Multimedia Tools and Applications
https://doi.org/10.1007/s11042-024-20136-8

Investigating the impact of sensor axis combinations on activity recognition and fall detection: an empirical study

Erhan Kavuncuoğlu[1] · Ahmet Turan Özdemir[2] · Esma Uzunhisarcıklı[3]

Received: 27 October 2023 / Revised: 11 June 2024 / Accepted: 18 August 2024

Abstract
Activity recognition is a fundamental concept widely embraced within the realm of healthcare. Leveraging sensor fusion techniques, particularly involving accelerometers (A), gyroscopes (G), and magnetometers (M), this technology has undergone extensive development to effectively distinguish between various activity types, improve tracking systems, and attain high classification accuracy. This research is dedicated to augmenting the effectiveness of activity recognition by investigating diverse sensor axis combinations while underscoring the advantages of this approach. In pursuit of this objective, we gathered data from two distinct sources: 20 instances of falls and 16 daily life activities, recorded through the utilization of the Motion Tracker Wireless (MTw), a commercial product. In this particular experiment, we meticulously assembled a comprehensive dataset comprising 2520 tests, leveraging the voluntary participation of 14 individuals (comprising 7 females and 7 males). Additionally, data pertaining to 7 cases of falls and 8 daily life activities were captured using a cost-effective, environment-independent Activity Tracking Device (ATD). This alternative dataset encompassed a total of 1350 tests, with the participation of 30 volunteers, equally divided between 15 females and 15 males. Within the framework of this research, we conducted meticulous comparative analyses utilizing the complete dataset, which encompassed 3870 tests in total. The findings obtained from these analyses convincingly establish the efficacy of recognizing both fall incidents and routine daily activities. This investigation underscores the potential of leveraging affordable IoT technologies to enhance the quality of everyday life and their practical utility in real-world scenarios.

Keywords Fall detection · Activity recognition · Wearable sensors · Sensor axis combinations · Machine learning

1 Introduction

Activity recognition and fall detection are critical for ensuring the safety and well-being of older adults. Technological advancements have significantly improved these capabilities. Innovative approaches utilizing mobile devices, wearable sensors, and artificial intelligence algorithms enable real-time activity classification and fall detection.

Extended author information available on the last page of the article

Published online: 12 September 2024

Activity recognition plays a vital role in monitoring a person's daily activities (ADLs) and gleaning valuable insights into their health status. Falls, however, pose a significant risk factor for older adults, potentially leading to severe injuries.

Demographic changes are fueling the recent surge in health technology advancements. According to World Health Organization (WHO) data, a key driver is the steadily increasing global elderly population [1]. Globally, the population aged 65 and over stood at 9% (688 million) in 2019. This figure is projected to rise to approximately 12% (1 billion) by 2030 and further increase to 16% (1.6 billion) by 2050 [2]. Considering the growing elderly and disabled population, the development of assistive technologies (ATs) to empower them in daily living activities (DLAs), promote their safety and independence, and reliably detect critical events like falls has emerged as a progressively crucial and indispensable research domain [3].

Research efforts have focused not only on reliable fall detection but also on monitoring and recognizing Activities of Daily Living (ADLs) to improve the quality of life for individuals at risk of falls. Given the strong correlation between falls and ADLs established in numerous studies, activity recognition systems hold significant potential for various applications. These applications encompass social-physical interaction [4], factory worker activity recognition [5], health and sports science domains [6], and even extend to the entertainment and interactive gaming sectors [7].

Numerous solutions have been proposed for automatic activity recognition and fall detection [8–11]. Classification of these solutions can be achieved based on the sensor technology employed, encompassing three primary categories: Ambient Sensor-Based (ASB), Wearable Sensor-Based (WSB), and Hybrid Sensor-Based (HSB) approaches [10–12].

- Ambient Sensor-Based (ASB) Technologies: Leveraging a diverse array of sensor modalities, including acoustic [13], infrared [14], vibration [15], and vision-based sensors [16], these technologies are seamlessly integrated into the environment (doors, walls, floors, furniture, etc.) to facilitate ADL recognition and fall detection [17].
- Wearable Sensor-Based (WSB) Technologies: At the core of WSB technologies lie sensors that capture motion parameters, including acceleration, velocity, and orientation [10, 11].
- Hybrid Sensor-Based (HSB) Technologies: HSB technologies seamlessly integrate both ASB and WSB approaches, often employing sensor pairs such as microphone-accelerometer or infrared microphone combinations [18].

Despite their substantial benefits for activity recognition and fall detection, these technologies are not without their inherent challenges, such as real-time analysis, integration with smart homes, high computational power requirements, data fusion across different sensors, and sensor synchronization needs [12, 18, 19].

In the nutshell, this study addresses limitations in existing activity recognition and fall detection research, aiming for a more robust and generalizable approach.

Key Improvements:

- Comprehensive Dataset: We address generalizability by constructing a diverse dataset encompassing a wider range of activities (36 total, including 20 falls) and balanced participant demographics.

- Optimized Model Performance: Hyperparameter analysis ensures optimal classification accuracy for the models.
- Real-World Applicability: A low-power sensor network with energy harvesting capabilities promotes long-term wearability.
- Effective Sensor Data Utilization: We investigate selecting appropriate sensor axis combinations, demonstrating high accuracy with carefully designed models.

Challenges:

- User-centered evaluation: Conducting pilot studies to assess user comfort, acceptance, and the system's effectiveness in real-world settings.
- Data expansion and analysis: Collecting more data encompassing a wider variety of situations to enhance model stability and investigate the effects of additional parameters.
- Integration with existing systems: Exploring seamless integration with other health monitoring wearables for a more comprehensive approach to health management.

By addressing these challenges, we can refine the sensor network architecture, optimize energy consumption, and enhance user experience. This will lead to a lightweight, cost-effective, and perpetually wearable fall detection system that significantly improves user quality of life.

Contribution to the Field:

This study contributes by:

- Assessing Sensor Combinations: Analyzing the impact of different sensor axis combinations on activity recognition performance in wearable-based sensors.
- Realistic Evaluation: Obtaining realistic results by working with a gender-balanced participant group.
- Comparison Standard: Providing a comparison standard for fall and activity recognition systems, improving their comparability.
- Foundation for Improvement: Laying a foundation for improving the design of fall and activity recognition devices.
- Next-Generation AI Algorithms: Building upon these advancements, the aim is to develop a device capable of real-time operation and an efficient artificial intelligence model. The results of this study will contribute to the development of next-generation AI algorithms for activity recognition and fall detection.

Overall, this work presents a significant step towards robust and reliable fall detection systems, promoting the safety, well-being, and independent living of the growing elderly population.

The organization of this paper is as follows: Sect. 2 delves into a comprehensive review of extant literature on the application of WSB technologies in activity recognition. Delving into the methodological aspects, Sect. 3 elaborates on the datasets utilized in the study and comprehensively outlines the methodology employed for training and evaluating the machine learning (ML) models. In Sect. 4, a detailed exposition of the study's findings are found. Section 5 explores the impact of using different sensor axis combinations on activity recognition performance. Finally, Sect. 7 concludes the paper by discussing potential future research directions in this field.

2 Related works

Researchers have introduced a diverse range of devices specifically designed for activity recognition and fall detection applications. However, evaluating the accuracy level of these devices is challenging as common activity datasets are not available. In previous studies (Table 1), research has been conducted using public datasets and self-created datasets [20, 21]. For instance, the PAMAP2 dataset contains the ADLs of nine elderly volunteers [22]. The SBHAR dataset includes data on six different activity types [23]. The MHealth dataset comprises the ADLs of 10 volunteers [24]. The MobiAct dataset contains the activities of 66 volunteers [25]. Additionally, publicly accessible datasets such as the Multimodal UP-Fall Detection Dataset are also available [26]. These datasets have accelerated the recognition of falls and daily activities, generating significant interest for research [27]. In general, these datasets have facilitated the development of a standard for research [28].

Numerous academic studies have focused on fall and activity recognition algorithms. For instance, Buber and Guvensan proposed a study for activity recognition [29]. Dernbach and colleagues conducted a study for the recognition of simple and complex activities [30]. Anjum and Ilyas presented a study on recognizing activities with a smartphone carried in different positions [31]. Saputri and colleagues conducted a study on activity recognition using a smartphone [32]. Bayat introduced a novel system capable of recognizing six distinct activity types [33]. Figueriedo and colleagues suggested a technique for recognizing falls [34]. Zhao and colleagues proposed a fall detection system based on smartphones [35]. Albert and colleagues collected acceleration data for ADLs [36]. Kansiz and colleagues conducted a study using a smartphone accelerometer to recognize activities [37]. Mehrang and colleagues used heart rate monitors and accelerometers for activity recognition [38]. Pavey and colleagues recognized activities using a wrist-worn accelerometer [39]. Hsu and colleagues identified ADLs using an inertial system [40]. Sok and colleagues proposed a method for fall detection [41]. Li and colleagues recognized activities using signal streams acquired from sensors [42].

In these studies in the literature, different datasets, sampling frequencies, activity types, numbers of volunteers, gender balance, and sensor configurations were used. Therefore, it is challenging to compare the results of these studies. Another issue is the lack of information on the performance of different sensor combinations.

3 Materials and methods

This section provides information about the system developed for activity tracking and fall detection (Activity Tracking Device—ATD) and the commercially available device (Motion Trackers Wireless—MTw). It also explains the general working principles regarding sensor types, the number of sensors, and configurations. Details about the experimental preparation process and information about the volunteers are also included in this section.

3.1 Systems used for Activity Recognition and Fall Detection (ATD and MTw)

This study employed ATD and MTw devices, encompassing a fusion of MEMS inertial sensors and magnetic sensors, to facilitate activity recognition and fall detection.

Table 1 Review of methods and performance results for falls and ADLs

Ref	Dataset	Sens	Vol	Act	Samp. (Hz)	Pos	Alg	Perf
Reiss [22]	PAMAP2	3xA (×2), 3xG, 3xM, HRM	9	18 ADL	100	Chest, wrist, arm	C4.5, *k*-NN, Boosted C4.5, Bagging C4.5, NB	Boosted C4.5: %99,69
Anguita [23]	SBHAR	3xA 3xG	30	6 ADL	50	Waist	multiclass SVM	multiclass SVM: %96
Memiş [24]	Mhealth	3xA, 3xG, 3xM, ECG	10	12 ADL	50	Chest, right wrist, left ankle	SVM, *k*-NN, NB, RF, Bagging, DTs	SVM: %98,4
Vavoulas [25]	MobiAct	3xA, 3xG	66	12 ADL 4 Fall	20	Trouser pocket	J48, LR, MLP, *k*-NN	*k*-NN: %97,1
Buber [29]	non-public	3xA	5	7 ADL	20	Front trouser pocket	J48, K-Star, BN, NB, RF, *k*-NN	*k*-NN: %94
Dernbach [30]	non-public	3xA, 3xG	10	8 ADL	80	User selection (position and orientation)	MLP, NB, BN, DTs, B-FT, *K*-star	MLP: %93
Anjum [31]	non-public	3xA, 3xG, GPS	10	7 ADL	15	Hand, trouser pocket, shirt pocket, handbag	NB, C4.5, *k*-NN, SVM	C4.5: %95,2
Saputri [32]	non-public	3xA	27	5 ADL	50	Front trouser pocket	ANN	93%
Bayat [33]	non-public	3xA	4	6 ADL	100	Hand	J48, *K*-Star, BN, NB, RF, *k*-NN	MLP&LB&SVM: % 91,15Acc
Figueriedo [34]	non-public	3xA, 3xG, 3xO	8	10 Fall 17 ADL	50, 100	Trouser pocket, belt	SVM. threshold algorithms	SVM %96,19
Zhao [35]	non-public	3xA, WiFi module	10	4 Fall 3 ADL	32	Waist	C 4.5, NB, SVM	C 4.5%100Precision %75,8Recall
Albert [36]	non-public	3xA	15	Fall-like event 4 ADL	20	Belt: set position and orientation	C 4.5, NB, RLR, *k*-NN, SVM	RLR: Detection: %98 Classification: %99,6
Kansiz [37]	non-public	3xA	8	4 Fall 6 ADL	20	Pocket	J48, *K*-Star, NB	*K*-Star Average recall %88,0
Mehrang [38]	non-public	3xA, HRM	20	4 ADL	25	Wrist	RF, SVM	RF % 89,2 SVM %85,6

