Journal of Reliable Intelligent Environments (2021) 7:189–213
https://doi.org/10.1007/s40860-021-00147-0

REVIEW

Trends in human activity recognition using smartphones

Anna Ferrari[1] · Daniela Micucci[1] · Marco Mobilio[1] · Paolo Napoletano[1]

Received: 15 April 2021 / Accepted: 24 June 2021 / Published online: 3 July 2021

Abstract

Recognizing human activities and monitoring population behavior are fundamental needs of our society. Population security, crowd surveillance, healthcare support and living assistance, and lifestyle and behavior tracking are some of the main applications that require the recognition of human activities. Over the past few decades, researchers have investigated techniques that can automatically recognize human activities. This line of research is commonly known as Human Activity Recognition (HAR). HAR involves many tasks: from signals acquisition to activity classification. The tasks involved are not simple and often require dedicated hardware, sophisticated engineering, and computational and statistical techniques for data preprocessing and analysis. Over the years, different techniques have been tested and different solutions have been proposed to achieve a classification process that provides reliable results. This survey presents the most recent solutions proposed for each task in the human activity classification process, that is, acquisition, preprocessing, data segmentation, feature extraction, and classification. Solutions are analyzed by emphasizing their strengths and weaknesses. For completeness, the survey also presents the metrics commonly used to evaluate the goodness of a classifier and the datasets of inertial signals from smartphones that are mostly used in the evaluation phase.

Keywords ADL · Human activity recognition · Machine learning · Deep learning · Smartphone

1 Introduction

The first work on human activity recognition dates back to the late ’90s [1]. During the last 30 years, the Human Activity Recognition (HAR) research community has been very active proposing several methods and techniques. In recent years, significant research has been focused on experimenting with solutions that can recognize Activities of Daily Living (ADLs) from inertial signals. This is mainly due to two factors: the increasingly low cost of hardware and the wide spread of mobile devices equipped with inertial sensors. The use of smartphones to both acquire and process signals opens opportunities in a variety of application contexts such as surveillance, healthcare, and delivering [2–4].

In the context of HAR, most of the classification methods rely on the Activity Recognition Process (ARP) protocol. As depicted in Fig. 1, ARP consists of five steps, acquisition, preprocessing, segmentation, feature extraction, and classification.

The *data acquisition* step is in charge of acquiring data from sensors. Data generally originate from sensors such as accelerometers, compasses, and gyroscopes. Data acquired from sensors typically include artifacts and noise due to many reasons, such as electronic fluctuation, sensors calibration, and malfunctions. Thus, data have to be processed.

The *preprocessing* step is responsible for removing artifacts and noise. Generally, preprocessing is based on filtering techniques. The output of the step is a set of filtered data that constitute the input for the next step.

The *data segmentation* step is responsible of splitting data into segments, also called windows. Data segmentation is a common practice which facilitates the next step.

The *feature extraction* step aims to extract the most significative portion of information from the data to be given to the classification algorithm while reducing data dimension.

The *classification* is the last step of the process. It consists in training and testing the algorithm. That is, the parameters of the classification model are estimated during the training procedure. Thereinafter, the classification performances of the model are tested in the testing procedure.

This paper presents a review of the techniques and methods commonly adopted in the steps of the ARP process. The

✉ Daniela Micucci
daniela.micucci@unimib.it

[1] Department of Informatics, Systems and Communication, University of Milano - Bicocca, Milan, Italy

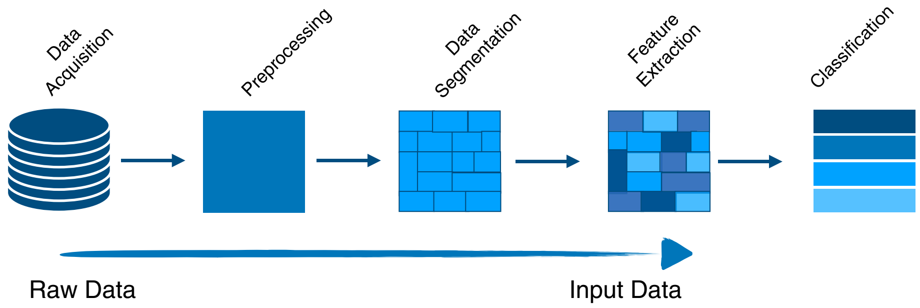

Fig. 1 Activity recognition process

focus is on techniques and methods that have been experimented and proposed for smartphone. Therefore, this review does not include other types of devices used in HAR. The choice to consider smartphones only is due to the increasing attention paid to these devices by the scientific community as a result of their valuable equipment and their wide diffusion.

The paper also provides an overview of the most used datasets for evaluating HAR techniques. Since this review is focused on smartphones, the datasets included are those of inertial signals collected using smartphones.

The analysis of the state-of-the-art encompasses scientific articles selected based on the following criteria and keywords:

- first 100 papers found in Google Scholar with keywords: *human activity recognition smartphone*,
- first 100 papers found in Google Scholar with keywords: *human activity recognition smartphone* staring from 2015,
- first 100 papers found in Google Scholar with keywords: *personalized human activity recognition smartphone*,
- first 100 papers found in Google Scholar with keywords: *personalized human activity recognition smartphone* staring from 2015.

The selection of the papers has been completed on March 2020.

We initially removed duplicates from the resulting articles. Then, we manually checked the remaining papers by reading the abstract, the introduction, and the conclusion sections to quickly eliminate those articles that are out of the scope of our survey. The articles that we excluded are those that rely on devices other than smartphones, those that use smartphones in conjunction with other devices, those that use sensors different from the inertial ones, and those that deal with complex ADLs such as preparing a meal, taking transportation, and so on.

The paper is organized as follows.

Section 2 introduces the problem related to human activity recognition. Section 3 describes the data acquisition step and, thus, the sensors that are mainly exploited in HAR for data acquisition. Section 4 presents the preprocessing activity that is normally performed on the raw data as acquired by the sensors. Sections 5 and 6 describe the commonly used segmentation strategies and features, respectively. Section 7 introduces the most recent classification methods, their strength, and weakness. Moreover, the Section discusses personalization and why it is important to improve the overall classification performance. Given the importance of datasets in the evaluation process of techniques and methods, Sect. 8 discusses the characteristics of a set of publicly available datasets often used in the evaluation of classifiers. Section 9 summarizes the lessons learned and provides some guidance on where the research should focus. Finally, Sect. 10 sketches the conclusions.

2 Background

This section is intended to provide a quick overview of the recognition process of activities of daily living. The details are then discussed in more detail in the respective sections.

The goal of human activity recognition (HAR) is to *automatically* analyze and understand *human actions* from *signals* acquired by multimodal *wearable* and/or *environmental* devices, such as accelerometer, gyroscope, microphones, and camera [5].

Recently, research has been shifting toward the use of wearable devices. There are several reasons that have led to this shift, which include lower costs as they do not require a special installation, the use also outside the home, and a greater willingness to use as perceived by users as less intrusive respecting their privacy.

Among wearable devices, recently, smartphones are the most widely used compared to ad hoc devices. This is mainly due to the fact that the smartphone is now widely used even in the older population and is always ‘worn’ without being perceived as an additional element of disturbance, because it is now integrated into the daily routine.

Figure 2 shows the recognition process. To the left are the sensors that are the source of the data required to recognize activities, whereas to the right are activities of daily living that are recognized by the ‘activity recognition’ chain (in the middle).

The potentially recognizable activities vary in complexity: walking, jogging, sitting, and standing are examples of the most simple ones; preparing a meal, shopping, taking a bus, and driving a car are examples of the most complex ones. Depending on the complexity, different techniques and types of signals are implemented. We are interested in activities that belong to the category of the simplest ones.

When the wearable device is a smartphone, the most commonly used sensors are the accelerometer, gyroscope, and magnetometer. Therefore, the first step of the Activity Recognition Process (ARP) introduced in Sect. 1 (*Data Acquisition*) requires to be able to interface with the sensors and to acquire the signals with the required frequencies. This step is detailed in Sect. 3.

As the signals are acquired, they undergo an elaboration process whose purpose is to remove the noise caused by the user and the sensors. Generally, high-pass, low-pass filters, and average smoothing methods are applied. This corresponds to the second step (*Preprocessing*) of the ARP that is detailed in Sect. 4.

The continuous pre-processed data stream is then split into segments whose dimensions and overlaps may vary according to several factors such as the technique used to classify, the type of activity to be detected, and the type of signals to be processed. This corresponds to the third step (*Segmentation*) of the ARP process that is detailed in Sect. 5.

The segments of pre-processed signals are then elaborated to extract significative features. This step (*Feature extraction* in the ARP process) is crucial for the performance of the final recognition. Two main types of features are commonly used: hand-crafted features (which are divided into time-domain and frequency-domain) and learned features that are automatically discovered. Feature extraction is detailed in Sect. 6.

The last step of the ARP process is *Classification*. For many years, this step was accomplished through the exploitation of traditional machine learning techniques. More recently, due to promising results in the field of video signal processing, deep learning techniques have also been used. More recently, due to the problem known as *population diversity* [6] (which is related to the natural users heterogeneity in terms of data), researchers have applied recognition techniques based on personalization to obtain better results. Classification is detailed in Sect. 7.

3 Data acquisition

Historically, human activity recognition techniques exploited both environmental devices and ad hoc devices worn by subjects [7]. Commonly used environment devices include cameras [8–11], and other sensors such as, for example RFID [12], acoustic sensors [13], and WiFi [14]. The ad hoc devices were worn by people on different parts of their bodies and included typically inertial sensors [7].

Over the past decade, a considerable progress in hardware and software technologies has modified habits of the entire population and business. On one hand, the micro-electro-mechanical systems (MEMS) have reduced sensors size, cost, and power needs of sensors, while capacity, precision, and accuracy have increased. On the other hand, Internet of Things (IoT) has enabled the spread of easy and fast connections between devices, objects, and environments. The pervasiveness and the reliability of these new technologies enables the acquisition and storage of a large amount of multimodal data [15].

Thanks to these technological advances, smartphones, smartwatches, home assistants, and drones are daily used and represent essential instruments for many economy businesses, such as remote healthcare, merchandise delivering, agriculture, and others [16]. These new technologies together with the large availability of data gained the attention from the research communities, including HAR.

The goal of this section is to present the most used wearable devices for data acquisition in HAR, which are a consequence of the technological advances discussed above.

Wearable devices encompass all accessories attached to the person's body or clothing incorporating computer technologies, such as smart clothing, and ear-worn devices [17]. They enable to capture attributes of interest as motion, location, temperature, and ECG, among others.

Nowadays, smartphones and smartwatches are the most used wearable devices among the population. In particular, the smartphone is one of the most used devices in people's daily lives and it has been stated that it is the first thing people reach for after waking up in the morning [18,19].

Smartphone's pervasiveness over last years is due mostly because it provides the opportunity to connect with people, to play games, to read emails, and, in general, to achieve almost all online services that a user needs. In particular, their high diffusion is a crucial aspect, because the more the users, the more the data availability. The more data availability, the more information and the more the possibility to create robust models.

A the same time, smartphones are preferable over other wearables, because a huge amount of sensors and softwares are already installed and permit to acquire many kind of data, potentially, all day long.

The choice of the sensors plays an important role for the activity recognition performances [20].

Accelerometers, gyroscopes, and magnetometers are the most used sensors for HAR tasks and classification.

- *Accelerometer*. The accelerometer is an electromechanical sensor that captures the rate of change of the velocity

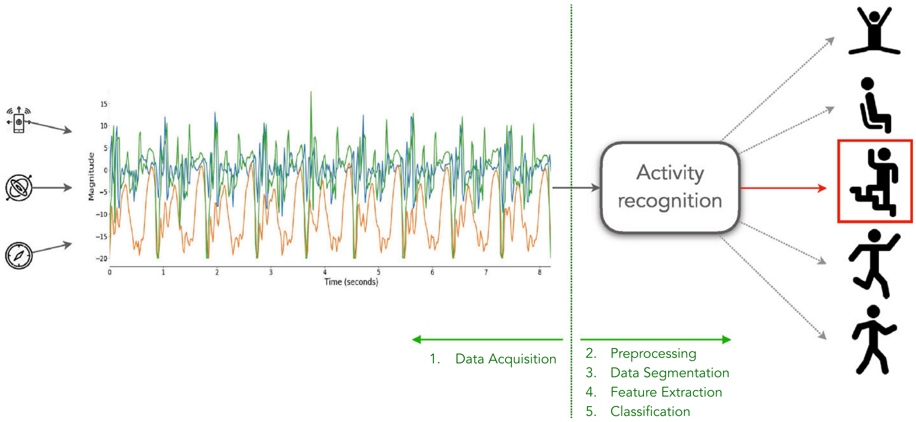

Fig. 2 An abstracted overview of the human activity recognition process

of an object over a time laps, that is, the acceleration. It is composed of many other sensors, including some microscopic crystal structures that become stressed due to accelerative forces. The accelerometer interprets the voltage coming from the crystals to understand how fast the device is moving and which direction it is pointing in. A smartphone records three-dimension acceleration, which join the reference devices axes. Thus, a trivariate time series is produced. The measuring unit is meters over second squared (m/s^2) or g forces.

– *Gyroscope*. The gyroscope measures three-axial angular velocity. Its unit is measured in degrees over second ($degrees/s$).
– *Magnetometer*. A magnetometer measures the change of a magnetic field at a particular location. The measurement unit is Tesla (T), and is usually recorded on the three axes.

