Unobtrusive human activity classification based on combined time-range and time-frequency domain signatures using ultrawideband radar

Mohamad Mostafa | Somayeh Chamaani

Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran

Correspondence
Somayeh Chamaani, Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran.
Email: Chamaani@kntu.ac.ir

Funding information
Alexander von Humboldt-Stiftung

Abstract
In this proposed approach to unobtrusive human activity classification, a two-stage machine learning–based algorithm was applied to backscattered ultrawideband radar signals. First, a preprocessing step was applied for noise and clutter suppression. Then, feature extraction and a combination of time-frequency (TF) and time-range (TR) domains were used to extract the features of human activities. Then, feature analysis was performed to determine robust features relative to this kind of classification and reduce the dimensionality of the feature vector. Subsequently, different recognition algorithms were applied to group activities as fall or non-fall and categorise their types. Finally, a performance study was used to choose the higher accuracy algorithm. The ensemble bagged tree and fine K-nearest neighbour methods showed the best performance. The results show that the two-stage classification was more accurate than the one-stage. Finally, it was observed that the proposed approach using a combination of TR and TF domains with two-stage recognition outperformed reference approaches mentioned in the literature, with average accuracies of 95.8% for eight-activities classification and 96.9% in distinguishing between fall and non-fall activities with efficient computational complexity.

1 | INTRODUCTION

Injuries and fatalities from human falls are a major public health problem. Falling is a main cause of unintentional injury deaths, especially among adults over age 65 years [1, 2]. According to the World Health Organization, ‘An estimated 646,000 fatal falls occur each year, making it the second leading cause of unintentional injury death, after road traffic injuries’ [1]. Furthermore, the fall-related mortality rate has increased dramatically owing to the ageing population [3]. However, it is reported that the fast detection of human falls can allow quick medical intervention, reduce risk possibility, and increase the probability of lives being saved after falls occur [4, 5]. Therefore, it is important to prevent falls or facilitate their early identification by developing automatic fall detection technology [6]. When detecting falls, other human activities must be considered to distinguish falls from non-fall activities.

Fall detection sensors or technologies are categorised into two main groups: wearable and non-wearable devices. First, there are different types of wearable sensors such as mobile phones [7, 8], smart watches [9], accelerometers and gyroscopes [6, 10, 11], and push-button alerts [12]. The major advantage of wearable devices is their low cost. However, they have two main shortcomings. First, some of these devices require the subject to be conscious after a fall, to push a button, for example. Second, subjects must remember to wear the sensor all of the time. Thus, non-wearable sensors are preferred, unobtrusive, and more acceptable. Some examples of non-wearable devices or ambient solutions [13] are smart floors [14], visual sensing or camera-based systems [15–17], acoustic equipment and microphones [18–20], pyroelectric infrared [21, 22], and radar-based fall detection and human activity classification (HAC) systems [7, 23–43]. However, visual and camera-based devices are affected by low lighting or enclosed conditions, which limit their results. Also, there are...
restrictions for acoustic-based techniques such as the requirement of noise-free acoustic environments and low multipath reflections. Both methods have the problem of installation in bathrooms owing to privacy issues for cameras and high multipath reflections for acoustic equipment. In this regard, among non-wearable techniques, ultrawideband (UWB) radar can be a good and effective technology for remote monitoring of human activities because of its inherent and unobtrusive features such as insensitivity to lighting and weather conditions, high-range resolution for positioning, penetration ability through obstacles in behind-wall, indoor and outdoor scenarios, good immunity against multipath interference, low power consumption, and capability of multiple target discrimination [44–46]. Therefore, UWB sensors can be used in health monitoring and human care systems, especially to enhance quality of care for elderly people and patients.

Human fall detection and HAC using UWB sensors is founded on the detection of motion caused by the activity itself; therefore, a specific signature related to each human activity is created. The transmitted radar signal is affected by these motions, which produce changes in frequency, phase, amplitude, and differences in time-of-arrival of backscattered signals from the target [46]. Therefore, a time-frequency (TF) and time-range (TR) domain analysis is needed for this approach to reveal human activity motion signatures as velocity, the Doppler effect of different human body parts in motion, and human position versus observation time. Briefly, all of these revealed signatures are considered basic and fundamental in HAC operation [24]. For further clarification, Figure 1 shows the general approach of HAC using UWB radar used in most of the literature. It is mainly divided into three steps:

1. Preprocessing to clean noisy and cluttered collected data with different methods such as bandpass filters, DC removal, and moving average filters [44–47];
2. Feature extraction using a predefined [48] or automated feature extraction method [49]; and
3. Classification of activities using machine learning (ML) and deep learning (DL) algorithms, which can be categorised as supervised algorithms [50] and unsupervised algorithms [51].

Numerous radar-based methods have been developed to address the HAC problem. TF representation is widely used for HAC. Kim et al. [23] used a support vector machine (SVM) to recognise seven human activities using six micro-Doppler signatures extracted from the TF representation (spectrogram). Kim et al. [24] applied a deep convolutional neural network (DCNN) algorithm instead of an SVM to classify the same activities. In Kim et al. [24] there is no need for manual feature extraction. Automatic feature extraction and recognition processes are done simultaneously using DL algorithms. Jokanovic et al. [26] improved the accuracy of micro-Doppler signature estimation by implementing a multindow TF analysis for radar returns using SVM for classification. Jokanovic et al. [27] evaluated a combination of TF and TR domains and demonstrated the superiority of using this combination over the use of a single domain in human motion recognition. Jokanovic et al. [28] again used this combination and proposed a deep learner approach consisting of two stacked autoencoders and a logistic regression classifier to improve activity recognition accuracy. In these publications, each domain (TR and TF) was used separately at each recognition stage. Sadreazami et al. proposed three different DL methods for fall detection [36–38]. They used radar time series data obtained by the sum of the range profile of UWB radar returns as input for a deep residual network for automatic feature learning [36]. In work by the same authors [37], the features were learned automatically using DCNN from the radar time series data. The authors [38] also proposed a capsule network-based fall detection (CapsFall) method in which the TR representation of the radar time series was fed into a CapsFall method for automatic feature learning. Du et al. [39] introduced a three-dimensional (3D) DL framework for human motion analysis in which reflected radar echoes were transformed into range-Doppler points to obtain the discrete representation of motion trajectory. Du et al. [40] proposed a segmented convolutional gated recurrent neural network (SGGRNN) method to classify different human activities determined from on micro-Doppler spectrograms. Ding et al. [41] addressed the HAC problem using a multilayer classification method. First, signatures in the TR domain were used to categorise the 12 human activities into in situ motions and non-in situ motions. After that, spectrograms were obtained using the weighted range-time-frequency transform (WRTFT) method. Then, the physical empirical and principal component analysis (PCA) features were extracted and used to classify the motions into in situ and non-in situ motions, respectively [41]. Li et al. [42] developed a hierarchical recognition technique in which each activity among the six motions was classified based on a one-versus-all binary classification. A combination of different domains was used, the extracted features were investigated, and the optimum features were selected for each stage. Qi et al. [43] proposed a multilayer approach by combining the conventional K-nearest neighbour (KNN) supervised ML technique and a convolutional neural network (CNN) to distinguish 12 kinds of human daily motion. First, the radial features with the KNN classifier were used to divide the motions into two categories: those with no radial displacement and those with radial displacement. Then, a CNN classifier used the feature spectrograms to categorise the two groups of activities into 12 different motions [43].

This work proposes an approach using a UWB radar (Xethru X4M03) [52] for fall detections and HAC using ML algorithms. First, data were collected from three participants performing fall and non-fall activities, as shown in Table 1.
parameter sensitivity was also investigated. Potential algorithms with less computational complexity.