Table 1 (continued)

Ref	Dataset	Sens	Vol	Act	Samp. (Hz)	Pos	Alg	Perf
Pavey [39]	non-public	3xA	21	4 ADL	30	Wrist	RF	Acc. % 80,1 (sedentary) % 95,7 (stationary) %91,7 (walking) % 93,7 (runing)
Hsu [40]	non-public	3xA, 3xG	10	10 ADL	100	Wrist	PNN	% 90,5 Acc
Sok [41]	non-public	3xA	13	6 ADL	-	Waist	HMM	% 88,9 Acc

B-FT Best-First Tree, *RF* Random Forest, C4.5 (WEKA J48), *ANN* Artificial Neural Network, C4.5 Decision Tree, *DTs* Decision Trees, ID3 Decision Tree, K-Star, *k-NN* k-Nearest Neighbors, *LR* Logistic Regression, *RLR* Regularized Logistic Regression, *BN* Bayes Net, *MLP* Multilayer Perceptron, *NB* Naive Bayes, *QDA* Quadratic Discriminant Analysis, *SVM* Support Vector Machine, *PNN* Probabilistic Neural Network, *HMM* Hidden Markov Model, *EELM* Ensemble Extreme Learning Machine, *ELM* Extreme Learning Machine, *LSTM* Long Short Time Memory, *HRM* Heart Rate Monitor. In the sensor column, *3xA* 3-axis accelerometer, *3xG*: 3-axis gyroscope, *3xM* 3-axis magnetometer. References (Ref.); Sensor (Sens.); Volunteer (Vol.); Activities (Act.); Sampling Frequency (Samp.); Positions (Pos.); Algorithms (Alg.); Performance (Perf.)

Fig. 1 MTw sensor and recording unit

Table 2 Fundamental detection components of MTw sensors

	A	G	M
Sensör Type	Digital	Digital	Digital
Full scale	± 120 m/s^2	± 1200 deg/s	± 1.5 Gauss
Noise	200 $\mu g/\sqrt{\text{Hz}}$	0.01 deg/s/$\sqrt{\text{Hz}}$	0.2 mGauss/Hz
Bandwith	180 Hz	180 Hz	10–60 Hz (var.)
Bias stability	0.1 mg	10 deg/hr	

3.1.1 Motion Trackers (MTw) development kit

Xsens Technologies, the renowned developer of motion tracking solutions, offers the MTw development kit [43]. This kit comprises both hardware and software components (see Fig. 1). The kit comprises two primary components: six MTw sensor units and an Awinda Receiver Station.

The MTw sensor unit comprises a suite of sensors, including a 3D accelerometer (A) for detecting 3D acceleration, a gyroscope (G) for measuring 3D angular velocity, a magnetometer (M) for gauging 3D magnetic field, a barometer to gauge atmospheric pressure, with a measurement range of 300–1100 hPa. While the barometer data is not utilized for classification purposes, as outlined in Table 2, the remaining sensors play a crucial role in capturing relevant motion and orientation information.

3.1.2 Activity Tracking Device (ATD)

The ATD architecture comprises six components, including four sensors, one controller, a battery, and an SD card reader. Four different sensor types, namely BMX055, BMP280, MAX30102, and GSR, are utilized within the ATD framework. Other components in the ATD architecture include the ESP32-WeMos-Lolin32 controller, a lithium-ion battery, and an SD card reader.

The BMX055 sensor is a 9-axis sensor used for motion, orientation, and magnetic direction detection. The BMP280 sensor is employed for absolute pressure measurement. The MAX30102 sensor is capable of pulse and oximetry measurements, while the GSR sensor is used to measure galvanic skin response (see Fig. 2). Table 3 details the specifications of the sensors employed in the ATD.

The ESP32-WeMos-Lolin32 controller has been employed as the controller in this setup. This module provides high processing power and low power consumption, along

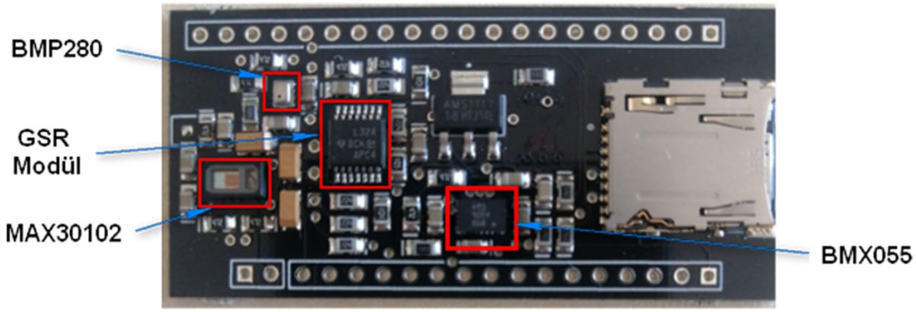

Fig. 2 Placement of sensors on the board

Table 3 Fundamental detection components of sensors in ATD Design

BMX055		
Accelerometer	Sensor Type	Digital
	Measurement Range (g)	±16
	Zero Offset (mg)	±80
	Noise Density (ug/$\sqrt{}$Hz)	150
	Resolution [bit]	12
Gyroscope	Sensor Type	Digital
	Measurement Range (derece/sn)	2000
	Zero Offset (derece/sn)	±1
	Noise Density (derece/s/$\sqrt{}$Hz)	
	Resolution [bit]	16
Magnetometer	Sensor Type	Digital
	Measurement Range (gauss)	12
	Resolution [bit]	12
BMP280		
Sensor Type		Digital
Pressure Resolution		0.16 Pa
Measurement Range		157 Hz üstü
MAX30102		
Sensor Type		Digital
ADC Resolution		14
Red ADC Count		26000
IR ADC Count		26000
Dark Current Count		0
DC Ambient Light Rejection		0
Grove - GSR		
Sensitivity		Adjustable with a Potentiometer
Input Signal		No Resistance, No Conductivity
Output Signal		Voltage, Analog Reading
Finger Touch Material		Nickel

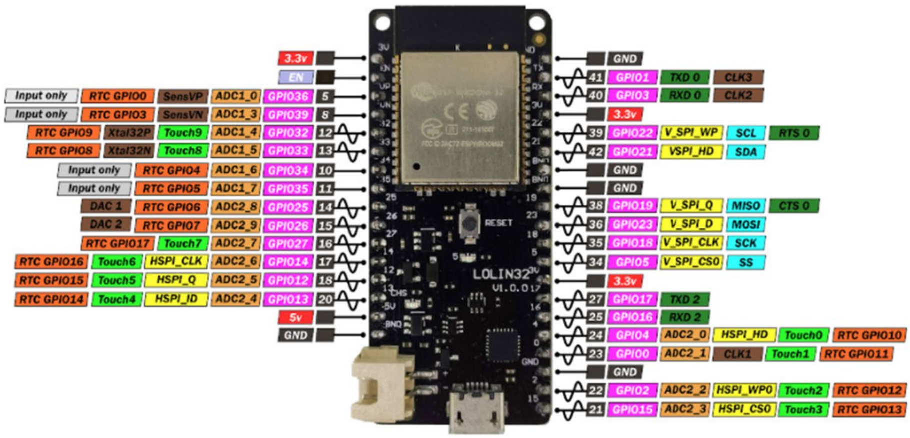

Fig. 3 ESP32 module and pinout specifications

Fig. 4 Battery used as the power supply for ATD

with Wifi, Bluetooth, and BLE capabilities. Furthermore, it is equipped with GPIO, UART, I^2C, and SPI interfaces for controlling various peripheral devices (see Fig. 3).

A 1450 mA lithium-ion battery has been used to provide power to the ATD device (see Fig. 4). Additionally, an SD card reader module has been employed for continuous data recording.

3.2 Experimental preparation process and volunteer information

This section provides information about the experimental stages conducted with MTw and ATD devices and details about the volunteers.

3.2.1 Experimental preparation process

The experimental setup involved a rigorous series of trials adhering to the established experimental protocol for fall event simulation [44]. The experiments conducted in this study involved human participants and were approved by the Ethics Committee of Erciyes

University (Approval Number: 2011/319). All the sensor units used in the research were adjusted and calibrated, ensuring that the datasets were accurate and reliable. A sampling frequency of 25 Hz was chosen to collect the data effectively and efficiently. Selecting an appropriate sampling rate is crucial for recognizing activities and detecting falls. In general, human activities exhibit a frequency range that typically falls between 0 and 20 Hz. Balancing power efficiency and data fidelity, a sampling frequency of 25 Hz was strategically selected for this study [45].

Experimental preparation process using MTw Experiments created using MTw were conducted with a total of 14 volunteers (7 females and 7 males). Female participants' demographic characteristics were captured as follows: mean age 21.5 ± 2.5 years, mean weight 58.5 ± 11.5 kg, and mean height 169.5 ± 12.5 cm. Among male participants, the mean age was 24 ± 3 years, the mean weight was 67.5 ± 13.5 kg, and the mean height was 172 ± 12 cm (see Table 4).

Experimental preparation process Using ATD The identification of ADL and fall actions within the ATD dataset was guided by the activity type labels extracted from the MTw device data. The study involved 30 participants (15 females and 15 males). Female participants were further characterized by an average age of 22.3 ± 6.5 years, weight of 60.3 ± 9 kg, and height of 164.4 ± 5.5 cm. For male participants, the age, weight, and height ranges were calculated as 28.9 ± 10.7 years, 80.1 ± 12.6 kg, and 177 ± 7.3 cm, respectively (see Table 5).

Table 4 Age, Gender, and Anthropometric Information of Volunteers (MTw)

Gender	Volunteer	Height (cm)	Weight (kg)	Age
Male	101	170	75	21
	102	174	81	21
	103	180	78	23
	104	176	67	27
	106	160	54	22
	107	175	72	21
	108	184	68	21
Female	203	170	51	21
	204	157	47	21
	205	169	51	20
	206	166	47	19
	207	165	60	20
	208	163	55	24
	209	182	70	22
	All Volunteers			
	Mean	170.79	62.57	21.64
	Standard Deviation	8.17	11.82	1.98

Table 5 Age, Gender, and Anthropometric Information of Volunteers (ATD)

Gender	Volunteer	Height (cm)	Weight (kg)	Age
Male	101	179	100	39
	102	174	96	37
	103	191	69	18
	104	178	86	20
	105	175	89	18
	106	168	93	50
	107	170	62	21
	108	168	90	37
	109	182	75	20
	110	192	60	18
	111	178	76	34
	112	176	71	21
	113	170	87	41
	114	174	81	37
	115	180	67	22
Female	201	163	50	34
	202	169	69	19
	203	166	53	21
	204	162	62	19
	205	156	73	41
	206	172	70	18
	207	163	45	19
	208	165	53	19
	209	162	59	23
	210	169	73	21
	211	159	48	21
	212	175	63	19
	213	157	64	20
	214	168	63	22
	215	160	60	18
	All Volunteers			
	Mean	170.70	70.23	25.57
	Standard Deviation	9.01	14.73	9.32

The Placement of ATD on Volunteers is Illustrated in Fig. 5.

