Wireless Personal Communications (2021) 120:1593–1635
https://doi.org/10.1007/s11277-021-08525-w

A Survey of Deep Learning Based Models for Human Activity Recognition

Nida Saddaf Khan[1] · Muhammad Sayeed Ghani[1]

Accepted: 13 April 2021 / Published online: 7 May 2021

Abstract
Human Activity Recognition (HAR) is a process of recognizing human activities automatically based on streaming data obtained from various sensors, such as, inertial sensors, physiological sensors, location sensors, camera, time and many more environmental sensors. HAR has proven to be beneficial in various fields of study especially in healthcare, aged-care, ambient living, personal care, social science, rehabilitation engineering and many other domains. Due to the recent advancements in computing power, deep learning-based algorithms have become most effective and efficient choice of algorithms for recognizing and solving HAR problems. In this survey, we categorize recent research work with respect to various factors and measures to investigate the recent trends in HAR using deep learning algorithms. The articles are analyzed in various aspects, such as those related to HAR, time series analysis, machine learning models, methods of dataset creation, and use of various other new trends such as transfer learning, active learning, etc.

Keywords Sensors applications · Human activity recognition · Sensor data analytics · Deep learning · Sensors · Human activity recognition datasets · HAR

1 Introduction

Human Activity Recognition (HAR) is a process of recognizing the physical actions of agents involved in performing an activity. When the task of recording activities is performed by sensors these are referred to as sensor-based activity recognition. These sensors could be of many types which are broadly categorized as vision-based and non-vision-based sensors. In the former type cameras are used as sensors and in the latter type, motion, physiological, sound and other sensors are used. A variety of research has been conducted in vision-based activity recognition tasks [1, 2]. However, these approaches have the critical problem of users' privacy as well as a greater computational complexity of modelling

✉ Nida Saddaf Khan
nskhan@iba.edu.pk

Muhammad Sayeed Ghani
sghani@iba.edu.pk

1 Telecommunication Research Lab, Department of Computer Science, Institute of Business Administration, Garden/ Kayani Shaheed Road, Karachi 74400, Pakistan

videos or still-images. On the other hand, non-vision-based approaches have gained greater acceptability in the research community due to the extensive developments in pervasive computing and sensor technologies [3]. They offer comparable recognition performance at a lower computational cost with privacy also ensured. Furthermore, with the development of wearable sensors the constraint of a fixed environment and stationary-setting limitations are alleviated as well which is often suffered by cameras [4]. These reasons have made non-vision-based approaches more powerful and effective in HAR. Hence, in this paper our focus is only on non-vision-based HAR and will be referred to as sensor-based HAR or simply HAR.

The process of HAR could be divided in three stages: stage A Data Collection Process, stage B machine learning (ML) based HAR process and stage C Framework support as shown in Fig. 1. Each stage is comprised of many steps for its performance. Stage A involves the tasks related with the process of dataset creation such as identifying the set of activities for which the dataset is to be created, the identification of number and type of sensors, devices and subjects, position of sensors, orientation of devices and other features of data. Some researchers create their own dataset according to the need of their research or the requirement of the application domain such as [5]–[8]. Majority of the researchers utilize the already existing datasets such as [4, 9–11].

Stage B of this process is the most widely researched area where the researchers present their ML based algorithms to recognize the activities from the required datasets. The emphasis is on the development of smart algorithms which can perform the recognition task with greater accuracy and lesser complexity. This stage involves the task of feature extraction and selection, type of model selection, evaluation techniques and metrics depiction etc.

Lastly, stage C is the Framework support which is a step ahead of dataset and model creation. Most of the research's end goal is to produce a well performing algorithm to recognize the under-consideration activities. But lately, the researchers have also started showing interest in utilizing the recognition in order to form an automated system. By which they can fully assess the usefulness of their technique in a real-world situation. For example, a medical support system which can not only recognize the suicidal intentions of a psychological patient but can also alert the authorities about the situation in runtime to avoid any mishap to happen.

Deep learning (DL) which is also referred as hierarchical learning or deep structured learning is based on the concept of Artificial Neural Network (ANN). Such models have been applied successfully in the fields of speech recognition, audio recognition, computer vision, natural language processing (NLP), drug discovery, social network filtering, bioinformatics, board game programs, where their performances are reported either superior or comparable to other algorithms [12–14]. With the development of processing and storing capabilities, it has become possible to use DL techniques successfully in dynamic and

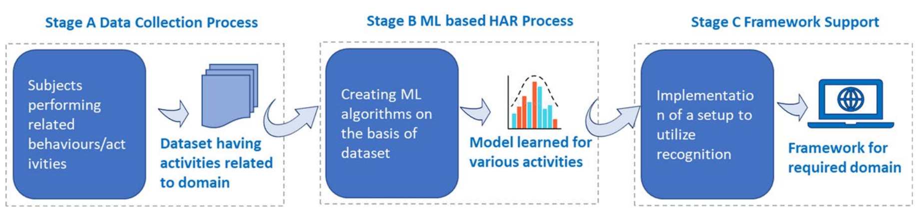

Fig. 1 The process of HAR

complex problems such as activity recognition than traditional ML techniques. This is due to the following limitations:

- The performance of many approaches rely on statistical features [15]. Some of the studies have used Fast Fourier Transformation (FFT) in frequency domain to extract features manually from signal sequences [16].
- The design and selection of manual features require domain knowledge of the system too [17]. The better the acquisition of domain knowledge the higher the performance of the model. But DL algorithms does not require to have domain knowledge and they extract the features which maps the signal data with the activity-class.
- By using conventional approaches, the features selection applied to sensors' data, can provide shallow features based on statistical or signal-processing specific features. These features have limitation in their performance to recognize the activities, as they are reported to be useful in identifying low-level or simple activities, but are not effective in dealing with high-level or complex activities [3].

Deep learning can overcome these limitations by automatically learning the features from raw sensors' data without the need of any feature engineering and domain knowledge. It can extract the high-level representation in the deep layers, making it suitable to be used for complex and dynamic scenarios. Due to these capabilities, DL structures have been used to extract the features from raw data and learn the complex and dynamic temporal relationships from time series generated by motion sensors. This field has shown rapid development in the field of HAR in past decade and its popularity in research community is continuously increasing. In [18], the trend of usage of traditional ML approaches and DL approaches have been shown, where it can be seen easily that after the year of 2013, the DL based approaches are showing increase in their usage curve and they raise further above the traditional ML algorithms usage curve.

This survey presents a detailed review of deep learning-based algorithms in terms of various rarely discussed technical aspects which are classified under the features of time series, machine learning, dataset, HAR, and other recent or special features. We believe the analysis is unique in its structure and provides a comprehensive guideline regarding implementation, recent trends and developments in the field, and will enable researchers to assess state-of-the-art techniques from a variety of aspects.

2 Related Work

There are some research surveys where human activity recognition has been studied from different aspects. These include the following categories of analysis: by specific methods [3]; by types of algorithms [19], by type of sensor [18, 20, 21]; using multi-sensory fusion [22], by type of activity (complex activity) [23, 24] or the type of device [25], whereas others have analysed HAR in more general way [26, 27].

In [3], the research articles are reviewed by following three aspects: application domain, modality of sensors and types of deep models. The survey summarized various models and datasets, but it lacks the analysis of other important and significant aspects of HAR such as technical details of type of activities, data processing, and evaluation techniques for various deep learning-based models. Furthermore, it does not analyse the datasets of HAR based on position and orientation of sensors.

Another research survey where the articles on AR are reviewed from the point of new trends in machine learning, i.e., active learning, deep learning, transfer learning and semantics [19]. This survey presents an overall analysis of research studies on above stated aspects and characterizes the studies on these bases, but it lacks other aspects which are related to implementation of these algorithms.

Another survey [18] presented a survey for HAR techniques using accelerometer sensor, and categorized them on the basis of types of algorithms. Foremost, the algorithms are broadly categorised as deep learning and traditional machine learning based algorithms. Forty-eight research articles are further summarized based on types of activities, classifiers, devices and recognition accuracies. Similarly, [20] presented a survey for HAR where the scope is limited to analyse the articles of accelerometer only. This survey compared the articles by sample rate, window size, and percentage of overlap for time series and presented a review on certain features extraction and the factors which affect HAR using accelerometer. The research studies [18] and [20] were based on one type of sensor only i.e., Accelerometer, as it is considered to be the most widely used sensor for non-vision based HAR. But there are other sensors such as gyroscope and magnetometer which have shown promising results when combined with Accelerometer and their role should be discussed in the creation of an efficient HAR algorithm. There is another research study [21] which has surveyed the field of HAR by using one type of sensor, i.e., radar. It highlighted the significance of deep learning-based models in learning the patterns of activities from various types of radars. The survey is useful but again limited in its scope to one type of sensor which is rarely available in common wearable devices such as smartphones and smartwatches. On the other hand, there exists a survey in HAR where the main focus is to review the various methods of fusing multiple type of sensors to get rich context about the activities [22]. In this survey, the authors investigated various methods and approaches presented by researchers to combine the information from more than one sensor with the aim of identifying research opportunities in this area. Although, the survey presented extensive and in-depth analysis of the articles, but it does not provide other detail related to implementation. The articles were categorized through the lens of multi-fusion modalities and lack the detail about various techniques such as modelling of time series, pre-processing and machine learning algorithms.

There are some surveys where the articles belonging to complex activities are surveyed [23, 24]. As stated, they belong to the complex activities only whereas there is a huge contribution in the literature about simple activities which should also be categorized in order to provide a guideline for HAR process. Another research survey has been presented which shows the evolution of HAR in last two decades using mobile phone. The authors have analysed almost all the details regarding dataset collection process such as nature of behaviours, position, frequency, and orientation of sensors, window size, feature extraction, dimension reduction, traditional machine learning algorithms, deep learning-based algorithm, and evaluation techniques. This is a thorough research survey which provides information about almost every aspect of HAR, however it is been carried out for one device only i.e., for smart phone. With the advancement in pervasive computing there are various other wearable devices which provide low-cost solutions such as IMU and fancy devices such as smart watches. These devices have been used extensively in many research studies in the field of HAR. Hence, they should also be included in the review of HAR techniques and approaches.

The research survey [26] presents the review of HAR articles on the basis of traditional machine learning algorithms (decision tree, support vector machine, etc.) and machine learning algorithm. On the other hand [27] is a recent survey in HAR where the authors' focus was on reviewing the device-free approach mostly based on RFID technology. In device-free approach, subjects are not required to wear or carry any device for sensors data to recognise activities. Rather the environment and the objects are tagged with certain sensors such as cameras and RFID. The device-free approach has its own merits, but they pose various limitations (need of environmental setup for data creation and hence not being able to be applied in real-life settings) and serious issues related with the privacy.

Another research survey [28] presented a review of HAR research articles, but it mainly focuses on the traditional machine learning techniques and does not provide the overview of DL models in the field. Moreover, they do not provide other important aspects of HAR such as analysis of window size, percentage of overlaps, type of activities (simple, complex, or both), and orientation of sensors etc., which are significant research challenges of this field. The comparative analysis of all mentioned surveys is given in Table 1 in the summarized form.

3 Survey Methodology

This survey is conducted to analyse and categorize the research articles related to DL in HAR by the following aspects: dataset features, time series features, machine learning features, HAR features and other features such as recent trends, speciality etc. which are addressed by some research works. The summarized picture of our analysis can be depicted by Fig. 2, whereas the detail of these features are discussed in subsequent headings.

3.1 Dataset Features

A Dataset is a collection of instances or samples which share a common set of characteristics known as attributes. For any ML problem, the dataset is of prime importance as the quality of the solution depends upon the features of the datasets. In HAR, features of dataset are related with the determination of type of activities, type of sensors, type of devices, sample rate, position of sensors, orientation of sensors, subjects' detail analysis, and type of application domains, which are explained in subsequent subsections.

3.1.1 Types of Activities

The activities are mainly of two types i.e., simple activities (S) and complex activities (C) [5]. Simple activities are easy to recognise through a sensor since they are repetitive in nature, such as sitting, standing, jogging, running etc. While on other hand, complex activities are not that easy to be recognised, because they are not very repetitive in nature and they require more data and time for their recognition as compared to simple activities such as eating, smoking, making coffee, etc. [5]. Some datasets are created for complex activities such as in [6, 29], some are created purely for simple activities such as [30–35], while some are created for both types of activities (B) such as [36–39].

3.1.2 Types of Sensors

There are various types of sensors available which have been used for HAR successfully. According to [3] and [37], these sensors can be classified broadly into three categories: object sensors (O), environment sensors (E) and wearable/body-worn sensors (BW).