- The two-stage recognition scheme for human activities consists of three separate models for classification founded on multiple-domain representation. The combination of binary classification in the first stage and two multiclass classifiers in the second stage was proposed using conventional ML techniques. With the selection of the optimum and robust feature set and classifier type for each stage, a competitive performance was achieved, compared with DL algorithms with less computational complexity.

- The most important novelty of our work is a good and versatile combination of multimodal features. These effective features help us to achieve good accuracy even with conventional ML methods instead of DL methods. An efficient two-stage classification with an optimal combination of multimodal features was used to improve performance compared with one-versus-all hierarchical schemes with less redundancy of data.

The remainder of the work is organised as follows. In Section 2, signal models for HAC and the data collection setup are described. The complete proposed ML-based algorithms applied to classify fall and non-fall human activities are explained in detail in Section 3. Results and discussion are presented in Sections 4 and 5, respectively. Concluding remarks are given in Section 6.

2 | MATERIALS AND METHODS

2.1 | UWB radar and data collection setup

The X4M03 UWB radar, shown in Figure 2a, was used for data collection in all samples for HAC. The X4M03 operates at 6–8.5 GHz, offering high spatial resolution. In addition, the sampling rate in propagation time (fast time) was equal to 23.328 GHz. Each sampling point is referred to as a range bin in which the length of each scan is 1536 range bins. The radar covers a distance range of 0.2–9.9 m with 5.35-cm range resolution. The duration of each measurement scenario was approximately 8~10 s (observation time or slow time), in which the frames per second is 400. The radar was placed 1 m above the floor level (Figure 2b), and 300 data samples were collected in different scenarios at different ranges (R0: 1~5 m), as presented in Table 2. Three healthy subjects (age 20–30 years, two males and one female) participated in the data collection process, performing numerous predefined fall and non-fall activities, as illustrated in Figure 3. The number of fall and non-fall activities and their categorisation in training and test data sets are given in Table 2. Schematics of different activities are shown in Figure 3.

2.2 | Signal modelling

To learn easily and become familiar with the modelling equation of the signal, it is essential to define two independent time scales of the signal: first, where wave propagation phenomena
into the environment towards the target (human body). If the condition of the measurement environment is considered an ideal case (that is, no loss, no noise, no antenna mismatch, and no angular dependency), the propagated signal hits the human body and the transmitted signal will be partially reflected towards the receiver antenna, obtaining a backscattered signal, \( s_{rx}(t) \), which can be written as [45]:

\[
s_{rx}(t) = \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \sigma_k(t)^* s_{tx}(t - nT_0),
\]

where \( n \in N, n = \{0, \ldots, N\} \) is the index of the transmitted signal, \( s_{tx}(t) \), \( K \) is the total number of detected targets, \( t \) is the fast time, and \(^*\) means convolution. \( \sigma_k(t) \) is the time-dependent propagation path of the \( k \)th reflection, which is described as:

\[
\sigma_k(t) = \gamma_{tx}(t)^* \gamma_k(t)^* \gamma_{rx}(t)
\]

\( \gamma_{tx}(t) \) and \( \gamma_k(t) \) are the impulse response functions (IRF) of the transmitting and receiving antenna, respectively, and \( \gamma_{rx}(t) \) is the IRF of the \( k \)th reflection.

For simplicity, let us first suppose that a simple target at \( k = i \) is a static or motionless target. The effect of \( \sigma_k(t) \) over the transmitted signal for static targets can be expressed as:

\[
s_{rx}(t) = a_i \sum_{n=0}^{N-1} s_{tx}(t - \tau_i - nT_0),
\]

where \( a \) is the complex amplitude, \( \tau_i \) is the round-trip time delay, \( c \) is the speed of light and \( R_i \) is the range of target \( i \).

When the target is not static and the main purpose is to determine the motion of target, we can rewrite Equation (7) as:

\[
s_{rx}(t) = a_i \sum_{n=0}^{N-1} s_{tx}(t - \tau_i(T) - nT_0),
\]

where \( \tau_i(T) \) is the round-trip time delay of the target in motion.

In radar, the variation of delay occurs only in the observation time; that is, no variation in delay can be observed in fast time during one pulse duration \( T_p \). Therefore, \( \tau_i \), the time-variant round-trip delay, can be written as:

\[
\tau_i(T) = \tau_0 - \frac{2\nu T}{c}
\]

where \( \tau_0 = \frac{2R_0}{c} \), \( R_0 \) is the initial range, and \( \nu \) is the radial velocity.

---

**FIGURE 2** (a) Xethru X4M03 ultrawideband radar; (b) Illustration of experimental setup and measurement schematic.
Following this, the equation of the received signal (including both stationary and non-stationary effects) can be represented as:

\[ s_{rx}(t) = a_s \sum_{n=0}^{N-1} s_{tx}(t - \tau_i - nT_0) + a_r \sum_{n=0}^{N-1} s_{tx}(t - \tau_i(T) - nT_0), \]

where the first and second \( \sum \) relate to stationary and non-stationary targets, respectively.

To study the Doppler signatures resulting from the human body in different motions, the equations of \( \tau_i \) should be rewritten with respect to the Doppler frequency. The relation between the Doppler frequency \( f_D \) and radial velocity \( v \) can be modelled as [53]:

\[ f_D = \frac{2v}{\lambda_0} = -\frac{2vf_0}{c} \]

where \( \lambda_0 \) is the transmitted wavelength and \( f_0 \) is the transmitted frequency. Then, the time-dependent round-trip delay is:

\[ \tau_i(T) = \tau_0 - \frac{\lambda_0}{2} f_D T \]

where \( f_D \) can take the value of \( f_{torso}, f_{limbs}, f_{head}, f_r \), and \( f_b \), which are the Doppler frequencies resulting from torso, limb, head, respiration, and heartbeat motions, respectively.

These Doppler signatures are caused by the human activity and vital signs, and are used as a feature in the next sections. Therefore, Equation (7) can be rewritten as:

\[ s_{rx}(t) = a_s \left\{ \sum_{n=0}^{N-1} s_{tx}(t - \tau_i - nT_0) \right\} + a_r \left\{ \sum_{n=0}^{N-1} s_{tx}(t + \frac{\lambda_0}{2} f_D T - \tau_0 - nT_0) \right\}, \]

In addition, an important parameter for UWB radar for HAC is the range resolution, which can be defined as [54]:

\[ \text{Range resolution} = \frac{c}{2BW} = \omega \frac{c}{2} \]

where \( BW \) is bandwidth and \( \omega \) is pulse width.

Finally, because there are no ideal conditions in real measurement scenarios, additional propagation phenomena should be considered when writing the model equation of the received signal. Therefore, the reflected signal from the target is buried by noise \( n(t,T) \) and clutter \( c(t,T) \) (both stationary and non-stationary) caused by many environment reflections (e.g., walls, furniture, moving objects), and antenna coupling. The received signal will be denoted as:

\[ r(t, T) = s_{rx}(t, T) + c(t, T) + n(t, T), \]

where \( c(t,T) \) and \( n(t,T) \) are clutter and noise, respectively. They should be removed from the received signal in the pre-processing stage. For further information about UWB radar...
signal modelling for human motion detection and classification, see Mostafa et al. [45], Sachs and Herrmann [55] and He et al. [56].

3 | PROPOSED METHODOLOGY

In this section, different steps of the proposed algorithm for HAC are described. First, an overall algorithm block diagram is presented in Figure 4. The algorithm is divided into three levels. The first level is a preprocessing stage that cleans the collected radar data by applying different signal processing techniques to remove noise and stationery and non-stationary clusters and prepare data for the next stage. The second level begins by representing the cleaned data in different domains (TR and TF) and extracting all predefined features. After that, several features are selected and passed to the first stage of classification depending on the supervised ML to distinguish between fall and non-fall activities. In the last level, the output results of the first stage are classified separately using two classifiers: one for fall activities and the other for non-fall activities.

3.1 | Preprocessing

The main purpose of this stage is to clean and prepare the collected data for classification.