3.3 Activity types, dataset, and classification techniques

This section provides an explanation of activity types, data collection, preprocessing, artificial intelligence techniques, and performance metrics for the detection of ADLs and falls.

Fig. 5 The Placement of ATD on Volunteers

3.3.1 Data collection process and activity types with MTw

For this investigation, we employed a dataset encompassing 2520 records (14 volunteers × 36 activities × 5 repetitions). The data was meticulously collected from 14 volunteers, encompassing 36 distinct activities (20 sets fall and 16 sets ADL), each performed five times. This dataset incorporates both fall and ADL event recordings. A detailed breakdown of the activity types is provided in Table 6.

The experiments involved the placement of three-axis sensors equipped with six sensor units (accelerometer, gyroscope, and magnetometer) on various sections of the volunteers' bodies. Recorded DLA and falls differ from activities recorded in a laboratory setting, as they mimic real-life occurrences [46].

3.3.2 Data collection process and activity types with ATD

In this study, a dataset comprising 1350 records (30 volunteers × 15 activities × 3 repetitions) collected from 30 volunteers was used. Encompassing both fall and ADL data, the dataset comprises a collection of activities performed by volunteers. These activities,

Table 6 List of Considered Activity Types in the Study (Falls and ADLs) (MTw)

Experimental ADLs		
#	Label	Description
1	Walking forward	Walking straight ahead
2	Walking backward	Reversal of the first movement
3	Slow jogging	Jogging slowly like athletes do for warming up
4	Squatting and standing up	Bending knees and standing up again
5	Bending at the waist	Bending at the waist with a 90-degree angle
6	Bending down to pick up an object	Bending down with a 90-degree angle to pick up an object from the floor
7	Limping	Walking as if one leg is defective
8	Stumble	Continuing to walk after stumbling upon an object while walking forward
9	Ankle sprain	Temporary rolling the foot inward while walking forward and continuing to walk
10	Coughing	Voluntary cough or sneeze
11	Sitting on a chair	Sitting on a chair at a level and comfort suitable
12	Sitting on a couch	Sitting on a couch at a level and comfort suitable
13	Sitting in a squat position	Sitting upright in a squat position
14	Sitting on a bed	Sitting on a bed at a level and comfort suitable
15	Lying down on a bed	Lying down vertically on a bed
16	Getting out of bed	Transitioning from a lying position to a sitting position
Experimental Falls		
#	Label	Description
17	Falling forward	Falling forward to the ground
18	Falling forward while protecting oneself	Falling forward while using hands to protect oneself
19	Falling towards the knees	Falling down on the knees
20	Falling onto the knees and then lying down	Falling onto the knees and then lying down on the floor
21	Getting up quickly after falling down	Quickly getting up after falling down
22	Getting up slowly after falling down	Slowly getting up after falling down
23	Falling forward and turning to the right	Falling forward and turning to the right side
24	Falling forward and turning to the left	Falling forward and turning to the left side
25	Sitting back	Sitting down backwards

Table 6 (continued)

26	Lying down backwards	Falling backwards to the ground
27	Falling backwards and turning to the right	Falling backwards and turning to the right side
28	Falling backwards and turning to the left	Falling backwards and turning to the left side
29	Falling onto the right side	Falling onto the right side
30	Getting up quickly after falling onto the right side	Quickly getting up after falling onto the right side
31	Falling onto the left side	Falling onto the left side
32	Getting up quickly after falling onto the left side	Quickly getting up after falling onto the left side
33	Falling off the bed	Falling off the bed
34	Falling off the podium	Falling off the podium
35	Fainting and falling to the ground	Fainting and collapsing to the ground
36	Collapsing against the wall and slowly falling down	Collapsing against the wall and gradually sliding down

Table 7 List of Considered Activity Types in the Study (Falls and ADLs) (ATD)

#	Label	Description
Experimental ADLs		
1	Walking straight ahead	Walking straight ahead
2	Brisk slow jogging	Jogging slowly, similar to warm-up for athletes
3	Squatting and standing up	Squatting down and standing up again
4	Bending and staying	Bending at the waist with a 90-degree angle and returning to a straight standing position
5	Stumbling	Continued walking after stumbling upon an object while walking forward
6	Sitting on a chair	Sitting on a chair at a level and comfort suitable (hard surface)
7	Stand up from a chair	Stand up from a chair at a level and comfort suitable (hard surface)
8	Lying down on a bed	Lying down vertically on a bed
Experimental Falls		
#	Label	Description
9	Falling forward	Falling forward to the ground
10	Falling backward	Falling backward to the ground
11	Falling onto the right side	Falling onto the ground with the right arm
12	Falling onto the left side	Falling onto the ground with the left arm
13	Falling off the bed	Falling to the floor while lying in bed
14	Falling off the podium	Slip and fall while walking on the podium
15	Fainting	Fainting and falling to the ground

categorized into seven fall sets and eight ADL sets, were each repeated three times to ensure data consistency and robustness. A detailed description of the activity types is provided in Table 7.

For data collection, sensor units equipped with tri-axial sensors (accelerometer, gyroscope, and magnetometer) were affixed to the waist regions of the participants. This choice was made considering studies [47, 48] that indicate the highest classification performance is achieved with sensors placed on the waist.

3.3.3 Dataset Preprocessing Procedure

Each movement was captured by the waist sensor unit for a period of 12–18 s. The highest acceleration value (A_{max}) was determined using the data gathered from the accelerometer [49, 50].

$$A_{max} = \sqrt{A_x^2 + A_y^2 + A_z^2} \quad (1)$$

To obtain the active motion range, a two-second time interval was used to collect data from before and after the peak acceleration. This resulted in a total of 101 samples, with 25 samples per second for a total of 2 s before and 2 s after the peak acceleration (2 s × 25 Hz for the peak acceleration + 1 sample + 2 s × 25 Hz). The remaining records were not used.

Following data acquisition from the accelerometer, gyroscope, and magnetometer sensors along the three axes, a 101 × 9 matrix was constructed by aggregating the sensor data [49]. The matrix consisted of 101 rows, representing individual samples, and 9 columns, representing sensor axes. Figure 6 illustrates the arrangement of the matrix.

3.3.4 Feature extraction

Feature extraction was performed from the datasets collected with MTw and ATD for ML techniques, which will be examined for activity classification performance.

The extracted features encompass the following [49, 51]:

- Minimum, maximum, mean, skewness, and kurtosis
- Five peak points of DFT (Discrete Fourier Transform)
- Frequency values
- Eleven values of the autocorrelation function

Consequently, 26 features were generated for each record, computed using the provided formulas below.

$$mean(d) : \mu = \frac{1}{N} \sum_{i=1}^{N} d_i \quad (2)$$

$$variance(d) : \sigma^2 = \frac{1}{N} \sum_{i=1}^{N} \left(d_i - \mu\right)^2 \quad (3)$$

$$skewness(d) = \frac{1}{N\sigma^3} \sum_{i=1}^{N} \left(d_i - \mu\right)^3 \quad (4)$$

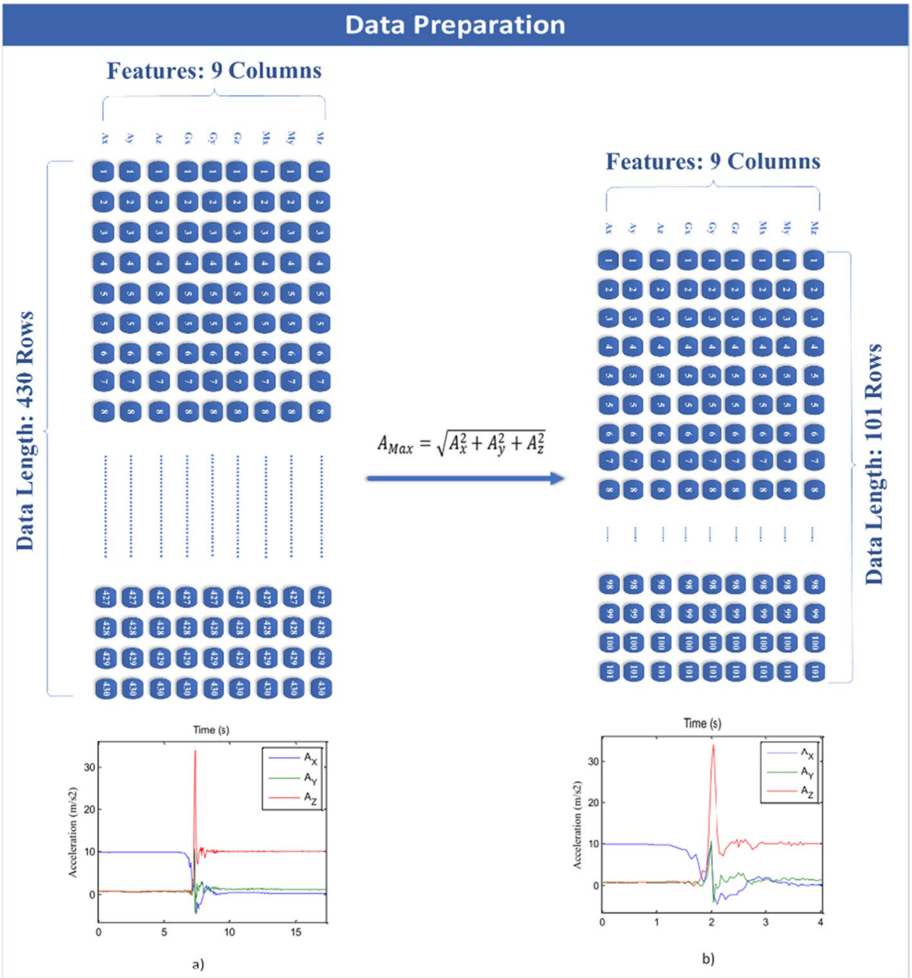

Fig. 6 Data Preprocessing Process. These two graphs belong to the first repetition of Activity 17 of subject 203. The data is collected from the sensor located on the waist. (**a**) 430 samples (more than 17 s of raw data collected at 25 Hz) are gathered, and (**b**) reduced to 101 samples (shortened to 4 s of data)

$$kurtosis(d) = \frac{1}{N\sigma^4} \sum_{i=1}^{N} (d_i - \mu)^4 \tag{5}$$

$$autocorrelation(d) : R_{ss}(\Delta) = \frac{1}{N-\Delta} \sum_{i=0}^{N-\Delta-1} \left(d_i - \mu\right)\left(d_{i-\Delta} - \mu\right), \Delta = 0,1,\dots,N-1 \tag{6}$$

$$\mathrm{DFT}(k) = \sum_{i=0}^{N-1} d_i e^{-\frac{j2\pi ki}{N}}, k = 0,1,\dots,N-1 \tag{7}$$

In this research, the performance of activity classification was examined using 72 different combinations. The study focused on the x, y, and z axes of sensor units for both MTw and ATD. Table 8 illustrates the seventy-two axis combinations for the sensor types.