Table 1 Comparative analysis of existing surveys of HAR

Survey paper	Primary focus	Time Series Features	Machine learning features	Dataset features	HAR features	Other features
Wang et al. [3]	Role of deep learning in HAR in sensor-based activities	No	No	Yes	No	No
Ramasamy et al. [19]	Categorized articles by active learning, deep learning, transfer learning and semantics-based techniques	No	No	No	No	Yes
Slim et al. [18]	State-of-the-art HAR architectures are summarized based on accelerometer sensor	Yes	Yes	No	No	Yes
Alrazzak et al. [20]	Survey of HAR based on accelerometer sensor	Yes	Yes	Yes	No	No
Li et al. [21]	Survey of deep learning in HAR using Radar	No	No	No	Partially	Yes
Aguileta et al. [22]	Survey of papers in HAR which has used fusion of various sensors	No	Yes	No	No	No
Sakr et al. [23]	Survey of HAR techniques for complex activities	No	Partially	Partially	No	Yes
Mobark et al. [24]	Survey of HAR techniques for complex activities using mobile phones	No	Partially	Partially	No	No
Sousa et al. [25]	Survey of HAR by various aspects using smartphone	Partially	Yes	Yes	No	No
Jobanputra et al. [26]	General HAR survey of various ML techniques	No	Yes	No	No	No
Hussain et al. [27]	Survey of HAR using device-free approach	No	Partially	No	Partially	Yes
Elbasiony et al. [28]	Survey of traditional ML techniques in HAR	No	Yes	Yes	No	No

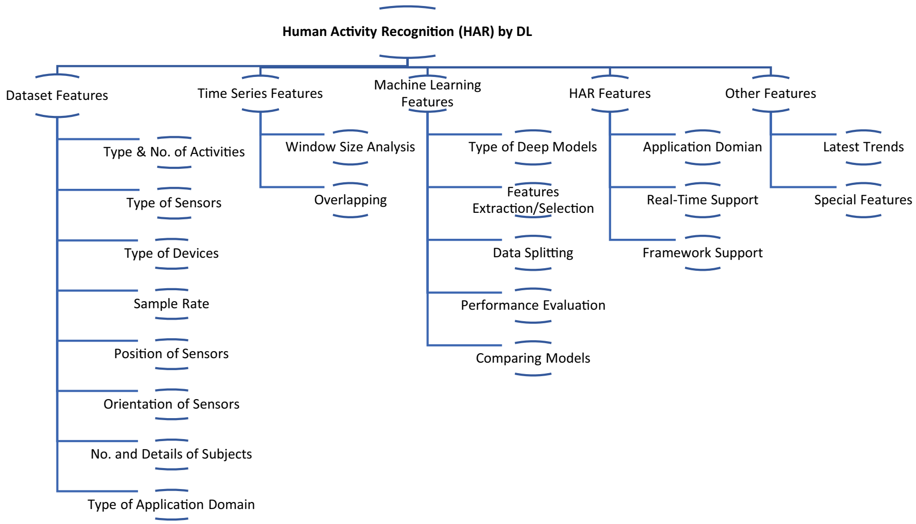

Fig. 2 Categorization of research on HAR

The object sensors are placed/attached with certain objects to record the patterns associated with the movement of these objects [36, 37]. Environmental sensors are placed in an environment to sense the information about that area such as temperature, humidity etc. [36, 37, 40]. Whereas the body-worn sensors are those which are worn/attached on the body of the subjects such as GPS etc. In the field of HAR, BW sensors are the most common type of sensors and among these sensors, accelerometer (A), gyroscope (G) and magnetometer (M) are the most widely used sensors by the research community [30, 32, 32–34]. The research studies have used various subtypes of these sensors which are given in Table 2 along with their acronyms by which they are referred in this study.

3.1.3 Types of Devices

Sensors are mounted or embedded in devices in which they can be operated. For example, smartphones (s-phone) contain various types of sensors such A, G, M, GPS etc. which have been used extensively by the research community in HAR [29–33]. With the rapid development and growth in smart devices, smartwatch (s-watch) has also become a popular choice where the study involves the data from hand movements. For example, if the pattern of the activities of eating or writing are needed then smartwatches can serve as the best choice. Various existing research work has used smartwatches for this purpose [30, 41–44].

Another important device is Inertial Measurement Unit (IMU), which are electronic devices to measure and report the accelerometer, gyroscope, magnetometer and sometimes other sensors to which they are attached [45]. They are low-cost dedicated devices for sensors with higher accuracy rate to measure physical movements and unlike smartwatches, can be mounted on different body parts easily. Due to these reasons, they have been widely

Table 2 Acronyms used to identify various sensors

Sensor	Acronyms	Sensors	Acronym
Accelerometer (BW)	A	Relative Magnetic Sensing (BW), (O)	RM
Gyroscope (BW)	G	Reed Switch (E), (O)	RS
Magnetometer (BW)	M	Radio Frequency Identification (O)	RFID
Linear Acceleration (BW)	LA	Float sensors to measure the toilet being flushed (E)	F
Angular Velocity (BW)	AV	Proximity Infrared (BW), (O)	Pr
GPS (BW)	GP	Pressure (O) (E)	P
Wi-Fi (BW)	W	Light (E)	Li
Motion (BW)	Mo	Humidity (E)	H
Temperature (E)	T	Electromyography Muscle (BW)	EMG
Door (E)	D	Electrooculography (BW)	EOG
Received Signal Strength Indicator (BW)	RSSI	Electroencephalogram (BW)	EEG
Heart Rate (BW)	HR	Functional near-infrared spectroscopy (BW)	fNIR
Spontaneous Blink Rate (BW)	SBR	Electrocardiography (BW)	ECG
Microphone (BW), (E)	Mi	Pressure Mats to measure sitting on a couch (E), (O)	PM
Camera (BW), (E)	C	Mercury Contacts to detect the movement of objects (E), (O)	MC
Stretch Sensor (Textile based) (BW)	SS	Passive infrared to detect motion in a specific area (E)	PIR

used for HAR problems [35, 38, 39, 46–53]. Other than these devices, some other commercial devices have also been successfully used [31, 36, 37, 40, 54, 55].

3.1.4 Sample Rates

Sample rate is measured in Hz which states the number of sensors' readings that are recorded in one second. The sample rate is kept according to the type of physical movement. For example, if the dataset has been collected for activities which expects various distinct physical movements within a second then probably a higher sample rate (around or greater than 100) is more appropriate such as in a production scenario [31, 46], or a scenario where instant reactions are studied [49]. In general, the sample rate of 50 Hz is suggested to be the appropriate frequency for data recording of simple activities and has been used most commonly [5, 34, 42, 43, 47, 48, 52, 55–61]. Some research studies have also explored less than 50 Hz for recording activities of daily living such as 40 Hz [55], 30 Hz [36, 37], 25 Hz [42, 51], 20 Hz [32], even less than 10 Hz [41]. On the other hand, higher sample rate (greater than 100) have also been reported for simple activities of daily living [50, 61].

3.1.5 Positions of Sensors

Determination of the most appropriate position of sensors is always been a research question for the researchers to mount, attach or wear a sensors [5]. The position to wear a sensor depends on the type of targeted activities. If the activities are simple in nature such as walking, sitting, jogging then sensors in pocket position are sufficient to learn these

Table 3 Acronyms used for position of sensors

Body Position	Acronym	References
Head	HE	[44]
Chest, Torso, Sternum, Breastbone	CH	[38, 39, 44, 46, 51, 54, 55]
Upper Arm	UA	[44, 52, 59]
Lower Arm	LA	[52]
Hand	HA	[31, 33, 38, 39]
Wrist	WR	[5, 30, 40–43, 46, 51, 59]
Waist	WA	[6, 30, 34, 43, 44, 48, 49, 57] [53]
Belt	BE	[59]
Back	BA	[52, 62]
Lower Back	LB	[35] [50]
Pocket	PO	[5] [6] [32,56,58] 59]
Hips	HI	[47]
Thigh	TH	[35, 44, 52, 62]
Knee	KN	[51, 63]
Shank, Shin	SH	[35, 44, 50, 52]
Ankle	AN	[38, 39, 63]
TOE	TO	[50]

activities [5]. On the other hand, if the activities are not simple and they require to capture the movement of other body parts then the sensors should be placed on those body parts. Some body positions which are used by different researchers are given in Table 3.

3.1.6 Orientations of Sensors

Besides position, orientation of sensors is also a question to deal with. By orientation of sensor, it is analysed that the sensor is worn/kept in a particular way or it is left on the choice of subject to keep it in any orientation. In some studies, the orientation is kept fixed [5, 34, 47, 49, 54, 59], in some cases it is kept different [30, 31, 52], while in some cases orientation information is not provided (NA) such as [32, 33, 35–40, 46].

3.1.7 Subject Detail

The number of subjects and the availability of their details are also considered as significant aspects to analyse the data collection process. Some studies emphasised on providing this information [32, 37–41, 43, 44, 47–51, 53, 55, 56–59, 62, 63], some simply ignore them [30–37, 42, 46, 52, 54, 56, 60, 61], while some have provided partial (Par.) detail of the subjects [5, 30]. But as the field is progressing towards advancement, researchers should also consider these details to see their impact on the overall performance.

3.1.8 Types of Application Domain

Another important feature of the dataset is the domain for which the activities are recorded. The most common domain in HAR is activities of daily living (ADL) in which the routine activities are studied such as walking, sitting etc. This is the domain for which majority of the datasets are studied and created [6, 30, 34, 43, 44, 47, 48, 50, 54–57, 58–61, 63]. Along with ADL, certain other household (HH) [37–40], cleaning (CA) [42] and kitchen activities (KA) [36, 37] have also been recorded in datasets. Another widely studied domain is related with the fitness (gym) [52, 53] and sports activities (SA) [38, 39, 51] such as tennis (T) [33] etc. The datasets have also been created for healthcare (HC) activities [6] such as fall detection (FD) [41, 58, 62], and freezing of gait (FoG) [35] and some psychological disorders like depression [7], anxiety disorder (AD) [54] and internalizing disorder (ID) [49]. Other application domains are order-picking (OP) scenario in a warehouse [46], car production scenario (CPS) in a factory [31] and scenario of transitional activities (TA) [53].

A summarized analysis of datasets by different features is given in Table 4. This table lists datasets that are created or used by the research articles analysed in this survey. Each dataset is assigned an ID by which they are referred in this survey for the sake of convenience.

3.2 Time Series Features

Time series features are related with the concept of time series modelling, since HAR is a time series problem in which the sensors' streams are recorded at a periodic interval. In order to process these streams, they need to be segmented into data windows.

Table 4 Analysis of datasets[1]

Dataset reference and assigned ID	Type of activities	No. of activities	Type of sensors	Sensors	Devices	Sample rate	Position of sensors[2]	Sensor orientation	No. of subjects	Availability of subjects' detail?	Application domain
D1: HHAR [30]	S	6	BW	A, G	S-phone, S-watch, Tablet	Highest possible	WA, B_WR	6 different orientations	9	Par	ADL
D2: Opportunity [36, 37]	B	27,000	E, BW, O	A, G, M, C, Mi, RM, RS, UWB, RFID, P, Pr, Li, T, EOG, EEG, fNIR, ECG, EMG	Various commercial devices	30 Hz	Various body positions	NA	12	No	ADL, KA
D3: Pamap2 [38, 39]	B	18	BW	HR, T, A, G, M	3 IMU	100 Hz	HA, CH, AN	NA	9	Yes	ADL, HH, SA
D4: Order Picking [46]	B	9	BW	A, G, M	3 IMU	100 Hz	B_WR, CH	NA	3	No	OP
D5: Skoda [31]	S	10	BW	A	ADXL330[3]	96 Hz	B_HA	Different orientation	1	No	CPS
D6: Actitracker [32]	S	5	BW	A	S-phone	20 Hz	PO (Front)	NA	29	No	ADL
D7: Bulling [33]	S	12	BW	A, G	S-phone	32 Hz	HA	NA	2	No	ADL, T
D8: WISDM (Laboratory-data) [32]	S	6	B	A	S-phone	20 Hz	PO (Front)	NA	29	Yes	ADL
D9: UCI-HAR [34]	S	6	B	A, G, M	S-phone	50 Hz	WA	Vertical position	30	No	ADL

Table 4 (continued)

Dataset reference and assigned ID	Type of activities	No. of activities	Type of sensors	Sensors	Devices	Sample rate	Position of sensors[2]	Sensor orientation	No. of subjects	Availability of subjects' detail?	Application domain
D10: USC-HAD [47]	S	12	B	A, G	IMU	50 Hz	R_HI (Front)	x-axis points towards ground	30	Yes	ADL
D11: Daphnet FOG [35]	S	2	BW	A	IMU	64 Hz	SH, TH, LB	NA	10	No	FoG
D12: Van Kasteran [40]	S	NA	E	RS, PM, MC, PIR, F	RFM DM 1810	NA	NA	NA	3	Yes	HH
D13: Zebin [48]	S	6	BW	A, G	IMU	50 Hz	WA	NA	20	Yes	ADL
D14: Shoaib [5]	B	13	BW	A, G, LA	S-phone	50 Hz	PO, R_WR	Portrait, screen towards body	10	P	ADL
D16: McGinnis et al. [49]	S	NA	BW	A	IMU	300 Hz, 100 Hz	WA	Fixed	63	Yes	ID
D17: Miranda et al. [54]	NA	NA	BW	HR, SBR	Band, Google Glass	NA	CH	Fixed	8	No	AD
D15: Osman [6]	S	10	BW	A	S-phone	NA	PO, WA	NA	10	No	ADL
	C	51	BW	RSSI for location, Time	location, time	NA	PO, WA	NA	3	No	HC
D18: Bhat et al. [63]	S	6	BW	A, SS	TI-CC2650	SS: 100 Hz, A: 250 Hz	KN, AN	NA	9	Yes	ADL

