Wireless Sensor Networks for Healthcare

Subjects: Health Care Sciences & Services
Contributor: Rani Baghezza

Wireless sensor networks for healthcare refers to the networks that help to monitor the physical conditions. This entry details the Real-Time Centralized Activity Recognition and Real-Time Distributed Activity Recognition.

Keywords: activity recognition ; machine learning

1. Introduction

Population aging comes with a set of problems that will have to be solved in the next decades. It is projected that nearly 2.1 billion people will be over the age of 60 by 2050 [1]. As disability [2], as well as dementia [3], rates increase with age, this segment of the population requires assistance in their daily lives.By capitalizing on the use of technology for ambient assisted living, it is possible to extend the autonomy and the quality of life of patients suffering from mild degrees of dementia. Using technology may also allow us to reduce the ever increasing costs of healthcare [3] in terms of human and monetary resources. Smart environments and smart homes were not initially thought of as healthcare oriented, and the latter were described as “[…] a residence equipped with computing and information technology which anticipates and responds to the needs of the occupants, working to promote their comfort, convenience, security and entertainment through the management of technology within the home and connections to the world beyond” by Harper [4]. The initial vision was to increase convenience and comfort for any resident, but soon enough, researchers understood the potential of an environment filled with sensors to remotely monitor patients, and made use of these wireless sensor networks to assist them. Virone [5] presented an architecture that collects sensor data inside a smart home and stores it in a database so that patients can be remotely monitored, combining sensors, back-end nodes, and databases to track a patient inside their home.In entry, we explore the evolution of activity recognition from basic offline implementations to fully distributed real-time systems. The motivation for this review comes from the transition we are reaching in our research work in the use of ambient intelligence for assisted living in smart homes [6][7]. With the technological advancement of the Internet of Things, we are now beginning to investigate the distribution and real-time implementation of smart technologies, not only to improve current technologies used in smart homes, but to extend assistance to vulnerable users beyond smart homes, and into smart cities [8]. In order to reach that goal, it is necessary to have a good understanding and a wide overview of the field of activity recognition in smart homes and its evolution, from its infancy using binary sensors and simple learning rules, to more advanced, real-time distributed implementations in Wireless Sensor Networks using real-time distributed machine learning.The switch from offline to real-time distributed activity recognition would allow us to bring timely assistance to semi-autonomous people by quickly detecting anomalies and being able to bring human assistance to patients when it is needed. Distributing the algorithm for activity recognition allows us to take advantage of the ever increasing capabilities of embedded devices and the IoT. With distribution, there is no need to run a costly distant server to process all the data, there is less need for long distance data communication, less privacy issues, and no single point of failure. However, there are still many challenges to overcome for real-time distributed activity recognition, which this survey attempts to cover.

2. Real-Time Centralized Activity Recognition

Real-time centralized activity recognition focuses on collecting data and recognizing activities in real-time, usually by aggregating data from the nodes of a wireless sensor network into a local computer or a distant server, and using machine learning algorithms to perform real-time or periodical classification of the performed activities. Switching from an offline to a real-time context is necessary when monitoring higher risk patients, or whenever the use-case requires an instantaneous feedback from the system. The sensors used are similar to the ones covered in the previous section: Environmental sensors, wearable sensors, smartphones, or any combination of these methods. As far as the architecture of the system goes, different network topologies can be used (mesh, star, partial mesh), as well as several main communication protocols (Wi-Fi, Bluetooth, ZigBee, ANT, LoRaWAN). Real-time systems sometimes provide a visualization tool to give feedback to the user or to the physician in the case of a healthcare application.This section reviews real-time centralized activity recognition systems for healthcare, with an emphasis on the architecture of the systems as well as the issues and challenges of real-time and near real-time machine learning in a streaming context. We compare local and cloud-based approaches, and conclude on the state of the art in the field and the limitations of these systems.

2.1. Architecture and Communication

Efficient real-time activity recognition systems need to rely on a robust architecture. Data collected from each sensor node in the system needs to reach the central node in a timely manner. In this section, we review and compare the main communication protocols used for real-time activity recognition, as well as the most used network topologies. We also discuss data collection and storage issues in a real-time system.

2.1.1. Communication Protocols

Wi-Fi, Bluetooth, ZigBee, and ANT (Adaptive Network Topology) are the main wireless technologies used for communication in Wireless Sensor Networks. We also cover the recently introduced LoRaWAN, which is aimed at long range, low energy communication. Cellular networks (GSM) can also be considered for smartphone based applications, but in the context of a WSN, they cannot be used to communicate with sensor nodes.When it comes to newer technologies, even if 5G has not been used for activity recognition applications inside smart homes to the best of our knowledge, it seems very promising for IoT applications [9], and has been used for the promotion of unobtrusive activities and collision avoidance in the city [10]. Chettry et al. [11] have writen a comprehensive survey on the use of 5G for IoT applications. More recent Wi-Fi protocols, such as IEEE 802.11af have been used for healthcare applications to collect health data, such as body temperature and blood pressure from wearable sensors [12]. Aust et al. [13] have foreseen the upcoming challenge of highly congested classical wireless spectrums (2.4 GHz/5 GHz) due to the rapid advancement of the IoT, and they have reviewed the advantages of using sub 1 GHz Wi-Fi protocols, such as IEEE 802.11ah for industrial, scientific, and medical applications.In terms of transfer speed, Wi-Fi offers the fastest solution. Newest standards such as 802.11ac advertise a maximum theoretical speed of to 1300 Mbps [14], with a maximum theoretical range of about 90 m outdoors, and 45 m indoors. Bluetooth 3.0, however, offers a maximum speed of 24 Mbps and a maximum theoretical range of 100 m. Starting from Bluetooth 2.0, a new standard was introduced as Bluetooth Low Energy (BLE), and was further improved in version 5.0, doubling its data rate [15]. The main goal of BLE is to reduce energy consumption in order to extend battery life of smartphones and wearables. However, the maximum transfer speed of BLE is 2 Mbps only, making it 12 times slower than regular Bluetooth. ZigBee has emerged in 2003 as a new standard particularly adapted to the Internet of Things (IoT) and sensor-based systems, with a focus on low latency and very low energy consumption [16]. LoRaWAN (Longe Range Wide Area Network) is a recently released (2016) network protocol built for LoRa compliant chips. It uses a star-of-stars topology and is advertised for long range, low energy consumption communication. Sanchez-Iborra et al. [17] have tested LoRaWAN's maximal range and have found that packets could travel up to 7 km in an urban scenario, and 19 km in a rural scenario, thanks to the absence of obstacles in the way.There are a lot of parameters to take into account for energy consumption, such as the type of packets used for Bluetooth, data transfer rate, sleep time, and transfer time for Wi-Fi. In each case, we picked the lowest consuming configuration presented in the papers in order to have an even ground for comparison. A comparative table can be found above (Table 1). In most cases, maximum range, transfer speed, number of nodes, and minimum power consumption cannot be achieved at the same time.Table 1. Comparison table for Wi-Fi, Bluetooth, BLE, ANT, and ZigBee standards in terms of speed, range, energy consumption, compatible topologies, and maximum number of nodes in a single network. The highest values in have been highlighted in the speed, range, and max number of nodes category, and the lowest value in the energy consumption category.

Protocol	Speed (Mbps)	Range (m)	Energy cons. (mW)	Topologies	Max Nodes
Wi-Fi	**1300**	**90**	**12.21**	**P2P, Star**	**250**
Bluetooth	**24**	**100**	**4.25**	**P2P, Broadcast**	**7 (active)**
BLE (5.0)	**2**	**240**	**0.07**	**P2P, Broadcast, Mesh**	**7 (active)**
ZigBee	**0.25**	**100**	**0.66**	**P2P, Star, Cluster Tree, Mesh**	**65,536**
ANT	**0.06**	**30**	**0.83**	**P2P, Star, Tree, Mesh**	**65,533**
LoRaWAN	**0.027**	**19,000**	**1.65**	**Star of stars**	**120**

The choice of a protocol depends on the application and the main constraints of the system. In the case of a Wearable Body Sensor Network (WBSN), sensor nodes are powered through small batteries and need to rely on energy-efficient protocols such as BLE. However, if the number of nodes needed is high, such as in a smart home containing dozens of environmental sensors, ANT and ZigBee might be better suited. In applications where energy is not the main concern, but a high throughput is needed, such as in video based real-time activity recognition, Wi-Fi or even wired alternatives will most likely be the best choice. Because of its long range and low speed properties, LoRaWAN is more suited for periodic, low speed exchanges over long distances, but it is not particularly suited for WBSN applications.