3.1.1 | Data collection

For HAC using UWB radar, the backscattered signals at different time of arrivals, or different ranges from the radar, can be successively gathered. Then, the reflected echoes \( r(t, T) \) are sampled and recorded in a 2D data matrix \( R \in \mathbb{R}^{m \times n} \) as:

\[
R = \{ R^{(m)}[n] : m = 1, \ldots, M; n = 1, \ldots, N \}, \quad (15)
\]

where \( m \) and \( n \) represent the propagation and observation time index, respectively.

The output of the sensor system is a data matrix \( R \) (radargram). The range profiles are stored in \( M \) range-bins, which is the number of columns of matrix \( R \). Each scan, representing the reflected signal, has a data vector of length \( M \). The radar scans are stored in \( N \) scans every 0.0025 s (400 scans per second), which represents the rows of matrix \( R \).

3.1.2 | DC removal and stationary clutter suppression

The main purpose of this section is to remove noise, DC, and stationary clutter. This is done by subtracting the mean of the collected data matrix in slow time. After that, the background subtraction method is used to remove stationary clutter from backscattered signal \( R \). This method is applied by subtracting the estimated background of each range bin over the observation time, that is \( z(m) = R(:, m) \) from the measured signal:

\[
x(m) = z(m) - p(m - 1), \quad (16)
\]

Estimated background \( p(m - 1) \) is calculated using the moving average method; the estimation technique uses the average of previous \( L \) samples to estimate the background of the reflected signal:

\[
p(m - 1) = \frac{1}{L} \sum_{j=m-L}^{m-1} r(j), \quad (17)
\]

The simplest case is a moving target indicator that considers \( L = 1 \), where simply, the background is assumed to be equal to the previous sample; \( L = 30 \).

3.1.3 | Non-stationary clutter suppression

In addition, non-stationary clutter should be suppressed. For this purpose, the resulting output of stationary clutter reduction is matrix \( X \), where \( X \in \mathbb{R}^{N \times M} \). SVD calculates the lag-covariance matrix \( C = XX^T \) and its singular decomposition. If \( N \geq M \), the SVD of \( X \) can be written as (18):

\[
X = U \Sigma V^T, \quad (18)
\]

where \( \Sigma = \begin{bmatrix} \Lambda & 0 \\ 0 & M \end{bmatrix} \), \( \Sigma \in \mathbb{R}^{N \times M} \). For \( N \leq M \), \( \Sigma = [\Lambda \ 0] \) is the diagonal matrix of eigenvalues, with \( \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_R) \); \( \sigma_i \) is the singular values sorted in descending order; \( U \in \mathbb{R}^{N \times M} \), \( U = (u_1, u_2, \ldots, u_N) \) and \( V \in \mathbb{R}^{N \times M} \), \( V = (v_1, v_2, \ldots, v_N) \) are the matrix of empirical orthogonal functions and the matrix of principal components, respectively, where \( U^T U = I, V^T V = I \).

The data matrix may be thought of as the summation of \( M \) individual singular-value matrices, each given by Equation (19):

\[
D_j = U \Sigma_j V^T, \quad (19)
\]

where \( D \in \mathbb{R}^{N \times M} \), \( j \in \{1, \ldots, M\} \), \( \Sigma_j \) is a matrix of the same size as \( \Sigma \) in which all elements are zero except \( \Sigma(j,j) \).

Data matrix \( X \) can be rewritten as Equation (20):

\[
X = X_{\text{cutter}} + X_{\text{target}} + X_{\text{noise}} = \sum_{a=1}^{A} U \Sigma_a V^T + \sum_{b=1}^{B} U \Sigma_b V^T + \sum_{c=1}^{C} U \Sigma_c V^T \quad (20)
\]
where clutter, target, and noise are represented by the set of eigentriples A, B, and C, respectively. SVD is used to detect the target signals and remove the contribution of $X_{clutter}$ and $X_{noise}$ that is, clutter (stationary and non-stationary) and noise, respectively. Clutter mainly results from environmental reflections occurring where measurements are performed, such as from walls, static objects, non-static objects, and furniture.

An SVD operation is applied iteratively and the data matrix is reconstructed after removing the singular values that represent the clutter. Our proposed method shows that second, third, fourth, and fifth singular values contain target information. When all other singular values are removed, the reconstructed data matrix is matrix $Y = y_{ij} \in \mathbb{R}^{N \times M}$. For further details, please refer to Qi et al. [43] and Mostafa et al. [44].

### 3.1.4 Summation and normalisation

The resulting output matrix of non-stationary clutter suppression is matrix $y_{ij} \in \mathbb{R}^{N \times M}$, where $N$ is the number of rows corresponding to observations recorded at different time intervals (slow time, across scans) and $M$ is the number of columns representing the spatial samples from different ranges (fast time, only one scan). To prepare the input time series for the proposed HAC ML algorithms and calculate the spectrogram, the average of each impulse response (over fast time) is calculated as:

$$Y_i = \sum_{j=1}^{M} \frac{y_{ij}}{\max_j |y_{ij}|}$$  \hspace{1cm} (21)

Using the X4M03 radar, we measure the reflected signals in different observation times and create a typical radargram (Figure 5). The vertical axis is the fast time and the horizontal axis is the slow time. Figure 5 shows the recorded radar data at different steps of the preprocessing stage of fall-forward human activity. The noise and clutter reduction can be clearly observed after applying denoising and clutter suppression signal processing techniques to the measured data shown in Figure 5a. The data are then prepared for the feature extraction step.

### 3.2 Feature extraction

After the preprocessing stage, feature extraction is the most critical step in ML-based classification. Received signal contains information about subject activity such as range and Doppler frequency. This information can be extracted from measured data represented in two different domains of TR and TF. In addition to statistical features in each domain, several useful features are extracted from the combination of the domains. The total number of extracted features is 56, but not all features are selected for the classification steps.
Only the minimum number of features, providing the more accurate classification, are chosen. Hence, the features are evaluated for each classification stage. Then, the selected features are used in the classification for the remaining data and tested data. Finally, the model is exported and the resulted features in Section 3.3 are used for classification. The overall scheme of feature extraction and evaluation is illustrated in Figure 6.

3.2.1 Features of TR domain

Figure 7 shows the TR representation of preprocessed data for all eight fall and non-fall activities. Here, the discussed features are those extracted from data represented in the TR domain. First, two main features can be calculated in this TR domain, which are range difference resulting from motion activities and the event width, as shown in Figure 7a. These two features provide important information about activities, because the duration of the event and the change in range differ from one activity to another, especially between fall and non-fall activities, as shown in Figure 7a–h. The range variation of fall activities is commonly greater than that of non-fall motions. Furthermore, the duration of fall activities is more than the time needed to sit, stand, or bend.

After that, a set of empirical and statistical features that characterised the distribution of the radar signal, are extracted: power of received signal, entropy (relative degree of randomness), mean, median, variance, skewness (measure the asymmetry of the data), kurtosis (measure of the degree of distribution peakedness), mobility, form factor, mean frequency, median frequency, root mean square, zero-crossing rate (the number of times the signal crosses the zero reference level in a specific time interval), and turns count (the number of zero crossings of the derivative of the signal).

The last type of features defined in this section is founded on a PCA that was applied to the data matrix in building the radargram shown in Figure 7. Because noise and clutter were removed in the preprocessing stage, the resulting data matrix mainly contains human motion information. Therefore, the most dominant eigenvalues are relevant to the human activities. The first principal component (PC) contains spatial information about the subject activity (greatest source of motion) in the room. The human body is positioned or distributed among the range profiles of the radar echoes for each posture. Because the energy contribution resulting from human motion may go beyond the first PC, other PCs have been considered. In the PCA:
The first eigenvalue was processed alone. This eigenvalue and its eigenvector were used to reconstruct the matrix that contained the spatial distribution of activity or motion and the mean, median, skewness, kurtosis were computed and used as features.

