconstruction error to classify anomaly events. Fig. 12b shows receiver operating characteristic (ROC) plot that demonstrates the good ability of the autoencoder to discriminate between normal and anomaly cases.

TABLE 8. Anomaly detection comparison of different techniques within each class of Cairo dataset.

No.	Activity (No. test)	Classifier	TP	FP	TN	FN	Accuracy	Precision	Recall	F1-score	MCC
1	BTlt (9)	SVM (linear)	1	0	8	0	1.00	1.00	1.00	1.00	1
		K-means (N=7)	0	0	8	1	0.89	0.44	0.50	0.47	-
		Proposed OCD-AE	1	0	8	0	1.00	1.00	1.00	1.00	1
2	BrKe (15)	SVM (linear)	-	-	-	-	-	-	-	-	-
		K-means (N=7)	0	0	15	0	1.00	1.00	1.00	1.00	-
		Proposed OCD-AE	0	0	15	0	1.00	1.00	1.00	1.00	-
3	Dinn (13)	SVM (linear)	1	2	10	0	0.85	0.67	0.92	0.70	0.53
		K-means (N=7)	0	0	12	1	0.92	0.46	0.50	0.48	-
		Proposed OCD-AE	0	0	13	0	**1.00**	**1.00**	**1.00**	**1.00**	-
4	Laun (3)	SVM (linear)	-	-	-	-	-	-	-	-	-
		K-means (N=7)	0	0	3	0	1.00	1.00	1.00	1.00	-
		Proposed OCD-AE	0	0	3	0	1.00	1.00	1.00	1.00	-
5	LHome (21)	SVM (linear)	0	0	20	1	0.95	0.48	0.50	0.49	-
		K-means (N=7)	0	0	20	1	0.95	0.48	0.50	0.49	-
		Proposed OCD-AE	0	0	20	1	0.95	0.48	0.50	0.49	-
6	Lund (12)	SVM (linear)	0	0	12	0	1.00	1.00	1.00	1.00	-
		K-means (N=7)	0	0	12	0	1.00	1.00	1.00	1.00	-
		Proposed OCD-AE	1	0	11	0	1.00	1.00	1.00	1.00	1.00
7	NiWa (21)	SVM (linear)	4	1	16	0	0.95	0.90	0.97	0.93	0.86
		K-means (N=7)	1	0	17	3	0.86	0.93	0.62	0.66	0.46
		Proposed OCD-AE	3	0	18	0	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
8	R1Slp (15)	SVM (linear)	-	-	-	-	-	-	-	-	-
		K-means (N=7)	0	0	14	1	0.93	0.47	0.50	0.48	-
		Proposed OCD-AE	0	0	15	0	**1.00**	**1.00**	**1.00**	**1.00**	-
9	R1Wa (16)	SVM (linear)	3	1	12	0	0.94	0.88	0.96	0.91	0.83
		K-means (N=7)	0	0	13	3	0.81	0.41	0.50	0.45	-
		Proposed OCD-AE	1	0	15	0	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
10	R1Wo (14)	SVM (linear)	0	0	14	0	1.00	1.00	1.00	1.00	-
		K-means (N=7)	0	0	14	0	1.00	1.00	1.00	1.00	-
		Proposed OCD-AE	2	0	12	0	1.00	1.00	1.00	1.00	1.00
11	R2Slp (16)	SVM (linear)	-	-	-	-	-	-	-	-	-
		K-means (N=7)	0	0	16	0	1.00	1.00	1.00	1.00	-
		Proposed OCD-AE	0	0	16	0	1.00	1.00	1.00	1.00	-
12	R2Me (14)	SVM (linear)	2	1	11	0	**0.93**	0.83	**0.96**	**0.88**	0.78
		K-means (N=7)	0	0	12	2	0.86	0.43	0.50	0.46	-
		Proposed OCD-AE	1	0	11	2	0.86	**0.92**	0.67	0.71	0.53
13	R2Wa (16)	SVM (linear)	-	-	-	-	-	-	-	-	-
		K-means (N=7)	0	0	15	1	0.94	0.47	0.50	0.48	-
		Proposed OCD-AE	1	0	15	0	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