2.1.2. Topology

Real-time activity recognition systems collect data continuously from environmental or wearable sensors. Regardless of the type of sensors used, each sensor node has to send data back to the central node, which is usually a computer or a smartphone.Suryadevara et al. [18] have used environmental sensors with ZigBee components organized in a star topology to determine the wellness of inhabitants based on their daily activities. The main advantage of a star topology is that each end node only has to know how to reach the central node, which simplifies communication. However, each sensor node has to be in reach of the central node, which might not always be the case for activity recognition in extended spaces. The central node also represents a single point of failure. Cheng et al. [19] have presented an architecture in which the failing central node is replaced with the next most powerful node available so that the system can keep running.In order to improve the flexibility of their system, Suryadevara et al. [20] have experimented with a mesh topology using XBee modules to forecast the behavior of an elderly resident in a smart home. A partial mesh topology is used, where 3 relay nodes can forward data from the end nodes to the central coordinator. The reliability of the network has been shown to be above 98.1% in the worst case scenario, with 2 hops between the end node and the central coordinator.Baykas et al. [21] have compared the efficiency of star and mesh topology for wireless sensor networks and shown that in wide area networks, mesh topology networks require 22% more relay nodes to support sensor traffic as reliably as the star topology equivalent. The nodes in a mesh topology also need 20% more bandwidth to deliver data in a timely manner.ZigBee and ANT also support cluster tree topologies, allowing the use of hubs that receive data from their end nodes and forward it to the other nodes of the network. This topology can be useful when it is not possible to connect every end node to a single central node directly, or when the use of intermediary relay nodes allows us to handle traffic in a more efficient way. Altun et al. [22] have used a relay node on the chest to forward data from the left wrist end node to the central XBus master node located on the user’s waist.Büsching et al. [23] have presented a disruption tolerant protocol to ensure that nodes in wireless body area networks could switch from online to offline storage should their connection to the central station be interrupted. After the connection is lost, each node will store the collected data on an SD card until the connection is restored. When the node is back online, it has to go through a synchronization procedure in order to send the backlog of data in the correct order to the central station.In the context of activity recognition in a single room or a small house, it seems preferable to rely on a star topology as long as all the end nodes are in range of the central coordinator to ensure optimal reliability with the least amount of nodes. The figure below (Figure 1) shows the main types of topologies used in the literature.

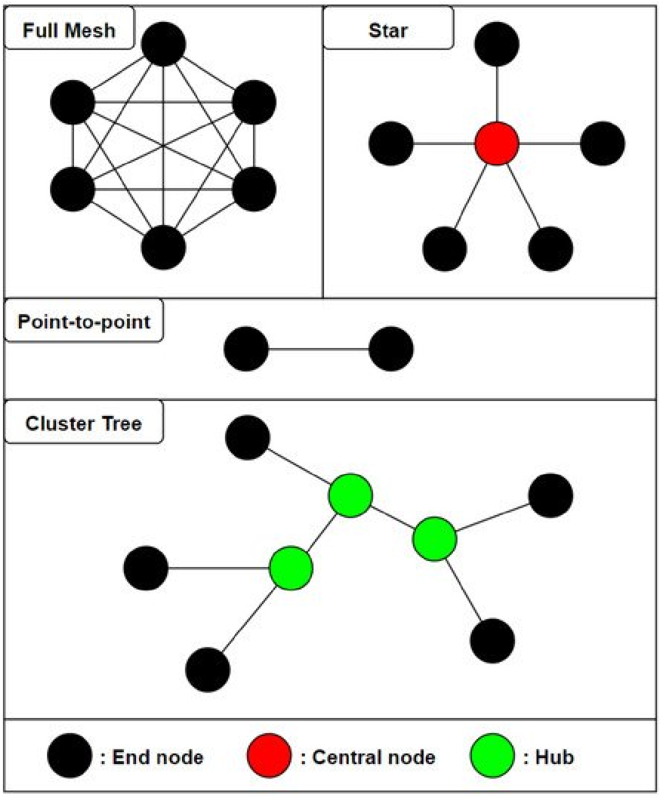

Figure 1. An overview of different network topologies used in wireless sensor networks for real-time activity recognition. In the full mesh (top left), any node can directly communicate with any other node in the network. In the star topology, all of the end nodes are connected to a single central node. In a point-to-point topology (middle), two nodes communicate strictly with one another. In a cluster tree (bottom), end nodes are connected to different hubs which are connected to each other.

2.1.3. Data Collection

In real-time systems, data can either be collected from sensor nodes at a fixed time interval with the associated timestamp, or in the case of binary sensors, the new state of the sensor can be collected whenever it changes. Nguyen et al. have used a fixed time interval of 1 min in their real-time activity recognition system in an office [24]. Every single minute, each sensor sends its state to the base station, that then determines the activity that is being performed by the worker in the office based on a set of binary rules.However, this method can lead to a lot of data being collected and stored, as well as redundant data when the sensors stay in the same state for an extended period of time. Suryadevara et al. [18] have instead opted to use an event-based approach for data collection, where the data collected from the sensors is only stored if the most recent state of the sensor differs from the last stored state. This allows us to store much less data, and to discard any repetitive data. This approach only works for binary, or to a certain extent, digital sensors. In most wearable sensor-based systems, data has to be continuously collected, as sensors such as gyroscopes or accelerometers usually operate in the 10–100 Hz sampling frequency range, and every single data point has to be stored to extract accurate features for activity recognition [25].In order to improve the efficiency of the system, Chapron et al. [26] have opted for a smart data compression based on the range of definition of data collected by the accelerometer. They have noticed that full float precision was not required to store accelerometer, magnetometer, and gyroscope data, and they have successfully compressed data so that each packet could contain the complete information from the 9-axis IMU and fit in a single BLE characteristic.After choosing the right sensors based on the activity recognition task, it is critical to build a suitable architecture and opt for an appropriate communication protocol for real-time application. These choices should allow for a fast, reliable, and energy efficient system. Data also has to be handled and stored in a way that minimizes storage and processing requirements of the central station.

2.2. Local and Cloud Processing

In a centralized activity recognition system, data is collected from sensor nodes and sent to a central station. That central station can either be a local computer, located in the lab or the house, or it could be a distant server that receives the data, processes it, and uses it for activity recognition. The main advantage of cloud computing is to offset the processing and complex calculations to powerful distant machines [27]. It also provides a very high availability, and a single set of machines can be used for different applications by splitting their processing power and resources amongst different tasks.

More recently, as another sign of the trend shifting from centralized to distributed and pervasive computing, the Fog computing model has been developed, where some of the processing is distributed to intermediary nodes between the end device and the distant servers [28].

2.2.1. Local Processing

Smartphones can also be used for local activity recognition, such as the system built by He et al. [29] relying on sensors wirelessly connected to a gateway, connected to a smartphone via USB. Kouris et al. [30] have used 2 body-worn accelerometers and a heart rate monitor sending data to a smartphone using Bluetooth. Similarly, Zhang et al. [31] have used accelerometers and gyroscopes communicating with a smartphone using Bluetooth. Biswas et al. [32] have used an accelerometer on the dominant wrist of a subject to perform arm movement recognition. The data is sent from the accelerometer to a local computer, which then transfers it to a FPGA board using a RS232 cable. The whole processing and machine learning is performed by the FPGA that is directly linked to the local computer.