In addition to accelerometers, gyroscopes, and magnetometers, other less common sensors are used in HAR. For example, Garcia-Ceja and Brena use a *barometer* to classify vertical activities, such as ascending and descending stairs [21]. Cheng et al. [22] and Foubert et al. [23] use *pressure* sensors arrays to detect respectively activities and lying and sitting transitions. Other researchers use biometric sensors. For example, Zia et al. use *electromyography* (EMG) for fine-grained motion detection [24], and Liu et al. use *electrocardiography (ECG)* in conjunction with accelerometer to recognize activities [25].

Accelerometer is the most popular sensor in HAR, because it measures the directional movement of a subject's motion status over time [26–31]. Nevertheless, it struggles to resolve lateral orientation or tilt, and to find out the location of the user, which are precious information for activity recognition.

For these reasons, some sensor combinations have been proposed as valid solution in HAR. In most of the cases, accelerometer and gyroscope are used conjointly to both acquire more information about the device movements, and to possibility infer the device position [32–36]. Moreover, Shoaib et al. demonstrated that gyroscope-based classification achieves better results than accelerometer for specific activities, such as walking downstairs and upstairs [35]. Furthermore, as afore mentioned, gyroscope data permit to infer device position that drastically impacts recognition performances [37,38].

Other studies combined accelerometer and magnetometer simultaneously [39], acceleration and gyroscope with magnetometer [40,41], accelerometer with microphone and GPS [6], and other combinations [42].

An important factor to consider in the acquisition step is the sample rate that influences the number of available samples for the classification step. The *sampling rate* is defined as the number of data points recorded in a second and is expressed in *Hertz*. For instance, if the sampling rate is equal to 50Hz, it means that 50 values per second are recorded. This parameter is normally set during the acquisition phase.

In the literature, different sampling rates have been considered. For instance, in [43], the sampling rate is set at 50 Hz, in [44] at 45 Hz, and from 30 to 32 Hz in [32]. Although the choice is not unanimous in the literature, 50 Hz define a suitable sampling rate that properly permits to model human activities [45].

4 Preprocessing

In a classification pipeline, data preprocessing is a fundamental step to prepare raw data for further steps.

Raw data coming from sensors often present artifacts due to instruments, such as electronic fluctuation or sensor calibration, or to the physical activity its self. Data have to be cleaned to exclude from the signals these artifacts.

Moreover, accelerometer signal combines the linear acceleration due to body motion and due to gravity. The presence of the gravity is a bias that can influence the accuracy of the classifier, and thus is a common practice to remove the gravity component from the raw signal.

For all the reasons mentioned above, a filtering procedure is normally executed. Filters are powerfully instruments which acting on frequency component of the signal.

The high-frequency component of the accelerometer signal is mostly related to the action performed by the subjects, while the low-frequency component of the accelerometer signal is mainly related to the presence of gravity [46–48].

Usually, a low-pass filter with cut-off frequency ranging between 0.1 and 0.5 Hz is used to isolate the gravity component. To find the body acceleration component, the result of the low-pass filtered signal is subtracted from the original signal [49–51].

Filtering is also used to clear raw data from artifacts. It is stated that a cut-off frequency of 15Hz is enough to capture human body motion which energy spectrum lies between 0 Hz and 15 Hz [49,52].

5 Data segmentation

Data segmentation partitions signals into smaller data segments, also called windows.

Data segmentation helps in overcoming some limitations due to acquisition and pre-proccessing aspects. First, data sampling: data recorded from different subjects may present different lengths in time which is generally a limit for the classification process. Second, time consumption: multidimensional data can lead to a very high computational time consumption. Splitting data into smaller segments helps the algorithm to face with high volumes of data. Third, it helps the computation of the features extraction procedure in terms of more simplicity and lower time consumed.

Window characteristics are influenced by: (a) the type of windowing, (b) the size of the window, and (c) the overlap among contiguous windows.

5.1 Window type

Three main types of windowing are mainly used in HAR: activity-defined windows, event-defined windows, and sliding windows [53].

In *activity-defined windowing*, the initial and end points of each window are selected by detecting patterns of the activity changes.

In *event-defined windowing*, the window is created around a detected event. In some studies, it is also mentioned as *windows around peak* [43].

In *sliding windowing*, data are split into windows of fixed size, without gap between two consecutive windows, and, in some cases, overlapped, as shown in Fig. 4. Sliding windowing is the most widely employed segmentation technique in activity recognition, especially for periodic and static activities [54].

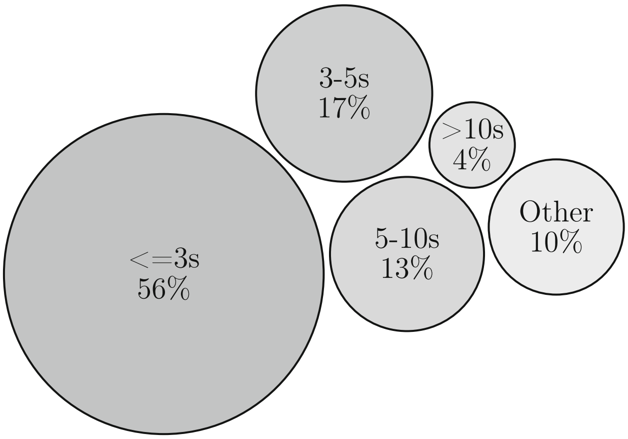

Fig. 3 State-of-the-art Sliding Window's Size

5.2 Window size

The choice of the window size influences the accuracy of the classification [55]. However, its choice is not trivial. Windows should be large enough to guarantee to contain at least one cycle of an activity and to differentiate similar movements. At the same time, incrementing its dimension does not necessarily improve the performance. Shoaib et al. show that 2 s is enough for recognizing basic physical activities [35].

Figure 3 shows the distribution of windows size among the state-of-the-art studies we considered. It is possible to notice that the most frequently used window size does not exceed 3 s.

The impact of windows sizes on the classification performance still remains a challenging task for the HAR community and continues to be largely studied in the literature [35,54,56].

5.3 Window overlap

Another parameter to consider is the percentage of overlap among consecutive windows. Sliding windows are often overlapped, which means that a percentage of a window is repeated in the subsequent window. This leads to two main advantages: it avoids noise due to the truncation of data during the windowing process, and increases the performance by increasing the data points number.

Generally, the higher the number of data points, the higher the classification performance. For these reasons, overlapped sliding windows are the most common choice in the literature.

Figure 5 shows the distribution of the percentage overlap in the state-of-the-art. In more than 50% of the proposals we selected, an overlap of 50% has been chosen. Some approaches avoid any overlap [29,32,44,57], claiming faster

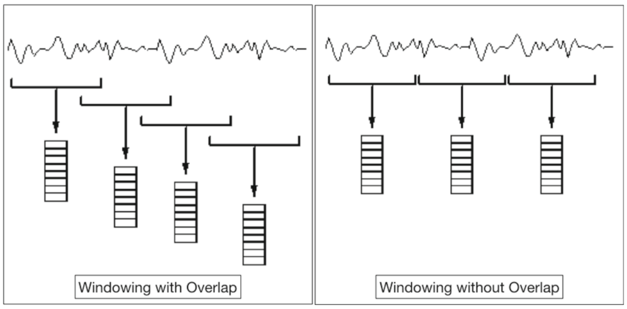

Fig. 4 Sliding windows with and without Overlap

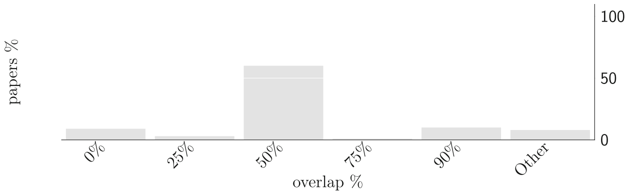

Fig. 5 Distribution of % of overlap in the state-of-the-art

responses in real-time environments and better performances in detection of short duration movements.

At the end of the segmentation step, data are organized in vectors $\mathbf{v}_i$ as follows:

$$\mathbf{v}_i = (\underbrace{x_1,\ x_2,\ \dots x_n,}_{x-dimension}\ \underbrace{y_1,\ y_2,\ \dots\ y_n,}_{y-dimension}\ \underbrace{z_1,\ z_2,\ \dots\ z_n}_{z-dimension}),$$

where x, y, z are the three-axis acceleration values, and $\mathbf{v}_i$ is a $1 \times (n \times k)$ vector that represents the ith window. k refers dimensionality of the sensor; for instance, a 3-axial acceleration has value of $k = 3$. The number n is the total length of the windows, and it depends on two factors: the size of the widows, normally in seconds, and the *sampling rate*.

6 Feature extraction

Theoretical analysis and experimental studies indicate that many algorithms scale poorly in domains with large number of irrelevant and\or redundant data [58].

The literature shows that using a set of features instead of raw data improves the classification accuracy [59]. Furthermore, features extraction reduces the data dimensionality while extracting the most important peculiarity of the signal by abstracting each data segment into a high-level representation of the same segment.

From a mathematical point of view, features extraction is defined as a process that extracts a set of new features from the original data segment through some functional mapping [60]. For instance, let be $\mathbf{x} = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^n$ a segment of data, an extracted feature f_i is given by

$$f_i = g_i(x_1, x_2, \dots, x_n) \quad \text{for } i = 1, \dots, m,$$

where $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is a map. The features space is of dimension $m \leq n$, which means that features' extraction reduces raw data space dimension.

In the classification context, the choice of g_i is crucial. In fact, in the recognition process, g has to be chosen, such that the original data are mapped in separated regions of the features space. In other words, the researcher assumes that in the feature space, data diversify better than in the original space.

The accuracy of activity recognition approaches dramatically depends on the choice of the features [55].

In the literature, the way features g_i are extracted is divided into two main categories: *hand-crafted features* and *learned features*.

6.1 Hand-crafted features

Hand-crafted features are the most used features in HAR [61–63]. The term “hand-crafted” refers to the fact that the features are selected from an expert using heuristics.

Hand-crafted features are themselves generally split in *time-domain* and *frequency-domain* features. The signal domain is changed from time to frequency based on the Fourier transformation.

Table 1 shows some of the most used time-domain and frequency-domain features.

Low computational complexity and calculation simplicity make hand-crafted features still a good practice for activity recognition.

Nevertheless, they present many disadvantages, such as a high dependency on the sensor choice and the reliance on the expert knowledge. Hence, a different set of features need to be defined for each different type of input data, that is, accelerometer, gyroscope, time domain, and frequency domain. In addition, hand-crafted features highly depend on experts' prior knowledge and manual data investigation and it is still not always clear which features are likely to work best.

It is a common practice to chose the features through empirical evaluation of different combinations of features or with the aid of feature selection algorithms [64].

Table 1 Hand-crafted features in time domain and frequency domain. $FFT(f)$ is the Fourier transformation of the signal f

Time domain features Feature name	Formula	Description
Minimum	$\min_{j=1,\ldots n}(x_j)$	Minimum value of a given segment in each dimension
Maximum	$\max_{j=1,\ldots n}(x_j)$	Maximum value of a given segment in each dimension
Mean	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$	Mean value of a given segment in each dimension
Median	$Me = x_{0.5} : F(x_{0.5}) \leq 0.5$	Median value of a given segment in each dimension
Standard Deviation	$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$	Standard deviation of a given segment in each dimension
Variance	$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$	Variance of a given segment in each dimension
Interquartile Difference	$ID = x_{0.75} - x_{0.25}$	Difference between third and first quartile of a given segment in each dimension
Skewness	$skw = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{s^3}$	Skewness value of a given segment in each dimension
Kurtosis	$kurt = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{s^4}$	Kurtosis value of a given segment in each dimension
Root mean square	$rms = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}$	Root mean square value of a given segment in each dimension
Total Sum	$ts = \sum_{i=1}^{n} x_i$	Total sum value of a given segment in each dimension
Range	$R = max - min$	Range of a given segment in each dimension
Mean of Peak's distance	$m_p = \frac{1}{s^2}\sum_{j=1}^{s}\sum_{i=1}^{s} d(p_i, p_j)$	Mean of distance between peaks of a given segment in each dimension
Fourth central moment	$m_4 = \frac{1}{n}\sum_{j=1}^{n}(x - \bar{x})^4$	Fourth central moment of a given segment in each dimension
Fifth central moment	$m_5 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^5$	Fifth central moment of a given segment in each dimension

Frequency domain features Feature name	Formula	Description
Entropy	$H(x) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i)$	Normalized information entropy of the discrete FFT components
Sum of the spectral power components	$ID = x_{0.75} - x_{0.25}$	Difference between third and first quartile of a given segment in each dimension
Mean of the spectral components	$\mu_f = \frac{1}{n}\sum_{j=1}^{n} FFT_j$	Mean of FFT distribution
Median of the spectral components	$Me_f = FFT_{0.5} : F(FFT_{0.5}) = 0.5$	Median of FFT distribution
First cepstral coefficient	$c(1) = \mathcal{F}^{-1}\{\log \lvert FFT(f) \rvert\}$	First coefficient of the cepstrum transformation

6.2 Learned features

The goal of feature learning is to automatically discover meaningful representations of raw data to be analyzed [65].