Table 4 (continued)

Dataset reference and assigned ID	Type of activities	No. of activities	Type of sensors	Sensors	Devices	Sample rate	Position of sensors[2]	Sensor orientation	No. of subjects	Availability of subjects' detail?	Application domain
D19: Wand et al. [56]	S	5	BW	A, G	S-phone	50 Hz	R_PO	NA	7	No	ADL
D20: Mauldin et al. [41]	S	8	BW	A	S-watch, s-phone	4 Hz, 1.25 Hz, 62.5 Hz, 31.25 Hz	L_WR	NA	7	Yes	ADL, FD
D21: Mauldin et al. [41]	S	11	BW	Notch	NA	NA	WR	NA	7	Yes	FD
D22: Klenk et al. [62]	S	NA	BW	A, G, M	Various	Various	BA, TH, others	NA	> 2000	Yes	FD
D23: Bevilacqua et al. [50]	S	4	BW	A, G	IMU	102.4 Hz	2 at SH, 2 at TO, 1 at LB	NA	19	Yes	ADL
D24: UCI Hapt, Ortiz et al. [57]	S	6	BW	A, G	S-phone	50 Hz	WA	NA	30	Yes	ADL
D25: Sanitation, Zhang et al. [42]	S	7	BW	A	S-watch	25 Hz	WR	NA	NA	No	ADL, CA
D26: UniMib Shar, Micucii et al. [58]	S	17	BW	A	Smartphone	50 Hz	PO (half time in L & half time in R)	NA	30	Yes	ADL, FD
D27: Shoaib et al. [59]	S	7	BW	A, G, M	S-phone	50 Hz	B_PO, B_BE, R_UA, R_WR	Landscape for belt, Portrait for rest	10	Yes	ADL

Table 4 (continued)

Dataset reference and assigned ID	Type of activities	No. of activities	Type of sensors	Sensors	Devices	Sample rate	Position of sensors[2]	Sensor orientation	No. of subjects	Availability of subjects' detail?	Application domain
D28: Khan et al. [43]	S	8	BW	A	S-phone, s-watch	50 Hz	WA, WR (dom. arm)	NA	15	Yes	ADL
D29: Altun et al. [51]	S	19	BW	A, G, M	IMU	25 Hz	B_KN, CH, B_WR	NA	8	Yes	Sports, ADL
D30: Banos et al. [52]	S	33	BW	A, G, M	IMU	50 Hz	L_UA, L_LA, R_UA, R_LA, L_TH, R_TH, L_SH, R_SH, BA	Different orientation	17	No	Gym
D31: Hassan et al. [60]	S	12	BW	A, G	S-phone	50 Hz	NA	NA	NA	No	ADL
D32: ActiveMiles, Ravi et al. [61]	S	7	BW	A, G	S-phones	50-200 Hz	Any position, unconstrained	NA	10	No	ADL
D33: Sztyler et al. [44]	S	8	BW	A	S-phone, s-watch	50 Hz	CH, forearm, HE, SH, TH, UR, WA	Different orientation	15	Yes	ADL
D34: Qi et al. [53]	S	12	BW	A, G, M	IMU	50 Hz	WA	NA	20	Yes	ADL, gym, TA
D35: Sample et al. [55]	S	4	BW	A, RFID	W^2ISP^4	40 Hz	CH	NA	14	Yes	ADL

[1] In this Table, NA represents not available information about the field

[2] The prefix L represents the position of the sensor as being on the left side of the body, similarly R represents right and B represents sensors on both left and right, for example, L_UA means Left Upper Arm position

[3] Analog Device of 3-axis acceleration measurement system. https://www.sparkfun.com/products/retired/692

[4] Wireless Identification & Sensing Platform. https://sensor.cs.washington.edu/WISP.html

The most common features related with data segmenting are the size of sliding window (*WS*) and the percentage of overlap between consecutive windows. The sliding window is segmenting of sensors' stream data into slices of certain No. of reading (*NoR*) so that this segment can be fed into a learning model. The analysis of window size is very important as it depends on various factors including the type of activities, sampling rate and the complexity of the model. For example, some applications require the recognition of one or a particular set of activities for which a *WS* can be determined which optimizes the recognition accuracy such as determining the number of glasses of water taken by an individual in a day. Moreover, there could be some applications where the prime concern is not just the recognition accuracy but also the speed of the recognition like violence detection. For such activities, a smaller *WS* is desirable so that a timely action can be taken to control the damage. A smaller *WS* also reduces the resources and energy requirements and hence produces a less complex model. But reducing the *WS* may affect the other performance measures of recognition such as, accuracy. Hence, a trade-off between the recognition accuracy and time, should be addressed by the designers of the system according to the type of activities and domain of the application [64].

The dataset of Opportunity (D2) [37] is extensively used by various researchers in the analysis of HAR. This dataset is collected for a sample rate of 30 Hz and comprised of simple and complex activities both which are related with the Activities of Daily Living (ADL) or Kitchen Activities (KA). The researchers have segmented this dataset for various window sizes (*WS*) ranging from 0.5 s (15 *NoR*) to 7.5 s (224 *NoR*). Since the application domain of this dataset is not of a nature where a fast response is needed, the researchers had the liberty to determine the optimum *WS* which can yield better performance. The research [42] segmented this dataset for 224 *NoR* and was able to achieve around 95% accuracy which is the highest accuracy reported for this dataset among the research works surveyed in this paper. Although another research work [65] was able to achieve around 94% accuracy with 15 *NoR* suggests that these activities could be recognised with smaller *WS* as well with more capable models and setups such as feature selection etc.

Another publicly available dataset (D9) [34] have also been used many times for HAR. It comprised of simple activities of ADL which are recorded with a sample rate of 50 Hz. Unlike the dataset D2, this dataset is not segmented for many different *WS* and most of the time it is been segmented with a *WS* of 2.56 s (128 NoR). Even a research work [66] has claimed to achieve 100% accuracy with the same *WS* using an innovative deep model (features learned by three parallel CNN are fused together). Similarly, relatively larger *WSs* are segmented and reported to yield good performance for datasets of D3, D5, and D8 by various research works of [9, 61], and [42] respectively. A research work even sliced the stream of sensors into the windows of 60 s to determine the abnormal behaviour in the simple house-hold activities [67].

From the evaluation of these articles, it is concluded that determination of *WS* is a crucial challenge for HAR problems and it should be determined by taking all the aspects of the required system into consideration such as the application domain, type of activities, recognition speed etc. A smaller *WS* can also yield better accuracy when designed with more capable models and other aspects of machine learning features which are explained in next subsection.

3.3 Machine Learning Features

To model HAR by a machine learning algorithm some features are important to be analysed by the researchers to create an effective model. The most important features are type of the model, feature extraction/selection techniques, evaluation measures like data splitting technique for training, validation and testing, evaluation metrics such as accuracy, precision, recall etc. There are various machine learning based algorithms such as Support Vector Machine (SVM), K Nearest Neighbour (KNN), Decision Tree (DT) and many others which have been used for HAR but after the breakthrough performance by deep learning-based algorithms in HAR, the rate of using traditional algorithms have reduced significantly. Researchers have explored effectiveness of various deep models such as deep Neural Network (DNN), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long-Short Term Memory (LSTM), Restricted Boltzmann Machine (RBM) and various hybrid deep models (combination of more than one deep structure). The selection of most appropriate deep model and other machine learning features depend upon the type and complexity of the problem. The research articles analysed in this work, have used many types of deep learning algorithms which are explained in next subsection and summarized in Table 5.

3.3.1 Deep Learning Models

Following are the most frequently used deep learning models.

3.3.1.1 Deep Neural Networks (DNN) Deep Neural Network (DNN) are created by artificial neural network (ANN). The reason they are called deep is because they have deep architecture by having many hidden layers in between input and output layers [3]. Traditional ANN which has fewer hidden layers are known as shallow networks and are also referred as multilayer perceptron (MLP). In machine learning DNN has been used directly and in the combination with CNN and RNN structures for pattern recognition. In [68], DNN is used for HAR along with some hand engineered features extracted from sensors. Similarly, another research work [69] has used PCA for features selection and DNN for learning the activities. In these studies, the deep structures are not fully utilized, and they are just used for classification. Their features learning capabilities were not explored and the network architecture were also not deep. Other research studies have shown that when the structure is deep with sufficient amount of data, activities can be recognized on raw sensor data without the need of any hand-crafted features [70].

Another research work [63] has proposed a DNN based online activity recognition system where the online training is achieved by using a policy gradient based reinforcement learning (RL) method. RL is a type of machine learning model, where the learning is performed through rewards and punishments in a stochastic environment. The agents take a sequence of decisions in such a way which maximizes the cumulative reward. Deep learning has also been used in RL for approximation of the policy and reward function [71]. Similarly, [72] highlights the latest research trends in RL based algorithms using multi-hop Q-routing in an energy harvesting enabled cognitive radio network.

3.3.1.2 Convolution Neural Networks (CNN) CNN are commonly used for their feature learning capability from raw sensors data. CNN is divided into two parts: convolution layer/s for feature extraction and fully connected layers for classification. A convolution layer is usually comprised of two operations, filtering and pooling through which features are learned. For example, if an image of a tiger is needed to be detected, then convolution is the operation by which the stripes, ears, eyes and other features of the tiger will be learnt. These learned features will then be forwarded to the fully connected (dense) layers to classify the image to the class of tiger.

There are various research studies in which human activities are recognized by CNN [4, 9, 11, 73]. In [9], the authors have used CNN to combine the process of feature extraction and classification in an end-to-end manner. They claimed to learn non-linear and temporal structures in simple and complex human movements through many stacked layers of convolution filters and pooling operations. The architecture contains many parallel branches, each having convolution layers for temporal relations, respective pooling operations and in the end a fully connected layer for combining all the local features and temporal relations in a global representation.

In [11], the authors have used CNN for activity recognition due to the following two advantages in image recognition problem i.e., local dependencies and scale invariance. They stated that CNN can capture the local dependencies in the signal because the nearby acceleration readings are likely to have strong relationship the same way a pixel is strongly correlated with nearby pixels in an image. Similarly, CNN offers features scale invariance because an activity performed by different user may have different force, strength, or pace. For example, the activity of running may have different pace by different participants.

Authors in [4] also highlighted the importance of CNN in extracting distinguishing features from raw sensors data. Their main contributions are to propose a solution by which feature extraction and classification can be performed together by one single model. This work was evaluated on two public datasets of simple activities [33, 39].

In [73], CNN is used for presenting an online and user-independent approach for recognizing human activities. Along with feature extraction, windows size is also investigated where it was found that larger window size does not lead to better performance in all cases of activity classification. A significant rise is found from 1 to 3 s, but slight improvements are noticed in its further growth by CNN and Random Forest (RF) while it degrades with other algorithms such as K-Nearest Neighbour (KNN) and PCA-based techniques. Later, window size of 1 s is used to enable real-time classification.

3.3.1.3 Recurrent Neural Networks (RNN) This class of deep learning algorithm is specifically known for its capability to label sequences or time series. RNN has the distinguishing feature of keeping the 'memory' from previous input sequence which in turn is utilized to influence the output of the current sequence. This feature makes RNN special because unlike traditional algorithms it does not assume that the data sequences are independent of each other as the information from the prior sequences is utilized to learn the current sequence. Due to this reason, RNN is commonly used to learn the temporal structures and dynamics of a time series. Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs) are two mostly used variants of RNN architecture. LSTM is proposed by [74] to address the problem of vanishing gradient in learning the long-term dependencies. As a solution LSTM has memory cells which comprise of three gates and parameters to model long-range dependencies in temporal sequences. These gates control the overall operation of memory cell by determining when to update states and forget previous hidden states. These gates are, input gate i_t, forget gate f_t and output gate o_t. The datasets which contain both simple and complex activities may need larger sequences of data to learn the repetitive pattern. Due to this reason, LSTM-based RNN is a better choice and have been previously reported

Table 5 Deep learning models

Model name	Acronyms	Description
Multilayer Perceptron	MLP	A multilayer perceptron is a class of feedforward artificial neural network (ANN)
Deep Neural Network	DNN	DNN is a deep architecture version of ANN, which works on the principle of biological neurons to loosely model the functionality of the brain
Convolutional Neural Network	CNN	Uses convolutional layers having filters to slide over the input data to learn features
Vanilla Recurrent Neural Network	VRNN/ RNN	Consists of loops which provides storage of information within the network. Their chain-like structure makes them useful for language translation, speech recognition, etc
Long Short-Term Memory	LSTM	A type of RNN with feedback connection. It is used mainly to preserve the gradients
Gated Recurrent Unit	GRU	A type of LSTM which lacks output gates, hence uses fewer parameters
Auto Encoder	AE	An unsupervised ANN used to learn data codings from samples
Restricted Boltzmann Machine	RBM	It is a generative type of stochastic NN which learns a probability distribution over the set of samples
Residual Network, Residual Layer	ResNet, ResLayer	It is based on pyramidal cells in the cerebral cortex. Skip connections are used to jump over some layers
Attention based CNN, Attention based LSTM	Att. CNN, Att. LSTM	The attention is a mechanism in deep learning in which greater attention is paid to certain factors in a sequence
Bi-directional LSTM	*biLSTM*	Bi-directional means that information flow in two directions, from past to future and from future to past
Reinforcement Learning	RL	A type of ML to figure how agents should take actions to maximize the cumulative reward
Extreme Learning Machine	ELM	It is a single-hidden layer feedforward NN without using any method to tune its parameters

to give better results for HAR problems. On the other hand, GRUs deals with the problem of short-term memory. It has two gates, a reset and an update gate and hidden states to control and regulate the flow of information.