• Eigenvalues 2–9 were chosen from the PCA in descending order because those eight dominant eigenvalues correspond to human motion.

In other words, large-scale movements such as human position changing lie in the first eigenvalue. Small scale movements and posture variation lie in eigenvalues 2–9.

3.2.2 | Features of TF domain

In addition to the TR domain, the TF domain is important and traditional for the representation of radar data collected from human subjects. The short-time Fourier transform (STFT) is the most applied technique to construct the spectrogram. It is typically used for TF representation. STFT performs a Fourier transform on the windowed signal [57]:

$$STFT (n, f_s) = \sum_{k=1}^{L} Y_s(k)w(n-k)e^{-j2\pi f_s k/L}$$  \hspace{1cm} (22)

where $w(k)$ denotes the window signal, $f_s$ is the sampling frequency on observation time and $L$ is the window length.

The spectrogram is the modulus squared of the STFT, as shown in Equation (23):

$$SPEC (n, f_s) = \left|STFT (n, f_s)\right|^2$$  \hspace{1cm} (23)

The spectrogram of the eight data for fall and non-fall activities are shown in Figure 8a–h. In addition, an important parameter, the energy burst curve (EBC), is calculated in Equation (24). The EBC with a lower-bound frequency $f_{lb} = 0$ Hz and upper-bound frequency $f_{ub} = 200$ Hz of eight different activities is presented in Figure 9:

$$EB (n) = \sum_{f_s = f_{lb}}^{f_{ub}} \left|SPEC (n, f_s)\right|$$  \hspace{1cm} (24)

Statistical features extracted from the EBC curves of Figure 9 are maximum, standard deviation, kurtosis, median and variance, and the maximum of EBC with $f_{lb} = 25$ Hz and $f_{ub} = 200$ Hz.

As mentioned, the Doppler and micro-Doppler frequency signatures of fall and non-fall activities, resulting from the motion of torso, limbs, head, arms, and the whole human body, are modelled in Equation (10) and can be extracted from the TF representation of data. To obtain a more reliable classification, the power of eight windows of the spectrogram is computed as part of an input feature. The frequency ranges of interest of these eight windows are $f_{lb} = \{0.2, 3, 10, 25, 45, 80, 120, 150\}$ Hz and $f_{ub} = \{3, 10, 25, 45, 80, 120, 150, 200\}$ Hz, respectively to cover all Doppler frequencies resulting from human activities and the motion of human body parts during the activities. For example, frequency ranges are \{[0.2 3], [3 10], ..., [150 200]\}. The intervals are selected according to information provided by the spectrogram about all activities in Figure 8. There is no need to detect the Doppler frequency of each activity precisely. As shown, fall activities contain higher Doppler frequency components that can reach, and in some cases exceed, 150 Hz (Figure 8a and 8b), where for non-fall activities, it is not greater than 80 Hz in all cases (Figure 8e–h). Besides, the Doppler signatures of walking are more distinguishable than other activities. In other words, the Doppler frequency components of walking are not restricted at only one time, but are distributed throughout the total time of measurement (Figure 8b). The power of these eight frequency intervals is

![Block diagram of both feature extraction and feature evaluation stages](image)
also considered a feature used for HAC, as shown in Figure 10. Furthermore, features are computed from the spectrogram, such as spectrogram entropy (statistical measure of randomness or uncertainty), the total power of the spectrogram, the power of frequency range of heartbeat and respiration, and the normalized rate with respect to the total power.

### 3.2.3 Feature evaluation

Feature evaluation is a step before the algorithm is generalized and exported for use in the tested data. After the feature extraction from TR and TF domains, not all features are considered. Furthermore, every classification step has its own features, but some are commonly used in the two stages. As shown, in Figure 6, the features extracted and used are those resulting in this section for each classifier. Figure 11a is a scatterplot showing the distribution of two activities, fall and non-fall, with respect to the values of two features or predictors (as an example: total power of spectrogram (EX), and spectrogram entropy). The scatterplot helps in understanding which features are more important for classification: that is, which provide more information about activities and separate the activities under classification. Consequently, it helps to choose or remove the investigated features. First, all features are selected; then, we remove features that do not help in classification, drawing on observation of the scatterplots (changing the pairs to get information about all the features) until we obtain the optimum result, which is best accuracy (accuracy and other metrics are defined in Section 4) with a minimum number of features. This method for feature selection is called sequential backward selection. Figure 11b–d shows the accuracy of the first, second, and last classifiers, respectively, with
Figure 11b shows that the first classifier can achieve the optimum accuracy with the selection of 27 of 56 features with 96.9% maximum accuracy. Furthermore, with a selection of only six features (power of the spectrogram, spectrogram entropy, range difference, and the power of fourth, fifth, and sixth spectrogram windows), 92% accuracy is achieved. For the second classifier (fall activities classifier), the best accuracy of 94.1% is achieved with a selection of 35 of 56 features. Here, the number of needed features is much more than that for the first classifier, because the classes are close to each other (the activities are more similar). As seen in Figure 11c, 18 features are needed to offer approximately 83% accuracy. The most robust features for the second classifier are range difference, entropy, event width, the power of fifth, sixth, seventh, and eighth spectrogram windows, eigenvalues 2 and 3, and the maximum and standard deviation of EBC. Regarding the last classifier (non-fall activities classifier), the best recorded result is 95.6% with 22 selected features; an accuracy of 85.6% with only six features can be obtained in which the features are entropy of spectrogram, entropy, the total power of spectrogram, and the power of fourth, fifth, and sixth spectrogram windows.

### 3.3 Classification

In this section, after performing feature extraction and selection, a resultant vector of selected features for each data sample is the input of the classifier. However, the classification stage is divided into two main steps. The first step consists of only one classifier, but before choosing the classifier type, a performance evaluation for different classifiers such as SVM and decision tree is performed. The same analysis is done for classifiers for the second
step. The purpose of the first classifier is to distinguish and categorise the fall activities separately from non-fall activities. In the next step, we have two distinct classifiers that can be used simultaneously. The two output groups of the first step classifier, fall activities and non-fall activities, are classified in simultaneous, independent, parallel ways to specify whether the motion type was fall forward, fall backward, fall left or fall right (fall group), or sit, stand, bend, or walk (non-fall group). The different types of classifiers are evaluated to choose the best classification method for each classifier:

- decision tree: fine tree, medium tree, and coarse tree
- discriminant: linear discriminant and quadratic discriminant
- logistic regression
- SVM: linear SVM, quadratic SVM, cubic SVM, fine Gaussian SVM, medium Gaussian SVM, and coarse Gaussian SVM.
- KNN: fine KNN, medium KNN, coarse KNN, cosine KNN, cubic KNN, and weighted KNN.

![Energy burst curve from 25 to 200 Hz for different human activities](image.png)

**Figure 9** Energy burst curve from 25 to 200 Hz for different human activities: (a) Fall forward; (b) Fall backward; (c) Fall left; (d) Fall right; (e) Sit; (f) Stand; (g) Bend; (h) Walk

| Activity | Frequency (Hz) |
|----------|----------------|
| Sit      | 25             |
| Stand    | 10             |
| Bend     | 5              |
| Walk     | 1              |

All experiments of HAC classification used the X4M03 UWB radar with a sampling frequency in observation time of 400 Hz to cover 0–200 Hz Doppler frequency components. The experimental setup of two different scenarios for HAC is presented in Figure 12. The next two subsections show the result of the training and test data, respectively. In what follows, the metrics used to rate the performance of classifier are:

\[
\text{Accuracy (AC)} = \frac{TP + TN}{TP + FP + TN + FN}
\]

\[
\text{Precision (PR)} = \frac{TP}{TP + FP}
\]

\[
\text{Sensitivity (SE)} = \frac{TP}{TP + FN}
\]

where TP and TN denote true positives and true negatives, respectively, and FP is the number of false positives and FN denotes the false negatives. The results of the preprocessing stage and feature evaluation were discussed in previous sections.