According to [66], the main features learning methods from sensor data are the following:

- *Codebooks* [67,68] considers each sensor data window as a sequence, from which subsequences are extracted and grouped into clusters. Each cluster center is a codeword. Then, each sequence is encoded using a bag-of-words approach using codewords as features.
- *Principal Component Analysis (PCA)* [69] is a multivariate technique, commonly used for dimensionality reduction. The main goal of PCA is the extraction of a set of orthogonal features, called principal component, which are linear combination of the original data and such as the variance extracted from the data is maximal. It is also used for features selection.
- *Deep Learning* uses Neural Networks engines to learn patterns from data. Neural Networks are composed from a set of layers. In each layer, the input data are transformed through combinations of filters and topological maps. The output of each layer becomes the input of the following layer and so on. At the end of this procedure, the result is a set of features more or less abstract depending on the number of layers. The higher the number of layers is, the more the features are abstract. These features can be used for classification. Different deep learning methods for features extraction have been used for time series analysis [70].

Features learning techniques avoid the issue of manually defining and selecting the features. Recently, promising results are leading the research community to exploit learned features in their analysis.

7 Classification

Over the last years, hardware and software development has increased wearable devices capability to face with complex applications and tasks. For instance, smartphones are, nowadays, able to acquire, store, share, and elaborate huge amount of data in a very short time. As a consequence of this technological development, new instruments related to the data availability, data processing, and data analysis are born.

The capability of a simple smartphone to meet some complex tasks (e.g., steps count and life style monitoring) is the result of very recent scientific changes regarding methods and techniques.

In general, more traditional data analysis methods, based on model-driven paradigms, have been largely substituted by more flexible techniques, developed during recent years, based on data-driven paradigms. The main difference between these two approaches is given by the a priori assumption about the relationship between independent and response variables. Thus, given a classification model, $\mathbf{y} = f(\mathbf{x})$, model-driven approaches state that f is (or can be) determined by assumptions on the distribution of the underlying stochastic process that generates $\mathbf{x}$. f is build through a set of rules, or algorithms, which choices depend on data with an unknown distribution. On the opposite, in data-driven paradigms, f is unknown and depends directly on the data and on the choice of the algorithm.

The strength and the success of data-driven approaches is due to their capability to manage and to analyze large amount of variables that characterize a phenomenon without assuming any a priori relation between the independent and response variables. From a certain point of view, this flexibility can be a weakness, because the lack of a well-known relation also can be interpreted as a lack of cause–effect knowledge.

In model-driven approaches, in contrast, cause–effect relation is known by definition. However, model-driven approaches loose in performance in estimating high-dimensionality relations.

In activity recognition context, model-driven approaches are less powerful and data-driven approaches are preferred [71].

Among data-driven algorithms, Artificial Intelligence (AI) have produced very promising results over the last years and have been largely used for data analysis, for information extraction, and for classification tasks. AI algorithms encompasses machine learning which, in turns, encompasses deep learning methods.

Machine learning uses statistical exploration techniques to enable the machine to learn and improve with the experiences without being explicitly programmed. Deep learning emulates human neural system to analyze and extract features from data. In this survey, we focus on machine learning and deep learning algorithms.

The choice of the classification algorithm drastically influences the classification performance, but up to now, there is no evidence of a best classifier and its choice still remains a challenging task for the HAR community.

In particular, machine learning and deep learning methods struggle to achieve good performances for new unseen users. This loss of performance is mostly caused by the subjects variabilities, also called *population diversity* [6], which is related to the natural users heterogeneity in terms of data. The following sections present both traditional state-of-the-art machine learning and deep learning techniques, and personalized machine learning and deep learning techniques as solutions to overcome the population diversity problem.

7.1 Traditional learning methods

Artificial Intelligence (AI) algorithms are based on the emulation of the human learning process. According to [72], the word *learning* refers to a process to acquire knowledge or skill about a thing. A thing can always be viewed as a system, and the general architecture of the knowledge of the thing follows the *FCBPSS* architecture, in which F is a function that refers to the role a particular structure plays in a particular context; C is a context that refers to the environment as well as pre-condition and post-condition surrounding a structure; B is a behavior that refers to causal relationships among states of a structure; P is a principle that refers to the knowledge that governs a behavior of a structure; S is a state that describes the property or character of a structure; S is a structure that represents elements or components of the system or thing along with their connections [73].

Machine learning and deep learning both refer to the word *learning* and, indeed, they are implemented, so that they emulate the human capability of learning.

Machine learning techniques used in HAR are mostly divided into supervised and unsupervised approaches. *Supervised* machine learning encompasses all techniques that rely on labeled data. *Unsupervised* machine learning are techniques which are based on data devoid of labels.

Let $\mathbf{x}$ and $\mathbf{y}$ be, respectively, a set of data and their corresponding labels. A *classification task* is a procedure whose goal is to predict the value of the label $\hat{\mathbf{y}}$ from the data input $\mathbf{x}$. In other terms, assuming that there exists a linear or non-linear relation f between $\mathbf{x}$ and $\mathbf{y}$, the goal of the classification is to find f such as the prediction's error, that is, the distance between $\mathbf{y}$ and $\hat{\mathbf{y}}$, is minimal.

In supervised machine learning, data and corresponding labels are known and the algorithm learns f by iterating a procedure until the global minimum of a loss function is reached. The loss function is again a measure about the prediction's error, estimated by the difference between $\mathbf{y}$ and $\hat{\mathbf{y}}$. The optimization procedure, that is, finding the loss global minimum, is computed on the *training dataset*, which is a subset of the whole dataset. Once the minimum is achieved, the model is ready to be tested on the *test dataset*. The *algorithm performance* measures the model's capability to classify new instances (Sect. 7.3 discusses details about the performance measures).

In unsupervised approaches, the labels $\mathbf{y}$ are unknown and the evaluation of the algorithm goodness bases on statistical indices, such as the variance or the entropy.

Consequently, the choice between supervised or unsupervised methods determines how the relation f between $\mathbf{x}$ and $\mathbf{y}$ is learnt. Since a human activity recognition system should return a label such as walking, sitting, running, and so on, most of HAR systems work in a supervised fashion. Indeed, it might be very hard to discriminate activities in a completely unsupervised context [7].

Table 2 Kernel in support vector machines

Kernel	Linear	Polynomial	RBF
Formula	$\mathbf{x}_i^T \mathbf{x}_j$	$(\mathbf{x}_i^T \mathbf{x}_j + c)^d$	$exp\left(\frac{\lVert\mathbf{x}_i - \mathbf{x}_j\rVert^2}{2\sigma^2}\right)$

Figure 6 shows the distribution of traditional machine learning and deep learning algorithms used for human activity recognition in the papers we selected for this survey. In the following paragraphs, we will describe the most used techniques in HAR with the related literature.

7.1.1 Traditional machine learning

Machine learning techniques have been largely used for activity recognition tasks. More and more sophisticated methods have been developed to face with the complexity related to activity recognition tasks. In this section, we describe traditional machine learning algorithms that have been mostly exploited for HAR, according to Fig. 6.

Support Vector Machine (SVM) belongs to *domain transform* algorithms. It implements the following idea: it is assumed that the input data $\mathbf{x}$ are not linearly separable with respect to the classes $\mathbf{y}$ in the data space, but there exists an higher dimensional space where the linearity is achieved. Once data are mapped into this space, a linear decision surface (or hyperplane) is constructed and used as recognition model. Thus, guided from the data, the algorithm searches for the optimal decision surface by minimizing the error function. The projection of the optimal decision surface into the original space marks the areas belonging to a specific class which is used for the classification [74]. The transformation of the original space into a higher dimensional space is made through a *kernel* which is defined as a linear or non-linear combination of the data, for example, polynomial kernel, sigmoid kernel, and radial basis function (RBF) kernel, see Table 2.

Originally, SVM have been implemented as two-class classifier. The computation of the multi-class SVM bases on two strategies: *one-versus-all* where one class is labeled with 0 and the other classes as 1, and *one-versus-one* where the classification is made between two classes at a time [75].

Among HAR classifiers, SVM is the most popular one [32,34,48,75–79].

***k*-Nearest Neighbors (*k*-NN)** is a particular case of *instance-based* method. The nearest-neighbor algorithm compares each new instance with existing ones using a distance metric (see Table 3), and the closest existing instance is used to assign the class to the new one. This is the simplest case where $k = 1$. If $k > 1$, the majority class of the closest

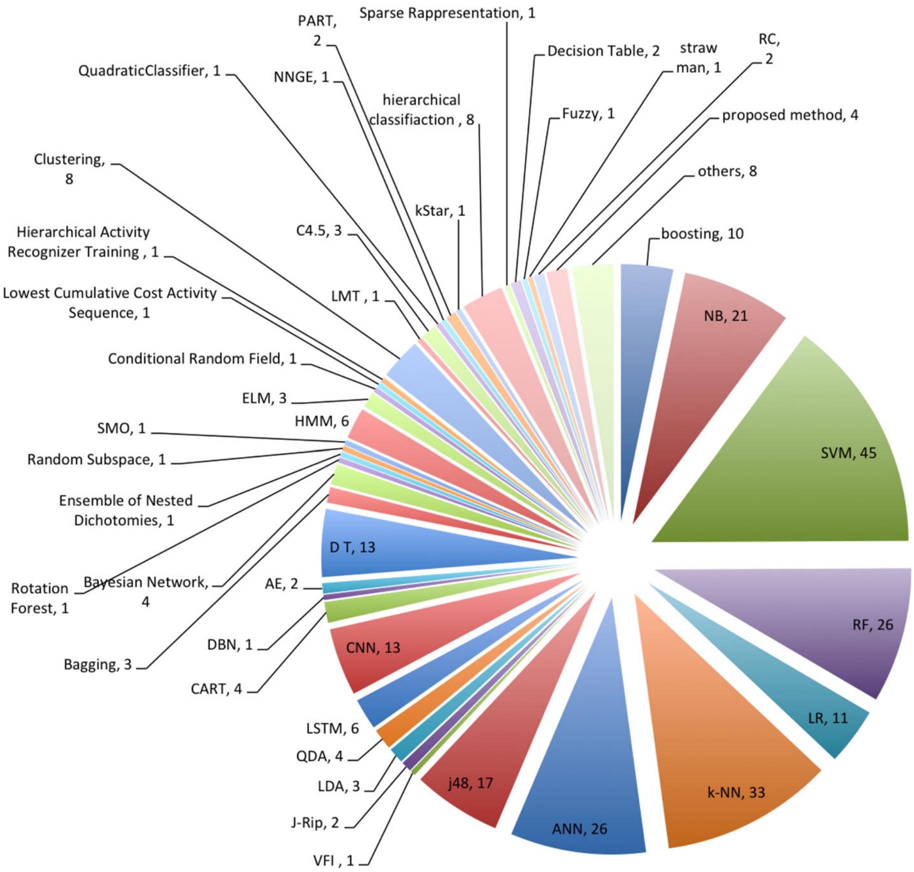

Fig. 6 Traditional machine learning and deep learning classifiers distribution

Table 3 Distance metrics in k-nearest neighbor

Distance	Formula
Euclidean	$\sqrt{\sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{x}_j)^2}$
City Block	$\sum_{i=1}^{n} \lvert\mathbf{x}_i - \mathbf{x}_j\rvert$
Chebychev	$\max_{i=1\ldots n} \lvert\mathbf{x}_i - \mathbf{x}_j\rvert$
Cosine	$1 - \frac{\mathbf{x}_i\mathbf{x}_j^T}{\sqrt{(\mathbf{x}_i\mathbf{x}_i^T)(\mathbf{x}_j\mathbf{x}_j^T)}}$
Correlation	$1 - \frac{(\mathbf{x}_i-\bar{\mathbf{x}}_i)(\mathbf{x}_j-\bar{\mathbf{x}}_j)^T}{\sqrt{(\mathbf{x}_i-\bar{\mathbf{x}}_i)(\mathbf{x}_i-\bar{\mathbf{x}}_i)^T}\sqrt{(\mathbf{x}_j-\bar{\mathbf{x}}_j)(\mathbf{x}_j-\bar{\mathbf{x}}_j)^T}}$
Mahalnobis	$\sqrt{(\mathbf{x}_i - \mathbf{x}_j)C^{-1}(\mathbf{x}_i - \mathbf{x}_j)^T}$ where C is the covariance matrix

k neighbors is assigned to the new instance [80]. It is a very simple algorithm and belongs to the lazy algorithms. Lazy algorithms have no parameters to learnt from the training phase [32,75–77,81]. k-NN depends only on the number k of the nearest neighbors.

Decision tree algorithms build a hierarchical model in which input data are mapped from the root to leafs through branches. The path between the root and a leaf is a classification rule [7]. Sometimes, the length of the trees has to be modified and growing and pruning algorithms are used. The construction of a tree involves determining split criterion, stopping criterion, and class assignment rule [82]. J48 and C4.5 are the most used decision tree in HAR [29,30,77,81].

Random Forest (RF) is a classifier consisting of a collection of tree-structured classifiers $\{h(\mathbf{x}, \Theta_k), k = 1, \ldots\}$ where the $\{\Theta_k\}$ are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input $\mathbf{x}$ [83]. Random Forest generally achieves high performance with high-dimensional data by increasing the number of trees [29,40,48,56,75,84, 85].