In [75], a deep RNN is being proposed which is based on long short-term memory (LSTM) for activity classification where the window size is of variable-length. The effectiveness of unidirectional and bidirectional RNNs are also being demonstrated specifically for the tasks of HAR. Its ability of having variable-length window size, gave it the power to classify the activity at any time stamp when working in real-time as contrasted with the other algorithms where an activity recognition has to wait till the end of the window size to be completed.

Another study [67] in which three types of RNN are investigated namely, Vanilla RNN (VRNN), LSTM and Gated Recurrent Unit (GRU). The usefulness of these three models have been explored for activity recognition and detection of abnormal behaviours of patients suffering from dementia. Since there is no dataset available for abnormal behaviours, the researchers brought the abnormality in an existing dataset through synthetic methods.

In [76], a deep architecture of stacked LSTM is proposed, with the capabilities of generalizing and learning temporal dynamics out of raw data. In this study, novelty is claimed in terms of usage of longer temporal sequences and batch normalization for RNN structures.

3.3.1.4 Hybrid Models These are special models which combine the deep structures of CNN and LSTM both. The reason for combining these two different deep learning structures is to obtain the dual advantage. As CNN are known for their feature extraction capability and LSTM are capable of learning temporal dynamics, these two models are combined to achieve both features. Such hybrid models have applications not only in HAR but also have been created in various application domain such as visual recognition [77], voice search task [78], computer vision and natural language processing [79].

Researchers have also used combinations of different deep learning models to achieve higher level of performance in HAR. In [10], a hybrid model of CNN and RNN is presented to exploit the advantages of CNN to extract local interactions and RNN to learn the temporal relationships of signal modalities. At first Fourier transformation is used as feature extraction, then a set of CNN layers are used to learn local dependencies, next RNN layers are used and in the last output layer is used according to the type of the problem. The model is referred as unified because it has been proposed and evaluated for both types of the problems i.e., regression and classification.

Another research where a hybrid deep model *DeepConvLSTM* is presented in [80] with the combination of CNN and RNN, exploiting the capabilities of CNN to automatically extract the distinguishing features and RNN to learn the complex temporal dynamics of time series.

3.3.2 Feature Extraction/Selection Techniques

In machine learning, feature extraction is one of the key steps towards making a successful model. It is the process of determining the crucial and relevant information from sensor stream through which one activity could be distinguished from other activities. Feature selection is the process of selecting certain features from the existing potential features. In HAR, determination and selection of the appropriate feature set is a challenging task in which the feature learning techniques are applied on raw sensors data and feature selection techniques are applied to pick the appropriate feature set to learn the patterns and reduce the dimension of the problem.

The feature extractions can be broadly categorized into two types: hand-crafted features and deep learning-based features. Hand-crafted features are those which are manually designed or engineered by researchers to extract attributes from time sequences. These techniques include time domain features (TDF) and frequency domain features (FDF). In TDF, information is extracted from sensors sequences about the behaviour of an object in time. While in FDF, the behaviour of an object is analysed in terms of the frequency, i.e., that given a set of frequency ranges, how much of a sequence/signal lies within a specific frequency range.

Regardless of the fact that hand-crafted features require domain specific knowledge, they are viably used in classification problems due to the advantages of lower computation complexity and simple setup requirement [81]. In such cases one needs to determine the appropriate trade-off between the accuracy and complexity efficiency.

In [63], both TDF and FDF based features are extracted from two sensors, i.e., accelerometer and stretch using DWT and FFT respectively. Features by FFT are learned from stretch sensors as this sensor captures the periodicity accurately and features by DWT are learned from accelerometer due to the chances of higher presence of noise in this sensor. In [60], at first the statistical features are extracted and then the feature reduction is carried out by KPCA and LDA (see Table 6 below) to obtain more robust features from raw sensors data. These features are combined with deep belief networks for activity recognition which was able to achieve an overall accuracy of 95.85%. Another research work [65] claimed to have achieved good performance just by using different types of hand-crafted features on two publicly available datasets.

With the progress in deep learning-based computation, the researchers do not need to apply explicit feature learning techniques. These features can be learned through deep learning-based models and hence are known as deep learned features. With deep learning, the researchers can combine the operations of feature learning and classification together in an end-to-end manner. This has also lifted the burden from the shoulders of researchers to acquire domain specific knowledge for feature engineering.

CNN is specifically known for its capability to learn features through convolution layers and extracting the local patterns using feature maps. Through multiple filters, high level interpretation from raw sensor data is extracted to represent generic attributes which in turn are used in classification. There are various research studies [4, 9, 43, 50, 66, 75, 82] which have used CNN on raw sensors' data to extract the features and were able to achieve comparable and even better performances in some cases than the state-of-the-art algorithms.

LSTM has also been used in this regard to learn the temporal dependencies and relationship from raw data associated with each activity. Many research works have explored its performance in HAR to determine the role of temporal relationship in sensor streams specially where a larger *WS* have been segmented [67, 75, 76, 83, 84].

CNN and LSTM learn different features from raw sensors' data. CNN extracts local patterns from feature maps and LSTM learns temporal dependencies from LSTM cells. For time series analysis both types of features are needed to be extracted to represent overall information. Similarly, other deep models have their own special capability to learn distinguishing features from raw data. For this reason, researchers have combined the speciality of multiple deep models to extract the features which represent the overall attributes of a system from different aspects. Different variants of such hybrid models have been proposed, evaluated and have proved to be successful in feature learning [80, 84–87].

Some research studies have also combined the hand-crafted features and deep learned features together to further enhance the system performance [10, 60, 61, 63, 65, 67, 88, 88]. Ignatov [73] has used deep features learned by CNN and statistical features to achieve good performance on two datasets. Similarly, [86, 89] claims to achieve better accuracy when the manually engineered features are combined with the deep learned features through CNN. Researchers have explored the performance of deep models with hand crafted features and without hand-crafted features. Some feature learning techniques used by the research articles surveyed in this paper are listed in Table 6.

3.3.3 Data Splitting Techniques

Data splitting techniques refer to the process by which the dataset is divided into training, validation and test data. For machine learning algorithms, a set of data is required to train the model for which training data is used. The validation data is used to validate the performance of the model during training epochs for tuning the hyperparameters. The test data is the held-out sample on which the performance of the model is evaluated after all the tuning of the hyperparameters is completed to find out the performance on a sample which has not participated in any way during the training process. The validation and test data have been referred interchangeably in the research community but some research emphasised on keeping test data separate and divide the training data into training and validation set, so that the test set can provide an unbiased evaluation [90].

In HAR, various types of techniques have been presented to split the data into training, validation and test set. The most common techniques reported by the research articles are summarized in Table 7. Usually, the application of the system determines the type of technique for data splitting. For example, if researchers focus to make a robust model which can adapt to the heterogeneity on subject level then, they opt for the split based on subjects. In this split, some of the subjects' data is used for training and the hold-out subjects are used for evaluation. Because every subject performs the activities according to their choice, style, posture and preferences which brings intra-class variations. So, the researchers aim to learn a model which is robust and give good performance despite the heterogeneity at subject level. The subject split could be also done by percentage in which a certain percentage of subjects is kept for testing/validation while the rest is used for training, referred as Percent Split (PS by subject) [53, 63, 66, 73, 76, 86]. On the other hand, subject split could also be achieved by numbers of subjects, i.e., a certain number is kept for evaluation and the rest is taken for training, referred as Subject Split (SS). Leave-one-out (LOO) or hold-one-out is a subtype of PS in which one subject is kept for validation/testing [9, 84, 91]. Leave-one-out could be based on other entities as well such as days, datasets, setup, units etc. For example, if the data collection is carried out in a scenario where more than one day is involved for a data collection of a subject, then one day data can be held-out for evaluation [4]. Similarly, hold-one-out could be done for other entities [84].

One more type of percent split is based on the data (PS by data), where the whole dataset is divided based on a percentage such as 70%-30, 80%-20, and so on. The larger portion is fed for training the model where the other portion is kept for evaluation [41, 42, 56, 75, 83, 85]. This is the mostly used splitting criteria in the general problems of machine learning and have been reported in HAR with success.

Another very popular approach is k-fold-cross validation (KFC) in which the data is divided into k numbers of equal folds. These folds could be obtained by number of samples [50], subjects [9, 10], days [67] etc. During training, one of the folds is used as hold-out for

Table 6 Feature extraction/selection techniques

Technique	Acronym	Detail
Raw Data	RD	The values generated by sensors' stream
Sensor based Features	SBF	For example, switching of sensor state, etc
Statistical Features	SF	The statistical measures such as Mean (M), variance (V), standard deviation (SD) etc
Frequency Domain Features	FDF	The analysis of signals with respect to the frequency, e.g., FFT
Fast Fourier Transformation	FFT	Used for conversion from time-based function to a frequency-based function
Discrete Wavelet Transformation	DWT	Used to convert pixels of an image into wavelets
Time Domain Features	TDF	Statistical feature analysis of signals in time domain, e.g., mean over time duration
Principal Component Analysis	PCA	Converts possibly correlated variables into uncorrelated principal components
Kernel Principal Component Analysis	KPCA	Extension of PCA reproducing kernel Hilbert Space
Linear Discriminant Analysis	LDA	A feature reduction technique based on finding the linear combinations of original features
Feature Reduction	FR	The process of reducing the number of features in a problem

validation and the rest are taken for training, so that every fold participates in both training and evaluation. KFC is suggested for smaller dataset problems to utilize each sample in training and validation.

Manual Selection (MS) has also been reported in the problems of HAR where the split can be performed by both subject and data. For example, a specific subject could be used, or a certain portion based on percentage of data could be kept for validation [4, 80, 82, 86, 92, 93].

3.3.4 Performance Evaluation Metrics

Performance evaluation metrics are measures by which it can be determined how well a model is performing on a given dataset in a system. Various types of performance measures have been used according to the type or the purpose of the problem. The most common and simple type of such metrics is confusion matrix. Confusion matrix is a table which has two dimensions one for representing the count of actual class labels and the other is for predicted class labels. Both dimensions have true positive (TP), false positive (FP), false negative (FN), and true negative (TN), where TP = the correctly predicted class 1, FP = the correct class is 0, predicted as 1, FN = the correct class is 0, predicted as 1, and TN = the correctly predicted class 0. It is illustrated by Fig. 3.

Another important metric is accuracy which is the most common measure used for the evaluation of the classification models. It is defined as the ratio of correct predictions over all predictions as given in Table 8. As stated earlier, it is the mostly used metric to evaluate the performance in the field of machine learning and same is the case in HAR as well [4, 9–11, 43, 84, 89]. It gives an overall performance of the system in the form of one quantity to compare it against other benchmarks specially where more than one dataset or learning models are compared [42, 65, 67, 75, 93].

Accuracy is a good measure where the dataset is balanced meaning that it has equal number of samples representing each class. But in case of unbalanced data, accuracy is not appropriate for performance evaluation since it can be misleading. For such cases, other measures are used such as precision, recall [61], f-measure [80, 85, 86, 88, 92, 94–96], and specificity [61] which are more appropriate. There definitions are given in Table 8.

Finally, we present in Table 9 some traditional machine learning algorithms which have served as benchmarks by our surveyed research articles against their proposed algorithms.

Table 7 Data Splitting Techniques for training, validation, and testing set

Technique	Acronym	Detail
Percent split	PS	*PS by subject* indicates splitting of data based on percentage of subjects *PS by data* indicates data samples are split based on percentage of data
K-Fold-Cross validation	KFC	KFC by Subject means folds are created based on subjects KFC by data means folds are created based on data
Leave-one -out	LOO	LOO means leaving one subject for validation/testing
Random selection	RS	Random Split by data or subject
Manual selection	MS	Manually selection of training, validation and test set on some basis

3.4 HAR Features

In the features of HAR, the characteristics of the study related with the overall HAR algorithm are analysed such as what is the application domain of the algorithm, did researcher created their own dataset or a publicly available dataset is used, has any framework is created to utilize the recognition by the model, or does it offer real-time support? Etc. As already stated, the importance of HAR in recent years has not only been on the recognition of activities but more importantly on the utilization of the recognition in resolving real world issues. Such features are included in our study to determine which studies are using them.