### 4 RESULTS

#### 4.1 Results of training data using fivefold cross-validation

The evaluation of HAC algorithms consisted of two main parts: feature evaluation (Section 3.2.3) and classifier
performance analysis. In this section, training data were used to evaluate different types of classifier. Only methods with the highest accuracy for each family method are shown in the results tables. Other evaluated methods with lower accuracy within the same family method are not presented. As an example, suppose the method family is 'tree'. The highest accuracy for this method family was achieved using a fine tree method. Therefore, only results for the fine tree method are shown in Tables 3 and 4. The same was applied for all family methods with the highest accuracy overall methods highlighted in green. Table 3 shows that the best classifier of the first step of the classification stage (to distinguish between fall and non-fall activities) was the ensemble bagged tree classifier with an averaged accuracy (AC) = 96.9%, precision (PR) = 96.4% and SE = 98.5% in a fivefold cross-validation with a training time of 3.188 s and prediction speed of approximately 800 observation/s. Hence, this method outperformed the accuracy of other methods at least by 5.3%. The results shown in Table 3 correspond to one run, and the averaged result over 10 runs was AC = 96.7%. The second step of classification consisted of two classifiers also in a fivefold cross-validation. The first was a fall activity classifier with four classes (fall forward, fall backward, fall left, and fall right) and the second was non-fall activities classifier with four classes (sit, stand, bend and walk). Table 4 shows the performance metrics of the different classification methods for the fall and non-fall activities classifiers. The fine KNN (k = 1) with AC = 94.1% (with a training time of 0.7545 s and prediction speed of approximately 2000 observation/s) and the ensemble bagged tree classifier with AC = 95.6% (with a training time of 2.7104 s and prediction speed of approximately 320 observation/s) were the optimum classification methods for fall and non-fall classifiers, respectively, outperforming the other methods by at least 4% accuracy. The required CPU time on an Intel core i5 2.2 GHz with 8 GB RAM was approximately 7 s. The confusion matrix of the three classifiers is shown in Tables 5–7. After evaluating the performance of classifiers, the two-stage classification approach was evaluated. To this end, its results were compared with the one-stage classification (Table 8) that directly classifies the activity into eight classes. The proposed two-stage approach with an AC of 95.8% outperformed the one-stage classifier by approximately 9%. The accuracy of the classifier

**FIGURE 11** (a) Distribution of fall and non-fall activities with respect to the total power of spectrogram and spectrogram entropy features. The accuracy of classification with respect to the number of selected features for: (b) Fall non-fall classifier; (c) Fall activities classifier; (d) Non-fall activities classifier.
of Table 8 is 86.7% using 44 selected features. A better result can be achieved using two-stage classification with fewer predictors or features. The green and red highlights in the confusion matrix mean true and false prediction, respectively.

4.2 | Results of validation data

The main goal is to export a model that can be used to categorise life activities of humans (using incoming or tested data), especially with regard to fall and non-fall events. After performance evaluation using training data with a fivefold cross-validation of the proposed approach of the HAC algorithm, the optimum features were selected for each classification step and the best classification methods were chosen for each classifier. The model can be exported using the performance analysis mentioned earlier, as seen in Figure 13. The results of text data classification using the exported model are listed in Tables 9–12. Table 9 presents the accuracy of the three classifiers used in the proposed approach with and without validation. The results of the test data approximately match the training data with some decrease in accuracy percentage because the test data were a small data set and errors or false predictions were accentuated in the results. In addition, the results of the fivefold cross-validation classifiers outperform that of the no-validation classifiers by approximately 6%. The

TABLE 5 Confusion matrix of ensemble bagged tree method first stage (using fivefold cross-validation)

| Predicted class | Fall | Non-fall |
|-----------------|------|----------|
| True class      |      |          |
| Fall            | 133  | 2        |
| Non-fall        | 6    | 84       |

TABLE 3 Results for each method by three metrics for first-stage classification using fivefold cross-validation

| Method family | Method                  | Averaged accuracy of fall activity classifier | Averaged accuracy of non-fall activity classifier |
|---------------|-------------------------|----------------------------------------------|--------------------------------------------------|
| Tree          | Fine tree               | 65.2                                         | 68.9                                             |
| Discriminant  | Linear discriminant     | 80.0                                         | 87.8                                             |
| Support vector machine (SVM) | Quadratic SVM       | 86.7                                         | 85.6                                             |
| K-nearest neighbour (KNN)   | Fine KNN               | 94.1                                         | 87.8                                             |
| Ensemble      | Bagged tree             | 82.2                                         | 95.6                                             |

TABLE 4 Results of each method for second-stage classification using fivefold cross-validation

| Method family | Method                  | Averaged accuracy | Averaged accuracy of non-fall activity classifier |
|---------------|-------------------------|-------------------|-----------------------------------------------|
| Tree          | Fine tree               | 65.2              | 68.9                                          |
| Discriminant  | Linear discriminant     | 80.0              | 87.8                                          |
| Support vector machine (SVM) | Quadratic SVM       | 86.7              | 85.6                                          |
| K-nearest neighbour (KNN)   | Fine KNN               | 94.1              | 87.8                                          |
| Ensemble      | Bagged tree             | 82.2              | 95.6                                          |
By 3%, compared with Kim and Ling [23], who used only TF data representation; but with the combination of DCNN for automatic feature learning and classification of seven activities, the obtained AC of 90.3% is less than our proposed method by approximately 5%. The training time for each fold with 400 epochs was about 1420 s on average, which is much longer than the training time of the proposed method (less than 7 s applying fivefold cross-validation), in which MATLAB is used for both to train the data. In Kim and Moon [24], the number of samples was 336; three spectrograms were extracted from each measurement, giving 1008 samples in total. The reference method used the whole spectrogram of 300 × 140 as input for DCNN classification, whereas in the proposed method the features from the spectrogram and radargram were used as input for a classifier, which reduced computational complexity.

By approximately 4%, compared with the method used in Jokanovic and Amin [27], in which data are first classified according to the range extent (two classes) and then by TF representation (four classes) and the accuracy was 91.25% (our approach has eight classes or activities). The computational time of the method used in Jokanovic and Amin [27] is 1.4 s.

By 4%, compared with the method used in Sadreazami et al. [38], in which only the TF domain (spectrogram) was used as input for a capsule module network (automatic feature learning) to differentiate fall and non-fall activities. The total number of samples for Jokanovic and Amin [28] is 208, whereas in the current work, 300 samples exist. This made the comparison more equitable regarding the dataset size and using the same performance metrics (accuracy and confusion matrix). There is no information about the computational complexity in Jokanovic and Amin [28].

By 4%, compared with the method used in Sadreazami et al. [36]. Sadreazami et al. used the residual network for automatic feature extraction and softmax for classification. This method was applied to 336 data samples (225 training data and 101 testing data), giving AC = 93.07% with no information about computational complexity.

By 4% (AC = 94.1%) with no cross-validation and by 4% with fivefold cross-validation; those authors used DCNN for automatic feature extraction and softmax classification. In terms of computational complexity, it required 30 s per training epoch.

By 2.5% (AC = 91.1%) with no cross-validation and by 6% with fivefold cross-validation; compared with the results obtained in CapsFall proposed approach in Sadreazami et al. [38], in which only the TF domain (spectrogram) was used as input for a capsule module network (automatic feature learning) to differentiate fall and non-fall activities. The comparison is done only between the CapsFall and the first-stage classification of the proposed method to compare performance regarding the classification of two classes. Sadreazami et al. [38] used the same radar Xethru X4M03 kit as here with 3360 samples.