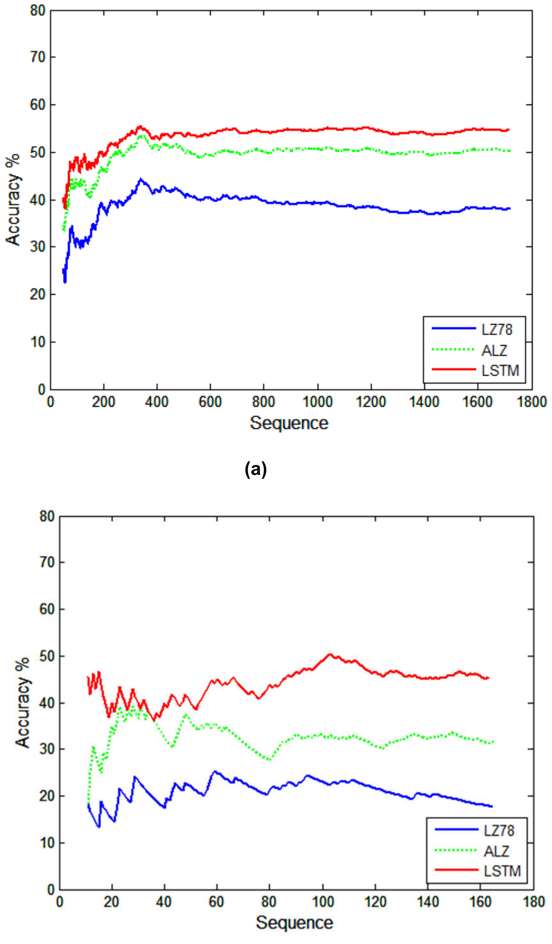
Table 8 shows the anomaly detection for Cairo dataset, the proposed OCD-AE performs the best in most of the activities and only it has been outperformed by SVM in activity R2_take_medicien (R2Me).

E. NEXT ACTIVITY PREDICTION

The activities performed by a resident can be arranged in a sequence to predict the next activity. Certain activities are easily predictable, such as ''leave home'' activity is always followed by ''enter home'', ''Sleeping'' activity is sometimes followed by ''bed_to_toilet'' activity, however, other activities need a technique to learn the pattern of activities performance. To predict the next activity, we arranged the normal activities as sequence and employed three algorithms such as LZ78, ALZ and LSTM to do the prediction. In the Aruba dataset, LZ78 generated a tree with 808 nodes while ALZ generated a tree of 20819 nodes using 70% of the sequence as training. The remaining 30% of the sequence is used for testing the accuracies of the algorithms. The accuracy of sequential prediction algorithms for the activities in Aruba and Cairo datasets are shown in Fig. 13. LZ78 and ALZ give accuracies of 38% and 50.3% respectively with a sequence length of 4. LSTM gives an accuracy of 54.6% with memory length of 4 using two hidden layers with 32 neurons each and 0.5 dropout layer between them. For the Cairo dataset, the best accuracy is achieved by LSTM with 45.4%. The accuracy of sequence prediction algorithms are low due to the limited length of sequence used for training the models, for example, the length of the training sequence in the Aruba dataset is greater than the Cairo dataset, therefore, we can notice an improvement in the algorithms accuracy of Aruba dataset.

V. CONCLUSION

This paper proposed a new sensor-based unified deep learning model for monitoring the elderly with cognitive impairments (such as dementia) living in a smart home. Our proposed method can recognize activities, detect anomalies and predict next activity by exploiting neural networks techniques such as deep neural network (DNN), overcomplete-deep autoencoder (OCD-AE) and long short-term memory (LSTM) network respectively. The accuracy of the activity recognition task has been increased by adding more features extracted from the dataset. Our results on activity recognition show great competition between DNN, Decision tree and SVM classifiers. Accuracies of 93% and 76% have been achieved in Aruba and Cairo datasets respectively by DNN. In addition, results on anomaly detection give



(a)

(b)

FIGURE 13. Sequence prediction accuracy achieved by different algorithms for (a) Aruba (b) Cairo datasets without anomaly instances.

promising results to detect most of the abnormal behaviors with accuracies of more than 90% detected using the proposed OCD-AE by considering boxplot outliers as ground truth. Finally, a clean sequence of activities has been constructed by discarding the anomalous instances. A sequence prediction algorithm such as LSTM has been used to predict the next activity. The proposed approach provides a comprehensive monitoring system that has the ability to recognize activities, detect anomalies and predict the next activity to assist elderly and medical team to identify health situations and generate preventive and corrective measures. A future work for this research paper can be extended to study approaches to handle the imbalance dataset to improve the deep learning models.

REFERENCES

[1] C. Zucchella, E. Sinforiani, S. Tamburin, A. Federico, E. Mantovani, S. Bernini, R. Casale, and M. Bartolo, "The multidisciplinary approach to Alzheimer's disease and Dementia. A narrative review of non-pharmacological treatment," *Frontiers Neurol.*, vol. 9, p. 1058, Dec. 2018, doi: 10.3389/fneur.2018.01058.