Naive Bayes (NB) belongs to *Bayesian* methods whose prediction of new instances is based on the estimation of the posterior probability as a product of the likelihood, which is a conditional probability estimated on the training set given the class, and a prior probability. In Naive Bayes, data are assumed independent given the class values. Thus, given y be a certain class and $\mathbf{x}_i \ldots \mathbf{x}_n$ the data, Naive Bayes classifier based on the Bayesian rules and the likelihood splits in the product of the conditional probabilities given the class

$$P(y|\mathbf{x}_1 \ldots \mathbf{x}_n) = \frac{P(y)P(\mathbf{x}_1, \ldots \mathbf{x}_n|y)}{P(\mathbf{x}_1, \ldots \mathbf{x}_n)} = \frac{P(y)\prod_{i=1}^{n} P(\mathbf{x}_i|y)}{P(\mathbf{x}_1, \ldots \mathbf{x}_n)}.$$

Decision rules is the maximum a posteriori (MAP) given by

$$\arg\max_{y} P(y|\mathbf{x}_1 \ldots \mathbf{x}_n) = \arg\max_{y} \prod_{i=1}^{n} P(\mathbf{x}_i|y).$$

Naive Bayes has been applied in activity recognition because of the simple assumption on the likelihood, which is usually violated in practice [29,77,81,86]

Adaboost is part of the *classifier ensembles*. Classifier ensembles encompass all algorithms that combine different classifiers together.

The combination between classifiers is meant in two ways: either using the same classifiers with different parameter's settings (e.g., random forest with different lengths), or using different classifiers together (e.g., random forest, support vector machines, and k-NN).

The ensemble classifiers encompass *bagging*, *stacking*, *blending*, and *boosting*. In bagging, n samplings are generated from training set and a model is created on each. The final output is a combination of each model's prediction. Normally, either the average or a quantile is used. In stacking, the whole training dataset is given to the multiple classifiers which are trained using the k-fold-cross-validation. After training, they are combined for final prediction. In blending, the same procedure as staking is performed, but instead of the cross-validation, the dataset is divided into training and validation. Finally, in boosting, the final classifier is composed of a weighted combination of models. The weights are initially equal for each model and are iteratively updated based on the models performance, as for Adaboost [6,7,87].

7.1.2 Traditional deep learning

Generally, the relation between input data and labels is very complex and mostly non-linear. Among Artificial Intelligence algorithms, Artificial Neural Network (ANN) is a set of supervised machine learning techniques which emulate human neural system with the aim at extracting non-linearity relations from data for classification.

Human neural system is composed by neurons (about 86 billions) which are connected with synapses (around 10^{14}). Neurons receive input signals from the outer (e.g., visual or olfactory) and based on the synaptic's strength they fire and produce some output signals to be transmit to other neurons. Artificial Neural Network bases on the same neurons and synapses concept.

In a traditional ANN, each data input value is associated with a neuron and its synapses strength is measured by a functional combination of input data $\mathbf{x}$ and randomly chosen *weights* $\mathbf{w}$. This value is passed to an *activation functions* σ which is responsable to determine the synapse strength and eventually to fire the neuron. The output of the activation function is given by $y = \sigma(\mathbf{w}^T\mathbf{x})$. If it fires, the output becomes the next neuron's input. Table 4 provides more details about activation functions.

A set of neurons is called *layer*. A set of layers and synapses is called *network*. The input data $\mathbf{x}$ are passed from the first layers to the last layer, called, respectively, the *input layer* and the *output layer*, through intermediary layers, called *hidden layers*. The term Deep Learning comes from the network's depth, that is, when the number of hidden layers grows.

Neurons belonging to same layers are not communicating to each others, while neurons belonging to different layers are connected and share the information passed through the activation function. If each neuron of the previous layer is connected to all neurons of the next layer, the former is called *fully connected* or *dense* layer. The output layer, also called *classification layers* in case of classification task or *regression layer* in case of continuous estimation, is responsable to estimate the predicted value $\hat{\mathbf{y}}$ of the labels $\mathbf{y}$. Once the last output is computed, the *feed-forward* procedure is completed.

Thereafter, an iterative procedure is computed to minimize the loss function. This procedure is called *back propagation* and is responsible to minimize the loss function with respect to the weights w_i. The weight's values, indeed, represent how strong is the relation between neurons belonging to different layers and how far the input information has to be transfer through the network. The minimization procedure bases on *gradient descent* algorithms, which iteratively search for weights, that reduce the value of the gradient of the loss until it meets the global minimum or a stopping criteria. In general, a greedy-wise tuning procedure over the hyper-parameters is performed to the aim at achieving the best network configuration. Most important hyper-parameters are: the number of layers, the kernel's number and size, the pooling's size, and the regularization parameter, such as the learning rate.

According to Fig. 6, most used deep learning algorithms are described in the following.

Multi-layer Perceptron (MLP) is the most widely used Artificial Neural Network (ANN). It is a collection or neurons organized in a layers' structure, connected in an acyclic graphs. Each neuron that belongs to a layer produces an output which becomes the input of the neurons of the next adjacent layer. Most common layer type is the fully connected layer, where each neurons share their output with each adjacent layer's neuron, while same layer's neurons are not connected. MLP is made up of the input layer, one (or more) hidden layer and the output layer [88]. Used in HAR as baseline for deep learning techniques, it has been often compared with machine learning, such as SVM [48,89], RF [48], k-NN [89], DT [89], and deep learning techniques, LSTM [90], CNN [89,90].

Convolutional Neural Networks (ConvNet or CNN) is a class of ANN based on convolution products between *kernels* and small patches of the input data of the layer. The input data are organized in channels if needed (e.g., in tri-axial accelerometer data, each axes is represented by one channel), and normally convolution is performed independently on each channel. The convolutional function is computed by sliding a convolutional kernel of the size of $m \times m$ over the input of the layer. That is, the calculation of the lth convolutional layer is given by

$$x_i^{l,j} = f\left(\sum_{a=1}^{m} w_a^j \cdot x_{i+a-1}^{l-1,j} + b_j\right),$$

where m is the kernel size, and $x_i^{l,j}$ is the jth kernel on the i-th unit of the convolutional layer l. w_a^j is the convolutional kernel matrix and b_j is the bias of the convolutional kernel. This value is mapped through the activation function σ. Thereafter, a pooling layer is responsable to compute the maximum or average value on a patch of the size $r \times r$ of the resulting activation's output. Mathematically, a local output after the max pooling or the average pooling process is given by

$$\text{max pooling} \quad x_i^{l,j} = \max_{a,b=1}^{r}(x_{a,b})$$

$$\text{average pooling} \ x_i^{l,j} = \tfrac{1}{r^2}\textstyle\sum_{a,b=1}^{r}(x_{a,b}).$$

The pooling layer is responsible to extracts important features and to reduces the data size's dimension. This convolutional-activation-pooling layers block can be repeated may time if necessary. The number of repetition time determines the depth of the network.

Generally, between the last block and the output layer one (or more), fully connected layer is added to perform a fusion of the information extracted from all sensor channels [88]. After the feed-forward procedure is ended, the back propagation is performed on the convolutional weights until the convergence to the global minimum or until a stopping criterion is met. Figure 7 depicts a CNN example in HAR, with six channels, corresponding to xyz-acceleration and xyz-angualr velocity data, two convolutional-activation-max pooling layers, one fully connected layer, and a soft-max layer which compute the class probability given input data.

CNN is a robust model under many aspects: in terms of local dependency due to the the signals correlation, in terms of scale invariance for different paces or frequencies, and in terms of sensor position [31,71]. For this reasons, CNN have been largely studied in HAR [91].

Additionally, CNN have been compared to other techniques. CNN outperforms SVM in [78] and baseline Random Forest in [27]. Roano et al. demonstrate that CNN outperforms state-of-the-art techniques, which are all using hand-crafted features [92]. More recently, ensemble classification algorithm with CNN-2 and CNN-7 shows better performance when compared with machine learning random forest, boosting, and traditional CNN-7 [40].

Residual Neural Networks (ResNet) is a particular convolutional neural network composed by blocks and skip connections which permit to increase the number of layers in the network. Success of Deep Neural network has been accredited to the additional layer, but He at al. empirically showed that there exists a maximum threshold for the network's depth without avoiding vanishing\explosion gradient's issues [93].

In Residual Neural Networks, the output x_{t-1} is both passed as an input to the next convolutional-activation-pooling block and directly added to the output of the block $f(x_{t-1})$. The former addiction is called *shortcut connection*. The resulting output is $x_t = f(x_{t-1}) + x_{t-1}$. This procedure is repeated many times and permit to deepen the network without adding neither extra parameters nor computation complexity. Figure 8 shows an example of ResNet. Bianco et al. state that ResNet represents the most performing network in the state of the art [94], while Ferrari et al. demonstrated that ResNet outperforms traditional machine learning techniques [59,95].

Long-Short-Term-Memory Networks is a variant of the Recurrent Neural Network which enables to store information over time in an internal memory, overcoming gradient's vanishing issue. Given a sequence of inputs $\mathbf{x} = \{x_1, x_2, ..., x_n\}$, LSTM's external inputs are its previous cell state c_{t-1}, the previous hidden state h_{t-1}, and the current input vector x_t. LSTM associates each time step with an input gate, forget gate, and output gate, denoted, respectively, as i_t, f_t, and o_t, which all are computed by applying an activation function of the linear combination of weights, input x_i, and hidden state h_{t-1}. An intermediate state $\tilde{c}_i$ is also computed through the $tahnh$ of the linear combination of weights, input x_i, and hidden state h_{t-1}. Finally, the cell and hidden state are updated as

$$c_t = f_t \cdot \tilde{c}_t + i_t \cdot \tilde{c}_t$$

$$h_t = o_t \cdot thanh(c_t).$$

The forget gate decides how much of the previous information is going to be forgotten. The input gate decides how to update the state vector using the information from the current input. Finally, the output gate decides what information to output at the current time step [30]. Figure 9 represents the network schema.

Fig. 7 Convolutional neural network schema

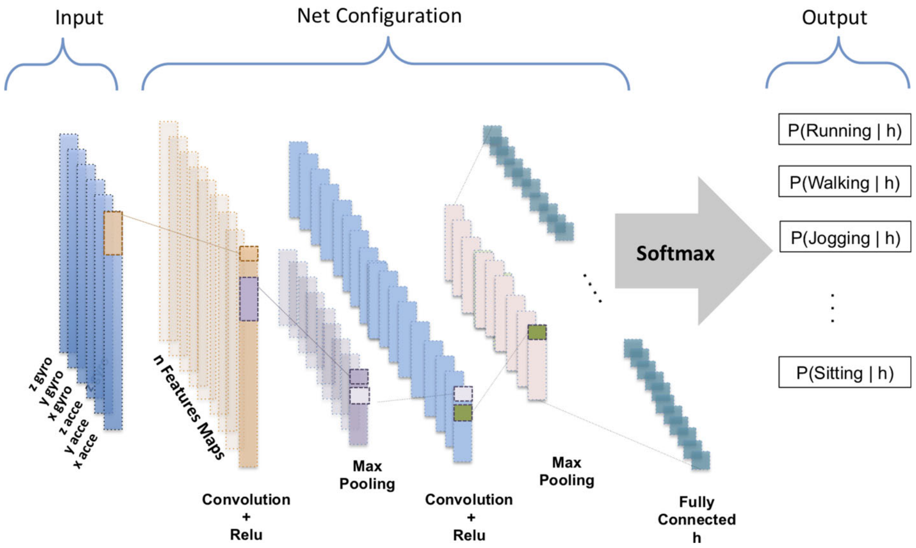

Table 4 Activation functions

Activation function	Step	Sigmoid	Tanh	ReLU
Formula	$\begin{cases} 0 \ if \ x < 0 \\ 1 \ if 0 \leq x \end{cases}$	$\frac{1}{1+e^{x}}$	$tan(x)$	$\max(0, x)$

Table 5 Loss functions for neural networks. M = number of classes, x = input data, y = class, $p_{x,y}$ = probability of being y given x

Loss function	Cross-entropy	Hinge	Euclidean	Absolute value
Formula	$-\sum_{y=1}^{M} y \cdot log(p_{x,y})$	$\max(0, 1 - \hat{y} \cdot y)$	$\sum_{y=1}^{M} (\hat{y} - y)^2$	$\sum_{y=1}^{M} \lvert\hat{y} - y\rvert$

Although LSTM is a very powerful techniques when data temporal dependencies have to be considered during classification, it takes into account only past information. Bidirectional-LSTM (BLSTM) offers the possibility to consider past and future information. Hammerla et al. illustrate how their results based on LSTM and BLSTM, verified on a large benchmark dataset, are the state-of-the-art [96].

7.1.3 Traditional machine learning vs traditional deep learning

Machine learning techniques have been demonstrated to be high performing even with low amount of labeled data and that are low time-consumption methods.

Nevertheless, machine learning techniques remain highly expertise-dependent algorithms. Input data feeding machine learning algorithms are normally features, a processed version of the data. Features permit to reduce data dimensionality and computational time. However, features are hand-crafted and are expert knowledge and tasks dependent.

Furthermore, engineered features cannot represent salient feature of complex activities, and involve time-consuming feature selection techniques to select the best features [97, 98].

Additionally, approaches using hand-crafted features make it very difficult to compare between different algorithms due to different experimental grounds and encounter difficulty in discriminating very similar activities [40].

In recent years, deep learning techniques are increasingly becoming more and more attractive in human activity recognition. First applied to 3D and 2D context in particular in vision computing domain [99,100], deep learning methods have been shown to be valid methods also adapted to the 1D case, that is, for time series classification [101], such as HAR.