By more than 9% compared with Du et al. [39], in which a 3D DL framework was applied to the discrete representation of motion trajectory for human motion analysis. The use of 3D representation with DL increased the complexity of the algorithm without providing high performance (87.16 average accuracy

5 | DISCUSSION

In this section, the results of the proposed algorithm are compared with results found in the state of the art. Table 13 shows the comparison of the proposed approach with the reference methods mentioned in the introduction. The proposed approach outperforms the approaches used in the reference papers in term of accuracy:

- By 3%, compared with the method used in Jokanovic and Amin [28], the same performance (AC approximately 97%) was obtained. The classification approach in Jokanovic and Amin [28] used DL algorithms for feature extraction and classified only two activities whereas our proposed approach distinguished among eight different types of fall and non-fall activities. The total number of samples for Jokanovic and Amin [28] is 208, whereas in the current work, 300 samples exist. This made the comparison more equitable regarding the dataset size and using the same performance metrics (accuracy and confusion matrix). There is no information about the computational complexity.

- Regarding Sadreazami et al. [37] by 1% approximately with no cross-validation and by 4% with fivefold cross-validation, those authors used DCNN for automatic feature extraction and softmax classification. In terms of computational complexity, it required 30 s per training epoch.

- By 2.5% (AC = 91.1%) with no cross-validation and by 6% with fivefold cross-validation; compared with the results obtained in CapsFall proposed approach in Sadreazami et al. [38], in which only the TF domain (spectrogram) was used as input for a capsule module network (automatic feature learning) to differentiate fall and non-fall activities. The comparison is done only between the CapsFall and the first-stage classification of the proposed method to compare performance regarding the classification of two classes. Sadreazami et al. [38] used the same radar Xethru X4M03 kit as here with 3360 samples.

- By more than 9% compared with Du et al. [39], in which a 3D DL framework was applied to the discrete representation of motion trajectory for human motion analysis. The use of 3D representation with DL increased the complexity of the algorithm without providing high performance (87.16 average accuracy

| TABLE 6 | Confusion matrix of fine K-nearest neighbour method for fall activities (using fivefold cross-validation) |
|---------|-------------------------------------------------------|
| Predicted class | Fall backward | Fall forward | Fall left | Fall right |
| True class | Fall backward | 36 | 0 | 0 | 0 |
|           | Fall forward | 1 | 42 | 0 | 2 |
|           | Fall left | 0 | 0 | 24 | 3 |
|           | Fall right | 0 | 0 | 2 | 25 |

| TABLE 7 | Confusion matrix of ensemble bagged tree method for non-fall activities (using fivefold cross-validation) |
|---------|-------------------------------------------------------|
| Predicted class | Bend | Sit | Stand | Walk |
| True class | Bend | 18 | 0 | 0 | 0 |
|           | Sit | 0 | 25 | 2 | 0 |
|           | Stand | 0 | 1 | 26 | 0 |
|           | Walk | 1 | 0 | 0 | 17 |
for radar dataset. Our proposed approach shows superiority even when a 2D representation has been used.

- By 5% more than the achieved accuracy in Du et al. [40]. SCGRNN was used to classify six different human activities using micro-Doppler spectrograms. Segmented features of spectrograms were extracted using a convolution operation. The weakness in Du et al. [39, 40] was the inability to capture the local structure of the micro-motion signature.

### TABLE 8 Confusion matrix of ensemble bagged tree method (accuracy of 86.7%) for all activities simultaneously (using fivefold cross-validation)

| Predicted class | Fall backward | Bend | Fall forward | Fall left | Fall right | Sit | Stand | Walk |
|-----------------|---------------|------|--------------|-----------|------------|-----|-------|------|
| True class      | 30            | 1    | 0            | 0         | 0          | 2   | 2     | 1    |
| Bend            | 1             | 17   | 0            | 0         | 0          | 0   | 0     | 0    |
| Fall forward    | 3             | 1    | 37           | 4         | 0          | 0   | 0     | 0    |
| Fall left       | 0             | 0    | 3            | 22        | 2          | 0   | 0     | 0    |
| Fall right      | 0             | 0    | 1            | 2         | 24         | 0   | 0     | 0    |
| Sit             | 2             | 0    | 0            | 0         | 0          | 25  | 0     | 0    |
| Stand           | 1             | 1    | 0            | 0         | 0          | 1   | 24    | 0    |
| Walk            | 1             | 0    | 1            | 0         | 0          | 0   | 0     | 16   |

### TABLE 9 Results of accuracy for tested data classification

| Classification stage | Method                          | Accuracy fivefold cross-validation | Accuracy no validation |
|----------------------|---------------------------------|-----------------------------------|------------------------|
| First                | Ensemble bagged tree            | 93.3                              | 92.0                   |
| Second fall activity | Fine K-nearest neighbour        | 86.7                              | 88.8                   |
| Second non-fall activity | Ensemble bagged tree           | 93.3                              | 70.0                   |
| Averaged accuracy    | For three classifiers           | 91.6                              | 85.7                   |

### TABLE 10 Confusion matrix of ensemble bagged tree method (tested data)

| Predicted class | Fall | Non-fall |
|-----------------|------|----------|
| True class      | 41   | 4        |
| Non-fall        | 1    | 29       |

### TABLE 11 Confusion matrix of fine K-nearest neighbour method for fall activities (test data)

| Predicted class | Backward | Forward | Left side | Right side |
|-----------------|----------|---------|-----------|------------|
| True class      | 11       | 0       | 1         | 0          |
| Forward         | 0        | 13      | 0         | 0          |
| Left            | 0        | 1       | 6         | 2          |
| Right           | 0        | 2       | 0         | 9          |

### TABLE 12 Confusion matrix of ensemble bagged tree method for non-fall activities (test data)

| Predicted class | Bend | Sit | Stand | Walk |
|-----------------|------|-----|-------|------|
| True class      | 6    | 0   | 0     | 0    |
| Sit             | 0    | 8   | 1     | 0    |
| Stand           | 0    | 1   | 8     | 0    |
| Walk            | 0    | 0   | 0     | 6    |
By 1% compared with Ding et al. [41]. The range domain was used for binary classification (first stage) into in situ motions and non-in situ motions with 99.9% accuracy (using SVM). After that, the physical empirical and PCA features were extracted from WR TFT spectograms and used to classify into in situ motions (KNN) and non-in situ motions (bagged trees), respectively.

By 0.4 compared with Li et al. [42]. A one-versus-all hierarchical classification was used with five separate models. Conventional ML algorithms, four SVM and one bagged tree, were used for classification. This approach has redundancy and computational complexity. The training data for the unclassified activities were used for each previous model before being classified.

The accuracy of the Qi et al. [43] was above 97%. Multistage classification and combined conventional machine KNN and GoogLeNet (CNN-based method) were used to distinguish 12 kinds of human activities.

Therefore, the use of multiple-domain and multistage classification improves accuracy for motion recognition. DL methods, especially neural network–based methods, show superiority in terms of classification accuracy, but they increase the complexity of learning. Referring to Qi et al. [43], the computational complexity of GoogLeNet, CNN–gated recurrent unit, and deep convolutional autoencoder is higher than SVM, KNN, bagged tree, and PCA classification. Subsequently, the complexity of the proposed method is less than the methods used in Qi et al. [43] because it uses the KNN and bagged tree as a classifier. The proposed method provides high classification performance with efficient complexity by defining and selecting a group of robust features from multiple domains and choosing the optimum classifier for each classification stage.