[2] D. Arifoglu and A. Bouchachia, "Activity recognition and abnormal behaviour detection with recurrent neural networks," *Proc. Comput. Sci.*, vol. 110, pp. 86–93, Feb. 2017, doi: 10.1016/j.procs.2017.06.121.

[3] J. Van Erum, D. Van Dam, and P. P. De Deyn, "Alzheimer's disease: Neurotransmitters of the sleep-wake cycle," *Neurosci. Biobehav. Rev.*, vol. 105, pp. 72–80, Oct. 2019.

[4] A. Aztiria, "Learning frequent behaviours of the users in intelligent environments," *J. Ambient Intell. Smart Environ.*, vol. 2, no. 4, pp. 435–436, 2010, doi: 10.3233/AIS-2010-0084.

[5] J. Sturge, M. Klaassens, D. Lager, G. Weitkamp, D. Vegter, and L. Meijering, "Using the concept of activity space to understand the social health of older adults living with memory problems and dementia at home," *Social Sci. Med.*, vol. 288, Nov. 2021, Art. no. 113208, doi: 10.1016/j.socscimed.2020.113208.

[6] E. De-La-Hoz-Franco, P. Ariza-Colpas, J. M. Quero, and M. Espinilla, "Sensor-based datasets for human activity recognition—A systematic review of literature," *IEEE Access*, vol. 6, pp. 59192–59210, 2018, doi: 10.1109/ACCESS.2018.2873502.

[7] R. A. Shahad, L. G. Bein, M. H. M. Saad, and A. Hussain, "Complex event detection in an intelligent surveillance system using CAISER platform," in *Proc. Int. Conf. Adv. Electr., Electron. Syst. Eng. (ICAEES)*, 2016, pp. 129–133, doi: 10.1109/ICAEES.2016.7888023.

[8] P. Rashidi, D. Cook, and L. Holder, "Discovering activities to recognize and track in a smart environment," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 4, pp. 527–539, Apr. 2011.

[9] T. V. Kasteren, A. Noulas, G. Englebienne, and B. Krose, "Accurate activity recognition in a home setting," in *Proc. Int. Conf. Ubiquitous Comput.*, Seoul, South Korea, 2008, pp. 1–9.

[10] A. Jarraya, A. Bouzeghoub, A. Borgi, and K. Arour, "DCR: A new distributed model for human activity recognition in smart homes," *Expert Syst. Appl.*, vol. 140, Feb. 2020, Art. no. 112849.

[11] N. Yala, B. Fergani, and A. Fleury, "Feature extraction for human activity recognition on streaming data," in *Proc. Int. Symp. Innov. Intell. Syst. Appl. (INISTA)*, Sep. 2015, pp. 1–6, doi: 10.1109/INISTA.2015.7276759.

[12] S. S. Akter and L. B. Holder, "Activity recognition using graphical features," in *Proc. 13th Int. Conf. Mach. Learn. Appl.*, Dec. 2014, pp. 165–170, doi: 10.1109/ICMLA.2014.31.

[13] L. G. Fahad, S. F. Tahir, and M. Rajarajan, "Activity recognition in smart homes using clustering based classification," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 1348–1353, doi: 10.1109/ICPR.2014.241.

[14] D. Arifoglu and A. Bouchachia, "Detection of abnormal behaviour for dementia sufferers using convolutional neural networks," *Artif. Intell. Med.*, vol. 94, pp. 88–95, Mar. 2019.

[15] M. R. Alam, M. B. I. Reaz, and M. A. M. Ali, "SPEED: An inhabitant activity prediction algorithm for smart homes," *IEEE Trans. Syst., Man, Cybern.-A, Syst. Hum.*, vol. 42, no. 4, pp. 985–990, Jul. 2012.

[16] K. A. Alaghbari, M. H. Md Saad, A. Hussain, R. A. Othman, and M. R. Alam, "A comparison of sequential prediction algorithms in IoT enabled smart environments," in *Proc. IEEE 12th Control Syst. Graduate Res. Colloq. (ICSGRC)*, Aug. 2021, pp. 238–243, doi: 10.1109/ICSGRC53186.2021.9515261.

[17] F. D. Casagrande, J. Torresen, and E. Zouganeli, "Predicting sensor events, activities, and time of occurrence using binary sensor data from homes with older adults," *IEEE Access*, vol. 7, pp. 111012–111029, 2019, doi: 10.1109/ACCESS.2019.2933994.

[18] M. R. Alam, M. Reaz, and M. Ali, "Statistical modeling of the resident's activity interval in smart homes," *J. Appl. Sci.*, vol. 11, pp. 3058–3061, Dec. 2011.