Table 8 Sensor Type Axis Combinations. Y: Yes, N: No

Combinations	Accelerometer			Gyroscope			Magnetometer		
	Ax	Ay	Az	Gx	Gy	Gz	Mx	My	Mz
Ax	**Y**	N	N	N	N	N	N	N	N
Ay	N	**Y**	N	N	N	N	N	N	N
Az	N	N	**Y**	N	N	N	N	N	N
Gx	N	N	N	**Y**	N	N	N	N	N
Gy	N	N	N	N	**Y**	N	N	N	N
Gz	N	N	N	N	N	**Y**	N	N	N
Mx	N	N	N	N	N	N	**Y**	N	N
My	N	N	N	N	N	N	N	**Y**	N
Mz	N	N	N	N	N	N	N	N	**Y**
AxAy	**Y**	**Y**	N	N	N	N	N	N	N
AxAz	**Y**	N	**Y**	N	N	N	N	N	N
AxGx	**Y**	N	N	**Y**	N	N	N	N	N
AxGy	**Y**	N	N	N	**Y**	N	N	N	N
AxGz	**Y**	N	N	N	N	**Y**	N	N	N
AxMx	**Y**	N	N	N	N	N	**Y**	N	N
AxMy	**Y**	N	N	N	N	N	N	**Y**	N
AxMz	**Y**	N	N	N	N	N	N	N	**Y**
AyAz	N	**Y**	**Y**	N	N	N	N	N	N
AyGx	N	**Y**	N	**Y**	N	N	N	N	N
AyGy	N	**Y**	N	N	**Y**	N	N	N	N
AyGz	N	**Y**	N	N	N	**Y**	N	N	N
AyMx	N	**Y**	N	N	N	N	**Y**	N	N
AyMy	N	**Y**	N	N	N	N	N	**Y**	N
AyMz	N	**Y**	N	N	N	N	N	N	**Y**
AzGx	N	N	**Y**	**Y**	N	N	N	N	N
AzGy	N	N	**Y**	N	**Y**	N	N	N	N
AzGz	N	N	**Y**	N	N	**Y**	N	N	N
AzMx	N	N	**Y**	N	N	N	**Y**	N	N
AzMy	N	N	**Y**	N	N	N	N	**Y**	N
AzMz	N	N	**Y**	N	N	N	N	N	**Y**
GxGy	N	N	N	**Y**	**Y**	N	N	N	N
GxGz	N	N	N	**Y**	N	**Y**	N	N	N
GxMx	N	N	N	**Y**	N	N	**Y**	N	N
GxMy	N	N	N	**Y**	N	N	N	**Y**	N
GxMz	N	N	N	**Y**	N	N	N	N	**Y**
GyGz	N	N	N	N	**Y**	**Y**	N	N	N
GyMx	N	N	N	N	**Y**	N	**Y**	N	N
GyMy	N	N	N	N	**Y**	N	N	**Y**	N
GyMz	N	N	N	N	**Y**	N	N	N	**Y**
GzMx	N	N	N	N	N	**Y**	**Y**	N	N
GzMy	N	N	N	N	N	**Y**	N	**Y**	N
GzMz	N	N	N	N	N	**Y**	N	N	**Y**
MxMy	N	N	N	N	N	N	**Y**	**Y**	N
MxMz	N	N	N	N	N	N	**Y**	N	**Y**

Table 8 (continued)

MyMz	N	N	N	N	N	N	N	**Y**	**Y**
AxGxMx	**Y**	N	N	**Y**	N	N	**Y**	N	N
AxGxMy	**Y**	N	N	**Y**	N	N	N	**Y**	N
AxGxMz	**Y**	N	N	**Y**	N	N	N	N	**Y**
AxGyMx	**Y**	N	N	N	**Y**	N	**Y**	N	N
AxGyMy	**Y**	N	N	N	**Y**	N	N	**Y**	N
AxGyMz	**Y**	N	N	N	**Y**	N	N	N	**Y**
AxGzMx	**Y**	N	N	N	N	**Y**	**Y**	N	N
AxGzMy	**Y**	N	N	N	N	**Y**	N	**Y**	N
AxGzMz	**Y**	N	N	N	N	**Y**	N	N	**Y**
AyGxMx	N	**Y**	N	**Y**	N	N	**Y**	N	N
AyGxMy	N	**Y**	N	**Y**	N	N	N	**Y**	N
AyGxMz	N	**Y**	N	**Y**	N	N	N	N	**Y**
AyGyMx	N	**Y**	N	N	**Y**	N	**Y**	N	N
AyGyMy	N	**Y**	N	N	**Y**	N	N	**Y**	N
AyGyMz	N	**Y**	N	N	**Y**	N	N	N	**Y**
AyGzMx	N	**Y**	N	N	N	**Y**	**Y**	N	N
AyGzMy	N	**Y**	N	N	N	**Y**	N	**Y**	N
AyGzMz	N	**Y**	N	N	N	**Y**	N	N	**Y**
AzGxMx	N	N	**Y**	**Y**	N	N	**Y**	N	N
AzGxMy	N	N	**Y**	**Y**	N	N	N	**Y**	N
AzGxMz	N	N	**Y**	**Y**	N	N	N	N	**Y**
AzGyMx	N	N	**Y**	N	**Y**	N	**Y**	N	N
AzGyMy	N	N	**Y**	N	**Y**	N	N	**Y**	N
AzGyMz	N	N	**Y**	N	**Y**	N	N	N	**Y**
AzGzMx	N	N	**Y**	N	N	**Y**	**Y**	N	N
AzGzMy	N	N	**Y**	N	N	**Y**	N	**Y**	N
AzGzMz	N	N	**Y**	N	N	**Y**	N	N	**Y**

Within each sensor type and for each axis, the extracted feature vectors consist of 26 elements. These elements encompass accelerometer data (Ax, Ay, Az), gyroscope data (Gx, Gy, Gz), and magnetometer data (Mx, My, Mz). With this inference, the feature vector length for single-axis combinations is calculated as 26, for two-axis combinations it is 52 (26 $\times$ 2), and for three-axis combinations, it is 78 (26 $\times$ 3).

3.3.5 Usage of artificial intelligence techniques

In this section, artificial intelligence (AI) techniques used for activity recognition are discussed. These techniques are employed to extract valuable information from input data and identify an activity. Eleven ML algorithms were employed in this study to accurately classify activities. These algorithms were applied with the features extracted in Sect. 2.3.4 from raw data collected by the sensors.

Logistic Regression (LR) LR model was used to find the optimal decision boundary between these two classes, assuming that ADL and fall data are linearly separable [52]. In this research, the LR model was tested using two sets of hyperparameters. The first set consisted of three optimization algorithms: "newton-cg", "lbfgs", and "liblinear". The second set included five regularization parameter values: "100, 10, 1.0, 0.1, and 0.01".

k-Nearest Neighbors Algorithm (k-NN) *k*-NN is a supervised ML algorithm that can be used for classification and regression tasks. It works by identifying the *k* most similar data points in the training set to a new data point and assigning the new data point to the same class as the majority of its k nearest neighbors [53]. For fall detection and activity recognition, it considers the falling or activity state of each training instance. When a new instance comes, it predicts its class by looking at the nearest k training instances. The value of k is an important hyperparameter that affects performance.. In this study, distance metrics such as ('euclidean', 'manhattan', 'chebyshev', 'minkowski'), the number of neighbors (*k*) as [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29], and the power parameter (p) for the minkowski metric as [1, 2] were tested with three different hyperparameters for all sensor and axis combinations.

Naive Bayes (NB) NB is a probabilistic classifier that takes into account the assumption that each feature makes an independent and equal contribution to the target category [54]. The NB classifier was used to perform classification based on the probabilities obtained from the sensor data, under the assumption that each feature contributes independently and equally to the target category. In this study, a hyperparameter, var_smoothing, was tested. Values ranging from 0 to -9 with logarithmically equidistant intervals were generated by adding the maximum variance part of all features to ensure computational stability, and the model was tested with a single hyperparameter.

Decision Trees (DT) DT is a non-parametric and hierarchical technique that is commonly used for both regression and classification problems. DT splits data at each level of hierarchy, with two, three, or multiple branching nodes [55]. For fall detection and activity recognition, it builds a tree structure according to the features (sensor data, motion features, etc.) that determine falling and activity states. Its interpretability is an advantage. In this study, different hyperparameters for creating the model were tested, including maximum depth (max depth) as [5, 10, 20, 40, 80, 100], minimum samples required to split a node (min samples split) as [2, 5, 10, 20, 40, 80, 100, 200], minimum samples required at a leaf node (min samples leaf) as [5, 10, 20, 40, 80, 100], maximum features (max features) as ['auto', 'sqrt', 'log2'], and criteria for splitting (criterion) as "gini" or "entropy."

Linear Discriminant Analysis (LDA) LDA is a statistical method that uses a linear combination of features to distinguish between two or more groups. It is a multivariate method, meaning that it takes into account the relationships between multiple features [56]. It attempts to find the combinations of features that best separate falling and activity states. Dimensionality reduction can be useful, especially for high-dimensional data. In this research, The parameter being tested was the solver parameter, which included three options: "svd", "lsqr", and "eigen".

Support Vector Machines (SVMs) SVM stand as a prominent supervised learning algorithm, renowned for their ability to maximize the margin between decision boundaries established by supporting points. SVMs were initially developed to tackle non-linear classification tasks by employing the kernel method [57]. In the context of fall detection and activity recognition, this method identifies the optimal hyperplane that effectively distinguishes between falling and activity states. Its ability to handle non-linear problems stems from its utilization of kernel functions, which enable transformation into higher dimensional spaces. In this research, an academic tested Support Vector Machines (SVM) using three sets of hyperparameters. These hyperparameters included different kernel types such as linear, poly, rbf, and sigmoid. The regularization parameter values tested were 0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.3, 1.5, 1.7, and 2.0. Additionally, the polynomial kernel function degrees tested were 2, 3, 4, and 5.

Ensemble AdaBoost (EAB) EAB is an ensemble algorithm that assigns weights to examples in the dataset according to their ease or difficulty in classification and makes the algorithm pay more or less attention to them in generating subsequent models based on these weights [58]. It combines the predictions of weak learners by weighting them. It can perform well in complex and non-linear problems for ADL and fall classes. To evaluate the performance of the EAB model, a comprehensive grid search was conducted using two key hyperparameters: the number of trees (n_estimators) and the learning rate. The number of trees was varied across a range of values, including [10, 50, 100, 500, 1000, 1500], while the learning rate was explored across values of [0.001, 0.01, 0.1, 0.2, 0.3].

Ensemble Gradient Boosting (EGB) EGB algorithm trains a sequential series of decision trees, where each tree is trained on the residuals of the previous tree [59]. For fall detection and activity recognition, it attempts to create a stronger classifier by minimizing the errors of previous learners. It may be preferred for its ability to capture complex relationships. To evaluate the performance of the EGB model, a hyperparameter tuning experiment was conducted investigating four key parameters: the number of trees (n_estimators) as [10, 50, 100, 500, 1000, 1500], the maximum depth (max depth) as [5, 10, 20, 40, 80, 100], the minimum number of samples required to split a node (min samples split) as [2, 5, 10, 20, 40, 80, 100, 200], and the maximum features (max features) as ’auto’, ’square’, ’log2’.

Ensemble Random Forest (ERF) ERF algorithm builds a collection of decision trees by training them on different subsets of the data. The predictions made by each tree are then combined to make a final prediction [60]. It trains many decision trees from randomly selected subsets of features and then combines their predictions. Its robustness against overfitting and suitability for parallel processing are advantages. To optimize the performance of the ERF model, a hyperparameter tuning process was performed evaluating four key parameters: the number of trees (n_estimators) as [10, 50, 100, 500, 1000, 1500], the maximum depth (max_depth) as [5, 10, 20, 40, 80, 100], the minimum number of samples required to split a node (min_samples_split) as [2, 5, 10, 20, 40, 80, 100, 200], and the maximum features (max_features) as ’auto’, ’square’, ’log2’.