3.4.1 Application Domain

Application domain of each research study is presented and earlier explained in subsection 3.1.8 in the context of datasets.

3.4.2 Real-Time Support

As the field of HAR is progressing, the research has expanded its scope towards using the recognition of activities in the form of a system. Real-time activity recognition is one of such applications where the activity recognition models are used to provide real-time detection [53, 61, 63, 73, 85]. There are some applications where real-time monitoring is required to control the system and prevent any mishap/accident to happen such as fall detection [41, 94].

3.4.3 Framework Support

Another important feature of new development in the field of HAR is the implementation of a framework where the recognition of human behaviours is utilized in making decisions or certain actions. For example, a mobile app which can recognize certain behaviours such as if a patient is suffering from dementia has taken his/her medicine on time or not. If the medicine is not taken on time, then a notification could be sent to the patient/loved ones/care takers. Many research studies in HAR now focus on this challenge to utilize the recognised behaviours in making an automated system. Bhat [63] has implemented a framework which supports online training, and inference using the

Fig. 3 Confusion matrix

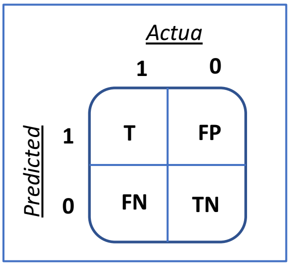

Table 8 Performance Metrics, where TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative

Performance measure	Formula	Description
Accuracy	$Acc = \frac{(TP+TN)}{(TP+TN+FP+FN)}$	Ratio of correct recognized samples to the total samples
Precision	$P = \frac{TP}{(TP+FP)}$	Ratio of correctly recognized samples to the total recognized as class 1
Recall/Sensitivity	$R = \frac{TP}{(TP+FN)}$	Ratio of correctly recognized samples to the samples in actual class 1
Specificity	$Spe = \frac{(TN)}{(TN+FP)}$	Ratio of actual class 0 to the correctly recognized as 0
F1 Score/F-measure	$F1 = \frac{2*(P*R)}{(P+R)}$	Weighted average of Precision and Recall

Table 9 Models used in comparison with deep learning-based models

Benchmarked ML algorithm	Acronyms	Research articles
Quadratic Discriminant Analysis	QDA	[42, 91]
Decision Tree	DT	[42, 63, 91, 93]
Naïve Bayes	NB	[41],
Support Vector Machine	SVM	[4, 10, 40–43, 60, 63, 65, 67, 75, 76, 87, 89, 91, 93]
K-Nearest Neighbor	KNN	[42]
Logistic Regression	LR	[91]
Random Forest	RF	[10, 63, 73, 75, 91]
Linear Classifier	LC	[61]
Mean Variance	MV	[4, 93]
Hidden Markov Model	HMM	[65, 67, 75]
Conditional Random Fields	CRF	[67, 87]

real-time data collected by sensors. Mauldin [41] has made an android app to connect a smartwatch with a smartphone to collect data and to detect fall in real-time. Similarly [61] implemented their proposed approach in the form of an android app and as an implementation for the Intel Edison Development Platform for visual demonstration of on-node classification of human activities.

In [94], the authors presented a system which not only supports real-time recognition but also provided the on-device Incremental Learning (IL) (a technique to update the model) and Active Learning (AL). AL is a ground truthing technique in machine learning by which a set of new data points can be labelled by querying an information source referred as oracle.

3.5 Other Features

Besides the features described in previous subsection, there are some other features which include but are not limited to the concepts of transfer learning, online/offline training or learning, power consumption analysis, active learning, etc. We have included these features to analyse the research articles to build up on existing surveys.

The research work [63], has not only provided real-time support but also has proposed an online training and inference mechanism by which recognition accuracy has been improved by 33% for new users. Similarly, [73] has also proposed and implemented an online system for HAR. Whereas [94] has highlighted the importance of on-device incremental learning (IL) to update the model using real-time data over the period of time. For this purpose, a ground truthing method like Active Learning (AL) is needed so that the new data points could be labelled. Hence, the concepts and techniques of IL and AL have been incorporated into traditional HAR method to explore and open new doors for research challenges in this field. Energy consumption analysis (ECA) is one of these challenges, specially for deep learning algorithms as they are power-hungry algorithms and the sensor-embedded devices are low-power so it is important to do this analysis to create deep learning algorithms which can be run on such devices [10, 61, 63].

Mauldin [41] has suggested innovation in terms of system architecture. They have implemented an android based app by creating a three-layered IoT based architecture. In the first layer a smartwatch is connected which collects real-time data from subjects and a smartphone is in the middle layer which takes real-time data and runs the main application named as *SmartFall*. Finally, the inner most layer is comprised of some very heavy-duty computation like web server to host applications, a database to store the sensors' data and the machine learning based services which analyze the data and continuously update the model for fall detection.

Another new concept in deep learning is Extreme Learning Machine (ELM). ELM is a modified version of fully-connected layers where the weights are updated by a generalized inverse operation and is well-known for its fast convergence, easy implementation, and good generalization ability [97]. Due to these characteristics, ELM is an ideal choice for HAR applications where usually a quick real-time response is needed [85, 29].

Another emerging new research dimension is Transfer Learning (TL) in machine learning. It is a process of storing and utilizing the previous knowledge to apply in solving other different but related problems to improve the generalization [98] and solve the insufficient training data problem [99]. This may prove to be beneficial in HAR problems where there are various types of heterogeneities involved in different contexts as achieved by [43] and [91]. In [43], an approach is proposed for CNN to utilize the properties learned in one context, in another context which can have different devices or positions, given the same set of activities, reducing the requirement of huge volume of data for training new context. Whereas in [91], TL is utilized for new users because every individual has different behavioral pattern. An approach is proposed which takes labels from new user for few data points and updates only the classification layer (top layer) to be more specific to the behavioral pattern of a new user.

Research is ongoing to investigate various other techniques in the deep learning based algorithms to make them more suitable for the task of recognizing activities, such as adoption of Bayesian optimization for hyperparameters [84], multi-level classification [65, 88], feature fusion strategies [66, 82], segmenting and analysis of *WS* [95], and dealing with the challenges of weakly labelled dataset [56].

Table 10 summarizes the detailed analysis of the research articles surveyed in this paper, based on time series features, machine learning features, HAR features and other features. We have also analysed and summarized the architectural details of proposed deep models so that it can give readers an overview of the type, structure and depth of the models. This analysis is given in Table 11.

4 Conclusion

This survey investigated several recent articles based on deep learning algorithms in the field of HAR that have used data from sensors. It provides a detailed analysis based on aspects that are rarely discussed in surveys such as time series modeling, machine learning based techniques, dataset features, HAR specific features and other latest trends which are acquiring wider adaptability and popularity as the field advances. The analysis presented in this work is unique in its structure and may serve as a guideline for practitioners in the implementation of HAR techniques.

Table 10 Analysis of all reviewed articles. In the column of Accuracy/Performance if metric is not specified then it is accuracy

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Rueda et al. [9]	D2, D3, D4	No	D2: *WS* = 720 ms (24 *NoR*), D3: *WS* = 3 s (100 *NoR*) D4: *WS* = 1 s (100 *NoR*)	D2: 50%, D3: 78%, D4: 99%	CNN	No FS	D2: LOO (by subject), D4: KFC (by subject)	D2: 92.22%, D3: 93.68%, D4: 70.80%, 69.36%	Simpler CNN [80]	ADL	No	No	–
Yang et al. [4]	D2, D7	No	NA	step size 3	CNN	No FS	D2: MS, D7: LOO via RS (by day)	D2: 87.7%, 83.0%, 86.7%, D7: 94.1%, 96.0%	SVM [100], 1NN [100], MV [33], DBN [17]	ADL, Tennis	No	No	–
DeepSense, Yao et al. [10]	D1	No	*WS* = 5 s, it is sub-segmented to 0.25 s for FFT	NA	CNN + GRU	FDF (FFT)	KFC (by subject)	94%	RF, SVM, RBM, Multi-RBM	ADL	No	No	Energy consumption analysis (ECA)
Zeng et al. [11]	D2, D5, D8	No	*WS* = 64 *NoR*	50%	CNN	No FS	KFC	D2: 76.83%, D5: 88.19%, D8: 96.88%	PCA-ECDF [17]	ADL, CPS	No	No	–
Ignatov et al. [73]	D8, D9	No	D8: *WS* = 1-10 s (20–200 *NoR*), D9: *WS* = 1 s & 2.56 s (50 & 128 *NoR*)	D8: Step size 20, D9: 50%	CNN	RD + SF	D8: PS 72–28% (by subject), D9: PS 70–30% (by subject)	D8: 93.32%, D9: 97.63%, Cross-dataset:, 82.76%	RF, PCA, KNN	ADL	No	Yes	Online

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Murad et al., [75]	D9, D10, D2, D11, D5	No	D9: *WS* = 128 *NoR*, D10: *WS* = 128 *NoR*, D2: *WS* = 24 *NoR*, D11: *WS* = 32 *NoR*, D5: *WS* = 128 *NoR*	NA	LSTM	No FS	PS 80–20%	D9: 96.7%, D10: 97.8%, D2: 92.5%, D11: 94.1%, D5: 92.6%	CNN [101], SVM [34], Seq. ELM [102], LS-SVM, RF, DBN, NN, HMM	ADL,FoG,CPS	No	No	–
Arifoglu et al. [67]	D12	No	WS = 60 s	NA	VRNN, LSTM, GRU	RD + SBF	LOO (by Day) KFC	LSTM Acc, DS A: 96.7%, DS B: 87.2%, DS C: 87.4%	NB, HMM, HSMM, CRF, SVM, All are mentioned in [40]	HH	No	No	–
Zebin et al. [76]	D13	No	WS = 128	NA	LSTM	No FS	PS 70–30% (by subject)	92%	SVM [48], DNN [48], *ConvLSTM* [80], Factorized LSTM	ADL	No	No	–
ConvLSTM, Ordonez et al. [80]	D2, D5	No	*WS* = 500 ms (15 *NoR*)	50%	CNN + RNN	No FS	D2: MS, D5: NA	D2: 0.93, 0.866 F1, D5: 0.958 F1	CNN [11], CNN [103], Baseline CNN	ADL, CPS	No	No	–

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Bhat et al. [63]	D18	Yes	*WS* = 1 s – 3 s	0.5 s overlap	MLP, RL using PGA	FFT, DWT	PS 80%-20% (by subjects)	97.7%	RF, KNN, SVM, DT	ADL	Yes	Yes	Online training & inference, ECA
Wang et al. [56]	D9, D19	Yes	*WS* = 2.56 s (128 *NoR*	50%	Att. CNN	No FS	PS 70–30%	D9: 93.41%, D19: 93.83%	Simple CNN, CNN [80], *ConvLSTM* [80]	ADL	No	No	Weakly labelled dataset
Gudur et al. [94]	D1, D21	No	Various *WS*	No overlap	CNN (bays.)	No FS	NA	D1: 86% Acc, D20: 0.969 F1	NA	ADL, FD	Yes	Yes	AL, On-device IL
Mauldin et al. [41]	D20, D21, D22	Yes	WS = 1.28 s (40 *NoR*)	50%	GRU	No FS	D20: PS 2/3–1/3, D21, D22: LOO	D20: 85%, D21: 99%, D22: 99%	NB, SVM	Fall	Yes	Yes	3-layered IoT system architecture
Bevilacqua et al. [50]	D23	Yes	WS ~ 2 s (204 *NoR*)	97%	CNN	No FS	KFC	Detailed results	NA	ADL	No	No	–
Pienaar et al. [83]	D8	No	*WS* = 10 s (200 *NoR*)	90%	LSTM	No FS	PS 80–20%	94%	NA	ADL	No	No	—

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Zhang et al. [42]	D8, D9, D24, D2, D25	Yes	*WS* = 224 *NoR*	No overlap	U-Net	No FS	PS 70–30%	D8 = 97%, D9 = 98.4%, D24 = 93.1%, D2 = 94.7%, D25 = 88.6%	SVM, KNN, DT, QDA, CNN	ADL	No	No	–
Dong et al. [89]	D9	No	*WS* = 2.56 s (128 *NoR*)	50%	CNN	RD, TDF, FDF	PS 70.-30%	96.9%	MC-SVM	ADL	No	No	–
Kasnesis et al. [82]	D9, D3	No	*WS* = 2.56 s (128 *NoR*)	50%	CNN	No FS	D9: MS, D3: LOO	D9: 97.25%, D3: 88.56%	CNN (early fusion), LSTM	ADL	No	No	Late fusion
Long et al. [95]	D2, D26	No	*WS* = 1 s-3 s (32, 64, 96 *NoR*)	NA	ResNet	No FS	NA	D2: 90.29 F1, D26: 77.23 F1	AE, MLP, CNN, LSTM, Hybrid, ResNet	ADL, FD	No	No	Uses parallelly the short and long windows
Zeng et al. [96]	D3, D11, D5	No	D3: *WS* = 5.12 s (170 *NoR*), D11: *WS* = 1 s (32 *NoR*), D5: NA	D3: 78%, D11: 50%, D5: NA	Att. LSTM	No FS	D3: LOO, D11: LOO, D5: PS 80–10-10% (by class)	D3: 0.8996 F1 D11: 0.8373 F1 D5: 0.8903 F1	LSTM, *ConvLSTM* [80], LSTM-S(sample-by-sample) [92]	ADL, HH, SP, FoG, CPS	No	No	–