The hyperparameter sensitivity was also investigated. The hyperparameters are the maximum number of splits and number of learners for the first and third classifiers, and the distance metric and distance weight for the second classifier. These hyperparameters were tuned and the corresponding classification accuracy obtained. Tables 14 and 15 show the accuracy of the first and third classifiers (fall-versus-non-fall and non-fall) when the maximum number of splits and number of learners have values of {50 100 150 200 224} and {5 10 15 20 25 30 35 40}, respectively, for the first classifier and {50 60 70 80 89} and {5 10 15 20 25 30 35 40}, respectively, for the

---

**Table 13** Performance comparison of different human activity classification algorithms and proposed method

| Publication       | Classification method                  | Feature extraction domain                        | Number of classes | Accuracy (%) |
|-------------------|----------------------------------------|-------------------------------------------------|-------------------|-------------|
| Kim et al. [23]   | Support vector machine                 | Time-frequency (TF) domain                       | 7                 | 92.8        |
| Kim et al. [24]   | Deep learning (DL)                     | TF domain                                       | 7                 | ~91.0       |
| Jokanovic et al.  [26] | DL                                      | Time-range (TR) then TF domain                   | 2                 | 97.1        |
| Jokanovic et al.  [27] | Principal component analysis classifier | TR then TF domain                               | 4                 | 91.2        |
| Jokanovic et al.  [28] | DL                                      | TR then TF domain                               | 2                 | 97.1        |
| Sadreazami et al. [36] | Softmax classifier                   | Time series (deep residual network)             | 2                 | 93.0        |
| Sadreazami et al. [37] | Softmax classifier                     | Deep convolutional neural network (CNN) to time series | 2                 | 95.8        |
| Sadreazami et al. [37] | Softmax classifier with fivefold cross-validation | Deep CNN to time series                          | 2                 | 92.7        |
| Sadreazami et al. [38] | CapsFall no cross-validation          | TF domain                                       | 2                 | 94.1        |
| Sadreazami et al. [38] | CapsFall fivefold cross-validation      | TF domain                                       | 2                 | 91.1        |
| Du et al. [39]    | Three-dimensional point net            | TF domain                                       | 6                 | ~92.0       |
| Du et al. [40]    | Segmented convolutional gated recurrent neural network | TF domain                                         | 6                 | ~90.0       |
| Ding et al. [41]  | K-nearest neighbour (KNN) and bagged trees | TR then TF domain                               | 12                | ~94.8       |
| Li et al. [42]    | Support vector machine and linear discriminant | TR and TF domain                               | 6                 | 95.4        |
| Qi et al. [43]    | KNN with CNN                           | TR then TF domain                               | 12                | 97.4        |
| Proposed first stage | Ensemble bagged tree                  | TF and TR simultaneously                        | 2                 | 96.9        |
| Proposed second-stage fall activity | Fine KNN                           | TF and TR simultaneously                        | 4                 | 94.1        |
| Proposed second-stage non-fall activity | Ensemble bagged tree             | TF and TR simultaneously                        | 4                 | 95.6        |
| Proposed, average accuracy | Fine KNN with ensemble bagged tree | TF and TR simultaneously                        | 8                 | 95.8        |
third classifier with 0.1 learning. The data show that the optimum values for the maximum number of splits and number of learners are [224, 30] and [89, 30] for the first and third classifiers, respectively. Furthermore, the highest accuracy performance is obtained for the second classifier (fall) when the distance metric and the distance weight are Euclidean and equal, respectively, with $AC = 94.1\%$. Distance metric and distance weight can be \{Euclidean, Chebyshev, cosine, cubic, city block, correlation, Hamming, Spearman, Jaccard\} and \{equal, inverse, squared inverse\}, respectively.

This study did not consider situations such as the presence of multiple targets or moving obstacles in the radar field, other daily life activities (not considered in the training data), and variations owing to irregular motion (because classification is founded on micro-Doppler features, such variations can cause changes in micro-Doppler signatures and lead to misclassification). In the future, the performance could be enhanced with a larger dataset compiled of actions from both young and older subjects and over a range of environmental conditions. Besides, in situ measurement could also be considered, such as by installing the radar in home and workplace settings and recording different actual daily life activities.

### 6 | CONCLUSION

This work proposes an approach to HAC applying a two-stage ML-based algorithm to data collected from backscatter UWB radar signals. A two-stage classification algorithm for HAC was proposed and evaluated. The signal modelling of HAC using UWB radar was discussed and presented. A preprocessing stage was applied to mitigate the contribution of clutter and noise. The cleaned data were represented in two domains: TF and TR. Then, 56 features were extracted from these two domains, containing range features, Doppler signatures features, and statistical features. Important features were range difference, event width, spectrogram entropy, power of spectrogram, and the power of the third, fourth, fifth, sixth, and seventh spectrogram windows. Features of each classifier were selected to provide the optimum accuracy. The best recognition method for each classifier was chosen as ensemble bagged tree, fine KNN, ensemble bagged tree for fall/non-fall classifier, fall classifier, and non-fall classifier, respectively. The proposed two-stage algorithm with eight classes outperformed by 9%, a single-stage algorithm that used the ensemble bagged tree classifier with 44 selected features. Finally, the performance of the proposed approach was compared with the literature, and the proposed approach showed better performance.

### ACKNOWLEDGEMENT

The authors thank the Alexander von Humboldt Foundation (Alexander von Humboldt-Stiftung) for helpful support.

### ORCID

Mohamad Mostafa  
[https://orcid.org/0000-0003-2687-0963](https://orcid.org/0000-0003-2687-0963)  
Somayyeh Chamaani  
[https://orcid.org/0000-0003-2077-599X](https://orcid.org/0000-0003-2077-599X)

### REFERENCES

1. World Health Organization: Falls (2019). [https://www.who.int/news-room/fact-sheets/detail/falls](https://www.who.int/news-room/fact-sheets/detail/falls)
2. Vynorste, S.B., Annest, J.L., Ryan, G.W.: Surveillance for fatal and nonfatal injuries—United States. MMWR, Surveill. Summ. 53(7), 1–57 (2001)
3. Centres for Disease Control and Prevention: Fatalities and injuries from falls among older adults - United States, 1993-2003 and 2001-2005. Morb Mortal Wkly Rep. 55(45), 1221–1224 (2006)
4. Moran, C.G., et al.: Early mortality after hip fracture: is delay before surgery important? JBJS. 87(3), 483–489 (2005)
5. Gurley, R.J., et al.: Persons found in their homes helpless or dead. N Engl J Med. 334(26), 1710–1716 (1996)
6. Evaluation of accelerometer-based fall detection algorithms on real-world falls, PloS One. 7(5), 37062 (2012)
7. Kansiz, A.O., Guvensan, M.A., Turkmen, H.I.: Selection of time-domain features for fall detection based on supervised learning. Proc. World Congr. Eng. Comput. Sci., USA. 2, 1–6 (2013)
8. Vallabh, P. et al.: Fall detection using machine learning algorithms. In: 2016 24th Int. Conf. Software, Telecommun. Comput (SoftCOM). Networks, pp. 1–9. (2016)
9. Ngwu, A. et al.: Fall detection using smartwatch sensor data with accessor architecture. In: Int. Conf. Smart heal, pp. 81–93, Springer, Cham (2017)
10. Huynh, Q.T., et al.: Optimization of an accelerometer and gyroscope-based fall detection algorithm. J. Sensors. (2015)
11. Rakhman, A. Z., Nugroho, I. E.: Fall detection system using accelerometer and gyroscope based on smartphone. In: 2014 1st Int. Conf. Inf. Technol. Comput. Electr. Eng., IEEE, pp. 99–104. (2014)

12. Porter, E. J.: Wearing and using personal emergency response system buttons. J. Gerontol Nurs. 31(10), 26–33 (2005)

13. Diraco, G., Leone, A., Siciliano, P.: A radar-based smart sensor for unobtrusive elderly monitoring in ambient assisted living applications. Biosensors. 7(4), 55 (2017)

14. Feng, G., et al.: Floor pressure image for fall detection using fibre-optic sensors. IEEE Pervasive Comput. 15(2), 40–47 (2016)

15. Zhang, C., Tian, Y.: RGB-D camera-based daily living activity recognition. J. Comput. Vis. Image Process. 2(4), 12 (2012)