Ensemble Extra Tee (EET) EET is an extension of ERF, an ensemble learning model. EET has a lower probability of overfitting compared to ERF because it randomly selects the best feature to split a node with the corresponding value. EET aims to split a node with the highest feature value [61]. The algorithm employs a random forest approach, training numerous decision trees with randomized feature and split-point selection. The predictions from these individual trees are then aggregated to achieve robust fall detection and activity recognition capabilities. To optimize the performance of the EET model, a hyperparameter tuning process was performed evaluating four key parameters: the number of trees (n_estimators) as [10, 50, 100, 500, 1000, 1500], the maximum depth (max_depth) as [5, 10, 20, 40, 80, 100], the minimum number of samples required to split a node (min_samples_split) as [2, 5, 10, 20, 40, 80, 100, 200], and the maximum features (max_features) as 'auto', 'square', 'log2'.

Ensemble Bagging Classifier (EBC) EBC is an ensemble algorithm that trains a collection of classifiers on different subsets of the data and combines their predictions to make a final prediction [62]. For fall detection and activity recognition, it trains multiple weak learners and combines their predictions. Its robustness against overfitting and suitability for parallel processing are advantages. To assess the performance of the EBC model, a hyperparameter tuning experiment focusing was conducted on a single parameter, the number of trees (n_estimators), as [10, 50, 100, 500, 1000, 1500].

3.3.6 Evaluating the performance of artificial intelligence techniques

Evaluating the performance of artificial intelligence techniques is a crucial step because it determines how well ML algorithms classify. Therefore, it is essential to select and apply performance evaluation criteria accurately. The chosen metrics significantly influence how algorithm performance is assessed and comparisons are made. However, the effectiveness of the performance evaluation process is also of great importance. Rigorous evaluation of a developed classification model necessitates testing it on unseen data with known class labels to ensure accurate performance assessment. This provides a realistic assessment of how well the model will perform on new data.

In this study, the dataset is divided into two parts obtained with MTw, consisting of data from 14 participants, 10 volunteers (1800 samples), and 4 volunteers (720 samples), and data obtained with ATD from 30 individuals, consisting of 20 volunteers (900 samples) and 10 volunteers (450 samples). To test the models, data with 4 volunteers for MTw and 10 volunteers for ATD are set as the test set. This strategy effectively isolates the test set examples from the training process, precluding the emergence of biased and artificially inflated performance metrics.

To evaluate the classification performance of the models, the k-fold cross-validation method is employed. With this method, the training and validation sets are cyclically changed using a specified k value. In this study, a k value of 10 has been chosen, resulting in the dataset being divided into equal parts. Each part is utilized as a validation set, while the remaining segments constitute the training set. This process allows for the creation of distinct performance sets for each model. Evaluation of model performance on the validation set is achieved through averaging the obtained results. Subsequently, the trained models are deployed on a distinct test set composed of test volunteers, enabling a comprehensive assessment of their classification capabilities.

Accuracy is one of the most common metrics used to evaluate the classification performance of algorithms, as shown in Eq. 8. It is calculated by dividing the number of correct predictions by the total number of predictions. A confusion matrix is a tool that can be used to visualize the performance of a classification algorithm.

$$Acc(Accuracy) = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} x100 \tag{8}$$

In the equation, T_n represents true negatives, T_p denotes true positives, F_p signifies false positives, and F_n indicates false negatives. In summary, the symbols used for binary classification, specifically for distinguishing between falls and activities of daily living (ADL), are as follows:

T_p: True positive; actually a fall, correctly classified.
T_n: True negative; actually not a fall (no fall), correctly classified.
F_p: False positive; The system incorrectly identified an event as a fall when no fall actually occurred.
F_n: False negative; An instance where a fall event occurred but was erroneously classified as non-fall.

In addition to accuracy, other metrics that are commonly used to evaluate the performance of a classification algorithm are sensitivity and specificity.

Sensitivity (Se), also known as recall, is a measure of how well the algorithm identifies positive examples (Eq. 9). It is calculated by dividing the number of correctly classified positive examples (the number of falls that are correctly classified) by the total number of positive examples.

$$Se(Sensitivity) = \frac{T_p}{T_p + F_n} x100 \tag{9}$$

The number of non-falls that are correctly classified in all negative examples is called specificity (Sp). This is calculated using Eq. 10:

$$Sp(Specificity) = \frac{T_n}{T_n + F_p} x100 \tag{10}$$

4 Results

In this section, the performance outputs of the artificial intelligence models developed to detect the contextual relationship between falls and ADLs, collected with MTw and ATD, for highly accurate classification among sensor axis combinations, which is the starting point of the study, are examined. This section consists of two parts. In the first part, the classification performance of artificial intelligence models developed for the data collected with MTw and, in the second part, for the data collected with ATD, is compared, taking into account the sensor axis combinations.

A total of 72 different data formats, composed of Ax, Ay, Az, Gx, Gy, Gz, Mx, My, Mz sensor axes for both activity tracking devices (MTw and ATD), were trained with 11 ML algorithms. At the end of the training, seven sensor axis combinations, yielding the highest accuracy rates, are presented along with the algorithm pairs.

The achievements of the developed artificial intelligence models within the scope of examination and evaluation were assessed considering classification performance in multi-class classification (MTw—36 Activities and ATD—15 Activities) and binary classification (falls and ADLs). In the training set, the k-fold technique was applied with k = 10, and the average and standard deviation (std) values for the validation set were obtained with different hyperparameter arrangements through 10 repetitions. Models were created for each combination using the hyperparameter values that achieved the highest accuracy rates on the validation set. In the final stage, the developed models were tested on an unseen test set, leading to generalized models and performance values for both binary and multi-class classification.

4.1 Classification Performance Metrics (MTw)

4.1.1 Binary Classification (ADLs and Falls)

Within the scope of this investigation, 11 distinct ML algorithms were employed to classify falls and ADLs utilizing a binary classification approach. The hyperparameter values contributing to the performance of each model during the training process are presented in Table 9.

The analysis of sensor axis combinations in different data formats has been utilized to evaluate classification performance using various metrics. Metrics such as the confusion matrix, accuracy, sensitivity, and specificity were calculated, and the results for algorithm-combination pairs are presented in Table 10. In the examination conducted on the test set, the generalized performance of classifiers used with seven different axis combination types was assessed. Among the investigated classifiers, SVM algorithm exhibited the superior performance, achieving the highest accuracy rate when employing the AxGy-AxGxMz axis combinations. The accuracy rate of the model developed with AxGy and AxGxMz combinations was determined to be 99.17%. Furthermore, the primary goal for binary classifications is to accurately classify fall cases to the highest extent. Therefore, using the AxGy combination and SVM pair, 100% sensitivity was achieved, and all fall test data was accurately classified. It is noted that a more acceptable performance was obtained compared to the AxGy combination.

In the scope of this study, the computational requirements of models for binary and multiclass classifications were examined. Leveraging a computational platform equipped with an 8-core Intel Core i7 processor operating at 2.60 GHz, 16 GB of RAM, an Nvidia GeForce GTX 950 M GPU (4 GB GDDR3), and a Microsoft Windows 10 (64-bit) operating system, the preprocessing and classification tasks were efficiently executed.

Table 10 compares the computational requirements and training, validation, and testing times for different axis type combinations in binary classification problems. Regarding training time, the algorithm with the highest computation time in axis combinations is SVM with the AxGxMz combination, while the algorithm with the lowest computation time is k-NN with the AxAy combination. Concerning validation time, k-NN with the AxAy combination has the highest computation time, while SVM with the AxGy combination has the lowest computation time. Regarding testing time, k-NN with the

Table 9 Machine Learning Models and Hyperparameter Values in Binary Classification with the MTw Dataset. Comb.: Combinations

Models	Comb	Parameters		
SVM	Ax	kernel: rbf	C: 1	degree: 3
k-NN	AxAy	metric: minkowski	k:5	p:2
SVM	AxAz	kernel: rbf	C: 1	degree: 3
SVM	AxGx	kernel: rbf	C: 1	degree: 3
SVM	AxGy	kernel: rbf	C: 1	degree: 3
k-NN	AxGxMx	metric: minkowski	k:5	p:2
SVM	AxGxMz	kernel: rbf	C: 1	degree: 3

Table 10 Comparison of Machine Learning Models and Axis Combination Pairs in Binary Classification with the MTw Dataset

		SVM				*k*-NN				SVM				SVM			
		Ax				**AxAy**				**AxAz**				**AxGx**			
		Validation Set		**Test Set**		**Validation Set**		**Test Set**		**Validation Set**		**Test Set**		**Validation Set**		**Test Set**	
		Classified Event															
		Fall	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**
Real Event	**Fall**	1000	0	389	11	1000	0	394	6	999	1	399	1	1000	0	391	9
	ADL	1	799	1	319	2	798	9	311	0	800	10	310	1	799	0	320
(%)	***Acc***	99,94		*Acc*	98,61	99,89		*Acc*	97,78	99,94		*Acc*	98,47	99,94		*Acc*	99,03
	Std	0,17				0,22				0,17				0,17			
	Sp	99,9		99,75		99,8		97,25		100		96,88		99,9		100	
	Se	100		97,19		100		98,44		99,9		99,75		100		97,81	
		Calculation Time (s)															
Train		0,034				0,026				0,152					0,115		
Validate		0,006				0,035				0,014					0,012		
Test		0,010				0,075				0,043					0,032		

		SVM				*k*-NN				SVM			
		AxGy				**AxGxMx**				**AxGxMz**			
		Validation Set		**Test Set**		**Validation Set**		**Test Set**		**Validation Set**		**Test Set**	
		Classified Event											
		Fall	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**
Real Event	**Fall**	999	1	400	0	1000	0	384	16	1000	0	392	8
	ADL	0	800	6	314	1	799	4	316	1	799	0	320
(%)	***Acc***	99,94		*Acc*	99,17	99,94		*Acc*	97,5	99,94		*Acc*	99,17
	Std	0,17				1,74				2,9			
	Sp	100		98,12		99,9		98,75		99,9		100	
	Se	99,9		100		100		95,94		100		98,12	
		Calculation Time (s)											
Train		0,040				0,038				0,195			
Validate		0,005				0,033				0,016			
Test		0,011				0,080				0,050			

AxGxMx combination has the highest computation time, whereas SVM with the Ax combination has the lowest computation time.

4.1.2 Multiclass Classification (36 Activities)

Within the scope of the study, 36 activities have been classified through multiclass classification using 11 different ML algorithms. The hyperparameter values contributing to the success of each model during the training process are presented in Table 11.

In order to assess the classification performance of data formats composed of sensor axis combinations, accuracy metrics were employed. Table 12 illustrates the generalized performance of classifiers utilized with seven different axis combination types.

The highest accuracy rate was achieved with the AxGxMz combination and the EET algorithm. The accuracy rate of the model developed with this axis combination was determined to be 77.64%. When analyzing the data, it becomes evident that the AxGxMz data format, in conjunction with the EET algorithm, resulted in the highest accuracy rate for 36 different activities.

The EET model's classification performance of 36 activities in the test set is presented in Table 13 through the confusion matrix. In this matrix, rows represent the actual activities, while columns depict the classification results obtained by the model. Values on the diagonal indicate correct classification, whereas values off the diagonal represent misclassifications. For instance, in the matrix, the value of 2 at the intersection of row 4 (Squat and Stand Up) and column 13 (Sitting in the Air) indicates that the model classified activity 4 as 13. Within the diagonal of the confusion matrix, the values represent the model's classification accuracy for each activity. Limited or zero values off the diagonal demonstrate that the model did not confuse activities with each other.