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Hammerla et al. [92]	D2, D3, D11	No	D2: WS = 1 s (30 *NoR*), D3: WS = 5.12 s (170 *NoR*), D11: WS = 1 s (32 *NoR*)	D2: 50%, D3: 78%, D11: 50%	DNN, CNN, LSTM, *biLSTM*	No FS	D2: MS, D3: MS, D11: MS	D2: 0.929 F1, D11: 0.745 F1, D3: 0.76 F1	DNN, CNN, *ConvL-STM*[80]	ADL, HH, SP, FoG	No	No	–
Buenaventura et al. [84]	D27	No	*WS* = 2 s (100 *NoR*)	50%	CNN, LSTM	No FS	LOO (by position & subject)	Pos. Ind LSTM = 97.17%, Sub. Ind LSTM = 80.2%	NA	ADL	No	No	Bayesian optimization for hyper parameter optimization
Khan et al. [43]	D28	Yes	*WS* = 2.5 s (128 *NoR*)	50%	CNN	No FS	KFC	87.40%	CNN, SVM	ADL	No	No	TL
Rokni et al. [91]	D29, D6	No	Various	Various	CNN	No FS	LOO	> 90%	DT, LR, RF, SVM, QDA	SP, ADL	No	No	TL
Sun et al. [85]	D2	No	*WS* = 500 ms (15 *NoR*)	50%	CNN + LSTM + ELM	No FS	PS 80–20%	0.918 F1	CNN, CNN-LSTM	ADL	No	Yes	Classification by ELM
Hassan et al. [60]	D31	No	*WS* = 2.56 s (128 *NoR*)	50%	RBM	SF, KPCA, LDA	PS ~ 70–30% (by events)	95.85%	ANN, multiclass SVM	ADL	No	No	–

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/Special features
Ravi et al. [61]	D32, D6, D11, D5	Yes	*WS* = 4 & 10 s	No overlap	CNN	Spectrogram	KFC	D32: 95.1%, D6: 98.2%, D5: 91.7%, D11: 71.9% (R), 96.7% (Spe)	LC, ANN	ADL, CPS, FoG	Yes	Yes	ECA
Almaslukh et al. [88]	D33	No	*WS* = 1 s (50 *NoR*)	50%	CNN	Raw data + TDF	Subject independent: LOO, Subject dependent: KFC	98% F1	NA	ADL	No	No	3 level classifiers, 1st static/dynamic, 2nd pos. 3rd activity
Aviles-cruz et al. [66]	D9	No	*WS* = 2.56 s (128 *NoR*)	50%	CNN	No FS	PS 70 – 30% (by subject)	100%	Anguita [34], San-Segando [104], Ignatov [73], Cho [65], Ronao [105]	ADL	No	No	3 CNN are used in parallel and their features are fused

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Cho et al. [65]	D2, D9	No	D2: *WS*=500 ms (15 *NoR*), D9: *WS*=2.56 s (128 *NoR*)	D2: 50%, D9: 50%	CNN	D2: SF, D9: FDF, TDF	MS	D2: 94.2%, D9: 97.62%	*ConvLSTM* [80], SVM, Ronao [105], DCNN [106], 3-stage-HMM [107]	ADL	No	No	2-stage classification
Qi et al. [53]	D34	Yes	*WS*=3×3 *NoR*	NA	CNN	TDF, FDF	PS 90–10% (by subject)	94.18%	LSTM, *BiLSTM*,	ADL, Gym	No	Yes	–
Zhao et al. [86]	D2, D9	No	*WS*=2.56 s (128 *NoR*)	50%	ResNet-*biLSTM*	Raw data, TDF	D2: MS, D9: PS 70—30% (by Subject)	D2: 90.5% F1, D9: 93.5% F1	CNN, 1NN, LSTM	ADL	No	No	–
San et al. [93]	D2, D7, D30	No	D2: NA, D7: NA D:30: *WS*=6 s (150 *NoR*)	D2: NA, D7: NA, D30: No overlap	CNN	No FS	D2: MS, D7: RS LOO, D30: MS	D2: 87.7%, 83%, 86.7%, D7: 94.1%, 96% D30: 92.8%	SVM, KNN, MV, DBN, DT	ADL, Tennis, Gym	No	No	–

Table 10 (continued)

Reviewed article	Dataset reference		Time series features		Machine learning features					HAR features			Other features
Author reference	Dataset references	Self-Created dataset?	Window size (*WS*)	Overlapping	Deep models	Feature Extraction/ Selection	Data splitting technique	Accuracy / Performance	Models compared	Application domain	Framework Support?	Real-Time Support?	Comment/ Special features
Abedin et al. [87]	D35, D8	No	D35: *WS* = various sizes, D8: *WS* = 10 s (200 *NoR*)	NA	MLP + Pooling + MLP	NA	KFC	D35: 94.51 F1, 96.97 F1, D8: NA	SVM, CRF, Bi-LSTM, *ConvL-STM*	ADL	No	No	–

Table 11 DNN models and their architectures

Study	Deep architecture
Yang et al. [4]	Input – Con(1 × 5) – Subsampling(1 × 2) – Con(1 × 5) – Subsampling(1 × 2) – Con(1 × 3) – Unification – FC –Classification
Rueda et al. [9]	Input (1) – Con(5 × 1) – Con(5 × 1) – pooling – Con(5 × 1) – Con(5 × 1) – polling – FC
	Input (N) – Con(5 × 1) – Con(5 × 1) – pooling – Con(5 × 1) – Con(5 × 1) – polling – FC
	{Combine all individual path from input 1 to input N} – FC – Classification
Yao et al. [10]	(Con-Con-Con) Individual – Flatten – (Con – Con – Con) Merging – Flatten – GRU – GRU – Classification
Zeng et al. [11]	Input (Channel wise) – Con – pooling – FC(Merge) – FC – Classification
Bhat et al. [63]	Input – FC – Classification
Wang et al. [56]	Input – Con1 – Con2 – Con3 – MaxPooling – Con4 – MaxPooling – Con5 – MaxPooling – FC1– Att1(Con3, FC1) – Att2(Con4, FC1) – Att3(Con5, FC1) – FC2 – Classification
Mauldin et al. [41]	Input – GRU Layer – FC – Classification
Bevilacqua et al. [50]	Input – Con2D – MaxPooling – Con2D – MaxPooling – Con2D – MaxPooling – FC – Classification
Khan et al. [43]	Input – Con (32) – Con (64) – MaxPooling – FC – Classification
Hassan et al. [60]	Input – FC (60) – FC (20) – Classification
Ravi et al. [61]	Input – Con (N) – FC – Classification
Qi et al. [53]	Input – Con2D – Con2D – Con2D – Con2D – FC – Classification
ConvLSTM, Ordonez et al. [80]	Input – Con – Con – Con – Con – LSTM – LSTM – Classification
Ignatov et al. [73]	Input – Con – MaxPooling – Flattening – FC – Classification
Murad et al. [75]	Input – LSTM (1) …LSTM (N) – Late fusion – Classification
Arifoglu et al. [67]	Input – VRNN (N) – Classification, Input – LSTM (N) – Classification, Input – GRU (N) – Classification
Zebin et al. [76]	Input – LSTM (30) – LSTM (30) – FC – Classification
Gudur et al. [94]	Input – Con1 – Con2 – Flatten – FC1 – FC2 – Classification
Pienaar et al. [83]	Input – LSTM (64) – LSTM (64) – LSTM (N) – Classification
Dong et al. [89]	Input – For each channel {Con – MaxPooling – Con – MaxPooling – Con – MaxPooling – Con – MaxPooling} – Concatenate – Flatten – Concatenate (TDF, FDF) – FC1 – FC2—Classification
Kasnesis et al. [82]	Input – Con – MaxPooling – Con – MaxPooling – Con2D – AvgPooling – FC – Classification
Long et al. [95]	Input – For shorter *WS* {Con – MaxPooling – Res1(64) – Res2(128) – Res3(256) – Res4(512)}
	Input—For Longer *WS* {Con – MaxPooling – Res1(64) – Res2(128) – Res3(256) – Res4(512)} – Concatenate (shorter, longer) – FC – Classification
Zeng et al. [96]	Input – Att. Sensor – Att. LSTM – Classification
Hammerla et al. [92]	Input – LSTM – LSTM – LSTM – Classification, Input – biLSTM – biLSTM – biLSTM – Classification, Input – Con – Con – Con – Classification, Input – FC – FC – FC – FC – FC – Classification,
Rokni et al. [91]	Input – Con N – MaxPooling N – FC – Classification
Sun et al. [85]	Input – Con – Con – Con – Con – LSTM – LSTM – FC – ELM Classification

Table 11 (continued)

Study	Deep architecture
Almaslukh et al. [88]	Input – Con – MaxPooling—Con – MaxPooling – Flatten – Concatenate (TDF) – FC1 – FC2 – FC3 – Classification
Aviles-cruz et al. [66]	Input – a {Con 1 – MaxPooling 1 – Con 2 – MaxPooling 2 – Con3 – MaxPooling3 – Con4 – MaxPooling4}
	Input – b {Con 1 – MaxPooling 1 – Con 2 – MaxPooling 2}
	Input – c {Con 1 – MaxPooling 1} Concatenate & Flatten (a, b, c) – FC – Classification
Cho et al. [65]	Level 1: Input – Con – Con – Con – Con – Con – Flatten – FC – Classification
	Level 1: Input – Con 1 – MaxPooling 1 – Con 2 – Flatten – FC – Classification
Zhao et al. [86]	Input – ResLayer (LSTM) – ResLayer (LSTM) – Classification
San et al. [93]	Input – Con 1 – SubSampling 1 – Con 2 – SubSampling 2 – Con 3 – Unification – FC – Classification
Abedin et al. [87]	Input – FC – FC – FC – Aggregate – FC – FC – FC – Classification

[1]In this Table, Con = Convolution layer (1D), Con2D = Convolution layer (2D), and FC = Fully Connected Layer

References

1. Zhang, H.-B., et al. (2019). A comprehensive survey of vision-based human action recognition methods. *Sensors, 19*(5), 1005. https://doi.org/10.3390/s19051005
2. Zhang, S., Wei, Z., Nie, J., Huang, L., Wang, S., Li, Z. (2017). A review on human activity recognition using vision-based method. *Journal of Healthcare Engineering*. https://www.hindawi.com/journals/jhe/2017/3090343/ (accessed Jul. 10, 2019).
3. Wang, J., Chen, Y., Hao, S., Peng, X., Hu, L. (2017). Deep learning for sensor-based activity recognition: a survey. *Pattern Recognition Letter*, Accessed: Dec. 07, 2018. [Online]. Available: /paper/Deep-Learning-for-Sensor-based-Activity-A-Survey-Wang-Chen/c2420b5f507015ba735f27f8709706cf01be1c2b.
4. Yang, J.B., Nguyen, M.N., San, P.P., Li, X.L., Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. *Twenty-Fourth International Joint Conference on Artificial Intelligence*, p. 7.
5. Shoaib, M., et al. (2016). Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors, 16*(4), 426. https://doi.org/10.3390/s16040426
6. Gani, M.O., Saha, A. K., Ahsan, G.M.T., Ahamed, S.I. (2018). A novel framework to recognize complex human activity, Accessed: Nov. 07, 2018. [Online]. Available: /paper/A-Novel-Framework-to-Recognize-Complex-Human-Gani-Saha/1024a5b27c233a452bbca412c322943ae02b9a7c.
7. Wahle, F., Kowatsch, T., Fleisch, E., Rufer, M., & Weidt, S. (2016). Mobile sensing and support for people with depression: a pilot trial in the wild. *JMIR MHealth UHealth*. https://doi.org/10.2196/mhealth.5960
8. Jeong, T., Klabjan, D., Starren, J. (2016). Predictive analytics using smartphone sensors for depressive episodes. *ArXiv160307692 Cs Stat*, Mar. 2016, Accessed: Mar. 11, 2019. [Online]. Available: http://arxiv.org/abs/1603.07692.
9. Moya Rueda, F., Grzeszick, R., Fink, G. A., Feldhorst, S., & ten Hompel, M. (2018). Convolutional neural networks for human activity recognition using body-worn sensors. *Informatics, 5*(2), 26. https://doi.org/10.3390/informatics5020026
10. Yao, S., Hu, S., Zhao, Y., Zhang, A., Abdelzaher, T. (2016). DeepSense: A unified deep learning framework for time-series mobile sensing data processing. *ArXiv161101942 Cs*. Accessed: Nov. 07, 2018. [Online]. Available: http://arxiv.org/abs/1611.01942.
11. M. Zeng *et al.*, (2014). Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors. *6th International Conference on Mobile Computing, Applications and Services*, Austin, United States. Doi: https://doi.org/10.4108/icst.mobicase.2014.257786.