Deep learning techniques have shown many advantages over the machine learning, among them the capability to automatically extract features. In particular, depending on the depth of the algorithm, it is possible to achieve a very high abstraction level for the features, despite machine learning techniques [71]. In these terms, deep learning techniques are considered valid algorithms to overcome machine learning dependency on the feature extraction procedure and show crucial advantages in algorithm performance.

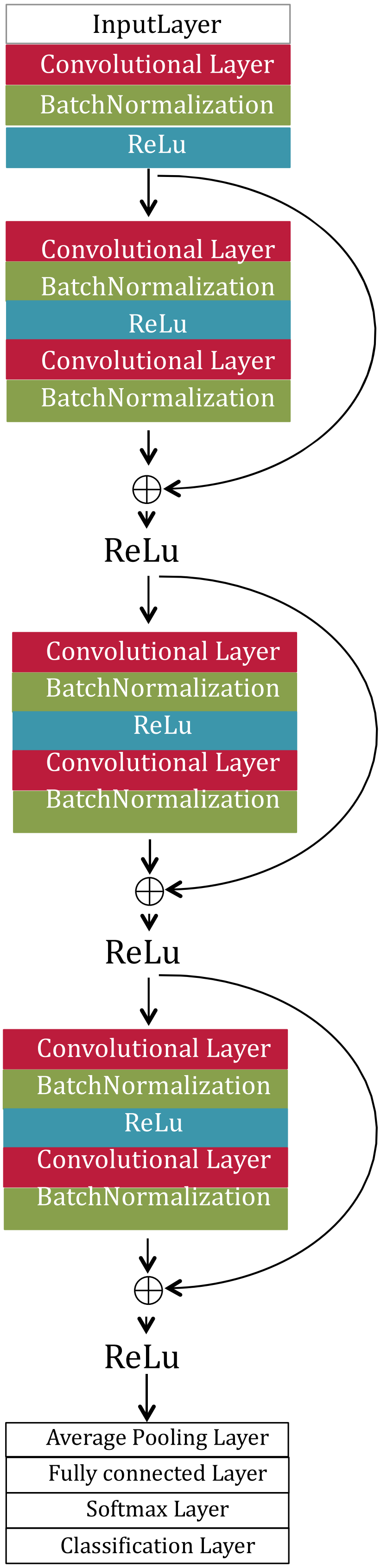

Fig. 8 ResNet full schema

However, deep learning techniques, unlike traditional machine learning approaches, require a large number of samples and an expensive hardware to estimate the model [95]. Large-scale inertial datasets with millions of signals recorded by hundreds of human subjects are still not available, and instead, several smaller datasets made of thousands of signals and dozens of human subjects are publicly available [102]. It is therefore not obvious in this domain, which method between deep and machine learning is the most appropriate, especially in those case where the hardware is low cost.

Scarcity of data results in an important limit of machine learning and deep learning approaches in HAR: the difficulties in being able to generalize the models against the variety of movements performed by different subjects [103]. This variety occurs in relation to heterogeneity in the hardware on which the inertial data are collected, different inherent capabilities or attributes relating to the users themselves, and differences in the environment in which the data are collected.

One of the most relevant difficulties to face with new situations is due to the *population diversity* problem [6], that is, the natural differences between users' activity patterns, which implies that different executions of the same activity are different. A solution is to leverage labeled user-specific data for a personalized approach to activity recognition [104].

7.2 Personalized learning methods

Traditional systems are limited in their ability to generalize to new users and/or new environments, and require considerable effort and customization to achieve good performance in a real context [105,106].

As previously mentioned, one of the most relevant challenges to face with new situations is due to the *population diversity* problem: as users of mobile sensing applications increase in size, the differences between people cause the accuracy of classification to degrade quickly [6].

Ideally, algorithms should be trained on a representative number of subjects and on as many cases as possible. The number of subjects present in the dataset does not just impact the quality and robustness of the induced model, but also the ability to evaluate the consistency of results across subjects [107].

Furthermore, although new technology potentially enables to store large amount of data from varied devices, the actual availability of data are scarce and public datasets are normally very limited (see Sect. 8). In particular, it is very difficult to source labeled data necessary to train supervised machine learning algorithms. To face this problem, activity classification models should be able to generalize as much as possible with respect to the final user.

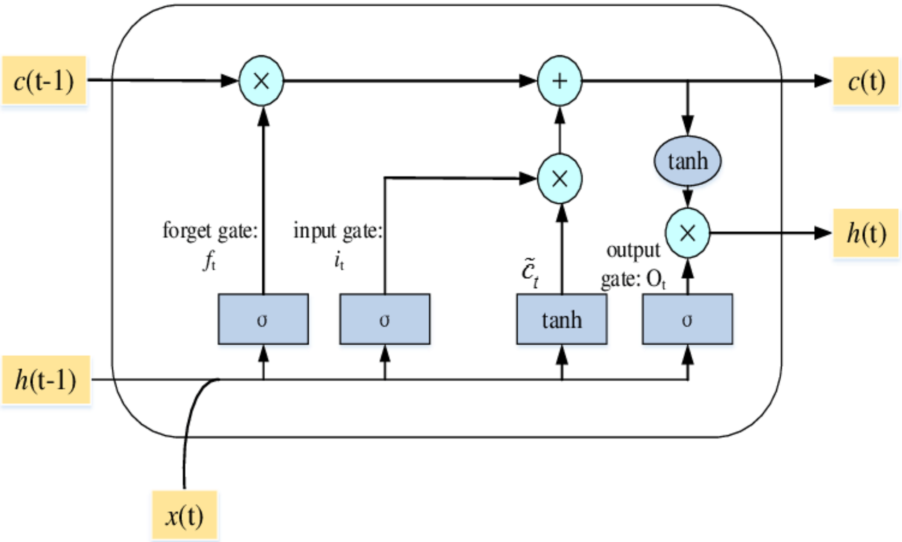

Fig. 9 LSTM networks schema. Source: "Nonlinear Dynamic Soft Sensor Modeling With Supervised Long Short-Term Memory Network", by X. Yuan, L. Li, and Y. Wang, 2020, *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3168–3176, copyright IEEE

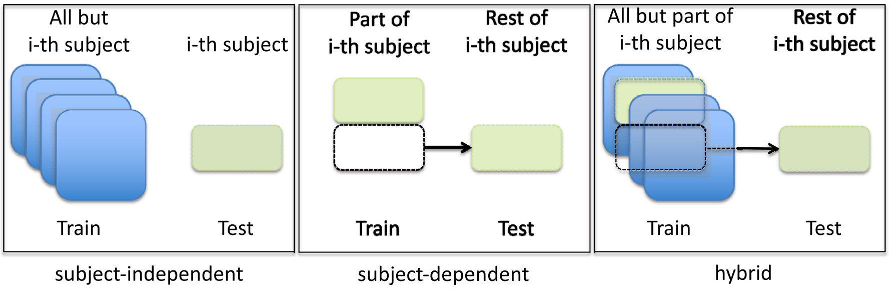

Fig. 10 A graphical representation of the three main classification models

Following sections discuss state-of-the-art results related to *population diversity* issue based on the personalization of machine learning and deep learning algorithms.

7.2.1 Personalized machine learning

To achieve generalizable activity recognition models based on machine learning algorithms, three approaches are mainly adopted in literature:

- **Data-based approaches** encompass three data split configurations: *subject-independent*, *subject-dependent*, and *hybrid*. The *subject-independent* (also called *impersonal*) model does not use the end user data for the development of the activity recognition model. It is based on the definition of a single activity recognition model that must be flexible enough to be able to generalize the diversity between users and it should be able to have good performance once a new user is to be classified.
  The *subject-dependent* (also called *personal*) model only uses the end user data for the development of the activity recognition model. The specific model, being built with the data of the final user, is able to capture her/his peculiarities, and thus, it should well generalize in the real context. The flaw is that it must be implemented for each end user [108].
  The *hybrid* model uses the end user data and the data of the other users for the development of the activity recognition model. In other words, the classification model is trained both on the data of other users and partially on data from the final user. The idea is that the classifier should recognize easier the activity performed by the final user. Figure 10 shows a graphical depiction of the three models to better clarify their differences. Tapia et al. [109] introduced the subject-independent and subject-dependent models, and later Weiss at al. [29] the hybrid model.
  The models were compared by different researchers and also extended to achieve better performance.
  Medrano et al. [110] demonstrated that the subject-dependent approach achieves higher performance and then subject-independent approach for falls detection, called respectively *personal* and *generic fall detector*.
  Shen et al. [111] achieved similar results for activity recognition and come to the conclusion that the subject-dependent (termed *personalized*) model tends to perform better than the subject-independent (termed *generalized*) one, because user training data carry her/his personalized activity information.

  Lara et al. [112] consider subject-independent approach more challenging, because in practice, a real-time activity recognition system should be able to fit any individual and they consider not convenient in many cases to train the activity model for each subject.

  Weiss at al. [29] and Lockhart et al. [61] compared the subject-independent and the subject-dependent (termed *impersonal* and *personal*, respectively) with the hybrid model. They concluded that the models built on the subject-dependent and the hybrid approaches achieve same performance and outperform the performance of the model based on the subject-independent approach.

  Similar conclusions are achieved by Lane et al. [6], who compare subject-dependent and subject-independent (respectively, named *isolated* and *single*) models with another model called *multi-naive*. In this case, subject-dependent approach outperformed the other two approaches as the amount of the available data increases.

  Chen et al. [75] compared the subject-independent, subject-dependent, and hybrid (respectively, called *rest-to-one*, *one-to-one*, and *all-to-one*) models, and once again the subject-dependent model outperforms the subject-independent model, whereas the hybrid model achieves the best performance. The authors also classify subject-independent and hybrid models as *generalized* models, while the subject-dependent model falls into the category of the *personalized* models.

  Same results have been achieved by Vaizman et al. [113], who compared the subject-independent, subject-dependent, and hybrid (respectively, called *universal*, *individual*, and *adapted*) models. Furthermore, they introduced context-based information by exploiting many sensors, such as, location, audio, and phone-state sensors.

- **Similarity-based approach** consider the similarity between users as a crucial factor for obtaining a classification model able to adapt to new situations.

  Sztyler et al. [114,115] proposed a personalized variant of the hybrid model. The classification model is trained using the data of those users that are similar to the final user based on signal patterns similarity. They found that people with same fitness level also have similar acceleration patterns regarding the running activity, whereas gender and physique could characterize the walking activity. The heterogeneity of the data is not eliminated, but it is managed in the classification procedure.

  A similar approach is presented by Lane et al. [6]. The proposed approach consists in exploiting the similarity between users to weight the collected data. The similarities are calculated based on signal pattern data, or on physical data (e.g., age and height), or on lifestyle index. The value of similarity is used as weight. The higher the weight, the more similar two users are and the more that signals from those users is used for classification.

  Garcia-Ceja et al. [116,117] exploited inter-class similarity instead of the similarity between subjects (called inter-user similarity) presented by Lane et al. [6]. The final model is trained using only the instances that are similar to the target user for each class.

- **Classifier-based approaches** obtain generalization from several combinations of activity recognition models.

  Hong at al. [105] proposed a solution where the generalization is obtained by a combination of activity recognition models (trained by a subject-dependent approach). This combination permits to achieve better activity recognition performance for the final user.

  Reiss et al. [118] proposed a model that consists of a set of weighted classifiers (experts). Initially, all the weights have the same values. The classifiers are adapted to a new user by considering a new set of suitable weights that better fit the labeled data of the new user.

Ferrari et al. have recently proposed a similarity-based approach that does not fall into the above classification [70]. The proposed approach is a combination of data-based and similarity-based approaches. Authors trained the algorithms by exploiting the similarity between subject and different data splits. They stated that hybrid models and similarity achieve best performance with respect to the state-of-the-art algorithms.

7.2.2 Personalized deep learning

Personalized deep learning techniques have been explored in the literature and mainly refer to two main approaches

- **Incremental learning** refers to recognition methods that can learn from streaming data and adapt to new moving style of a new unseen person without retraining [119]. Yu et al. [120] exploited the hybrid model and compare it to a new model called *incremental hybrid model*. The latter is trained first with the subject-independent approach, and then, it is incrementally updated based on personal data from a specific user. The difference from the hybrid is that the incremental hybrid model gives more weights to personal data during training.

  Similarly, Siirtola et al. [41] proposed an incremental learning method. The method initially uses a subject-independent model, which is updated with a two-step feature extraction method from the test subject data. Afterwards, the same authors proposed a 4 steps subject-dependent model [39]. The proposed method initially uses a subject-independent model, collects and labels the data from the user based on the subject-independent model, trains a subject-dependent model on the collected and labeled data, and classifies activity based on the subject-dependent model.

Vo et al. [121] exploited a similar approach. The proposed approach first trains a subject-dependent model from data of subject A. The model of subject A is then transferred to subject B. Then, the unlabeled samples of subject B are classified to the model of subject A. These data are finally used to adjust model for the subject B.

Abdallah et al. [122] propose an incremental learning algorithm based on clustering procedure which aims at tuning the general model to recognize a given user's personalized activities.

- **Transfer learning** bases on pre-trained network; it adjusts weights using new user's data. This procedure permits to reduce the time consumption of the training phase. In addition, it is a powerful method for when scarcity of labeled data does not permit to train a network from scratch.