Table 12 compares the computational requirements and training, validation, and testing times for classification algorithms in terms of axis type combinations for multiclass classification problems. When considering the training time, the combination AxGyMz—EET has the highest time requirement, while the combination AxMy—EET has the lowest time requirement. In terms of validation time, the combination AxGxMz—EET has the longest computation time, whereas the combination AxAz—SVM has the shortest computation time. For testing time, the combination AxGxMz—EET has the highest time requirement, and the combination AxMy—EET has the lowest time requirement.

Table 11 Machine Learning Models and Hyperparameter Values in Multiclass Classification with the MTw Dataset. Comb.: Combinations

Models	Comb	Parameters			
EET	AxAy	n_estimators:100	max_depth: 40	min_samples_split: 2	max_features:auto
SVM	AxAz	kernel: linear	C: 1,5	degree: 3	
EET	AxGx	n_estimators:500	max_depth: 20	min_samples_split:5	max_features: log2
EET	AxMy	n_estimators:50	max_depth: 40	min_samples_split:5	max_features:sqrt
EET	AxGxMy	n_estimators:1000	max_depth: 40	min_samples_split:20	max_features:auto
EET	AxGxMz	n_estimators:1500	max_depth: 40	min_samples_split:2	max_features:sqrt
EET	AxGyMy	n_estimators:1500	max_depth: 40	min_samples_split:2	max_features:log2

Table 12 Comparison of Machine Learning Models and Axis Combination Pairs in Multi-Class Classification Using the MTw Dataset

		EET			**SVM**			**EET**			**EET**		
		AxAy			**AxAz**			**AxGx**			**AxMy**		
		Validation Set	**Test Set**		**Validation Set**	**Test Set**		**Validation Set**	**Test Set**		**Validation Set**	**Test Set**	
(%)	*Acc*	92,39	*Acc*	75,97	91,28	*Acc*	75,42	91,28	*Acc*	77,08	91,94	*Acc*	72,92
	Std	5,82			6,97			5,63			5,18		
		Calculation Time (s)											
Train		0,308			1,453			1,116			0,138		
Validate		0,059			0,015			0,250			0,027		
Test		0,061			0,059			0,349			0,032		
		EET			**EET**			**EET**					
		AxGxMy			**AxGxMz**			**AxGyMy**					
		Doğrulama Seti	**Test Seti**		**Doğrulama Seti**	**Test Seti**		**Doğrulama Seti**			**Test Seti**		
(%)	*Acc*	91,61	*Acc*	75,42	91,22	*Acc*	77,64	91,78			*Acc*	76,11	
	Std	6,04			6,24			5,9					
		Calculation Time (s)											
Train		2,022			3,719			4,983					
Validate		0,420			1,484			1,303					
Test		0,548			1,731			1,714					

Table 13 Confusion Matrix Representation of the Classification Results for 36 Activities using the EET Model in the AxGxMz Data Format

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**	**32**	**33**	**34**	**35**	**36**
1	**13**	**5**	0	0	0	0	0	0	**2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	**16**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**4**
3	0	0	**15**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5**	0	0	0	0
4	0	**1**	0	**13**	0	0	0	0	0	0	0	0	**2**	0	0	**4**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	**15**	0	0	0	0	0	0	0	0	0	0	**5**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	**15**	0	0	0	0	0	0	0	0	**5**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	**3**	**5**	0	0	0	0	**7**	**3**	**2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	**15**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5**	0	0	0	0
9	0	0	0	0	0	0	**4**	**9**	**7**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	**3**	0	0	0	0	0	0	0	0	**17**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**11**	**4**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5**	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5**	**10**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5**	0	0	0
19	0	**2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**13**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5**
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	**14**	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**4**	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**2**	**2**	0	**1**	**4**	**6**	0	0	0	0	0	0	0	0	0	0	**2**	**2**	**1**	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**3**	**4**	0	0	**3**	**5**	0	0	0	0	0	0	0	0	0	0	**2**	0	**3**	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**1**	0	0	0	0	**13**	0	0	0	0	0	0	0	0	0	**3**	0	**2**	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	0	0	**19**	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0	0	0	0

Table 13 (continued)

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**	**32**	**33**	**34**	**35**	**36**
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**19**	0	**1**	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**4**	**16**	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**9**	**11**	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**9**	**11**	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0
36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**

4.2 Classification Performance Metrics (ATD)

4.2.1 Binary Classification (ADLs and Falls)

Within the scope of this investigation, fall and ADL classification was performed utilizing 11 distinct ML algorithms in a binary classification framework. The hyperparameter values contributing to the success of each model during the training process are presented in Table 14.

In order to examine the effects of axis combinations on classification performance, results obtained using the confusion matrix, accuracy, sensitivity, and specificity metrics are presented in Table 15..

Classifier models using seven different axis combinations were evaluated on the test data set. According to the results obtained, models developed with the Ax, AxAz, AxGy, AxMz, and AxGxMx axis combinations achieved the highest accuracy rates. Using the k-NN algorithm with these combinations, 100% accuracy, 100% sensitivity, and 100% specificity were achieved.

However, it is important to note a limitation of this study. The analysis revealed the situation where the model overfits the data and the results obtained on the test data set are higher than those on the validation data set. This may lead to low accuracy in classifying new data sets.

Additionally, the computational requirements of classification algorithms were compared in this research. The effects on training, validation, and testing times were examined. According to the results, the EGB algorithm with the AxAy axis combination had the highest training time, while the k-NN algorithm with the AxAz and AxGxMx axis combinations had the lowest training time. When looking at validation time, the k-NN algorithm with the Ax axis combination had the highest time, while the EGB algorithm with the AxAy axis combination had the lowest time. As for testing time, the k-NN algorithm with the AxMz axis combination had the highest time, and the EGB algorithm with the AxAy axis combination had the lowest time.

Table 14 Machine Learning Models and Hyperparameter Values in Binary Classification with the ATD Dataset. Comb.: Combinations

Models	Comb	Parameters			
k-NN	Ax	metric: minkowski	k:11	p:2	
EGB	AxAy	n_estimators: 100	max_depth: 3	min_samples_split:2	max_features: None
k-NN	AxAz	metric: minkowski	k:5	p:2	
k-NN	AxGx	metric: euclidean	k:3		
k-NN	AxGy	metric: euclidean	k:13		
k-NN	AxMz	metric: minkowski	k:11	p:2	
k-NN	AxGxMx	metric: euclidean	k:13		

Table 15 Comparison of Machine Learning Models and Axis Combination Pairs in Binary Classification with the ATD Dataset

		k-NN				**EGB**				**k-NN**				**k-NN**			
		Ax				**AxAy**				**AxAz**				**AxGx**			
		Validation Set		**Test Set**		**Validation Set**		**Test Set**		**Validation Set**		**Test Set**		**Validation Set**		**Test Set**	
		Classified Event															
		Fall	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**
Real Event	**Fall**	420	0	210	0	420	0	206	4	420	0	210	0	419	1	207	3
	ADL	2	478	0	240	2	478	1	239	2	478	0	240	2	478	0	240
(%)	***Acc***	99,78		*Acc*	100	99,78		*Acc*	98,89	99,78		*Acc*	100	99,67		*Acc*	99,33
	Std	0,67				0,67				0,67				0,71			
	Sp	99,58		100		99,58		99,58		99,58		100		99,58		100	
	Se	100		100		100		98,1		100		100		99,76		98,57	
		Calculation Time (s)															
Train		0,041				1,528				0,013				0,015			
Validate		0,063				0,002				0,028				0,022			
Test		0,058				0,004				0,051				0,044			

		k-NN				**k-NN**				**k-NN**			
		AxGy				**AxMz**				**AxGxMx**			
		Validation Set		**Test Set**		**Validation Set**		**Test Set**		**Validation Set**		**Test Set**	
		Classified Event											
		Fall	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**	**Fall**	**ADL**
Real Event	**Fall**	420	0	210	0	420	0	210	0	420	0	210	0
	ADL	1	479	0	240	2	478	0	240	3	477	0	240
(%)	***Acc***	99,89		*Acc*	100	99,78		*Acc*	100	99,67		*Acc*	100
	Std	0,33				0,67				1			
	Sp	99,79		100		99,58		100		99,38		100	
	Se	100		100		100		100		100		100	
		Calculation Time (s)											
Train		0,020				0,019				0,013			
Validate		0,022				0,023				0,017			
Test		0,038				0,066				0,036			

4.2.2 Multiclass Classification (15 Activities)

Within the scope of this study, 15 activities were compared using 11 different ML algorithms through multiclass classification. The hyperparameter values contributing to the success of each model during the training process are presented in Table 16.

To assess the classification performance of various data formats with different axis combinations, accuracy metrics were employed. The outcomes of the algorithm-combination pairs are displayed in Table 17. Generalized performance of classifiers using seven axis combinations was examined on the test set. Amongst the evaluated classifiers, the combination of AxAy axes yielded the superior accuracy rate. Specifically, the combination of AxAy sensors and the EGB algorithm achieved an accuracy rate of 94.00%.

Moreover, Table 18 presents the confusion matrix for axis combinations, providing a detailed evaluation of the classification performance for each activity type. When evaluating the classification performance of the EGB model on the test set for 15 activities, the confusion matrix presented in Table 18 is examined. Within the matrix, the numbers that are not located on the diagonal can be observed to be either smaller or equal to zero. A value of zero signifies that the developed model successfully distinguished between activities without any confusion.

Table 17 presents a comparison of the computational requirements for different ML algorithms when applied to multiclass classification problems and considering axis type combinations. The table provides information about the training, validation, and testing times required by each algorithm. Concerning training time, the algorithm with the highest computation time in axis combinations is EGB when combined with the AxGyMy axes. On the other hand, the algorithm with the lowest computation time is ERF when combined with the AxGxMx axes.When considering validation time, the algorithm with the highest computation time among the various axis combinations is EET with the AxGxMy combination. Conversely, ERF algorithm emerged as the most computationally efficient, exhibiting the lowest execution time when utilizing the AxGxMx data combination. In terms of testing time, the EGB algorithm displayed the greatest computational burden, particularly when combined with the AxGyMy axes.On the other hand, LR with the G axes has the lowest computation time.

5 Discussion

This study encompasses a comprehensive analysis of the classification performance of ML models developed for activity recognition and fall detection. The research's objective is to construct a well-structured dataset [20, 44, 63], and make the research conducted with this dataset comparable to others. A critical limitation addressed in this study is the paucity of diverse datasets utilized in current activity recognition and fall detection research. Many researchers have relied on datasets with a limited range of activities[23, 29, 30, 33, 35–37], some consisting solely of ADL data [22–24, 29–33, 38–41], a homogeneous pool of participants [25, 34–37], and insufficient information about the number of activity repetitions [29, 31, 33, 34, 37]. While these studies may report high classification accuracies for basic movements, their real-world applicability is questioned. Real-life scenarios are often more complex and noisy, posing challenges that these models might struggle to overcome, leading to potential misclassifications.

To address this limitation, a comprehensive dataset was constructed, encompassing a diverse spectrum of activities. This dataset comprises 36 activity types, including 20 fall events and 16 ADLs, and a subset of 15 activity types, consisting of 7 falls and 8 ADLs. Moreover, the dataset is characterized by an even representation of male and female participants, fostering gender balance and enhancing the generalizability of the research outcomes. This comprehensive approach enhances the dataset's ability to capture the inherent variability present in real-world settings, thereby improving the robustness and reliability of the developed models.