[19] E. Nazerfard, Z. Atashgahy, and A. Nadali, "Abnormal activity detection for the elderly people using ConvLSTM autoencoder," Res. Square, Tech. Rep., 2021, doi: 10.21203/rs.3.rs-693084/v1.

[20] A. Farayez, M. B. I. Reaz, and N. Arsad, "SPADE: Activity prediction in smart homes using prefix tree based context generation," *IEEE Access*, vol. 7, pp. 5492–5501, 2019, doi: 10.1109/ACCESS.2018.2888923.

[21] L. Fahad and S. Tahir, "Activity recognition and anomaly detection in smart homes," *Neurocomputing*, vol. 423, pp. 362–372, May 2021.

[22] D. Cook, "Learning setting-generalized activity models for smart spaces," *IEEE Intell. Syst.*, vol. 27, no. 1, pp. 32–38, 2012, doi: 10.1109/MIS.2010.112.

[23] R. A. Shahad and K. M. Pusat, "Suspicious loitering detection from annotated CCTV feed using CEP based approach," *J. Kejuruteraan*, vol. 30, no. 1, pp. 83–91, Jun. 2018.

[24] D. Cook, A. S. Crandall, B. L. Thomas, and N. C. Krishnan, "CASAS: A smart home in a box," *Computer*, vol. 46, no. 7, pp. 62–69, Jul. 2013.

[25] T. L. M. van Kasteren and G. Englebienne, "Human activity recognition from wireless sensor network data: Benchmark and software," in *Activity Recognition in Pervasive Intelligent Environments*, vol. 4. Paris, France: Atlantis Press, 2011, pp. 165–186.

[26] K. Gopalratnam and D. J. Cook, "Active LeZi: An incremental parsing algorithm for sequential prediction," *Int. J. Artif. Intell. Tools*, vol. 13, no. 4, pp. 917–930, 2004.

[27] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Trans. In. Theory*, vol. IT-24, no. 5, pp. 530–536, Sep. 1978.

[28] A. Ng, "Sparse autoencoder," *CS294A Lect. Notes*, vol. 72, pp. 1–19, Apr. 2011.

[29] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lect. IE*, vol. 2, pp. 1–18, Feb. 2015.

[30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[31] D. G. Altman and J. M. Bland, "Statistics notes: Diagnostic tests 1: Sensitivity and specificity," *BMJ*, vol. 308, no. 6943, p. 1552, Jun. 1994, doi: 10.1136/bmj.308.6943.1552.

[32] Y. Sasaki, "The truth of the F-measure," *Teach Tutor mater*, vol. 1, no. 5, pp. 1–5, 2007.

[33] B. W. Matthews, "Comparison of the predicted and observed secondary structure of t4 phage lysozyme," *Biochimica Biophys. Acta (BBA)-Protein Struct.*, vol. 405, no. 2, pp. 442–451, Oct. 1975.

KHALED A. ALAGHBARI received the B.Eng. degree (Hons.) in electronics engineering majoring in telecommunication, and the M.Eng.Sc. and Ph.D. degrees from Multimedia University (MMU), Melaka, Malaysia, in 2011, 2014, and 2020, respectively. He is currently a Postdoctoral Researcher at the Institute of IR 4.0, Universiti Kebangsaan Malaysia (UKM). His research interests include signal processing and machine learning.

MOHAMAD HANIF MD. SAAD is currently an Associate Professor and the Deputy Director of the Institute of IR4.0, Universiti Kebangsaan Malaysia (UKM). He is the Program Leader of INTEGRASI, UKM, IR4.0 Competence Center in System Integration. His research interests include research, development, and commercialization of applied intelligent systems, smart engineering, mechatronics, and complex event processing.

AINI HUSSAIN is currently a Professor and the Deputy Dean (Research and Innovation) of the Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia (UKM). Her research interests include intelligent systems, artificial intelligence, human–computer interaction, machine learning, pattern recognition, and image processing.

MUHAMMAD RAISUL ALAM received the Ph.D. degree in electrical and computer engineering from Carleton University, Canada. He is currently a Postdoctoral Fellow at the University of Toronto and the Toronto Rehabilitation Institute. He is also affiliated with the Vector Institute and AGE-WELL NCE. His research interests include the problems of planning and learning in smart homes. More specifically, he desires to explore the areas of machine learning, optimization, embedded systems, the Internet of Things and robotics to develop human-centered intelligent systems and assistive technologies for smart homes. He has been developing algorithms and systems to create context-aware, adaptive and personalized smart homes for the past 12 years. His research goal is to develop the next generation of smart homes that can provide healthcare support, increase energy efficiency, and improve quality of life.