16. Miao, S. G., Sung, P. H., Huang, C. Y.: A customised human fall detection system using omnica-camera images and personal information. In: 1st Transdiscipl. Conf. Distrib. Diagnosis home heal, D2H2, IEEE, pp. 1/39–42. (2006)

17. Tao, J., et al.: Fall incidents detection for intelligent video surveillance. In: 5th Int. Conf. Inf. Commun. Signal process, pp. 1599–1594 (2005)

18. Li, Y., Ho, K. C., Popescu, M.: Efficient source separation algorithms for acoustic fall detection using a Microsoft Kinect. IEEE Trans Biomed Eng. 61(3), 745–755 (2013)

19. Li, Y., et al.: Acoustic fall detection using a circular microphone array. Annu. Int. Conf. IEEE Eng. Med. Biol. 2242–2245 (2010)

20. Popescu, M., et al.: An acoustic fall detector system that uses sound height information to reduce the false alarm rate 30th. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. 4628–4631 (2008)

21. Liu, T., Guo, X., Wang, G.: Elderly-falling detection using distributed direction-sensitive pyroelectric infrared sensor arrays. Multidimens Syst Signal Process. 23(4), 451–467 (2012)

22. Yazar, A., Ersden, F., Gurin, A. E.: Multi-sensor ambient assisted living system for fall detection. Proc. IEEE Int. Conf. Acoust. Speech, Signal Process., 1–3 (2014)

23. Kim, Y., Ling, H.: Human activity classification based on multi-Doppler signatures using support vector machine. IEEE Trans. Geosci. Remote Sens. 47(5), 1328–1337 (2009)

24. Kim, Y., Moon, T.: Human detection and activity classification based on multi-Doppler signatures using deep convolutional neural networks. IEEE Trans. Geosci. Remote Sens. 13(1), 8–12 (2016)

25. Erol, B., Amin, M. G., Boashash, B.: Range-Doppler radar sensor fusion for fall detection, IEEE Radar Conf., pp. 819–824. (2017)

26. Jokanovic, B., et al.: Multi-window time-frequency signature reconstruction from undersampled continuous-wave radar measurements for fall detection, IET Radar. Sonar Navig. 9(2), 173–183 (2014)

27. Jokanovic, B., Amin, M. G.: Suitability of data representation domains in expressing human motion radar signals. IEEE Geosci. Remote Sens. Lett. 14(12), 2370–2374 (2017)

28. Jokanovic, B., Amin, M. G.: Fall detection using deep learning in range-Doppler radar. IEEE Trans. Aerosp. Electron. Syst. 54(1), 180–189 (2017)

29. Jokanovic, B., Amin, M. G., Ahmad, E.: Effect of data representations on deep learning in fall detection, Sens. Array Multichannel signal process. Work (SAM), pp. 1–5. (2016)

30. Gaddde, A., et al.: Fall detection and classifications based on time-scale radar signal characteristics. Radar Sens. Technol. XVIII(9077), 907712 (2014)

31. Jokanovic, B., et al.: Radar fall detection using principal component analysis. Radar Sens. Technol. XX9829), 982919 (2016)

32. Jokanovic, B., Amin, M. G., Erol, B.: Multiple joint-variable domains recognition of human motion. IEEE Radar Conf., pp. 948–952 (2015)

33. Liu, L., et al.: Doppler radar sensor positioning in a fall detection system. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc., 256–259 (2012)

34. Sadreazami, H., et al.: Compressed domain contactless fall incident detection using UWB radar signals. In: 18th IEEE International New Circuits and Systems Conference, pp. 90–93. (2020)

35. Nejadgholi, I., et al.: Classification of Doppler radar reflections as preprocessing for breathing rate monitoring. IET Signal Process. 13(1), 21–28 (2018)

36. Sadreazami, H., Bolte, M., Rajan, S.: Residual network-based supervised learning of remotely sensed fall Incidents using ultra-wideband radar. In: IEEE Int. Symp. Circuits Syst. IEEE, pp. 1–4. (2019)

37. Sadreazami, H., Bolte, M., Rajan, S.: Fall detection using Standoff radar-based sensing and deep convolutional neural network. IEEE Trans. Circuits Syst. II, 197–201 (2019). Express Briefs 67

38. Sadreazami, H., Bolte, M., Rajan, S.: CapsFall: fall detection using ultra-wideband radar and capsule network. IEEE Access. 7, 55336–55343 (2019)

39. Du, H., et al.: A three-dimensional deep learning framework for human Behaviour analysis using range-Doppler time points. Geosci Rem Sens Lett IEEE. 17(4), 611–615 (2019)

40. Du, H., et al.: Segmented convolutional gated recurrent neural networks for human activity recognition in ultra-wideband radar. Neurocomputation. 396, 451–464 (2020)

41. Ding, C., et al.: Non-contact human motion recognition based on UWB radar. IEEE Journal on Emerging and Selected Topics in Circuits and Systems. 8(2), 306–315 (2018)

42. Li, X., et al.: Hierarchical radar data analysis for activity and Personnel recognition. Rem Sens. 12(14), 2237 (2020)

43. Qi, R., et al.: Multi-classification algorithm for human motion recognition based on IR-UWB radar. IEEE Sensor J. 20(21), 12848–12858 (2020)

44. Mostafa, M., Chamaani, S., Sachs, J.: Applying singular value decomposition for clutter reduction in heartbeat estimation using M-sequence UWB Radar. Proc. Int. Radar Symp. 1–10 (2018)

45. Mostafa, M., Chamaani, S., Sachs, J.: Singular Spectrum Analysis based algorithm for vitality monitoring using M-sequence UWB Sensor. IEEE Sens. J. 20(9), 4787–4802 (2019)

46. A. Nezir, O. A. Yavrov, O. L. Lighthart, Signal processing for improved detection of Trapped Victims using UWB radar, IEEE Trans. Geosci. Remote Sens. 48, pp. 114–114 (2010)

47. Liang, X., et al.: Improved denoising method for through-wall vital sign detection using UWB impulse radar. Digit Signal Process. 74, 72–93 (2018)

48. Wu, Q., et al.: Radar-based fall detection based on Doppler time-frequency signatures for assisted living, IET Radar. Sonar Navig. 9(2), 164–172 (2014)

49. Sadreazami, H., Bolte, M., Rajan, S.: On the use of ultra wideband radar and stacked LSTM-RNN for at home fall detection. In: IEEE Life Sci. Conf. pp. 255–258. (2018)

50. Amin, M. G., et al.: Radar signal processing for elderly fall detection. IEEE Signal Process Mag. 33(2), 71–80 (2016)

51. Diraco, G., Leone, A., Siciliano, P.: AI-based early change detection in smart living environments. Sensors. 19(16), 3549 (2019)

52. Novelda’s XeThru X4M03, (n.d.). https://www.xethru.com/x4-radar-chip.html/(accessed February 25, 2019)

53. Narayan, R. M., Smith, S., Gallagher, K. A.: A multifrequency radar system for detecting humans and characterising human activities for short-range through-wall and long-range foliage penetration applications. In: IEEE Int. Microw. Sci. Technol. (2014)

54. Rane, S. A., et al.: Clutter suppression techniques to detect behind the wall static human using UWB Radar. In: Ieee Int. Conf. Recent Trends Electron. Inf. Commun. Technol. (RTEICT), IEEE, pp. 1325–1329. (2016)

55. Sachs, J., Herrmann, R.: M-sequence-based ultra-wideband sensor network for vitality monitoring of elders at home, IET Radar. Sonar Navig. 9(2), 125–137 (2015)

56. He, Y., Li, X., Jing, X.: Towards deep learning-based human target analysis. I UWB Technol. Its Appl., IntechOpen (2018)

57. Portmoff, M.: Time-frequency representation of Digital signals and systems based on short-time Fourier analysis. IEEE Trans. Acoust. 28(1), 55–69 (1980)