Another strength of this research lies in the comprehensive hyperparameter analyses conducted for both fall detection and activity recognition approaches. Meticulous parameter tuning, often overlooked in many studies, can lead to suboptimal model performance. By carefully optimizing the hyperparameters, the models have been ensured to operate at their full potential, maximizing their classification accuracy.

Furthermore, this study addresses a crucial aspect of sensor networks: power consumption optimization. By adopting an approach that minimizes power requirements and developing energy-harvesting methods, this research aims to create a simple, affordable, low-power, real-time, and long-lasting device for fall and ADL detection. This innovative approach addresses the environmental dependence issue, a significant limitation in existing solutions, and paves the way for more practical and sustainable solutions for individuals at risk of falling.

In addition to these strengths, the effectiveness of selecting appropriate axis combinations from datasets consisting of different activity types has been showcased. The results demonstrate that carefully curated models can achieve high classification accuracy, further solidifying the practical relevance of this research. To the best of our knowledge, no such a comprehensive study study has investigated sensor axis combination on activity recognition and fall detection (11 ML algorithms × 72 sensor axis combination on both of MTw and ATD = Total 1584 evaluate combinations).

In conclusion, this study addresses several critical limitations in the existing literature by constructing a diverse and comprehensive dataset, conducting rigorous hyperparameter analyses, and developing innovative solutions for power optimization and real-time operation. These contributions not only advance the field of activity recognition and fall detection but also pave the way for more robust, reliable, and practical solutions that can positively impact individuals' lives.

Table 16 Machine learning models and hyperparameter values in multiclass classification with the ATD Dataset. Comb.: Combinations

Models	Comb	Parameters			
EGB	AxAy	n_estimators:500	max_depth: 100	min_samples_split:20	max_features:log2
EET	AxGx	n_estimators:500	max_depth: 80	min_samples_split:5	max_features:log2
ERF	AxGxMx	n_estimators:100	max_depth: 40	min_samples_split:2	max_features:sqrt
EET	AxGxMy	n_estimators:1500	max_depth: 40	min_samples_split:2	max_features:log2
EGB	AxGyMy	n_estimators:1000	max_depth: 40	min_samples_split:20	max_features:sqrt
EGB	AxGzMy	n_estimators:1000	max_depth: 5	min_samples_split:10	max_features:sqrt
EGB	AxGzMz	n_estimators:1000	max_depth: 5	min_samples_split:10	max_features:sqrt

Table 17 Comparison of machine learning models and axis combination pairs in multiclass classification with the ATD Dataset

		EGB			**EET**					**ERF**					**EET**				
		AxAy			**AxGx**					**AxGxMx**					**AxGxMy**				
		Validation Set	**Test Set**		**Validation Set**		**Test Set**			**Validation Set**			**Test Set**		**Validation Set**	**Test Set**			
(%)	*Acc*	96	*Acc*	94	94,89	*Acc*		90,22	94,56				*Acc*	89,33	94,56		*Acc*		92,22
	Std	2			3,45				3,8						3,35				
		Calculation Time (s)																	
Train		48,851			1,030					0,223					2,072				
Validate		0,041			0,124					0,024					0,267				
Test		0,275			0,203					0,035					0,444				
		EGB			**EGB**										**EGB**				
		AxGyMy			**AxGzMy**										**AxGzMz**				
		Validation Set	**Test Set**		**Validation Set**						**Test Set**				**Validation Set**			**Test Set**	
(%)	*Acc*	94,22	*Acc*	89,56	94,22						*Acc*	91,33			94,22			*Acc*	89,78
	Std	4			3,21										3,87				
		Calculation Time (s)																	
Train		129,333			53,08										48,258				
Validate		0,101			0,039										0,028				
Test		0,629			0,154										0,129				

Table 18 Confusion Matrix Representation of Classification Results for 15 Activities Using the EGB Model in the AxAy Data Format

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	**29**	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0
2	0	**30**	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	**30**	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	**30**	0	0	0	0	0	0	0	0	0	0	0
5	**2**	**3**	0	0	**25**	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	**16**	**14**	0	0	0	0	0	0	0	0
7	0	0	0	0	0	**2**	**28**	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	**30**	0	0	0	0	0	0	0
9	0	0	0	0	**1**	0	0	0	**28**	0	**1**	0	0	0	0
10	0	0	0	0	0	0	0	0	0	**30**	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	**30**	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	**30**	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	**30**	0	0
14	0	0	0	0	0	0	0	0	0	0	**3**	0	0	**27**	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**30**

6 Conclusion and recommendations

In this study, two datasets were utilized and analyzed using various classification algorithms. Seven axis combinations that provided the highest performance were selected, and the models' performance on the test set was examined. High success rates were achieved in both binary classification (falls and ADLs) and multiclass classification (MTw—36 Activities and ATD—15 Activities).

When evaluating the impact of axis combinations on binary classification, it is challenging to draw a definitive conclusion. However, in multiclass classification problems, it was observed that the axis combinations had an effect on classification performance. As the number of axis types decreased, a general decrease in classification accuracy rates was observed. The classifiers chosen have been validated to successfully differentiate between falls and activities of daily living (ADLs) with high accuracy.

This study presents a low-power sensor network with energy harvesting capabilities for real-world fall detection applications. The proposed system demonstrates versatility across diverse settings, including elderly care in remote areas, home monitoring, workplace safety, fall prevention in rehabilitation centers, and personal fall detection. The use cases highlight the potential of this approach in safeguarding individuals at risk of falls.

Future research should prioritize incorporating practical use case data into the design and development process. Pilot studies within these scenarios can provide valuable data on several key aspects:

- User-Centered Evaluation: Here, pilot studies can assess user comfort, acceptance, and the system's effectiveness in real-world settings.
- Data Expansion and Analysis: Implementing data collection procedures to enhance data quantity and variability is essential. Expanding the dataset size can further investigate model stability and explore the effects of various parameters.Integration with Existing Systems: Exploring seamless integration with other health monitoring wearables can facilitate comprehensive health management.

- Sensor Performance: Evaluation should focus on the accuracy and sensitivity of fall detection algorithms.
- User Acceptance: Assessing user comfort, compliance, and overall satisfaction with the wearable sensor system is crucial.
- System Effectiveness: Measuring the impact of the system on fall prevention rates and intervention response times will provide valuable insights.

By integrating these findings, researchers can refine the sensor network architecture, optimize energy consumption, and enhance user experience.

Furthermore, body energy harvesting presents an exciting opportunity for extending battery life and achieving true long-term system autonomy. Collaborations with commercial entities can facilitate the development of wearable devices that harvest energy from human movement, further reducing reliance on external power sources.

This integrated approach, informed by practical use case data and harnessing body energy, paves the way for a lightweight, cost-effective, and perpetually wearable fall detection system. Such a device, seamlessly integrated with other health monitoring systems, has the potential to become a ubiquitous companion. This empowers individuals to manage their health and safety proactively, ultimately contributing to a significant improvement in overall quality of life.

Author contributions Erhan Kavuncuoğlu implemented the ML classifiers and contributed to the writing and editing of the manuscript. Esma Uzunhisarcıklı reviewed the manuscript and provided suggestions for corrections. Ahmet Turan Özdemir supervised the study, coordinated the experiments, offered insights on ML techniques, and made significant contributions to the writing and editing of the manuscript.

Funding This work was supported by the Erciyes University Scientific Research Project Coordination Department under Grant Number FDK-2018–8329.

Availability of data and material The dataset used in the work is shared as open source from the link below.

- https://archive.ics.uci.edu/ml/datasets/Simulated+Falls+and+Daily+Living+Activities+Data+Set (Mtw Dataset)
- https://drive.google.com/file/d/1ItVeUknM6Er7KIuBKHQgezHlObS65cVk/view?usp=sharing (ATD Dataset)

The classification performance of activities for other algorithms is shared at the link below for researchers to review.

- https://drive.google.com/drive/folders/1Jl8N_t6VzfI3AeZrpC9DTfE3UZ0GyrjG?usp=sharing (MTw Dataset)

- **https://drive.google.com/drive/folders/1sh1p4ch5m3MkC5EqKIZSpAmyVqthFR6b?usp=sharing (ATD Dataset)Code availability** Not applicable.

Declarations

Competing interests The authors declare no conflict of interest.

References

1. World Report on Ageing and Health (2015) World Health Organization, Geneva, Switzerland
2. United Nations, Department of Economic and Social Affairs, Population Division (2019) World Population Prospects 2019: Highlights
3. Carmeli E, Imam B, Merrick J (2016) Assistive technology and older adults. In: Health Care for People with Intellectual and Developmental Disabilities Across the Lifespan, pp 1465–1471. https://doi.org/10.1007/978-3-319-18096-0_117
4. Augimeri A, Fortino G, Rege MR et al (2010) A cooperative approach for handshake detection based on body sensor networks. In: Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics. pp 281–288. https://doi.org/10.1109/ICSMC.2010.5641696
5. Huang JY, Tsai CH (2007) A wearable computing environment for the security of a large-scale factory. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol LNCS 4551, pp 1113–1122. https://doi.org/10.1007/978-3-540-73107-8_122
6. Zhou B, Sundholm M, Cheng J et al (2017) Measuring muscle activities during gym exercises with textile pressure mapping sensors. Pervasive Mob Comput 38:331–345. https://doi.org/10.1016/j.pmcj.2016.08.015
7. Terada T, Tanaka K (2010) A framework for constructing entertainment contents using flash and wearable sensors. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol LNCS 6243, pp 334–341. https://doi.org/10.1007/978-3-642-15399-0_35
8. Aarthi S, Juliet S (2021) A comprehensive study on Human Activity Recognition. 2021 3rd Int Conf Signal Process Commun ICPSC 2021 59–63. https://doi.org/10.1109/ICSPC51351.2021.9451759
9. Usmani S, Saboor A, Haris M et al (2021) Latest Research Trends in Fall Detection and Prevention Using Machine Learning: A Systematic Review. Sens 2021 Vol 21 Page 5134 21:5134. https://doi.org/10.3390/S21155134
10. Singh A, Rehman SU, Yongchareon S, Chong PHJ (2020) Sensor Technologies for Fall Detection Systems: A Review. IEEE Sens J 20:6889–6919. https://doi.org/10.1109/JSEN.2020.2976554
11. Ramachandran A, Karuppiah A (2020) A Survey on Recent Advances in Wearable Fall Detection Systems. BioMed Res Int 2020:. https://doi.org/10.1155/2020/2167160
12. Wang Y, Cang S, Yu H (2019) A survey on wearable sensor modality centred human activity recognition in health care. Expert Syst Appl 137:167–190. https://doi.org/10.1016/j.eswa.2019.04.057
13. Salman Khan M, Yu M, Feng P et al (2015) An unsupervised acoustic fall detection system using source separation for sound interference suppression. Signal Process 110:199–210. https://doi.org/10.1016/j.sigpro.2014.08.021
14. Redmond SJ, Zhang Z, Narayanan MR, Lovell NH (2014) Pilot evaluation of an unobtrusive system to detect falls at nighttime. In: 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014, pp 1756–1759. https://doi.org/10.1109/EMBC.2014.6943948
15. Litvak D, Zigel Y, Gannot I (2009) Fall detection of elderly through floor vibrations and sound. In: Conference proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference, pp 4632–4635. https://doi.org/10.1109/IEMBS.2008.4650245
16. De Miguel K, Brunete A, Hernando M, Gambao E (2017) Home camera-based fall detection system for the elderly. Sens Switz 17:1–21. https://doi.org/10.3390/s17122864
17. Cheng L, Zhao A, Wang K et al (2020) Activity recognition and localization based on UWB indoor positioning system and machine learning. In: 11th Annual IEEE Information Technology, Electronics and Mobile Communication Conference, IEMCON 2020. Institute of Electrical and Electronics Engineers Inc., pp 528–533. https://doi.org/10.1109/IEMCON51383.2020.9284937
18. Koshmak G, Loutfi A, Linden M (2016) Challenges and issues in multisensor fusion approach for fall detection: Review paper. J Sens 2016:. https://doi.org/10.1155/2016/6931789
19. Chaccour K, Darazi R, El Hassani AH, Andres E (2017) From Fall Detection to Fall Prevention: A Generic Classification of Fall-Related Systems. IEEE Sens J 17:812–822. https://doi.org/10.1109/JSEN.2016.2628099
20. Noury N, Rumeau P, Bourke AK et al (2008) A proposal for the classification and evaluation of fall detectors. IRBM 29:340–349. https://doi.org/10.1016/j.irbm.2008.08.002
21. Rasheed MB, Javaid N, Alghamdi TA et al (2015) Evaluation of human activity recognition and fall detection using android phone. In: Proceedings - International Conference on Advanced Information