  Rokni et al. [123] propose to personalize their HAR models with transfer learning. In the training phase, a CNN is first trained with data collected from a few participants. In the test phase, only the top layers of the CNN are fine-tuned with a small amount of data for the target users.

A recent personalization approach is proposed by Ferrari et al. that relies on similarity among the subjects. The similarity is used to select the m most similar ones to the target. The algorithm is trained with their data [124].

7.3 Metrics for performance evaluation

In supervised machine learning algorithms, the classification uses three sets of data: the training, the validation, and the test datasets. The training set is designed to estimate the relation between input and output, together with the model parameters. The validation set is designed to affine and tune the model parameters and hyper-parameters. With hyper-parameters, it is meant the parameters that are not necessarily directly involved in the model, but define the structure of the algorithm, such as, for instance, the number of the channels in a deep network. Finally, the test set is used to evaluate the classification performance of the resulted classification model.

Training, validation, and test sets are generally defined as a partition of the original dataset and mostly representing, for instance, the 70%, 20%, and 10% of the number of the samples.

It is a common practice to perform the *k-fold cross-validation* procedure [48,81,125]. The k-fold-cross-validation is a procedure that helps to achieve more robust results and helps to avoid that the algorithm specializes on a specific partitions of the original dataset. In particular, it consists in split the training and test set in k-folds. The entire classification procedure is performed on each split, k times. Thus, k models are estimated, and their performances are evaluated and averaged.

Table 6 Confusion matrix

Groundtruth	**Estimated**	
	1	0
1	True positives (TP)	False negatives (FP)
0	False positives (FN)	True negatives (TN)

The classification performance is calculated through heuristic metrics based on the correctly classified samples. In particular, these metrics are all derived from the *confusion matrix*.

In supervised machine learning, the *confusion matrix* compares the groundtruth (the observed labels) with the estimated labels. The binary case is shown in Table 6.

True positives are observed 1-class samples which are classified as 1. True negatives represent the number of observed 0-class samples which are classified as 0. False negatives are 0-class samples which are classified as class 1. Viceversa, False positives represent the number of samples classified as 1-class but which truly belongs to 0-class.

The confusion matrix can be extended to the multi-class classification problem. In this case, on the principal diagonal are displayed the number of correctly classified samples, while out of the principal diagonal miss-classified samples are listed.

The classification performance can be measured by focusing either on the number of correct classified samples or by giving more importance on the miss-classification. The choice of the evaluation metric accentuates either one or the other aspect of the classification.

In the context of HAR, the *accuracy* is the most used metric for the evaluation of the classification performance [6,33,40,126]. According to the confusion matrix showed in Table 6, accuracy (Acc) is defined as follows:

$$Acc = \frac{TP + TN}{TN + FP + FN + TP}.$$

It calculates the percentage of correctly classified samples over the total number of the samples. The accuracy highlights the correct classification performance and gives more emphasis to the classification of the true positives and of the true negatives.

In some cases, it is required that the evaluation of the classification performance accentuates the mis-classifications, such as either false positives or false negatives cases. For instance, in the case of falls detection, the algorithm should be more penalized if it does not recognize a fall when it occurs (false negative) with respect to the cases where it does recognize a normal behavior as fall (false positive).

An appropriate metric for this case is the F_β-*score*. It is defined as function of *recall* and *precision*.

Recall is also called sensitivity or true positive rate and is calculated as the number of correct positive predictions divided by the total number of positives; the best value corresponds to 1, the worst to 0.

Precision is also called positive predictive value and is calculated as the number of correct positive predictions divided by the total number of positive prediction; the best precision is 1, whereas the worst is 0.

Formula are given by

$$precision = \frac{TP}{TP+FP}$$
$$recall = \frac{TP}{TP+FN}$$
$$F_\beta = \frac{(1+\beta^2)\cdot precision\cdot recall}{(\beta^2\cdot precision)+recall}$$

If $\beta = 1$, F_1-score is the harmonic mean of the precision and the recall.

The *specifictiy*, also called true negative rate, is calculated as the number of correct negative predictions divided by the total number of negatives. Best value corresponds to 1, while the worst is 0. Together with the sensitivity, the specificity helps to determine the best parameter value when a tuning procedure is computed. A common practice is to calculate the area under the curve (AUC) created by plotting values of the sensitivity and 1-specificity. This curve is called Receiver-Operating Characteristic curve (ROC). The value of the parameter which maximizes the classification performance corresponds the point on the ROC curve where AUC is maximal.

8 Datasets

In recent years, the spread of wearable devices has lead to a huge availability of physical activity data. Smartphones and smartwatches have become more and more pervasive and ubiquitous in our everyday life. This high diffusion and portability of wearable devices has enabled researchers to easily produce plenty of labeled raw data for human activity recognition.

Several public datasets are open to the HAR community and are freely accessible on the web, see for instance the UC Irvine Machine Learning Repository [127].

Table 7 shows the main characteristics of the most used datasets in the state-of-the-art.

Most of the datasets used contain signals recorded by smartphones. Some datasets also contain signals from both smartphones and IMUs, and from both smartphones and smartwatches (datasets D03, D010, D11, and D16).

In Table 7, each dataset has assigned an ID (column *ID*). Columns *Dataset* and *Reference* specify the official name and the bibliographic reference of each dataset respectively.

Column *# Activities* specifies the number of ADLs present in the dataset. Usually, each dataset contains 6–10 ADLs and in some cases, both ADLs and Falls data are considered, as in datasets D08, D09, D11.

Column *# Subjects* reports the number of subjects that performed the activities. Considering a restricted number of subjects in the analysis does not just impact the quality and robustness of the classification, but also the ability to evaluate the consistency of the results across subjects [107]. In other words, the number of the subjects included in the training set of the algorithm is crucial in terms of generalization capabilities of the model to classify a new unseen instance. Nevertheless, the difference between people, also called *population diversity*, could lead to poor classification, as largely discussed in [6]. Unfortunately, most of the datasets are limited in terms of subject numerousness.

To overcome this issues, recently, several HAR research groups implemented strategies for merging datasets [102, 134]. Other techniques, such as transfer learning and personalization, have been investigated for robustness of results [61,123,135].

Column *Devices* reports typologies and number of devices that have been used to collect the data. In particular, datasets D03, D04, D05, D06, D11, D12 collected data from several wearable devices at the same time, which is due to the following reasons. First, the device position influences the performance of the classification. Several works investigated which position leads to the best classification [35,136]. Furthermore, it is also challenging to investigate devices fusion, which has a not negligible positive effects on the classification performances and reflects realistic situation where users employ multiple smart devices at once [30,56,63,114].

Position-aware and position-unaware scenarios have been presented in [35]. In position-aware scenarios, the recognition accuracy on different positions is evaluated individually, while in position-unaware scenarios, the classification performance of the combination of devices positions is measured. It is shown that the latter approach highly improves the classification performances for some activities, such as walking, walking upstairs, and walking downstairs. Almaslukh et al. exploited deep learning technique for classification and demonstrated its capability to produce an effective position-independent HAR.

Column *Sensors* lists the sensors exploited in data collection. Tri-axial acceleration sensor (A) is the most exploited inertial sensor among the literature [7]. Datasets D9, D14, and D15 even collected just acceleration data. Acceleration is very popular, because it both directly captures the subjects' physiology motion status and it consumes low energy [137].

Table 7 Public HAR dataset collection inertial signals recorded from smartphone

ID	Dataset	# Activity	# Subject	# Devices	Sensors	Sampling rate (Hz)	Metadata	Reference
D01	UCI HAR	6 ADL	30	SP(1)	A,G	50	No	[49]
D02	Smartphone-based recognition of human Activities and postural transitions data set	6 ADL	30	SP(1)	A,G	50	No	[49]
D03	HHAR	6 ADL	9	SP(8),SW(4)	A,G	H	No	[128]
D04	Physical activity recognition dataset using Smartphone sensors	6 ADL	4	SP(4)	A,G,M	50	No	[20]
D05	Sensors activity dataset	7 ADL	10	SP(5)	AG,M,LA	50	No	[35]
D06	Complex human activities dataset	13 ADL	10	SP(2)	A,G,LA	50	No	[81]
D07	Motions sense	6 ADL	24	SP(1)	A,G,AT	50	Gender, Age Height,Weight	[129]
D08	MobiAct	11 ADL, 4 F	67	SP(1)	A,G,OR	87	Gender, Age Height,Weight	[130]
D09	UniMiB-SHAR	9 ADL, 8 F	30	SP(1)	A	50	Gender, Age Height,Weight	[43]
D10	UMAFall	12 ADL, 3 F	19	SP(1),IMUs(4)	A,G,M	200,20	Gender, Age Height,Weight	[131]
D11	Real world	8 ADL	15	SP(6),SW(1)	A,G,GPS,L,M,S	50	Gender, Age Height,Weight	[37]
D12	WISDM	6 ADL	29	SP (1)	A	20	No	[62]
D013	Smartphone dataset for HAR in Ambient assisted living (AAL) data Set	6 ADL	30	SP(1)	A,G	50	No	[49]
D14	Daily activity dataset	5 ADL	8	SP (1)	A	40	No	[132]
D15	HASC2010	6 ADL	96	SP(1)	A	[10-100]	Gender,Height Weight,Shoes Floor,Place	[133]
D016	Extrasensory dataset	7 ADL + 109 Specific activities	60	SP(1), SW	A,G,M,CO,LO,S,SM,ST	40,25	No	[113]

ADL = Activity of daily living, F = Falls; A = Accelerometer, LA = Linear Acceleration Sensor G = Gyroscope, M = Magnetometer, AT = attitude, OR = orientation, L = light, S = sound, SM = sound magnitude GPS = Global Positioning System,CO= compass, LO = location, ST = phone state, H = highest frequency as possible, SP = smartphone, SW = smartwatch, and IMU = inertial measurement unit

Acceleration has been combined with other sensors, such as gyroscope, magnetometer, GPS, and biosensors, with the aim of improving activity classification performance.

In general, data captured from several sensors carry additional informations about the activity and about the device settings. For instance, information derived from the gyroscope is used to maintain reference direction in the motion system and permits to determine the orientation of the smartphone [32,51].

Performances comparisons between gyroscope, acceleration, and their combination for human activity recognition have been explored in many studies. For example, Ferrari et al. showed that accelerometer is more performing than the gyroscope and their combination leads to an overall improvement of about 10% [36]. Shoaib et al. stated that in situations where accelerometer and gyroscope individually perform with low accuracies, their combination improved the overall performances, while when one of the sensors performs with higher accuracy, the performances are not impacted by the combination of the sensors [35].

Column *Sampling Rate* shows the frequency at which the data are acquired. The sampling rate has to be high enough to capture most significant behavior of data. In HAR, the most commonly used sampling rate is 50 Hz when recording inertial data (see Table 7).

Column *Metadata* lists characteristics regarding the subjects that performed the activities. In D07–11, D15 physical characteristics are annotated. In D15, environmental characteristics have been also stored, such as the kind of shoes worn, floor characteristics, and places where activities have been preformed. As discussed in Sect. 7, metadata are precious additional information, which help to overcome the population diversity problem.

9 Lessons learned and future directions

In this study, we covered the main steps of the Activity Recognition Process (ARP). For each phase of the ARP pipeline, we highlighted what are the key aspects that are being considered and that are more challenging in the field of HAR.

Specifically, when considering the Data Acquisition phase, we noted that the number and kind of available devices is constantly increasing and new devices and sensors are introduced every day. To take advantage of this aspect, new sensors should be experimented in HAR applications to determine whether or not they can be employed to recognize actions. Moreover, new combinations (data fusion) are possible, which may again increase the ability of the data to represent the performed activities.

This increase in sensor numbers and types, while ensuring the availability of more data sources, may pose a challenge in terms of heterogeneity as not all the devices and sensors share the exact same specifications. As an example, some accelerometers may output signals including the low frequencies of the gravity acceleration, while other may exclude it internally. For this reason, the preprocessing phase is of paramount importance to reduce signal differences due to heterogeneous sources and improve the consistency between the in vitro training (usually performed with specific devices and sensors) and real-world use, where the devices and sensors may be similar, but not equal, to the ones used when the models have been trained.

Moreover, we covered the fact that the way the signal is segmented and fed to the classification model may have a significant impact on the results. In the literature, sliding windows with a 50% overlap is the most common choice.

Another aspect we highlighted during this study is the importance of the features used to train the model, as they have a significant impact on the performances of the classifiers. Specifically, hand-crafted features may better model some already known traits of the signals, but automatically extracted features are free of bias and may uncover unforeseen patterns and characteristics.

New and improved features that are able to better represent the peculiar traits of the human activities are needed: ideally, they would combine the domain knowledge of the experts given in the hand-crafted features and the lack of bias provided by the automatically generated features.

Finally, regarding the Classification phase, we highlighted how Model-Driven approaches are being replaced by Data-Driven approaches as they are usually better performing.

Among the Data-Driven approaches, we find that both traditional ML approaches and more modern DL techniques can be applied to HAR problems. Specifically, we learned that while DL methods outperform traditional ML most of the time and are able to automatically extract the features, they require significant amounts of computational power and data more than traditional ML techniques, which makes the latter still a good fit for many use cases.