2.2.2. Cloud Processing

Cloud processing is particularly suited to healthcare applications, as it allows to build a system where the data and results of activity recognition are stored online. These results can then be accessed by the patients directly, as well as the physicians, and any abnormal behavior can be remotely spotted in real-time.Ganapathy et al. [33] have presented a system made of a set of body sensors, such as blood pressure, heart rate, respiration rate, ECG, and SPO2 sensor sending data to a smartphone using Bluetooth. These data are then wirelessly sent to a distant server. Predic et al. [34] have used the inertial sensors of a smartphone, combined with air quality data collected through environmental sensors. All of the data is then sent to a cloud to be stored and processed. Mo et al. [35] have used a smartphone to collect data from body worn sensors with energy harvesting capabilities, and to send it over to a distant server for processing and activity recognition. Serdaroglu et al. [36] have used a wrist worn watch with a built-in accelerometer to collect data in order to monitor patients' daily medication intake. The data is sent wirelessly from the watch to a gateway, connected to a computer via USB, before being sent over to a web server. The monitoring application is cloud-based, and both patients and doctors can access it. Khan et al. [37] have also used on-body worn accelerometers with an emphasis on the accurate positioning of the nodes on the subject's body in order to improve the activity recognition accuracy. They have used energy-based features and a cloud-based architecture to perform activity classification. Fortino et al. [38] have presented the CASE (Cloud-Assisted Agent-Based Smart Home Environment) system where data is collected from both environmental and wearable sensors. Environmental sensors send data to a base station, such as a Raspberry Pi or a Beagle Bone, and wearable sensors send collected data to a smartphone. These data are then forwarded to the cloud-based architecture for processing.Cheng et al. [19] have compared the efficiency of a local Python machine learning script running on a local computer and a distant Matlab algorithm running on the cloud. They have found that the accuracy of SVM seems to be better on the cloud (97.6% vs. 95.7%) and the accuracy of k-NN is better using the local Python script (97.9% vs. 95.9%). As cloud-based solution often times require the use of proprietary software and additional costs to rent storage space and processing power, Cheng et al. have decided to rely mainly on the localized algorithm, and only use the cloud algorithm if the local one fails.The context of application of a real-time activity recognition system is the main criteria to determine whether to rely on a local or cloud based architecture. In the case of a healthcare system where remote monitoring from the doctors is necessary, a cloud architecture seems to be more suited. It also allows the patient themselves to access their history, as well as family members, either from their computer or an application on their smartphone or tablet. However, cloud-based systems are usually more costly, because of the monthly fee and the maintenance needed, as well as the possible use of proprietary software, and the difference in activity recognition accuracy alone usually cannot justify the extra cost.The diagram below summarizes the main architectures used in the field of real-time activity recognition (Figure 2). The dashed line makes the difference between local and cloud based systems in order to compress the representation. The first approach uses sensor nodes wired to a gateway node that communicates with a computer. This approach is generally used with wearables in Wireless Body Sensor Networks (WBSN), where nodes are attached on different parts of the user's body, and a gateway node is used to store and transfer the collected data to a local station that could be connected to the cloud. This approach allows for a faster and more reliable communication between the sensor nodes and the gateway, but it makes the system less practical and more invasive as the number of nodes increases because of all the wires involved. The second approach uses wireless communication between the sensor nodes and the gateway node. This approach is generally used with environmental sensors, as using wires between every single sensor node and the gateway node would be impractical in a smart home. The gateway node is directly plugged into the central station, which could process the data locally or send it over to a cloud server [36]. In the last presented architecture, all of the sensor nodes are sending data wirelessly to a smartphone, acting as a gateway node, sending data to the local station. In some applications, the smartphone sends data over directly to cloud servers [33]. Two different gateway nodes can also be used in the same system [38].

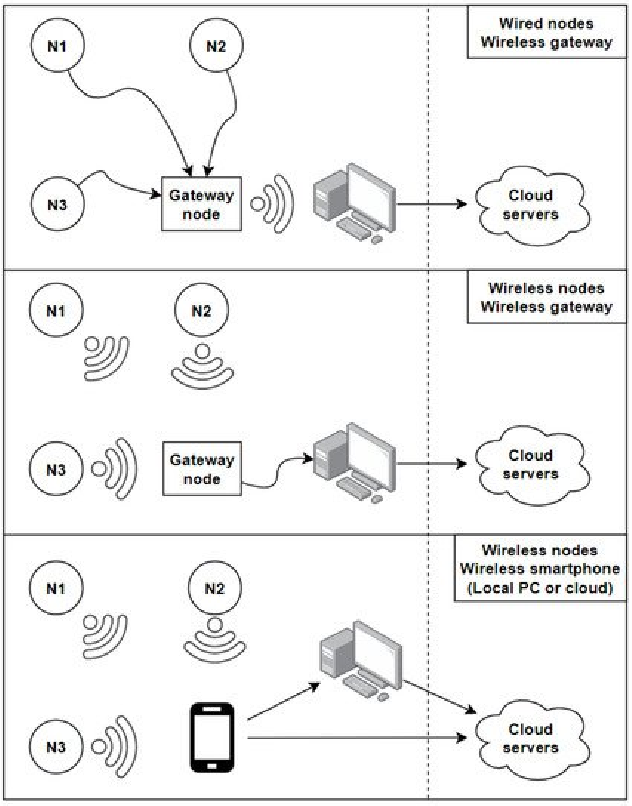

Figure 2. Diagram representing the three main local and cloud architectures used for real-time centralized activity recognition. Sensor nodes can be connected to a gateway node using USB/RS232 cables. The gateway node sends the collected data wirelessly to a central station that can process the data locally or send it over to a cloud (top). In the second configuration, sensor nodes can send the collected data wirelessly to a gateway node, connected to the central station (middle). In the third configuration, all communication is performed wirelessly between the sensor nodes, a smartphone used as a gateway node, and a central station or cloud servers (bottom).

2.3. Real-Time Machine Learning

Supervised machine learning algorithms are the most commonly used class of algorithm in activity recognition. Data are collected, manually labeled, and used to train a model. Once the model has been trained, it can be used to perform activity recognition. In real-time applications, the simplest approach is to collect data and train the model offline. Once the model is trained, it can be used online to classify new instances and perform activity recognition. It is also possible to train some models in real-time by using smaller batches from the continuous flow of incoming data in a streaming context. In both cases, the main challenge is to label data and evaluate the true performance of an algorithm. If activities are performed in real-time without a human observer to provide the ground truth for verification, the accuracy of the model cannot be evaluated for supervised machine learning based systems.In this section, we explore some proposed techniques of automatic activity labeling, highlight the main challenges of real-time activity in a streaming context, and review some of the approaches that have been applied to activity recognition applications.

2.3.1. Activity Labeling

In offline machine learning systems, labeling is done by hand, either directly by an observer, after the experiment by using video recording, or by the subject themselves with the help of a form or an application.Naya et al. [39] have provided nurses in a hospital with a voice recorder to allow them to describe the activity they were performing in real-time. However, the recording still has to be interpreted by a human in order to turn it into a usable activity label that can be used to train a machine learning model. In real-time systems, activity recognition can either happen in a closed or open universe. In a closed universe, a complete set of activities is defined from the start [24], and any new collected data forming a feature vector will be classified in one of the classes of this set, or remain an unclassified instance in some cases [40]. In an open universe, new activities can be discovered as the system runs, and irrelevant activities can be discarded.Suryadevara et al. [18] have created a table mapping the type and location of each sensor, as well as the time of the day to a specific activity label. Using this technique, they have achieved 79.84% accuracy for real-time activity annotation compared to the actual ground truth collected from the subjects themselves. This approach is efficient in systems where a single sensor or a set of sensor can be discriminative enough to narrow down the activity being performed. In this closed context, no new activities are discovered. Fortino et al. [38] have used the frequent itemset mining algorithm Apriori to find patterns in collected data. Events represented by quadruples containing a date, a timestamp, the ID of a sensor and its status are recorded. These events are then processed to form a list of occupancy episodes in the form of another quadruple containing a room ID, a start time, a duration, and the list of used sensors. The idea behind this quadruple is to automatically represent activities that emerge as a function of the sensors firing, the time and duration of their activation, as well as the room in which they are located. Apriori is used to find the most frequent occupancy episodes, which are then clustered. Clusters can change throughout the system's lifecycle, and each cluster acts as the representation of an unknown activity. The name of the activity itself cannot automatically be determined, and human intervention is still necessary to properly label it. This method is useful when there is a high correlation between time, location, and the observed activity.Through active learning [41], it is possible to provide the user with an interface that allows them to give feedback over the automatic label suggested by the system. If the label is correct, the user specifies it is, and the new learned instance is added to the base of knowledge. Semi-supervised learning [42] can be used together with active learning to compare activity annotation predictions with the ground truth provided directly by the user. The model is first trained with a small set of labeled activities. Classification results for unknown instances are then checked using active learning, and added to the training set if they have been correctly classified, thus progressively allowing the training set to grow, and making the model more accurate and versatile in the case of activity discovery.The smaller the set of activities to classify is, the easier it is to link a sensor to an activity. However, a sensor could be firing, letting us know that the sink is running, but it would be impossible to determine if the subject is washing their hands, brushing their teeth, shaving, or having a drink. The more complex the system gets, and the more sensors are added, the more difficult it becomes to establish a set of rules that link sensor activation to human activities. Most real-time systems have to be periodically re-trained with new ground truth in order to include new activities, and take into account the fact that the same activities could be performed in a slightly different way over time.