Networking and Applications, AINA. Institute of Electrical and Electronics Engineers Inc., pp 163–170. https://doi.org/10.1109/AINA.2015.181
22. Reiss A, Stricker D (2012) Introducing a new benchmarked dataset for activity monitoring. In: Proceedings - International Symposium on Wearable Computers, ISWC. IEEE, pp 108–109. https://doi.org/10.1109/ISWC.2012.13
23. Anguita D, Ghio A, Oneto L et al (2013) A public domain dataset for human activity recognition using smartphones. In: Proc. European Symp. Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013), Bruges, Belgium
24. Memis G, Sert M (2018) The effectiveness of feature selection methods on physical activity recognition. In: 26th IEEE Signal Processing and Communications Applications Conference, SIU 2018. Institute of Electrical and Electronics Engineers Inc., pp 1–4. https://doi.org/10.1109/SIU.2018.8404406
25. Chatzaki C, Pediaditis M, Vavoulas G, Tsiknakis M (2017) Human daily activity and fall recognition using a smartphone's acceleration sensor. In: Communications in Computer and Information Science. Springer Verlag, pp 100–118. https://doi.org/10.1007/978-3-319-62704-5_7
26. Martínez-Villaseñor L, Ponce H, Brieva J, et al (2019) UP-Fall Detection Dataset: A Multimodal Approach. Sens 2019 Vol 19 Page 1988 19:1988. https://doi.org/10.3390/S19091988
27. Ponce H, Martínez-Villaseñor L (2020) Approaching Fall Classification Using the UP-Fall Detection Dataset: Analysis and Results from an International Competition. Stud Syst Decis Control 273:121–133. https://doi.org/10.1007/978-3-030-38748-8_6
28. Ponce H, Martínez-Villaseñor L, Brieva J, Moya-Albor E (2020) Challenges and Trends in Multimodal Fall Detection for Healthcare. 273:. https://doi.org/10.1007/978-3-030-38748-8
29. Buber E, Guvensan AM (2014) Discriminative time-domain features for activity recognition on a mobile phone. In: IEEE ISSNIP 2014 - 2014 IEEE 9th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Conference Proceedings. IEEE Computer Society. https://doi.org/10.1109/ISSNIP.2014.6827651
30. Dernbach S, Das B, Krishnan NC, et al (2012) Simple and complex activity recognition through smart phones. In: Proceedings - 8th International Conference on Intelligent Environments, IE 2012, pp 214–221. https://doi.org/10.1109/IE.2012.39
31. Anjum A, Ilyas MU (2013) Activity recognition using smartphone sensors. In: 2013 IEEE 10th Consumer Communications and Networking Conference, CCNC 2013, pp 914–919. https://doi.org/10.1109/CCNC.2013.6488584
32. Saputri TRD, Khan AM, Lee S-W (2014) User-Independent Activity Recognition via Three-Stage GA-Based Feature Selection. Int J Distrib Sens Netw 10:706287. https://doi.org/10.1155/2014/706287
33. Bayat A, Pomplun M, Tran DA (2014) A study on human activity recognition using accelerometer data from smartphones. In: Procedia Computer Science. Elsevier B.V., pp 450–457. https://doi.org/10.1016/j.procs.2014.07.009
34. Figueiredo IN, Leal C, Pinto L et al (2016) Exploring smartphone sensors for fall detection. MUX J Mob User Exp 5:1–17. https://doi.org/10.1186/s13678-016-0004-1
35. Zhao Z, Chen Y, Wang S, Chen Z (2012) FallAlarm: Smart phone based fall detecting and positioning system. In: Procedia Computer Science. Elsevier B.V., pp 617–624. https://doi.org/10.1016/j.procs.2012.06.079
36. Albert MV, Kording K, Herrmann M, Jayaraman A (2012) Fall classification by machine learning using mobile phones. PLoS ONE 7:e36556. https://doi.org/10.1371/journal.pone.0036556
37. Kansiz AO, Guvensan MA, Turkmen HI (2013) Selection of time-domain features for fall detection based on supervised learning. In: Lecture Notes in Engineering and Computer Science, vol 2, pp 796–801
38. Mehrang S, Pietila J, Tolonen J et al (2017) Human activity recognition using a single optical heart rate monitoring wristband equipped with triaxial accelerometer. In: IFMBE Proceedings, pp 587–590. https://doi.org/10.1007/978-981-10-5122-7_147
39. Pavey TG, Gilson ND, Gomersall SR et al (2017) Field evaluation of a random forest activity classifier for wrist-worn accelerometer data. J Sci Med Sport 20:75–80. https://doi.org/10.1016/J.JSAMS.2016.06.003
40. Hsu YL, Lin SL, Chou PH, et al (2017) Application of nonparametric weighted feature extraction for an inertial-signal-based human activity recognition system. Proc 2017 IEEE Int Conf Appl Syst Innov Appl Syst Innov Mod Technol ICASI 2017 1718–1720. https://doi.org/10.1109/ICASI.2017.7988270
41. Sok P, Xiao T, Azeze Y et al (2018) Activity recognition for incomplete spinal cord injury subjects using hidden markov models. IEEE Sens J 18:6369–6374. https://doi.org/10.1109/JSEN.2018.2845749
42. (2022) ActiGraph wGT3X-BT | ActiGraph. https://actigraphcorp.com/actigraph-wgt3x-bt/. Accessed 2024

43. (2024) Xsens Technologies B.V., Enschede, the Netherlands, MTw awinda user manual and technical documentation. Available online: https://www.xsens.com
44. Abbate S, Avvenuti M, Corsini P et al (2010) Monitoring of human movements for fall detection and activities recognition in elderly care using wireless sensor network: A Survey. In: Wireless Sensor Networks: Application-Centric Design. https://doi.org/10.5772/13802
45. Antonsson EK, Mann RW (1985) The frequency content of gait. J Biomech 18:39–47. https://doi.org/10.1016/0021-9290(85)90043-0
46. Özdemir AT, Barshan B (2014) Detecting falls with wearable sensors using machine learning techniques. Sens Switz 14:10691–10708. https://doi.org/10.3390/s140610691
47. Özdemir AT (2016) An analysis on sensor locations of the human body for wearable fall detection devices: Principles and practice. Sens Switz 16:s16081161. https://doi.org/10.3390/s16081161
48. Ntanasis P, Pippa E, Özdemir AT et al (2017) Investigation of sensor placement for accurate fall detection. Lect Notes Inst Comput Sci Soc-Inform Telecommun Eng LNICST 192:225–232. https://doi.org/10.1007/978-3-319-58877-3_30
49. Kavuncuoğlu E, Uzunhisarcıklı E, Barshan B, Özdemir AT (2021) Investigating the performance of wearable motion sensors on recognizing falls and daily activities via machine learning. Digit Signal Process 103365. https://doi.org/10.1016/J.DSP.2021.103365
50. Uzunhisarcıklı E, Kavuncuoğlu E, Özdemir AT (2022) Investigating classification performance of hybrid deep learning and machine learning architectures on activity recognition. Comput Intell 38:1402–1449. https://doi.org/10.1111/coin.12517
51. Pippa E, Zacharaki EI, Özdemir AT et al (2018) Global vs local classification models for multi-sensor data fusion. In: ACM International Conference Proceeding Series, pp 1–5. https://doi.org/10.1145/3200947.3201034
52. Gudivada VN, Irfan MT, Fathi E, Rao DL (2016) Cognitive Analytics: Going Beyond Big Data Analytics and Machine Learning. Handb Stat 35:169–205. https://doi.org/10.1016/BS.HOST.2016.07.010
53. Duda RO, Hart PE, Stork DG (2001) Pattern classification, 2nd edn. John Wiley & Sons Inc, New York, NY, USA
54. Krishnan S (2021) Machine learning for biomedical signal analysis. Biomed Signal Anal Connect Healthc 223–264. https://doi.org/10.1016/B978-0-12-813086-5.00006-2
55. (2020) Classification and Regression Trees - Leo Breiman, Jerome Friedman, Charles J. Stone, R.A. Olshen - Google Kitaplar. https://books.google.com.tr/books/about/Classification_and_Regression_Trees.html?id=JwQx-WOmSyQC&redir_esc=y. Accessed 4 Dec 2020
56. Dohnálek P, Gajdoš P, Peterek T, Snášel V (2013) An overview of classification techniques for human activity recognition. Vibroengineering PROCEDIA 2:117–122
57. Sain SR, Vapnik VN (2006) The Nature of Statistical Learning Theory, 2nd edn., Springer. https://doi.org/10.1007/978-1-4757-3264-1
58. (2020) Introduction to Machine Learning - Ethem Alpaydin - Google Kitaplar. https://books.google.com.tr/books?hl=tr&lr=&id=tZnSDwAAQBAJ&oi=fnd&pg=PR7&ots=F3RW617tze&sig=dT_gV1jWla8rcWJNOb0AKER_9qw&redir_esc=y#v=onepage&q&f=false. Accessed 6 Dec 2020
59. Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189–1232. https://doi.org/10.1214/aos/1013203451
60. Cutler A, Cutler DR, Stevens JR (2012) Random Forests. In: Zhang C, Ma Y (eds) Ensemble Machine Learning. Springer, New York, NY. https://doi.org/10.1007/978-1-4419-9326-7_5
61. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Mach Learn 63:3–42. https://doi.org/10.1007/s10994-006-6226-1
62. John V, Liu Z, Guo C et al (2016) Real-time lane estimation using deep features and extra trees regression. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Verlag, pp 721–733. https://doi.org/10.1007/978-3-319-29451-3_57
63. Noury N, Fleury A, Rumeau P et al (2007) Fall detection - Principles and methods. In: Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings, pp 1663–1666. https://doi.org/10.1109/IEMBS.2007.4352627

Authors and Affiliations

Erhan Kavuncuoğlu[1] · **Ahmet Turan Özdemir**[2] · **Esma Uzunhisarcıklı**[3]

✉ Erhan Kavuncuoğlu
ekavuncuoglu@cumhuriyet.edu.tr

Ahmet Turan Özdemir
aturan@erciyes.edu.tr

Esma Uzunhisarcıklı
esma.uzunhisarcikli@bozok.edu.tr

1 Department of Computer Technology, Gemerek Vocational School, Cumhuriyet University, Sivas TR-58840, Turkey

2 Department of Electrical and Electronics Engineering, Engineering Faculty, Erciyes University, Kayseri TR-38039, Turkey

3 Department of Electrical and Electronics Engineering, Yozgat Bozok University, Engineering Faculty, Yozgat TR-66900, Turkey