Regardless of the classification method, we discussed how Population Diversity may impact the performances of HAR applications. To alleviate this problem, we mentioned some recents trends regarding the personalization of the models. Personalizing a classification model means identifying only a portion of the population that is similar to the current subject under some perspective and then only use this subset to train the classifier. The resulting model should be better fitting to the end user. This, however, may exacerbate the issue of data scarcity, since only small portions of the full datasets may be used to train the model for that specific user.

To solve this issue, more large-scale data collection campaigns are needed, as well as further studies in the field of dataset combination and preprocessing pipelines to effectively combine and reduce differences among data acquired from different sources.

10 Conclusions

This paper surveyed the state-of-the-art and new trends in human activity recognition using smartphones. In particular, we went through the activity recognition process: the data acquisition, preprocessing, data segmentation, feature extraction, and classification steps.

Each step has been analyzed by detailing the objectives and discussing the techniques mainly adopted for its realization.

We conclude the review by providing some considerations on the state of maturity of the techniques currently employed in each step and by providing some ideas for future research in the field.

We do not claim to have included everything that has been published on human activity recognition, but we believe that our paper can be a good guide for all those researchers and practitioners that approach this topic for the first time.

Funding Open access funding provided by Università degli Studi di Milano - Bicocca within the CRUI-CARE Agreement.

References

1. Foerster F, Smeja M, Fahrenberg J (1999) Detection of posture and motion by accelerometry: a validation study in ambulatory monitoring. Comput Hum Behav 15(5):571
2. Sun S, Folarin AA, Ranjan Y, Rashid Z, Conde P, Stewart C, Cummins N, Matcham F, Dalla Costa G, Simblett S et al (2020) Using smartphones and wearable devices to monitor behavioral changes during COVID-19. J Med Intern Res 22(9):
3. Mukherjee D, Mondal R, Singh PK, Sarkar R, Bhattacharjee D (2020) EnsemConvNet: a deep learning approach for human activity recognition using smartphone sensors for healthcare applications. Multimed Tools Appl 79(41):31663
4. Iyengar K, Upadhyaya GK, Vaishya R, Jain V (2020) COVID-19 and applications of smartphone technology in the current pandemic. Diabetes Metab Syndrome: Clin Res Rev 14(5):733
5. Shoaib M, Bosch S, Incel O, Scholten H, Havinga P (2015) A survey of online activity recognition using mobile phones. Sensors 15(1):2059
6. Lane ND, Xu Y, Lu H, Hu S, Choudhury T, Campbell AT, Zhao F (2011) Enabling large-scale human activity inference on smartphones using community similarity networks (csn). In: Proceedings of the International Conference on Ubiquitous Computing (UbiComp)
7. Lara OD, Labrador MA et al (2013) A survey on human activity recognition using wearable sensors. IEEE Commun Surv Tutor 15(3):1192
8. Peng X, Wang L, Wang X, Qiao Y (2016) Bag of visual words and fusion methods for action recognition: comprehensive study and good practice. Comput Vis Image Underst 150:109
9. Shou Z, Chan J, Zareian A, Miyazawa K, Chang SF, (2017) Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos, in *Proceedings of the IEEE conference on computer vision and pattern recognition* , pp. 5734–5743
10. Zhang S, Wei Z, Nie J, Huang L, Wang S, Li Z (2017) A review on human activity recognition using vision-based method. J Healthcare Eng 2017
11. Gonzàlez J, Moeslund TB, Wang L et al (2012) Semantic understanding of human behaviors in image sequences: from video-surveillance to video-hermeneutics. Comput Vis Image Underst 116(3):305
12. Buettner M, Prasad R, Philipose M, Wetherall D (2009) Recognizing daily activities with RFID-based sensors. In: Proceedings of the 11th international conference on Ubiquitous computing, pp. 51–60
13. Ofli F, Chaudhry R, Kurillo G, Vidal R, Bajcsy R (2013) Berkeley mhad: a comprehensive multimodal human action database. In: 2013 IEEE Workshop on Applications of Computer Vision (WACV) (IEEE, 2013), pp 53–60
14. Wang W, Liu AX, Shahzad M, Ling K, Lu S (2017) Device-free human activity recognition using commercial WiFi devices. IEEE J Sel Areas Commun 35(5):1118
15. Qi J, Yang P, Waraich A, Deng Z, Zhao Y, Yang Y (2018) Examining sensor-based physical activity recognition and monitoring for healthcare using Internet of Things: a systematic review. J Biomed Inf 87:138
16. Sreenilayam SP, Ahad IU, Nicolosi V, Garzon VA, Brabazon D (2020) Advanced materials of printed wearables for physiological parameter monitoring. Mater Today 32:147
17. Godfrey A, Hetherington V, Shum H, Bonato P, Lovell N, Stuart S (2018) From A to Z: wearable technology explained. Maturitas 113:40
18. Chotpitayasunondh V, Douglas KM (2016) How "phubbing" becomes the norm: the antecedents and consequences of snubbing via smartphone. Comput Hum Behav 63:9
19. Perlow LA (2012) Sleeping with your smartphone: how to break the 24/7 habit and change the way you work. Harvard Business Press, Harvard
20. Shoaib M, Scholten H, Havinga PJ (2013) Towards physical activity recognition using smartphone sensors. In: 2013 IEEE 10th international conference on ubiquitous intelligence and computing and 2013 IEEE 10th international conference on autonomic and trusted computing (IEEE, 2013), pp 80–87
21. Muralidharan K, Khan AJ, Misra A, Balan RK, Agarwal S (2014) Barometric phone sensors: more hype than hope!. In: Proceedings of the 15th workshop on mobile computing systems and applications, pp 1–6
22. Cheng J, Sundholm M, Zhou B, Hirsch M, Lukowicz P (2016) Smart-surface: large scale textile pressure sensors arrays for activity recognition. Pervas Mob Comput 30:97
23. Foubert N, McKee AM, Goubran RA, Knoefel F (2012) Lying and sitting posture recognition and transition detection using a pressure sensor array. In: 2012 IEEE international symposium on medical measurements and applications proceedings (IEEE, 2012), pp 1–6
24. Rehman M, Ziaur Waris A, Gilani SO, Jochumsen M, Niazi IK, Jamil M, Farina D, Kamavuako EN, (2018) Multiday EMG-based

classification of hand motions with deep learning techniques. Sensors 18(8):2497
25. Liu J, Chen J, Jiang H, Jia W, Lin Q, Wang Z (2018) Activity recognition in wearable ECG monitoring aided by accelerometer data. In: 2018 IEEE international symposium on circuits and systems (ISCAS) (IEEE, 2018), pp 1–4
26. Bao L, Intille SS (2004) Activity recognition from user-annotated acceleration data. In: International conference on pervasive computing, Springer, New York, pp 1–17
27. Lee SM, Yoon SM, Cho H (2017) Human activity recognition from accelerometer data using convolutional neural network. In: 2017 IEEE International conference on big data and smart computing (BigComp) (IEEE, 2017), pp 131–134
28. Shakya SR, Zhang C, Zhou Z (2018) Comparative study of machine learning and deep learning architecture for human activity recognition using accelerometer data. Int J Mach Learn Comput 8:577
29. Weiss GM, Lockhart JW (2012) The impact of personalization on smartphone-based activity recognition. In: Proceedings of the AAAI workshop on activity context representation: techniques and languages
30. Milenkoski M, Trivodaliev K, Kalajdziski S, Jovanov M, Stojkoska BR (2018) Real time human activity recognition on smartphones using LSTM networks. In: 2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO) (IEEE, 2018), pp 1126–1131
31. Almaslukh B, Artoli AM, Al-Muhtadi J (2018) A robust deep learning approach for position-independent smartphone-based human activity recognition. Sensors 18(11):3726
32. Alruban A, Alobaidi H, Clarke N, Li F (2019) Physical activity recognition by utilising smartphone sensor signals. In: 8th International conference on pattern recognition applications and methods, SciTePress, pp 342–351
33. Hernández F, Suárez LF, Villamizar J, Altuve M (2019) Human activity recognition on smartphones using a bidirectional LSTM network. In: 2019 XXII symposium on image, signal processing and artificial vision (STSIVA) (IEEE, 2019), pp 1–5
34. Hassan MM, Uddin MZ, Mohamed A, Almogren A (2018) A robust human activity recognition system using smartphone sensors and deep learning. Fut Gen Comput Syst 81:307
35. Shoaib M, Bosch S, Incel OD, Scholten H, Havinga PJ (2014) Fusion of smartphone motion sensors for physical activity recognition. Sensors 14(6):10146
36. Ferrari A, Micucci D, Mobilio M, Napoletano P (2019) Human activities recognition using accelerometer and gyroscope. In: European conference on ambient intelligence, Springer, New York, pp 357–362
37. Sztyler T, Stuckenschmidt H (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) (IEEE, 2016), pp 1–9
38. Bharti P, De D, Chellappan S, Das SK (2018) HuMAn: complex activity recognition with multi-modal multi-positional body sensing. IEEE Trans Mob Comput 18(4):857
39. Siirtola P, Koskimäki H, Röning J (2019) From user-independent to personal human activity recognition models exploiting the sensors of a smartphone. arXiv:1905.12285
40. Zhu R, Xiao Z, Li Y, Yang M, Tan Y, Zhou L, Lin S, Wen H (2019) Efficient human activity recognition solving the confusing activities via deep ensemble learning. IEEE Access 7:75490
41. Siirtola P, Koskimäki H, Röning J (2019) Personalizing human activity recognition models using incremental learning. arXiv:1905.12628
42. Li F, Shirahama K, Nisar MA, Köping L, Grzegorzek M (2018) Comparison of feature learning methods for human activity recognition using wearable sensors. Sensors 18(2):679
43. Micucci D, Mobilio M, Napoletano P (2017) Unimib shar: a dataset for human activity recognition using acceleration data from smartphones. Appl Sci 7(10):1101
44. Khan AM, Lee YK, Lee SY, Kim TS (2010) Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis. In: 2010 5th international conference on future information technology (IEEE, 2010), pp 1–6
45. Ravi N, Dandekar N, Mysore P, Littman ML (2005) Activity recognition from accelerometer data. In: Proceedings of the conference on innovative applications of artificial intelligence (IAAI)
46. Lester J, Choudhury T, Borriello G (2006) A practical approach to recognizing physical activities. In: International conference on pervasive computing, Springer, New York, pp 1–16
47. Gyllensten IC, Bonomi AG (2011) Identifying types of physical activity with a single accelerometer: evaluating laboratory-trained algorithms in daily life. IEEE Trans Biomed Eng 58(9):2656
48. Bayat A, Pomplun M, Tran DA (2014) A study on human activity recognition using accelerometer data from smartphones. Proc Comput Sci 34:450
49. Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL (2013) A public domain dataset for human activity recognition using smartphones. In: Proceedings of the European symposium on artificial neural networks, computational intelligence and machine learning (ESANN13)
50. Bo X, Huebner A, Poellabauer C, O'Brien MK, Mummidisetty CK, Jayaraman A (2007) Evaluation of sensing and processing parameters for human action recognition. In: 2017 IEEE International Conference on Healthcare Informatics (ICHI) (IEEE, 2017), pp 541–546
51. Su X, Tong H, Ji P (2014) Activity recognition with smartphone sensors. Tsinghua Sci Technol 19(3):235
52. Antonsson EK, Mann RW (1985) The frequency content of gait. J Biomech 18(1):39
53. Quigley B, Donnelly M, Moore G, Galway L (2018) A comparative analysis of windowing approaches in dense sensing environments. In: Multidisciplinary Digital Publishing Institute Proceedings, vol 2, p 1245
54. Banos O, Galvez JM, Damas M, Pomares H, Rojas I (2014) Window size impact in human activity recognition. Sensors 14(4):6474
55. Chen K, Zhang D, Yao L, Guo B, Yu Z, Liu Y (2020) Deep learning for sensor-based human activity recognition: overview, challenges and opportunities. arXiv:2001.07416
56. Janidarmian M, Roshan Fekr A, Radecka K, Zilic Z (2017) A comprehensive analysis on wearable acceleration sensors in human activity recognition. Sensors 17(3):529
57. Capela NA, Lemaire ED, Baddour N (2015) Improving classification of sit, stand, and lie in a smartphone human activity recognition system. In: 2015 IEEE international symposium on medical measurements and applications (MeMeA) proceedings, IEEE, pp 473–478
58. Langley P (1996) Elements of machine learning. Morgan Kaufmann, New York
59. Ferrari A, Micucci D, Marco M, Napoletano P (2019) Handcrafted features vs residual networks for human activities recognition using accelerometer. In: Proceedings of the IEEE international symposium on consumer technologies (ISCT)
60. Liu H, Motoda H (1998) Feature extraction, construction and selection: A data mining perspective, vol 453, Springer, New York
61. Lockhart JW, Weiss GM (2014) The benefits of personalized smartphone-based activity recognition models. In: Proceedings of the 2014 SIAM international conference on data mining (SIAM, 2014), pp 614–622