12. Ciresan, D., Meier, U., Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, pp. 3642–3649. Doi: https://doi.org/10.1109/CVPR.2012.6248110.
13. Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., pp. 1097–1105.
14. Google's AlphaGo AI wins three-match series against the world's best Go player | TechCrunch. https://techcrunch.com/2017/05/24/alphago-beats-planets-best-human-go-player-ke-jie/ (accessed Jun. 25, 2019).
15. Bao, L., Intille, S.S. (2004). Activity recognition from user-annotated acceleration data. *Pervasive Computing*, pp. 1–17.
16. Krause, A., Siewiorek, D. P., Smailagic, A., Farringdon, J. (2003). Unsupervised, dynamic identification of physiological and activity context in wearable computing. In *Seventh IEEE International Symposium on Wearable Computers. Proceedings.*, Oct. 2003, pp. 88–97. Doi: https://doi.org/10.1109/ISWC.2003.1241398.
17. Plötz, T., Hammerla, N.Y., Olivier, P. (2011). Feature learning for activity recognition in ubiquitous computing. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence,* volume 2, Barcelona, Catalonia, Spain, pp. 1729–1734. Doi: https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-290.
18. Slim, S., Atia, A., & M. M.A., and M.-S. Mostafa, . (2019). *Survey on human activity recognition based on acceleration data*. International Journal of Advanced Computer Science. https://doi.org/10.14569/IJACSA.2019.0100311
19. Ramasamy Ramamurthy, S., & Roy, N. (2018). Recent trends in machine learning for human activity recognition-A survey. *Reviews of Data Mining Knowledge Discovery., 8*(4), e1254. https://doi.org/10.1002/widm.1254
20. Alrazzak, U., Alhalabi, B. (2019). A survey on human activity recognition using accelerometer sensor. In *Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, Spokane, WA, USA, pp. 152–159. Doi: https://doi.org/10.1109/ICIEV.2019.8858578.
21. Li, X., He, Y., & Jing, X. (2019). A survey of deep learning-based human activity recognition in radar. *Remote Sensor, 11*(9), 1068. https://doi.org/10.3390/rs11091068
22. Aguileta, A. A., Brena, R. F., Mayora, O., Molino-Minero-Re, E., & Trejo, L. A. (2019). Multi-sensor fusion for activity recognition—a survey. *Sensors, 19*(17), 3808. https://doi.org/10.3390/s19173808
23. Sakr, N.A., Abu-Elkheir, M., Atwan, A., Soliman, H.H. (2018). Current trends in complex human activity recognition. 14, 20.
24. Mobark, M., & Chuprat, S. (2018). Recognition of complex human activity using mobile phones: a systematic literature review. *Journal of Theoretical and Applied Information Technology, 96*(12), 24
25. Sousa Lima, W., Souto, E., El-Khatib, K., Jalali, R., & Gama, J. (2019). Human activity recognition using inertial sensors in a smartphone: an overview. *Sensors, 19*(14), 3213. https://doi.org/10.3390/s19143213
26. Jobanputra, C., Bavishi, J., & Doshi, N. (2019). Human activity recognition: A survey. *Procedia Computing Science, 155*, 698–703. https://doi.org/10.1016/j.procs.2019.08.100
27. Hussain, Z., Sheng, M, Zhang, W. E. (2019). Different approaches for human activity recognition: a survey. *ArXiv190605074 Cs* Accessed: Mar. 07, 2020. [Online]. Available: http://arxiv.org/abs/1906.05074.
28. R. Elbasiony and W. Gomaa, (2020). A Survey on Human Activity Recognition Based on Temporal Signals of Portable Inertial Sensors. *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019)*, Cham, vol. 921, pp. 734–745. Doi: https://doi.org/10.1007/978-3-030-14118-9_72.
29. Qi, W., Su, H., Yang, C., Ferrigno, G., De Momi, E., & Aliverti, A. (2019). A fast and robust deep convolutional neural networks for complex human activity recognition using smartphone. *Sensors*. https://doi.org/10.3390/s19173731
30. Stisen, A. et al. (2015). Smart devices are different: assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems - SenSys '15*, Seoul, South Korea, pp. 127–140. Doi: https://doi.org/10.1145/2809695.2809718.
31. Zappi ,P. et al. (2008). Activity recognition from on-body sensors: accuracy-power trade-off by dynamic sensor selection. *Wireless Sensor Networks*, pp. 17–33.
32. Kwapisz, J. R., Weiss, G. M., & Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *SIGKDD Exploring Newsletter, 12*(2), 74–82. https://doi.org/10.1145/1964897.1964918

33. Bulling, A., Blanke, U., Schiele, B. (2014). A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Survey*, 46(3), 33:1–33:33. Doi: https://doi.org/10.1145/2499621.
34. Anguita, D., Ghio, A., Oneto, L., Parra, X., Reyes-Ortiz, J. L. (2013). A public domain dataset for human activity recognition using smartphones. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Bruges Belgium, [Online]. Available: https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones.
35. Bächlin, M., et al. (2010). Wearable assistant for parkinson's disease patients with the freezing of gait symptom. *Information Technology Biomedical IEEE Transactions On, 14*, 436–446
36. Roggen, D. et al. (2010). Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh International Conference on Networked Sensing Systems (INSS)*, pp. 233–240. Doi: https://doi.org/10.1109/INSS.2010.5573462.
37. Chavarriaga, R., et al. (2013). The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition. *Pattern Recognition Letter, 34*(15), 2033–2042. https://doi.org/10.1016/j.patrec.2012.12.014
38. Reiss, A., Stricker, D. (2012). Introducing a new benchmarked dataset for activity monitoring. In *16th International Symposium on Wearable Computers*, pp 108–109. Doi: https://doi.org/10.1109/ISWC.2012.13.
39. Toshev, A., Szegedy, C. (2014). DeepPose: Human pose estimation via deep neural networks. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1653–1660. Doi: https://doi.org/10.1109/CVPR.2014.214.
40. van Kasteren, T.L.M., Englebienne, G., Kröse, B.J.A. (2011). Human activity recognition from wireless sensor network data: benchmark and software. *Activity Recognition in Pervasive Intelligent Environments*, L. Chen, C. D. Nugent, J. Biswas, and J. Hoey, Eds. Paris: Atlantis Press, pp. 165–186.
41. Mauldin, T. R., Canby, M. E., Metsis, V., Ngu, A. H. H., & Rivera, C. C. (2018). SmartFall: A Smartwatch-based fall detection system using deep learning. *Sensors*. https://doi.org/10.3390/s18103363
42. Zhang, Y., Zhang, Y., Zhang, Z., Song, Y. Human activity recognition based on time series analysis using U-Net. *ArXiv preprint arXiv:1809.08113*, p. 21.
43. Khan, M.A.A.H., Roy, N., Misra, A., Scaling, (2018). Human activity recognition via deep learning-based domain adaptation. In *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Athens, pp. 1–9. Doi: https://doi.org/10.1109/PERCOM.2018.8444585.
44. T. Sztyler and H. Stuckenschmidt, (2016). On-body localization of wearable devices: An investigation of position-aware activity recognition. in *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Sydney, Australia, pp. 1–9. Doi: https://doi.org/10.1109/PERCOM.2016.7456521.
45. Ribeiro, N.F., Santos, C.P. (2017). Inertial measurement units: A brief state of the art on gait analysis. In *2017 IEEE 5th Portuguese Meeting on Bioengineering (ENBENG)*, pp. 1–4. Doi: https://doi.org/10.1109/ENBENG.2017.7889458.
46. Grzeszick, R., Lenk, J. M., Rueda, F. M., Fink, G. A., Feldhorst, S., ten Hompel, M. (2017). Deep neural network based human activity recognition for the order picking process. In *Proceedings of the 4th international Workshop on Sensor-based Activity Recognition and Interaction - iWOAR '17*, Rostock, Germany, pp. 1–6. Doi: https://doi.org/10.1145/3134230.3134231.
47. Zhang, M., Sawchuk, A. A. (2012). USC-HAD: a daily activity dataset for ubiquitous activity recognition using wearable sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12*, Pittsburgh, Pennsylvania, p. 1036. Doi: https://doi.org/10.1145/2370216.2370438.
48. Zebin, T., Scully, P., Ozanyan, K. (2017). Evaluation of supervised classification algorithms for human activity recognition with inertial sensors, pp. 1–3. Doi: https://doi.org/10.1109/ICSENS.2017.8234222.
49. McGinnis, R. S., et al. (2019). Rapid detection of internalizing diagnosis in young children enabled by wearable sensors and machine learning. *PLoS ONE, 14*(1), e0210267. https://doi.org/10.1371/journal.pone.0210267
50. Bevilacqua, A., MacDonald, K., Rangarej, A., Widjaya, V., Caulfield, B., Kechadi, T. (2019). Human activity recognition with convolutional neural netowrks. *ArXiv190601935 Cs Stat*, vol. 11053, pp. 541–552, 2019. Doi: https://doi.org/10.1007/978-3-030-10997-4_33.
51. Altun, K., Barshan, B., & Tunçel, O. (2010). Comparative study on classifying human activities with miniature inertial and magnetic sensors. *Pattern Recognition, 43*(10), 3605–3620. https://doi.org/10.1016/j.patcog.2010.04.019
52. Baños, O., Damas, M., Pomares, H., Rojas, I., Tóth, M.A., Amft, O. (2012). A benchmark dataset to evaluate sensor displacement in activity recognition. *Proc. 2012 ACM Conf. Ubiquitous*

Comput. - UbiComp 3912, Accessed: Apr. 05, 2020. [Online]. Available: https://www.academia.edu/13057715/A_benchmark_dataset_to_evaluate_sensor_displacement_in_activity_recognition.
53. Qi, W., Su, H., Yang, C., Ferrigno, G., De Momi, E., & Aliverti, A. (2019). A fast and robust deep convolutional neural networks for complex human activity recognition using smartphone. *Sensors, 19*(17), 3731. https://doi.org/10.3390/s19173731
54. Miranda, D., Calderón, M., Favela, J. (2014). Anxiety detection using wearable monitoring. *Proceedings of the 5th Mexican Conference on Human-Computer Interaction*, New York, NY, USA, 2014, p. 34:34–34:41. Doi: https://doi.org/10.1145/2676690.2676694.
55. Sample, A., Ranasinghe, D., Shi, Q. (2013). Sensor enabled wearable rfid technology for mitigating the risk of falls near beds.
56. wang, K., he, j, & zhang, l. (2019). attention-based convolutional neural network for weakly labeled human activities recognition with wearable sensors. *IEEE Sensors Journal, 19*(17), 7598–7604. https://doi.org/10.1109/JSEN.2019.2917225
57. Reyes-Ortiz, J.-L., Oneto, L., Samà, A., Parra, X., & Anguita, D. (2016). Transition-aware human activity recognition using smartphones. *Neurocomputing, 171*, 754–767. https://doi.org/10.1016/j.neucom.2015.07.085
58. Micucci, D., Mobilio, M., Napoletano, P. (2018). UniMiB SHAR: a new dataset for human activity recognition using acceleration data from smartphones. *ArXiv161107688 Cs*, Accessed: Oct. 30, 2018. [Online]. Available: http://arxiv.org/abs/1611.07688.
59. Shoaib, M., Bosch, S., Incel, O. D., Scholten, H., & Havinga, P. J. M. (2014). Fusion of smartphone motion sensors for physical activity recognition. *Sensors*. https://doi.org/10.3390/s140610146
60. Hassan, M. M., Uddin, Md. Z., Mohamed, A., & Almogren, A. (2018). A robust human activity recognition system using smartphone sensors and deep learning. *Future Generation Computer System, 81*, 307–313. https://doi.org/10.1016/j.future.2017.11.029
61. Ravi, D., Wong, C., Lo, B., Yang, G.-Z. (2016). Deep learning for human activity recognition: A resource efficient implementation on low-power devices. In *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pp. 71–76. Doi: https://doi.org/10.1109/BSN.2016.7516235.
62. Klenk, J., et al. (2016). The FARSEEING real-world fall repository: a large-scale collaborative database to collect and share sensor signals from real-world falls. *European Review of Aging and Physical Activity, 13*(1), 8. https://doi.org/10.1186/s11556-016-0168-9
63. Bhat, G., Chaurasia, V. V., Shill, H., Ogras, U. Y. (2018). Online human activity recognition using low-power wearable devices. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, San Diego, CA, pp. 1–8, Accessed: Mar. 30, 2020. [Online]. Available: https://www.researchgate.net/publication/327260776_Online_Human_Activity_Recognition_using_Low-Power_Wearable_Devices.
64. Banos, O., Galvez, J.-M., Damas, M., Pomares, H., & Rojas, I. (2014). Window size impact in human activity recognition. *Sensors*. https://doi.org/10.3390/s140406474
65. Cho, H., & Yoon, S. M. (2018). Divide and conquer-based 1D CNN human activity recognition using test data sharpening. *Sensors*. https://doi.org/10.3390/s18041055
66. Avilés-Cruz, C., Ferreyra-Ramírez, A., Zúñiga-López, A., & Villegas-Cortéz, J. (2019). Coarse-fine convolutional deep-learning strategy for human activity recognition. *Sensors, 19*(7), 1556. https://doi.org/10.3390/s19071556
67. Arifoglu, D., & Bouchachia, A. (2017). Activity recognition and abnormal behaviour detection with recurrent neural networks. *Procedia Computing Science, 110*, 86–93. https://doi.org/10.1016/j.procs.2017.06.121
68. Vepakomma, P., De, D., Das, S. Bhansali, S. (2015). A-Wristocracy: Deep learning on wrist-worn sensing for recognition of user complex activities. Doi: https://doi.org/10.1109/BSN.2015.7299406.
69. Walse, K. H., Dharaskar, R. V., & Thakare, V. M. (2016). PCA based optimal ANN classifiers for human activity recognition using mobile sensors data. *Proceedings of First International Conference on Information and Communication Technology for Intelligent Systems:, 1*, 429–436
70. Bengio, Y. (2013). Deep learning of representations: Looking Forward. *ArXiv13050445 Cs*, Accessed: Jul. 15, 2019. [Online]. Available: http://arxiv.org/abs/1305.0445.
71. Yang, T., Zhao, L., Li, W., & Zomaya, A. Y. (2020). Reinforcement learning in sustainable energy and electric systems: a survey. *Annual Reviews in Control, 49*, 145–163. https://doi.org/10.1016/j.arcontrol.2020.03.001