2.3.2. Machine Learning

There are two main distinctions to be made when it comes to real-time machine learning: Real-time training and real-time classification. The latter represents the simplest case of real-time machine learning: A model is trained offline with a fixed dataset, in the same way offline activity recognition is performed, and it is then used in real-time to classify new instances. For most supervised learning based methods, classification time is negligible compared to training time. Models that require no training, such as k-NN require a higher classification time. Nugyen et al. [24] have used binary rules that map sensor states to an activity label to classify new instances in near real-time (5 min time slices). Other straightforward approaches use real-time threshold based classification [43] or a mapping between gyroscope orientation and activities [32].Cheng et al. [19] have used both local and cloud based SVM and k-NN implementation for real-time activity recognition on an interactive stage where different lights are turned on depending on the activity of the speaker. k-NN [44] requires no training time, as it relies on finding the k nearest neighbours of the new data instance to be classified. However, in its original version, it has to compute the distance between the new instance and every single data point in the dataset, making it a very difficult algorithm to use for real-time classification.Altun et al. [22] have compared several algorithms in terms of training and storage time for activity recognition using wearable sensors. Algorithms such as Bayesian Decision Making (BDM), Rule Based Algorithm (RBA), decision tree (DT), K-Nearest Neighbor (k-NN), Dynamic Time Warping (DTW), Support Vector Machine (SVM), and Artificial Neural Network (ANN) are trained using 3 different methods: Repeated Random Sub Sampling (RRSS), P-fold, and Leave one out (L1O) cross validation. Using P-fold cross validation, DT has been shown to have the best training time (9.92 ms), followed by BDM (28.62 ms), ANN (228.28 ms), RBA (3.87 s), and SVM (13.29 s). When it comes to classification time, ANN takes the lead (0.06 ms), followed by DT (0.24 ms), RBA (0.95 ms), BDM (5.70 ms), SVM (7.24 ms), DTW (121.01 ms, taking the average of both DTW implementations), and k-NN in last position (351.22 ms). These results show that DT could be suited for both real-time training and classification, as it ranks high in both categories. Very Fast Decision Tree (VFDT) based on the Hoeffding bound have been used for incremental online learning and classification [45]. Even though ANN requires the most training time, it performs the quickest classification out of all the algorithms compared in this paper, and could therefore be used in a real-time context with periodic offline re-training. The ability of neural networks to solve more complex classification problems and automatically extract implicit features could also make them attractive for real-time activity recognition. These results were obtained for classification of 19 different activities in a lab setting, after using PCA to reduce the number of features.Song et al. [46] have explored online training using Online Sequential Extreme Learning Machine (OS-ELM) for activity recognition. Extreme Learning Machine is an optimization learning method for single-hidden layer feedforward neural network introduced by Huang et al. [47]. This online sequential variation continuously uses small batches of newly acquired data to update the weights of the neural network and perform real-time online training. ELM is particularly adapted to online learning as it has been crafted to deal with the issue of regular gradient-based algorithms being slow, and requiring a lot of time and iterations to converge to an accurate model. ELM has been shown to train NN thousands of times faster than conventional methods [47]. OS-ELM has been compared to BPNN [46], and has achieved an average activity recognition rate of 98.17% accuracy with a training time of about 2 s, whereas BPNN stands at 82.56% accuracy with a 55 s training time. This result show that neural networks could be a viable choice for online training as well as online classification, as long as the training procedure is optimized.Palumbo et al. [48] have used Recurrent Neural

Networks (RNN) implemented as Echo State Networks (ESN) coupled with a decision tree to perform activity recognition using environmental sensors coupled with a smartphone's inertial sensor. The decision tree constitutes the first layer, and possesses 3 successive split nodes based on the relative value of collected data. Each leaf of this decision tree is either directly an activity, or an ESN that classifies the instance between several different classes. ESN is a particular implementation of the Reservoir Computing paradigm, that is well suited to process streams of real-time data, and requires much less computation than classical neural networks. The current state of a RNN is also affected by the past value of its input signal, which allows it to learn more complex behavior variations of the input data, and is especially efficient for activity recognition, as the same recurrent nature of certain behaviors can be found in human activity recognition (present activities can help inferring future activities). On a more macroscopic scale, Boukhechba et al. [49] have used GPS data from a user's smartphone, and an online, window-based implementation of K-Means in order to recognize static and dynamic activities.In a streaming context with data being collected and used for training and classification, several issues can arise. Once the architecture and communication aspects have been sorted as described in the previous sections, the nature of the data stream itself becomes the issue. As time goes on, it can be expected that data distribution will evolve over time and give rise to what is referred to as concept drift [50]. Any machine learning model trained on a specific distribution of input data would see its performance slowly deteriorate as the data distribution changes. As time goes on, new concepts could also start appearing in data (new activities), and some could disappear (activity no longer performed). These are called concept evolution and concept forgetting. The presence of outliers also has to be handled, and any new data point that does not fit in the known distribution does not necessarily represent a new class.Krawczyk et al. [51] have reviewed ensemble learning based methods for concept drift detection in data streams. They have also identified different types of concept drift such as incremental, gradual, sudden, and recurring drift. Ensemble learning uses several different models to detect concept drift, and to re-train a model when concept drift is detected. The freshly trained model can be added to the ensemble or replace the currently worst performing model if the ensemble is full. Concept drift is usually detected when the algorithm's performance starts to drop significantly and does not return to baseline. Some of the challenges of concept drift detection are to keep the number of false alarm to a minimum, as well as to detect concept drift as quickly as possible. Various methods relying on fixed, variable size and a combination of different window sizes have been described in [51]. The figure below illustrates the process of concept drift detection and model retraining (Figure 3).

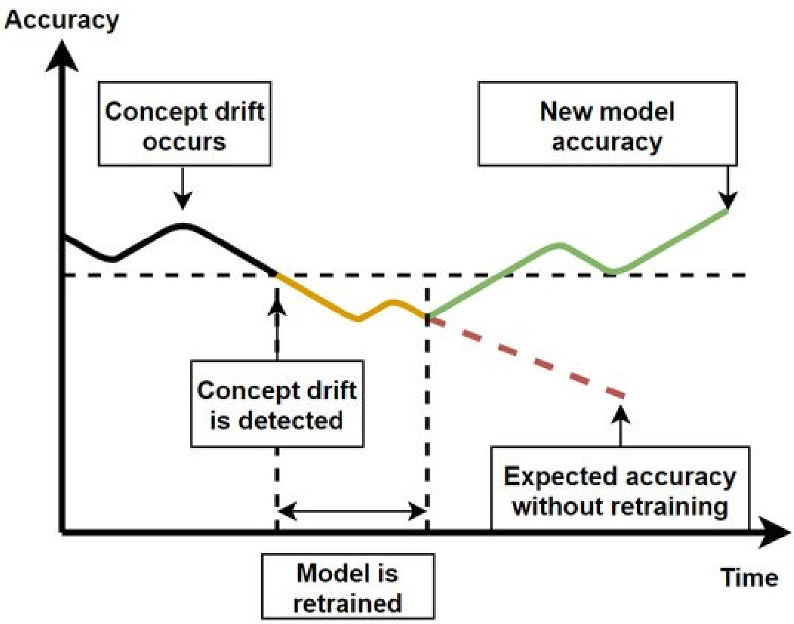

Figure 3. Diagram of the evolution of a model's accuracy over time as concept drift occurs in two cases: With retraining and without retraining.Additionally, in high speed data-streams with high data volumes, each incoming example should only be read once, the amount of memory used should be limited, and the system be ready to predict at any time [52]. Online learning can either be performed using a chunk-by-chunk or one-by-one approach. Each new chunk or single instance is used to test the algorithm first, and then to train it, as soon as the real label for each instance is specified. This comes back to the crux of real-time activity recognition, which is the need to know the ground truth as soon as possible to ensure continuous re-training of the model. Ni et al. [53] have addressed the issue of dynamically detecting window starting positions with change point detection for real-time activity recognition in order to minimize necessary human intervention the segment data before labeling it.In limited resources environments, such as when machine learning is performed on smartphone, a trade-off often has to be found between model accuracy and energy consumption as shown by Chetty [54] and He [29]. This is especially true for distributed real-time processing, which we cover in the next section.

2.2. Discussion

In this section, we have covered the main steps to follow in order to build a real-time activity recognition system, starting from the physical architecture, to the communication protocols, data storage, local and remote processing, activity labeling, machine learning in data streams, and the interaction with different actors of the system. Types and location of sensors have been covered in offline activity recognition section.Many challenges still have to be faced for real-time activity recognition, such as real-time activity labeling in real-time, adaption to concept drift and evolution, timely online training and classification, as well as memory efficient algorithms. We have reviewed several real-time algorithms and highlighted the importance of semi-supervised and unsupervised approaches to reduce necessary human intervention.With portable and embedded devices becoming more and more powerful by the year, the Internet of Things becoming the new standard, and the convenience of pervasive computing, it seems natural to transition from centralized to distributed activity recognition, and to explore the new research challenges and opportunities that rise with it.