62. Kwapisz JR, Weiss GM, Moore SA (2011) Activity recognition using cell phone accelerometers. ACM SIGKDD Explor Newsl 12(2):74
63. Altun K, Barshan B, Tunçel O (2010) Comparative study on classifying human activities with miniature inertial and magnetic sensors. Pattern Recogn 43(10):3605
64. Sani S, Massie S, Wiratunga N, Cooper K (2017) Learning deep and shallow features for human activity recognition. In: International conference on knowledge science, engineering and management, Springer, New York, pp 469–482
65. Plötz T, Hammerla NY, Olivier PL (2011) Feature learning for activity recognition in ubiquitous computing. In: Twenty-second international joint conference on artificial intelligence
66. Lago P, Inoue S (2019) Comparing Feature Learning Methods for Human Activity Recognition: Performance study in new user scenario. In: 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR) (IEEE, 2019), pp 118–123
67. Wang J, Liu P, She MF, Nahavandi S, Kouzani A (2013) Bag-of-words representation for biomedical time series classification. Biomed Signal Process Control 8(6):634
68. Shirahama K, Grzegorzek M (2017) On the generality of codebook approach for sensor-based human activity recognition. Electronics 6(2):44
69. Abdi H, Williams LJ (2010) Principal component analysis. Wiley Interdiscip Rev Comput Stat 2(4):433
70. Ferrari A, Micucci D, Mobilio M, Napoletano P (2020) On the personalization of classification models for human activity recognition. IEEE Access 8:32066
71. Wang J, Chen Y, Hao S, Peng X, Hu L (2019) Deep learning for sensor-based activity recognition: a survey. Pattern Recogn Lett 119:3
72. Zhang W, Yang G, Lin G, Ji C, Gupta MM (2018) On definition of deep learning. In: 2018 World Automation Congress (WAC), IEEE, pp 1–5
73. Lin Y, Zhang W (2004) Towards a novel interface design framework: function-behavior-state paradigm. Int J Hum Comput Stud 61(3):259
74. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273
75. Chen Y, Shen C (2017) Performance analysis of smartphone-sensor behavior for human activity recognition. IEEE Access 5:3095
76. Amezzane I, Fakhri Y, El Aroussi M, Bakhouya M (2018) Towards an efficient implementation of human activity recognition for mobile devices. EAI Endorsed Trans Context-Aware Syst Appl 4(13)
77. Vaughn A, Biocco P, Liu Y, Anwar M (2018) Activity detection and analysis using smartphone sensors. In: 2018 IEEE International Conference on Information Reuse and Integration (IRI), IEEE, pp 102–107
78. Xu W, Pang Y, Yang Y, Liu Y (2018) Human activity recognition based on convolutional neural network. In: 2018 24th International conference on pattern recognition (ICPR), IEEE, pp 165–170
79. Jalal A, Quaid MAK, Hasan AS (2018) Wearable sensor-based human behavior understanding and recognition in daily life for smart environments. In: 2018 International conference on frontiers of information technology (FIT), IEEE, pp 105–110
80. Witten IH, Frank E, Hall MA (2005) Practical machine learning tools and techniques. Morgan Kaufmann, pp 578
81. Shoaib M, Bosch S, Incel OD, Scholten H, Havinga PJ (2016) Complex human activity recognition using smartphone and wrist-worn motion sensors. Sensors 16(4):426
82. Rokach L, Maimon OZ (2008) Data mining with decision trees: theory and applications. Data mining with decision trees: theory and applications, vol. 69, World scientific, Singapore
83. Breiman L (1999) 1 RANDOM FORESTS–RANDOM FEATURES
84. Polu SK (2018) Human activity recognition on smartphones using machine learning algorithms. Int J Innovat Res Sci Technol 5(6):31
85. Bansal A, Shukla A, Rastogi S, Mittal S (2018) Micro activity recognition of mobile phone users using inbuilt sensors. In: 2018 8th international conference on cloud computing, data science & engineering (confluence), IEEE, pp 225–230
86. Antal P (1998) Construction of a classifier with prior domain knowledge formalised as bayesian network. In *IECON'98. Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society (Cat. No. 98CH36200)*, vol 4, IEEE, pp 2527–2531
87. Nguyen H, Tran KP, Zeng X, Koehl L, Tartare G (2019) Wearable sensor data based human activity recognition using machine learning: a new approach. arXiv:1905.03809
88. Yu T, Chen J, Yan N, Liu X (2018) A multi-layer parallel LSTM Network for Human Activity Recognition with Smartphone Sensors. In: 2018 10th International conference on wireless communications and signal processing (WCSP), IEEE, pp 1–6
89. Suto J, Oniga S, Lung C, Orha I (2018) Comparison of offline and real-time human activity recognition results using machine learning techniques. In: Neural computing and applications, pp 1–14
90. Nair N, Thomas C, Jayagopi DB (2018) Human activity recognition using temporal convolutional network. In: Proceedings of the 5th international workshop on sensor-based activity recognition and interaction, pp 1–8
91. Demrozi F, Pravadelli G, Bihorac A, Rashidi P (2020) Human activity recognition using inertial, physiological and environmental sensors: a comprehensive survey. arXiv:2004.08821
92. Ronao CA, Cho SB (2015) Deep convolutional neural networks for human activity recognition with smartphone sensors. In: International conference on neural information processing, Springer, pp 46–53
93. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 770–778
94. Bianco S, Cadene R, Celona L, Napoletano P (2018) Benchmark analysis of representative deep neural network architectures. IEEE Access 6:64270
95. Ferrari A, Micucci D, Mobilio M, Napoletano P (2019) Handcrafted features vs residual networks for human activities recognition using accelerometer. In: 2019 IEEE 23rd international symposium on consumer technologies (ISCT), IEEE, pp 153–156
96. Hammerla NY, Halloran S, Plötz T (2016) Deep, convolutional, and recurrent models for human activity recognition using wearables. arXiv:1604.08880
97. Friday NH, Al-garadi MA, Mujtaba G, Alo UR, Waqas A (2018) Deep learning fusion conceptual frameworks for complex human activity recognition using mobile and wearable sensors. In: 2018 International conference on computing, mathematics and engineering technologies (iCoMET), IEEE, pp 1–7
98. Yang J, Nguyen MN, San PP, Li XL, Krishnaswamy S (2015) Deep convolutional neural networks on multichannel time series for human activity recognition. In: Proceedings of the international joint conference on artificial intelligence (IJCAI 15)
99. Coşar S, Donatiello G, Bogorny V, Garate C, Alvares LO, Brémond F (2016) Toward abnormal trajectory and event detection in video surveillance. IEEE Trans Circ Syst Video Technol 27(3):683

100. Mabrouk AB, Zagrouba E (2018) Abnormal behavior recognition for intelligent video surveillance systems: a review. Expert Syst Appl 91:480
101. LeCun Y, Bengio Y et al (1995) Convolutional networks for images, speech, and time series. Handb Brain Theory Neural Netw 3361(10):1995
102. Siirtola P, Koskimäki H, Röning J (2018) OpenHAR: A Matlab toolbox for easy access to publicly open human activity data sets. In: Proceedings of the ACM international joint conference and international symposium on pervasive and ubiquitous computing and wearable computers (UbiComp18)
103. Bianchi V, Bassoli M, Lombardo G, Fornacciari P, Mordonini M, De Munari I (2019) IoT wearable sensor and deep learning: an integrated approach for personalized human activity recognition in a smart home environment. IEEE Internet of Things J 6(5):8553
104. Burns DM, Whyne CM (2020) Personalized activity recognition with deep triplet embeddings. arXiv:2001.05517
105. Hong JH, Ramos J, Dey AK (2016) Toward personalized activity recognition systems with a semipopulation approach. IEEE Trans Hum-Mach Syst 46(1):101–112
106. Igual R, Medrano C, Plaza I (2015) A comparison of public datasets for acceleration-based fall detection. Med Eng Phys 37(9):870
107. Lockhart JW, Weiss GM (2014) Limitations with activity recognition methodology & data sets. In: Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing: adjunct publication, pp 747–756
108. Berchtold M, Budde M, Schmidtke HR, Beigl M (2010) An extensible modular recognition concept that makes activity recognition practical. In: Annual conference on artificial intelligence (AAAI)
109. Tapia EM, Intille SS, Haskell W, Larson K, Wright J, King A, Friedman R (2007) Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In: Proceeding of the IEEE international symposium on wearable computers (ISWC)
110. Medrano C, Igual R, Plaza I, Castro M (2014) Detecting falls as novelties in acceleration patterns acquired with smartphones. PLoS One 9(4):e94811
111. Shen C, Chen Y, Yang G (2016) On motion-sensor behavior analysis for human-activity recognition via smartphones. In: 2016 Ieee International Conference on Identity, Security and Behavior Analysis (Isba), IEEE, pp 1–6
112. Lara OD, Pérez AJ, Labrador MA, Posada JD (2012) Centinela: a human activity recognition system based on acceleration and vital sign data. Pervasiv Mob Comput 8(5):717
113. Vaizman Y, Ellis K, Lanckriet G (2017) Recognizing detailed human context in the wild from smartphones and smartwatches. IEEE Pervasive Comput 16(4):62
114. Sztyler T, Stuckenschmidt H (2017) Online personalization of cross-subjects based activity recognition models on wearable devices. In: Proceedings of the IEEE international conference on pervasive computing and communications (PerCom)
115. Sztyler T, Stuckenschmidt H, Petrich W (2017) Position-aware activity recognition with wearable devices. Pervasiv Mob Comput 38:281
116. Garcia-Ceja E, Brena R (2015) Building personalized activity recognition models with scarce labeled data based on class similarities. In: International conference on ubiquitous computing and ambient intelligence, Springer, New York, pp 265–276
117. Garcia-Ceja E, Brena R (2016) Activity recognition using community data to complement small amounts of labeled instances. Sensors 16(6):877
118. Reiss A, Stricker D (2013) Personalized mobile physical activity recognition. In: Proceeding of the IEEE international symposium on wearable computers (ISWC)
119. Siirtola P, Röning J (2019) Incremental learning to personalize human activity recognition models: the importance of human AI collaboration. Sensors 19(23):5151
120. Yu T, Zhuang Y, Mengshoel OJ, Yagan O (2016) Hybridizing personal and impersonal machine learning models for activity recognition on mobile devices. In: Proceedings of the EAI international conference on mobile computing, applications and services (MobiCASE)
121. Vo QV, Hoang MT, Choi D (2013) Personalization in mobile activity recognition system using K-medoids clustering algorithm. Int J Distrib Sens Netw 9(7):315841
122. Abdallah ZS, Gaber MM, Srinivasan B, Krishnaswamy S (2015) Adaptive mobile activity recognition system with evolving data streams. Neurocomputing 150:304
123. Rokni SA, Nourollahi M, Ghasemzadeh H (2018) Personalized human activity recognition using convolutional neural networks. In: Thirty-second AAAI conference on artificial intelligence
124. Ferrari A, Micucci D, Mobilio M, Napoletano P (2020) On the personalization of classification models for human activity recognition. arXiv:2009.00268 (2020)
125. Ronao CA, Cho SB (2014) Human activity recognition using smartphone sensors with two-stage continuous hidden Markov models. In: 2014 10th International conference on natural computation (ICNC), IEEE, pp 681–686
126. Su X, Tong H, Ji P (2014) Accelerometer-based activity recognition on smartphone. In: Proceedings of the 23rd ACM international conference on conference on information and knowledge management, pp 2021–2023
127. Bay SD, Kibler D, Pazzani MJ, Smyth P (2000) The UCI KDD archive of large data sets for data mining research and experimentation. ACM SIGKDD Explor Newsl 2(2):81
128. Stisen A, Blunck H, Bhattacharya S, Prentow TS, Kjaergaard MB, Dey A, Sonne T, Jensen MM (2015) Smart devices are different: assessing and mitigating mobile sensing heterogeneities for activity recognition. In: Proceedings of the 13th ACM conference on embedded networked sensor systems, pp 127–140
129. Malekzadeh M, Clegg RG, Cavallaro A, Haddadi H (2018) Protecting sensory data against sensitive inferences. In: Proceedings of the workshop on privacy by design in distributed systems (W-P2DS18)
130. Vavoulas G, Chatzaki C, Malliotakis T, Pediaditis M, Tsiknakis M (2016) The MobiAct dataset: recognition of activities of daily living using smartphones. In: Proceedings of Information and Communication Technologies for Ageing Well and e-Health (ICT4AgeingWell16)
131. Casilari E, Santoyo-Ramón JA, Cano-García JM (2017) UMAFall: a multisensor dataset for the research on automatic fall detection. Procedia Comput Sci 110:32
132. Siirtola P, Röning J (2012) Recognizing human activities user-independently on smartphones based on accelerometer data. IJIMAI 1(5):38
133. Kawaguchi N, Watanabe H, Yang T, Ogawa N, Iwasaki Y, Kaji K, Terada T, Murao K, Hada H, Inoue S et al (2012) Hasc2012corpus: large scale human activity corpus and its application. In: Proceedings of the second international workshop of mobile sensing: from smartphones and wearables to big data, pp 10–14
134. Ferrari A, Mobilio M, Micucci D, Napoletano P (2019) On the homogenization of heterogeneous inertial-based databases for human activity recognition. In: 2019 IEEE world congress on services (SERVICES), IEEE, pp 295–300
135. Ferrari A, Micucci D, Marco M, Napoletano P (2019) On the homogenization of heterogeneous inertial-based databases for human activity recognition. In: Proceedings of IEEE services workshop on big data for public health policy making
136. Krupitzer C, Sztyler T, Edinger J, Breitbach M, Stuckenschmidt H, Becker C (2018) Hips do lie! a position-aware mobile fall detec-

tion system. In: 2018 IEEE international conference on pervasive computing and communications (PerCom), IEEE, pp 1–10

137. Huynh DTG (2008) Human activity recognition with wearable sensors, Human activity recognition with wearable sensors. Ph.D. thesis, Technische Universitat

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