72. Paul, A., & Maity, S. P. (2020). Outage analysis in cognitive radio networks with energy harvesting and Q-Routing. *IEEE Transactions on Vehicular Technology, 69*(6), 6755–6765. https://doi.org/10.1109/TVT.2020.2987751
73. Ignatov, A. (2018). Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. *Applied Soft Computing, 62*, 915–922. https://doi.org/10.1016/j.asoc.2017.09.027
74. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-term Memory. *Neural Computation, 9*, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
75. Murad, A., & Pyun, J.-Y. (2017). Deep recurrent neural networks for human activity recognition. *Sensors, 17*(11), 2556. https://doi.org/10.3390/s17112556
76. Zebin, T., Peek, N. Casson, A., Sperrin, M. (2018). Human activity recognition from inertial sensor time-series using batch normalized deep LSTM recurrent networks, 2018. Doi: https://doi.org/10.1109/EMBC.2018.8513115.
77. Donahue J. *et al.*, (2016). Long-term recurrent convolutional networks for visual recognition and description *ArXiv14114389 Cs*. Accessed: Dec. 11, 2019. [Online]. Available: http://arxiv.org/abs/1411.4389.
78. Sainath, T. N., Vinyals, O., Senior, A., Sak, H. (2015). Convolutional, long short-term memory, fully connected deep neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4580–4584. Doi: https://doi.org/10.1109/ICASSP.2015.7178838.
79. Vinyals, O., Toshev, A., Bengio, S. Erhan, D. (2015). Show and tell: a neural image caption generator. *ArXiv14114555 Cs*, Apr. 2015, Accessed: Dec. 11, 2019. [Online]. Available: http://arxiv.org/abs/1411.4555.
80. Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors, 16*(1), 115. https://doi.org/10.3390/s16010115
81. $$$F. Li, K. Shirahama, M. A. Nisar, L. Köping, and M. Grzegorzek, (2018). Comparison of feature learning methods for human activity recognition using wearable sensors. *Sensors*, vol. 18, no. 2, Art. no. 2, Feb. 2018. Doi: https://doi.org/10.3390/s18020679.
82. Kasnesis, P., Patrikakis, C. Z., & Venieris, I. S. (2018). PerceptionNet: A deep convolutional neural network for late sensor fusion. *Intelligent Systems and Applications, Cham, 868*, 101–119. https://doi.org/10.1007/978-3-030-01054-6_7
83. Pienaar, S. W., & Malekian, R. (2019). Human activity recognition using lstm-rnn deep neural network architecture. *2019 IEEE 2nd Wireless Africa Conference (WAC). Pretoria, South Africa, Aug., 2019*, 1–5. https://doi.org/10.1109/AFRICA.2019.8843403
84. Buenaventura, C.V.S., Tiglao, N. M. C., Atienza, R. O. (2018). Deep learning for smartphone-based human activity recognition using multi-sensor fusion. S*pringerprofessional.de*, Cham, 2018, pp. 65–75, Accessed: Apr. 03, 2020. [Online]. Available: https://www.springerprofessional.de/en/deep-learning-for-smartphone-based-human-activity-recognition-us/16376826.
85. Sun, J., Fu, Y., Li, S., He, J., Xu, C., & Tan, L. (2018). Sequential human activity recognition based on deep convolutional network and extreme learning machine using wearable sensors. *J. Sens., 2018*, 1–10. https://doi.org/10.1155/2018/8580959
86. Zhao, Y., Yang, R., Chevalier, G., Xu, X., & Zhang, Z. (2018). Deep residual Bidir-LSTM for human activity recognition using wearable sensors. *Mathematical Problems in Engineering, 2018*, 1–13. https://doi.org/10.1155/2018/7316954
87. Abedin, A., Rezatofighi, S. H., Shi, Q., Ranasinghe, D. C. (2019). SparseSense: Human activity recognition from highly sparse sensor data-streams using set-based neural networks. *ArXiv190602399 Cs Stat* Accessed: Apr. 06, 2020. [Online]. Available: http://arxiv.org/abs/1906.02399.
88. Almaslukh, B., Artoli, A., & Al-Muhtadi, J. (2018). A robust deep learning approach for position-independent smartphone-based human activity recognition. *Sensors, 18*(11), 3726. https://doi.org/10.3390/s18113726
89. Dong, M., Han, J., He, Y., & Jing, X. (2018). HAR-Net: Fusing deep representation and hand-crafted features for human activity recognition. *Signal and Information Processing, Networking and Computers, 550*, 32–40. https://doi.org/10.1007/978-981-13-7123-3_4
90. Russell, R., & Norvig, P. (1999). *Artificial Intelligence: A Modern Approach*. (3rd ed.). One Lake Street Upper Saddle River, NJUnited States: Prentice Hall Press.
91. Rokni, S.-A., Nourollahi, M., Ghasemzadeh, H. Personalized human activity recognition using convolutional neural networks, p. 2.
92. Hammerla, N. Y., Halloran, S., Plotz, T. (2020). Deep, convolutional, and recurrent models for human activity recognition using wearables. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence* Accessed: Apr. 03, 2020. [Online]. Available: https://dl.acm.org/Doi/https://doi.org/10.5555/3060832.3060835.

93. San, P.P., Kakar, P., Li, X.-L., Krishnaswamy, S., Yang, J.-B., Nguyen, M. N. (2017). Chapter 9—Deep learning for human activity recognition. *Big Data Analytics for Sensor-Network Collected Intelligence*, H.-H. Hsu, C.-Y. Chang, and C.-H. Hsu, Eds. Academic Press, pp. 186–204.
94. Gudur, G. K., Sundaramoorthy, P., Umaashankar, V. (2019). ActiveHARNet: Towards On-Device deep bayesian active learning for human activity recognition, pp. 7–12. Doi: https://doi.org/10.1145/3325413.3329790.
95. Long, J., Sun, W., Yang, Z., & Raymond, O. I. (2019). Asymmetric residual neural network for accurate human activity recognition. *Information*. https://doi.org/10.3390/info10060203
96. Zeng, M., Gao, H., Yu, T., Mengshoel, O.J., Langseth, H., Lane, I. and Liu, X., (2018). Understanding and improving recurrent networks for human activity recognition by continuous attention. In *Proceedings of the 2018 ACM International Symposium on Wearable Computers*. https://dl.acm.org/Doi/https://doi.org/10.1145/3267242.3267286.
97. Chen, C., Li, K., Duan, M., Li, K. (2017). Chapter 6—extreme learning machine and its applications in big data processing. *Big Data Analytics for Sensor-Network Collected Intelligence*, H.-H. Hsu, C.-Y. Chang, and C.-H. Hsu, Eds. Academic Press, pp. 117–150.
98. Yang, F., Zhang, W., Tao, L., Ma, J. (2020). Transfer learning strategies for deep learning-based PHM algorithms. *Applied Science*, 10(7), Art. no. 7. Doi: https://doi.org/10.3390/app10072361.
99. Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C. (2018). A survey on deep transfer learning. *ArXiv180801974 Cs Stat* Accessed: Feb. 08, 2021. [Online]. Available: http://arxiv.org/abs/1808.01974.
100. Cao, H., Nguyen, M.N., Phua, C., Krishnaswamy, S., Li, X.-L. (2012). An integrated framework for human activity classification. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, New York, NY, USA, pp. 331–340. Doi: https://doi.org/10.1145/2370216.2370268.
101. Jiang, W., Yin, Z. (2015). Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd ACM international conference on Multimedia* (pp. 1307–1310). [Online]. Available: https://dl.acm.org/citation.cfm?id=2806333.
102. Kumar, R.C., Bharadwaj, S.S., Sumukha, George, K. (2016). Human activity recognition in cognitive environments using sequential ELM. In *2016 Second International Conference on Cognitive Computing and Information Processing (CCIP)* (pp. 1–6). Doi: https://doi.org/10.1109/CCIP.2016.7802880.
103. Alsheikh, M. A., Selim, A., Niyato, D., Doyle, L., Lin, S., & Tan, H.-P. (2016). *Deep Activity Recognition Models with Triaxial Accelerometers*. (p. 6). AAAI Workshop Artif: Intell. Appl. Assist. Technol. Smart Environ.
104. San-Segundo, R., Montero, J. M., Barra-Chicote, R., Fernández, F., & Pardo, J. M. (2016). Feature extraction from smartphone inertial signals for human activity segmentation. *Signal Processing, 120*, 359–372. https://doi.org/10.1016/j.sigpro.2015.09.029
105. Ronao, C., & Cho, S.-B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Systems with Applications,* https://doi.org/10.1016/j.eswa.2016.04.032
106. Jiang, W., Yin, Z. (2015). Human activity recognition using wearable sensors by deep convolutional neural networks. pp. 1307–1310. Doi: https://doi.org/10.1145/2733373.2806333.
107. Zhu, X., & Qiu, H. (2016). High accuracy human activity recognition based on sparse locality preserving projections. *PLoS ONE, 11*, e0166567. https://doi.org/10.1371/journal.pone.0166567

Nida Saddaf Khan received her BS degree in computer science and MBA (finance) degree from University of Karachi, Pakistan, in 2007 and 2010, respectively, and the MS degree in computer science from Institute of Business Administration (IBA), Karachi, Pakistan, in 2013. She is currently a Ph.D. scholar and has been teaching in the department of Computer Science, IBA, Karachi since 2014. She is co-supervising a research based project where the deep learning based algorithms can be used to understand and treat certain mental disorders such as anxiety. She has also served as a research assistant in Artificial Intelligence Lab at IBA, Karachi for 2 years to create the core engine of a fully-funded research project for Anti Money Laundering (AML) solution. She has also taught as a visiting faculty in the Department of Computer Science, University of Karachi from 2008 to 2013. Her research interests include but not limited to deep learning, machine learning, IoT, smart home/building/city, sensor data analytics, probabilistic reasoning and optimization techniques.

Muhammad Sayeed Ghani has a B.S. from MIT (1984) and M.S. and Ph.D from Columbia University in Electrical Engineering (1986 and 1990). His field of expertise is applications of AI and Machine Learning to the Internet of Things. Dr. Ghani is presently professor in the department and has in the past been Associate Dean of the Faculty of Computer Science and been a member of the Board of Governors of IBA, and served as interim Acting Executive Director, IBA in 2016 and 2020. He has over 12 years of industry experience prior to joining IBA in 2001 as Chairman of the Center for Computer Studies. Dr. Ghani has worked as a Member of Technical Staff at NYNEX Science & Technology (NYNEX became part of Verizon Communications) in the US, where he was heavily involved in the Standards process of high-speed voice and data services. He has also been employed as IT Head for ABN AMRO Bank and State Life Insurance Corporation of Pakistan. Dr. Ghani has been a pioneer in introducing VSAT and Internet services in Pakistan and has been involved in various software development projects throughout his career. He has also worked at GTE Labs (now Verizon LTE Innovation Center) and the MIT Laboratory for Computer Science (now MIT CS & AI Lab) in the US and Unilever in Pakistan. Dr. Ghani has a number of publications in leading IEEE journals and conferences and has presented papers and conducted seminars in various forums in both the US and Pakistan. His current research interests at IBA include simulation and performance analysis of networks and Wireless Sensor Networks, Internet of Things, and applications of Machine Learning in Wireless Networks and IoT. He is presently also supervising five PhD students.