3. Real-Time Distributed Activity Recognition

With IoT boards becoming smaller and more powerful, distributed activity recognition is the next logical step to weave technology more profoundly in everyday life. In an ideal case and a fully distributed system, no node is essential and there is no single point of failure, as opposed to centralized activity recognition that relies on a local computer or a distant server for processing and classification. Distributed activity recognition also allows the creation of fully autonomous systems that do not rely on an external internet connection to keep performing their tasks. A local system also implies a higher degree of privacy, as no single node stores all of the data.However, distribution comes with a whole new set of problems, especially in a real-time context. In this section, we first cover the on-node processing aspects of distributed activity recognition, then we move on to communication between nodes, and we conclude on distributed machine learning. In each section, we review some of the methods that have been found in the literature and we identify the main considerations to take into account when building a distributed activity recognition system.

3.1. On-Node Processing

The main difference between centralized and distributed systems is the nodes' ability to perform some processing before communicating with other nodes. In a distributed system, basic feature extraction can be performed on low-power nodes, which allows it to save a lot of energy on communication, as features extracted on a data window are generally more compact than raw data [55].However, because of the nodes' limited processing power, feature extraction has to be as efficient as possible. Lombriser et al. [56] have compared features in terms of the information gain they provide to the classifier versus their computational cost, and they have found that the mean, energy, and variance were the most efficient features for on-body activity recognition with DT and k-NN, using accelerometer data.Roggen et al. [57] have used a limited memory implementation of a Warping Longest Common Subsequence algorithm to recognize basic movement patterns. In order to reduce necessary processing power and code space, the algorithm relies on integer operations rather than floating point operations. They have implemented the algorithm on a 8-bit AVR microcontroller and a 32-bit ARM Cortex M4 microcontroller, and they have shown that a single gesture could be recognized using only 0.135 mW of power, with the theoretical possibility of recognizing up to 67 gestures simultaneously on the AVR, and 140 on the M2. Because of the nature of the algorithm they have used, Roggen et al. have chosen to not extract any features and find the Longest Common Subsequence directly using raw data. This leads to more complex movement recognition being less efficient, but it reduces the computational burden on the nodes.Lombriser et al. [56] have observed that bigger window sizes and wider overlaps yielded higher activity recognition accuracy, but came with a higher computation burden as well. They have settled for a middle ground with 2.5 s windows and a 70% overlap. Roggen et al. [57] have reached similar conclusions as they have found that the shorter the sequence to match is, the lighter the computational burden is at the expense of losing specificity on activities to be recognized. Indeed, shorter patterns are less discriminative, and the shorter they are, the further away they are from describing a full gesture.Jiang et al. [58] have focused on rechargeable sensor networks, using RF waves from RF readers to supply energy to Wireless Identification and Sensing Platform tags (WISP tags). They have compared different scheduling methods to distribute energy amongst the nodes at runtime in order to achieve the highest Quality of Monitoring (QoM) possible, defined by the ratio between occurred and captured events. They have found that using a hybrid method with scheduling on both the reader and the WISP tags allowed for the best QoM results.In order to save energy whilst still maintaining a high activity recognition accuracy, Aldeer et al. [59] have used a low energy, low reliability motion sensor (ball-tube sensor) together with a high energy, high reliability sensor (accelerometer) for on-body acitivity recognition. The idea is to put the accelerometer to sleep when the user is mostly static, and to use cues from the ball-tube accelerometer to detect the beginning of a movement in order to enable accelerometer data collection. This allows to reduce energy consumption by a factor of 4 during periods where the user is static.

3.2. Communication

Communication is one of the main sources of energy loss in WSN together with high sampling sensors. Lombriser et al. [56] have found that radio communication accounted for 37.2% of the total node's energy consumption, which is the second biggest energy consumption after the microphone (45.8%) and far more than the CPU (3.5%). It is therefore crucial to find ways to save more energy by communicating efficiently and minimizing packet loss as much as possible.Considerable packet loss can be experienced in WBSN when a node tries to communicate with another node that is not in Line of Sight (LOS), such as shown in [56] where the transmission rate from a node on the waist to a node on the ankle falls to 78.93% when the subject is sitting down. Zang et al. [60] have designed the M-TPC protocol to address this issue in the specific case of the walking activity with a wrist worn accelerometer and a smartphone in the opposite pocket. They have noticed a negative correlation between acceleration values picked up by accelerometers and packet loss. Indeed when the body of the subject stands between the node and the smartphone, more data loss is experienced. M-TPC is designed to send packets of data only when acceleration is at its lowest, meaning that the subject's arm is either in front or behind their body, at the peak of the arm's swinging motion. This protocol allows to reduce transmission power by 43.24% and reduces packet loss by 75%.Xiao et al. [61] have used Sparse Representation based Classification with distributed Random Projection (SRC-RP) to recognize human activity. By randomly projecting data to a lower dimensional subspace directly on the nodes, they can efficiently compress data and save on transmission costs while still maintaining a high activity recognition accuracy. With a 50% data compression rate, they have achieved an activity recognition accuracy of 89.02% (down from 90.23% without any compression) whilst reducing energy consumption by 20%. The authors have compared RP to PCA, and found that RP yields a very close accuracy, whilst being less computationally expensive than PCA, as well as being data independent.De Paola et al. [62] have used a more centralized approach, but they have explored the issue of optimal sensor subselection in order to save energy in a WSN. Each node collects data, and a central Dynamic Bayesian Network is used to perform activity recognition based on environmental sensors in a smart home. The information gain of each sensor is computed, and if the state of the system is not expected to change much, the least relevant sensors are set to sleep mode to save energy. Using this adaptive method, they have achieved 79.53% accuracy for activity recognition, which is similar to the accuracy using all sensors, but with a power consumption three times lower.

3.3. Machine Learning

In an ideal distributed activity recognition system, processing and classification tasks should be equally split between all the nodes. However, most approaches in the literature still rely on a central node to perform the final classification steps. There are different levels of distribution, starting from minimal on-node processing, to partial on-node classification, to fully distributed classification. A summary of the different degrees of distribution observed in the reviewed literature can be found below (Figure 4).

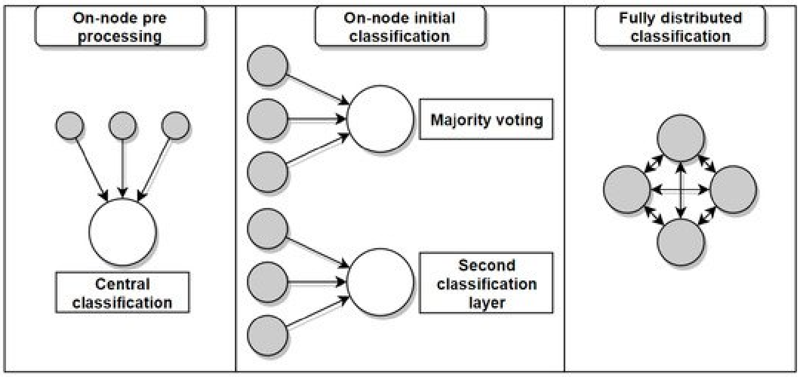

Figure 4. Diagram of different degrees of distribution for distributed machine learning in WSNs. On-node pre-processing is performed before sending the data to a central node for classification (left). Initial classification is performed on the sensor nodes before performing a second classification on the central node (middle). The classification process is distributed among the nodes (right).Farella et al. [63] have used on-body accelerometers for activity recognition. Their approach is mostly centralized, but they have used a lookup table that matches specific accelerometer values to the orientation of the node. This table is directly implemented on the nodes and allows a first basic pre-processing step to be carried out locally. Bellifemine et al. [64] have presented an agent-oriented implementation of the SPINE framework, designed to facilitate signal processing based tasks, such as activity recognition. Each node is able to perform basic feature extraction for accelerometer values based on a split function (mean, minimum, maximum for each axis), and uses an aggregation function to group up the features and send them to a central station.Roggen et al. [57] have used majority voting in their LM-WLCSS system to find the best overall match among the nodes. Zappi et al. [65] have trained a HMM on each node and used a central node to weigh in the contribution of each node and perform classification using either majority voting or a Naive Bayesian fusion scheme. Bayesian voting proved more efficient, especially when faced with noise in the data, as

the contribution of noisy nodes is reduced. Palumbo et al. [66] have used wearable and environmental sensors for distributed activity recognition using Received Signal Strength (RSS) values and Recurrent Neural Networks. In the distributed implementation for the walking activity, each node collects acceleration and RSS data, discards bad packets and sends data to the feature extraction module. Each node uses these features to make an activity prediction using the Echo State Network. Each prediction is sent to a gateway that performs majority voting as the final classification step. They have achieved a 91.11% accuracy for two activities (standing up and sitting down) using a centralized version of the algorithm. The authors have also shown that their algorithm only requires 2kB of RAM, which makes it suitable for embedded applications.Wang et al. [67] have used a 2-layer classification system where gesture recognition is performed directly on the sensor nodes worn on the body and around the wrists, and sent to a central smartphone that performs higher level activity recognition. On each node, K-Medoid clustering is used to find templates for each gesture. At activity recognition time, Dynamic Time Warping (DTW) is used to find the closest matching sequence to the training set using a 1 s sliding window on incoming data. Each gesture has a different pattern, and the body and wrist nodes try to match different patterns. All sensors send the results of this initial classification to the central node that merges it into a bitmap. Emerging Pattern (EP) is then used to recognize higher level activities. The average accuracy of the system is 92.9%, however, it is not as suited for interleaved activity recognition because of the importance of pattern temporality. They have also shown that performing the first stage of classification directly on the nodes allowed for 60.2% savings in energy consumption when compared to sending raw data for central processing.Atalah et al. [68] have used a 2-stage Bayesian classifier using data from an ear-worn sensor combined with environmental sensors. The ear-worn sensors first classifies the performed activity into a different category of activities based on the heart rate of the user. This first estimation is sent to the central node that receives additional data from the environmental sensor and uses it to perform the second classification stage. The authors have found that using the combination of both types of sensors allowed to reduce class confusion rates by up to 40% for some subjects, rather than relying only on wearable sensors.Amft et al. [55] have also used a 2-stage classifier in their distributed user activity sequence recognition system. The first layer of classification happens directly on the nodes where atomic activities are recognized by finding the closest match to a known pattern in the data. These atomic activities are organized in an alphabet, and different sequences of atomic activities correspond to a composite activity. Using body-worn accelerometers and environmental sensors in a car assembly scenario, the authors have identified 47 atomic and 11 composite activities, and achieved a 77% activity recognition accuracy. They have also observed a 16% data loss when sending raw data from the nodes to a central coordinator, which highlights the advantage of on-node processing for reduced data transmission in distributed systems.Fukushima et al. [69] have fully distributed a Convolutional Neural Network (CNN) in a WSN. Each node of the network is responsible for the computations of all the layers for a specific unit of the CNN, the same way CNN are used for image recognition, where a unit is a pixel or a group of pixels, except each node collects temperature or motion data in the two presented experiments. The CNN consists of an input layer, T hidden layers, a fully-connected layer and an output layer. Each hidden layer contains a convolutional layer and an optional pooling sublayer. Each convolutional layer has K filters. To ensure a lighter computational load, the number and size of the filters are limited. When a node receives all the necessary inputs from neighboring nodes, it goes through the layers, computes the output, and advertises the output for the units in the following layer. When a sensor node obtains ground truth, it begins the distributed backpropagation process, where each unit updates its weights based on the propagation from the previous units, therefore, there is no global optimization of the weights. MicroDeep has achieved 95.57% accuracy for temperature discomfort recognition, whereas a standard CNN running on a computer has achieved 97.1% accuracy. Using optimal parameters, MicroDeep achieves the same accuracy as the standard CNN, but its communication cost is multiplied by 8 as opposed to using a feasible configuration. Since the CNN forms a 2-D grid, there might be cells without an associated sensor. These cells are referred to as holes, and the authors have shown that they can still maintain a 92.4% accuracy even with 20% holes. This is an example of a fully distributed, real-time activity recognition system, which relies on a high number of sensor nodes to distribute processing.Bhaduri et al. [70] have presented a distributed implementation of a Decision Tree in a P2P network using misclassification error as a gain function. Each node has a set of training examples and the aim of the algorithm is to select the best split based on each node’s decision, using majority voting, in order to build an optimal tree. The authors have also reduced communication costs between the node by introducing a parameter that controls how essential an event has to be in order for a node to send a message to its neighbouring nodes. Another parameter is introduced to enforce a minimum delay between the transmission of two consecutive messages on each node. An event could be the introduction or disappearance of a node in the network, the change of state of a node, or additional data received, that could possibly change the structure of the tree. Decision Trees and Random Forest are an interesting choice for real-time activity recognition as discussed in the previous section, thanks to their acceptable training and classification time, and generally good performance in the field. To the best of our knowledge, no fully distributed implementation of DT or RF have been used for activity recognition in a healthcare context.Navia-Vazquez et al. [71] have addressed the distribution of the Support Vector Machine algorithm. They have presented a naïve approach where a local SVM is trained at every node using local data, the support vectors are sent to the other nodes, a new training set is built at each node using the support vectors received from other nodes, and new support vectors are computed and exchanged until convergence. Another method using Semiparametric Support Vector Machine is presented; it allows to reduce the communication costs compared to the naïve alternative. Both methods achieve a much better accuracy than a SVM only using locally available data on a single node. Support Vector Machine is not the fastest algorithm to train, but distributed optimizations such as the second method presented in that paper could allow dynamic retraining for a limited communication cost in sufficiently powerful WSNs.In an embedded context, the main considerations are to keep processing as close as possible to the data source, to optimize sensor usage and scheduling in order to save energy, to compress data and promote efficient node communication, and to use a higher number of nodes if accuracy is the priority. The use of local optimization procedures is also mandatory to reduce communication costs, such as Fukushima's local backpropagation for a distributed CNN [69].

3.2. Discussion

Real-time distributed activity recognition in a streaming context remains a very interesting field with many challenges to overcome. The six optimization angles of processing, memory, communication, energy, time, and accuracy leave no room for error in the conception of an efficient activity recognition system.Different degrees of distribution can be implemented depending on the size of the WSN, as well as the algorithm and the context of the experiment. A higher number of nodes generally allows for a better accuracy through voting, and a better noise resilience, whereas a smaller number of nodes allows for less intrusive systems that can easily mesh in daily life activities. Processing can be reduced by using approximations, such as using integers instead of floats [57], extracting features with the highest information gain to extraction cost ratio [56], or downsizing the windows and the length of patterns to be matched [57], which also reduces the memory overhead necessary for gesture recognition. Optimizing sensors sampling rate allows both processing and memory savings [59]. Using integers also allows to save a lot of memory at the cost of a loss in accuracy. Compressing data using random projection is also a way to reduce memory usage as well as communication costs [61], but comes with extra on-node processing.Communication can be optimized through the use of efficient protocols [60] as well as parameters controlling the importance of a message before sending it, as well as enforcing a minimum delay between consecutive transmission from the same node [70].Energy is the variable that dictates most of the design and implementation choices for distributed activity recognition, especially in a healthcare context where it is expected to be able to monitor patient around the clock. Any processing optimization usually reduces energy consumption, whereas any extra processing to save on memory or communication will increase it. Rechargeable sensors can be a viable option in some cases [58]. In a sustainable development perspective, it would be interesting to include renewable, portable energy sources, such as small solar panels or energy harvesting modules based on body motion, whilst still making sure the system is as lightweight and non-intrusive as possible, especially in WBSNs.Time is at the centre of real-time activity recognition, and any algorithm should be able to provide a classification result in a timely manner. Depending on the context of application, a one minute delay could be acceptable (activity of daily living monitoring) or it could be way too long (self-driving car).The accuracy of the algorithm should be as high as possible despite all of these constraints, and a lot of time, a trade off has to be found between time and accuracy, as well as energy and processing. It is expected that distributed training based algorithm will not perform as good as good as their offline, batch training based equivalents, but it is a necessary compromise for real-time activity recognition. Instead, the trend goes toward higher degrees of distribution with dozens or hundreds of nodes to compensate the accuracy lost while training using partial data with an increase in data sources and a better modeling of the environment or the subject in dense WBSNs.All of these angles are usually coupled, and trying to optimize one of them can result in a loss of performance in another one. A summary of some of the approaches used in the literature have been summarized in the table above (Table 2).Table 2. Comparison of the impact of different optimization methods used in the literature on processing, memory, communication, energy, time, and accuracy of the system. A + symbol (green cell) means a positive impact, a = symbol (yellow cell) means no noticeable impact, a - symbol (red cell) means a negative impact, and a ∼ symbol (gray cell) means an impact that could be positive or negative based on the method implementation.

Method	Processing	Memory	Comm.	Energy	Time	Acc.
Integers instead of floats	+	+	+	+	+	-
Highest gain features	+	=	=	=	=	-
Smaller windows/patterns	+	+	+	+	+	-
Lower sensor sampling rate	+	+	+	+	=	-
Compressing data on node	-	+	+	∼	=	-
Comm. reduction protocols	-	-	+	∼	-	-

Method	Processing	Memory	Comm.	Energy	Time	Acc.
Rechargeable sensors	-	=	-	+	=	=
Sensor subselection	-	=	+	+	-	-
More nodes	-	-	+	-	=	+
Majority voting	=	=	+	+	=	+
2-layer classification	+	=	-	-	=	+
Local optimization	+	+	+	+	+	-

References

1. United, N. World Population Ageing; Technical Report; United Nations, Department of Economic and Social Affairs, Pop ulation Division: New York, NY, USA, 2017.
2. Smith, S.K.; Rayer, S.; Smith, E.A. Aging and disability: Implications for the Housing Industry and Housing Policy in the United States. J. Am. Plan. Assoc. 2008, 74, 289–306.
3. Gaugler, J.; James, B.; Johnson, T.; Scholz, K.; Weuve, J. 2016 Alzheimer's disease facts and figures. Alzheimer's De ment. 2016, 12, 459–509.
4. Harper, R. Inside the Smart Home. In Interactive Marketing; Springer Science & Business Media: Berlin/Heidelberg, Ge rmany, 2006; Volume 5, pp. 307–308.
5. Virone, G.; Wood, A.; Selavo, L.; Cao, Q.; Fang, L.; Doan, T.; He, Z.; Stoleru, R.; Lin, S.; Stankovic, J.A. An assisted livi ng oriented information system based on a residential wireless sensor network. In Conference Proceedings—1st Trans disciplinary Conference on Distributed Diagnosis and Home Healthcare, D2H2 2006, Arlington, VA, USA, 2–4 April 200 6; IEEE: Piscataway, NJ, USA, 2006; pp. 95–100.
6. Maitre, J.; Bouchard, K.; Bertuglia, C.; Gaboury, S. Recognizing activities of daily living from UWB radars and deep lear ning. Expert Syst. Appl. 2021, 164.
7. Fortin-Simard, D.; Bilodeau, J.S.; Bouchard, K.; Gaboury, S.; Bouchard, B.; Bouzouane, A. Exploiting Passive RFID Tec hnology for Activity Recognition in Smart Homes. IEEE Intell. Syst. 2015, 30, 7–15.
8. Baghezza, R.; Bouchard, K.; Bouzouane, A.; Gouin-Vallerand, C. Activity Recognition in the City using Embedded Syst ems and Anonymous Sensors. Procedia Comput. Sci. 2020, 170, 67–74.
9. Shafique, K.; Khawaja, B.A.; Sabir, F.; Qazi, S.; Mustaqim, M. Internet of things (IoT) for next-generation smart system s: A review of current challenges, future trends and prospects for emerging 5G-IoT Scenarios. IEEE Access 2020, 8, 23 022–23040.
10. Jahn, A.; Morold, M.; David, K. 5G Based Collision Avoidance-Benefit from Unobtrusive Activities. In Proceedings of the 2018 European Conference on Networks and Communications, EuCNC 2018, Ljubljana, Slovenia, 18–21 June 2018; p p. 352–356.
11. Chettri, L.; Bera, R. A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems. IEEE Internet T hings J. 2020, 7, 16–32.
12. Bayrakdar, M.E. Priority based health data monitoring with IEEE 802.11af technology in wireless medical sensor netwo rks. Med. Biol. Eng. Comput. 2019, 57, 2757–2769.
13. Aust, S.; Prasad, R.V.; Niemegeers, I.G. IEEE 802.11ah: Advantages in standards and further challenges for sub 1 GH z Wi-Fi. IEEE Int. Conf. Commun. 2012, 6885–6889.
14. Lochan, V.; Fakharzadeh, M.; Choi, S. Wi-Fi on Steroids: 802.11 ac and 802.11 ad. IEEE Wirel. Commun. 2013, 20, 30 –35.
15. Yaakop, M.B.; Malik, I.A.A.; Bin Suboh, Z.; Ramli, A.F.; Abu, M.A. Bluetooth 5.0 throughput comparison for internet of th ing usability a survey. In Proceedings of the 2017 International Conference on Engineering Technology and Technopren eurship (ICE2T), Kuala Lumpur, Malaysia, 18–20 September 2017; pp. 1–6.
16. Kinney, P. ZigBee Technology: Wireless Control that Simply Works. Commun. Des. Conf. 2003, 2, 1–7.
17. Sanchez-Iborra, R.; Sanchez-Gomez, J.; Ballesta-Viñas, J.; Cano, M.D.; Skarmeta, A.F. Performance evaluation of lora considering scenario conditions. Sensors 2018, 18, 772.

18. Survadevara, N.K.; Mukhopadhyay, S.C.; Rayudu, R.K. Applying SARIMA time series to forecast sleeping activity for wellness model of elderly monitoring in smart home. In Proceedings of the International Conference on Sensing Technology, ICST, Kolkata, India, 18–21 December 2012; pp. 157–162.
19. Cheng, L.; Li, Y.; Guan, Y. Human Activity Recognition based on Compressed Sensing. In Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 9–11 January 2017; pp. 1–7.
20. Suryadevara, N.K.; Mukhopadhyay, S.C.; Wang, R.; Rayudu, R.K. Forecasting the behavior of an elderly using wireless sensors data in a smart home. Eng. Appl. Artif. Intell. 2013, 26, 2641–2652.
21. Baykas, T.; Goratti, L.; Rasheed, T.; Kato, S. On the spectrum efficiency of mesh and star topology wide area wireless sensor networks. In Proceedings of the 2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC), Washington, DC, USA, 2–5 September 2014; pp. 1819–1823.
22. Altun, K.; Barshan, B.; Tunçel, O. Comparative study on classifying human activities with miniature inertial and magnetic sensors. Pattern Recognit. 2010, 43, 3605–3620.
23. Busching, F.; Bottazzi, M.; Pottner, W.B.; Wolf, L. DT-WBAN: Disruption tolerant wireless body area networks in healthcare applications. In Proceedings of the 2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Lyon, France, 7–9 October 2013; pp. 196–203.
24. Nguyen, T.A.; Aiello, M. Beyond Indoor Presence Monitoring With Simple Sensors. In Proceedings of the 2nd International Conference on Pervasive Embedded Computing and Communication Systems, Rome, Italy, 24–26 February 2012; pp. 5–14.
25. Lin, H.C.; Chiang, S.Y.; Lee, K.; Kan, Y.C. An activity recognition model using inertial sensor nodes in a wireless sensor network for frozen shoulder rehabilitation exercises. Sensors 2015, 15, 2181–2204.
26. Chapron, K.; Plantevin, V.; Thullier, F.; Bouchard, K.; Duchesne, E.; Gaboury, S. A more efficient transportable and scalable system for real-time activities and exercises recognition. Sensors 2018, 18, 268.
27. Hayes, B. Cloud Computing. Commun. ACM 2008, 51, 9–11.
28. Mohan, N.; Kangasharju, J. Edge-Fog Cloud: A Distributed Cloud for Internet of Things Computations. In Proceedings of the 2016 Cloudification of the Internet of Things (CIoT), Paris, France, 23–25 November 2016; pp. 1–6.
29. He, Z.; Bai, X. A wearable wireless body area network for human activity recognition. In Proceedings of the International Conference on Ubiquitous and Future Networks, ICUFN, Shanghai, China, 8–11 July 2014; pp. 115–119.
30. Kouris, I.; Koutsouris, D. Activity Recognition Using Smartphones and Wearable Wireless Body Sensor Networks. In Proceedings of the International Conference on Wireless Mobile Communication and Healthcare, Kos, Greece, 5–7 October 2011; pp. 32–37.
31. Zhang, Y.; Sapir, I.; Markovic, S.; Wagenaar, R.; Little, T. Continuous Functional Activity Monitoring Based on Wearable Tri-axial Accelerometer and Gyroscope. In Proceedings of the 2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops, Dublin, Ireland, 23–26 May 2011; pp. 370–373.
32. Biswas, D.; Ajiwibawa, G.J.; Maharatna, K.; Cranny, A.; Achner, J.; Klemke, J.; Jobges, M. Real-time arm movement recognition using FPGA. In Proceedings of the IEEE International Symposium on Circuits and Systems, Lisbon, Portugal, 24–27 May 2015; pp. 766–769.
33. Ganapathy, K.; Vaidehi, V.; Poorani, D. Sensor based efficient decision making framework for remote healthcare. J. Ambient Intell. Smart Environ. 2015, 7, 461–481.
34. Predic, B.; Yan, Z.; Eberle, J.; Stojanovic, D.; Aberer, K. ExposureSense: Integrating daily activities with air quality using mobile participatory sensing. In Proceedings of the 2013 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2013, San Diego, CA, USA, 18–22 March 2013; pp. 303–305.
35. Mo, L.; Lu, X.; Feng, Z.; Hua, W. Online Human Daily Activity Recognition with Rechargeable Wearable Sensors. Sens. Mater. 2017, 29, 1353–1365.
36. Serdaroglu, K.; Uslu, G.; Baydere, S. Medication intake adherence with real time activity recognition on IoT. In Proceedings of the 2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Abu Dhabi, United Arab Emirates, 19–21 October 2015; pp. 230–237.
37. Khan, M.U.S.; Abbas, A.; Ali, M.; Jawad, M.; Khan, S.U.; Li, K.; Zomaya, A.Y. On the Correlation of Sensor Location and Human Activity Recognition in Body Area Networks (BANs). IEEE Syst. J. 2018, 12, 82–91.
38. Fortino, G.; Gravina, R.; Russo, W. Activity-aaService: Cloud-assisted, BSN-based system for physical activity monitoring. In Proceedings of the 2015 IEEE 19th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2015, Calabria, Italy, 6–8 May 2015; pp. 588–593.

39. Naya, F.; Ohmura, R.; Miyamae, M.; Noma, H.; Kogure, K. Wireless sensor network system for supporting nursing context-awareness. Int. J. Auton. Adapt. Commun. Syst. 2011, 4, 361–381.

40. Hao, J.; Bouzouane, A.; Bouchard, B.; Gaboury, S. Activity inference engine for real-time cognitive assistance in smart environments. J. Ambient Intell. Humaniz. Comput. 2018, 9, 679–698.

41. Settles, B. Active Learning Literature Survey; Technical Report; University of Wisconsin, Computer Science Department: Madison, WI, USA, 2009.

42. Zhu, X. Semi-Supervised Learning Literature Survey; Technical Report; University of Wisconsin: Madison, WI, USA, 2005.

43. Alwakeel, S.S.; Alhalabi, B.; Aggoune, H.; Alwakeel, M. A machine learning based WSN system for autism activity recognition. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA 2015, Miami, FL, USA, 9–11 December 2015; pp. 771–776.

44. Weinberger, K.Q.; Saul, L.K. Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res. 2009, 10, 207–244.

45. Domingos, P.; Hulten, G. Mining high-speed data streams. In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, USA, 20–23 August 2000; pp. 71–80.

46. Song, Y.; Liu, Z.; Wang, J. Human daily activity recognition based on online sequential extreme learning machine. In Proceedings of the World Congress on Intelligent Control and Automation (WCICA), Guilin, China, 12–15 June 2016; pp. 3226–3229.

47. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: Theory and applications. Neurocomputing 2006, 70, 489–501.

48. Palumbo, F.; Gallicchio, C.; Pucci, R.; Micheli, A. Human activity recognition using multisensor data fusion based on Reservoir Computing. J. Ambient Intell. Smart Environ. 2016, 8, 87–107.

49. Boukhechba, M.; Bouzouane, A.; Bouchard, B.; Gouin-Vallerand, C.; Giroux, S. Online recognition of people's activities from raw GPS data. In Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments, Corfu, Greece, 1–3 July 2015; pp. 1–8.

50. Abdallah, Z.S.; Gaber, M.M.; Srinivasan, B.; Krishnaswamy, S. Activity recognition with evolving data streams: A review. ACM Comput. Surv. (CSUR) 2018, 51, 1–36.

51. Krawczyk, B.; Minku, L.L.; Gama, J.; Stefanowski, J.; Woźniak, M. Ensemble learning for data stream analysis: A survey. Inf. Fusion 2017, 37, 132–156.

52. Stefanowski, J.; Brzezinski, D. Stream classification. Encycl. Mach. Learn. 2017, 1191–1199.

53. Ni, Q.; Patterson, T.; Cleland, I.; Nugent, C. Dynamic detection of window starting positions and its implementation within an activity recognition framework. J. Biomed. Inform. 2016, 62, 171–180.

54. Chetty, G.; White, M. Body sensor networks for human activity recognition. In Proceedings of the 3rd International Conference on Signal Processing and Integrated Networks, SPIN 2016, Noida, Delhi, 11–12 February 2016; pp. 660–665.

55. Amft, O.; Lombriser, C.; Stiefmeier, T.; Tröster, G. Recognition of User Activity Sequences Using Distributed Event Detection. Smart Sens. Context 2007, 126–141.

56. Lombriser, C.; Bharatula, N.B.; Roggen, D.; Tröster, G. On-body activity recognition in a dynamic sensor network. In Proceedings of the 2nd International Conference on Body Area Networks 2007, Florence, Italy, 11–13 June 2007.

57. Roggen, D.; Cuspinera, L.P.; Pombo, G.; Ali, F.; Nguyen-Dinh, L.V. Limited-Memory Warping LCSS for Real-Time Low-Power Pattern Recognition in Wireless Nodes. In Proceedings of the European Conference on Wireless Sensor Networks, Porto, Portugal, 9–11 February 2015; pp. 151–167.

58. Jiang, F.; He, S.; Cheng, P.; Chen, J. On optimal scheduling in wireless rechargeable sensor networks for stochastic event capture. In Proceedings of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS 2011, Valencia, Spain, 17–22 October 2011; pp. 69–74.

59. Aldeer, M.M.; Martin, R.P.; Howard, R.E. Tackling the Fidelity-Energy Trade-Off in Wireless Body Sensor Networks. In Proceedings of the 2017 IEEE 2nd International Conference on Connected Health: Applications, Systems and Engineering Technologies, CHASE 2017, Philadelphia, PA, USA, 17–19 July 2017; pp. 7–12.

60. Zang, W.; Li, Y. Motion aware transmission power control scheme in wireless body area network. In Proceedings of the IEEE Workshop on Signal Processing Systems, SiPS: Design and Implementation, Hangzhou, China, 14–16 October 2015.

61. Xiao, L.; Li, R.; Luo, J.; Xiao, Z. Energy-efficient recognition of human activity in body sensor networks via compressed classification. Int. J. Distrib. Sens. Netw. 2016, 12.

62. De Paola, A.; Ferraro, P.; Gaglio, S.; Lo Re, G.; Das, S.K. An Adaptive Bayesian System for Context-Aware Data Fusion in Smart Environments. IEEE Trans. Mob. Comput. 2017, 16, 1502–1515.

63. Farella, E.; Pieracci, A.; Benini, L.; Rocchi, L.; Acquaviva, A. Interfacing human and computer with wireless body area sensor networks: The WiMoCA solution. Multimed. Tools Appl. 2008, 38, 337–363.

64. Bellifemine, F.; Fortino, G. ASPINE: An agent-oriented design of SPINE. In Proceedings of the 10th Workshop on Objects and Agents, WOA'09, Parma, Italy, 9–10 July 2009; pp. 9–10.

65. Zappi, P.; Stiefmeier, T.; Farella, E.; Roggen, D.; Benini, L.; Tröster, G. Activity recognition from on-body sensors by classifier fusion: Sensor scalability and robustness. In Proceedings of the 2007 International Conference on Intelligent Sensors, Sensor Networks and Information Processing, ISSNIP, Melbourne, Australia, 3–6 December 2007; pp. 281–286.

66. Palumbo, F.; Barsocchi, P.; Gallicchio, C.; Chessa, S.; Micheli, A. Multisensor Data Fusion for Activity Recognition Based on Reservoir Computing. Commun. Comput. Inf. Sci. 2013, 386, 24–35.

67. Wang, L.; Gu, T.; Chen, H.; Tao, X.; Lu, J. Real-time activity recognition inwireless body sensor networks: From simple gestures to complex activities. In Proceedings of the 16th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, RTCSA 2010, Macau, China, 23–25 August 2010; pp. 43–52.

68. Atallah, L.; Lo, B.; Ali, R.; King, R.; Yang, G.Z. Real-time activity classification using ambient and wearable sensors. IEEE Trans. Inf. Technol. Biomed. 2009, 13, 1031–1039.

69. Fukushima, Y.; Miura, D.; Hamatani, T.; Yamaguchi, H.; Higashino, T. MicroDeep: In-network deep learning by micro-sensor coordination for pervasive computing. In Proceedings of the 2018 IEEE International Conference on Smart Computing, SMARTCOMP 2018, Taormina, Italy, 18–20 June 2018; pp. 163–170.

70. Bhaduri, K.; Wolff, R.; Giannella, C.; Kargupta, H. Distributed decision-tree induction in peer-to-peer systems. Stat. Anal. Data Mining: ASA Data Sci. J. 2008, 1, 85–103.

71. Navia-Vazquez, A.; Gutierrze-Gonzalez, D.; Parrado-Hernandez, E.; Navarro-Abellan, J.J. Distributed Support Vector Machines. IEEE Trans. Neural Netw. 2006, 17, 1091–1098.