Sensor-based Human Activity Recognition in a Multi-resident Smart Home Environment

DONG CHEN

A thesis submitted to Auckland University of Technology in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Primary Supervisor: Dr. Sira Yonchareon

Secondary Supervisor: Prof. Edmund Lai

July 2021

School of Engineering, Computer and Mathematical Sciences

Abstract

Human Activity Recognition (HAR) uses various types of sensors to collect data and recognise human motions. HAR can prevent potentially unsafe movements and behaviours such as falling or forgetting to take a medication. Many practical applications of HAR have emerged in recent years, such as ambient assisted living (AAL) systems and smart home monitoring. The vast majority of current behaviour recognition solutions in smart home environments are designed for single residents. However, in real life, living environments are occupied not only by the target individual, but also by their family and/or guests, which complicates the task of activity recognition. Therefore, multi-resident activity recognition is demanded in a smart home environment.

This research addresses three sub-problems of the multi-resident activity recognition problem: segmentation, classification, and online learning. To solve the segmentation problem, we first propose a novel hybrid fuzzy C-means segmentation method based on change point detection (CPD) for sensor events, which improves the performance of multi-resident activity recognition. Next, we propose a new Transformer with Bidirectional Gated Recurrent Unit (Bi-GRU) deep-learning method called TRANS-BiGRU, which efficiently learns and recognises the complex activities of multi residents. Finally, we propose a novel Locally-weighted Ensemble Detection-based Adaptive Random Forest Classifier (LED-ARF) for online analysis of multi-resident identification. After comprehensive experiments, we found that our proposed algorithm effectively solves the three problems of multi-resident action recognition.

Keywords: Multi-resident activity recognition, Activity sequence segmentation, Deep learning, Online learning

Table of Contents

List of Figures

List of Tables

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgments, nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature:

Date: 01/07/2021

Co-authored Work

All co-authors in the following table have approved these chapters for inclusion in Dong Chen's doctoral thesis.

Chapter	Author %
Chapter 2: A Survey on Non-intrusive Sensor-based Human Activity Recognition in a Multi-Resident Environment. Manuscript submitted to IEEE Internet of Things Journal.	DC = 80 SY = 10 EL = 5 JY = 5
Chapter 3: Hybrid Fuzzy C-means CPD-based Segmentation for Improving Sensor-based Multi-resident Activity Recognition. Manuscript published in IEEE Internet of Things Journal.	DC = 80 SY = 10 EL = 4 JY = 3 QZS = 3
Chapter 4: Transformer with Bidirectional GRU for Non-intrusive, Sensor-based Activity Recognition in a Multi-Resident Environment Manuscript submitted to IEEE Internet of Things Journal.	DC = 80 SY = 10 EL = 4 JY = 3 QZS = 3
Chapter 5: Locally-weighted Ensemble Detection-based Adaptive Random Forest Classifier for Sensor-based Online Activity Recognition for Multiple Residents Manuscript submitted to IEEE Internet of Things Journal.	DC = 80 SY = 10 EL = 4 JY = 3 QZS = 3
Dong Chen (DC), Sira Yongchareon (SY), Edmund Lai (EL), Jian Yu (JY), Quan Z Sheng (QZS)	

We, the undersigned, hereby agree to the percentages of participation to the chapters identified above:

Sira Yongchareon   Edmund Lai   Jian Yu   Quan Z Sheng   Dong Chen

Acknowledgment

I would like to express my gratitude to all those who helped me during the writing of this thesis. My deepest gratitude goes first and foremost to Dr. Sira Yongchareon, my supervisor, for his constant encouragement and guidance. He has walked me through all the stages of the writing of this thesis. Without his consistent and illuminating instruction, this thesis could not have reached its present form.

Second, I would like to express my heartfelt gratitude to Professor Edmund Lai for his insightful comments and encouragement. I am also greatly indebted to the professors and lecturers at the Department of Engineering, Computer, and Mathematical Sciences, who have instructed and helped me a lot in the past two years.

Last, my thanks would go to my beloved family for their loving considerations and great confidence in me all through these years. I also owe my sincere gratitude to my friends and my classmates who gave me their help and time in listening to me and helping me work out my problems during the challenging course of the thesis.

DONG CHEN

01/07/2021

Chapter 1 Introduction

In this chapter, we focus on the overall introduction of this thesis, and the chapter contains five sections. The first section introduces the research background of Human activity recognition (HAR) and motivation of studying multi-resident activity recognition. Related research questions are presented in second section. The third section introduces the contributions. The fifth section will lay out the structure of this thesis.

1.1 Background and Motivation

The global population is now entering the aging phase and the number of elderly people is rising rapidly in almost every country. By 2050, the number of people over 60 years old worldwide will exceed 2.1 billion [1]. As they age, many of them will likely suffer from geriatric diseases such as dementia or Alzheimer's, which decrease their ability to live independently [2]. The predicted shortage of caregivers and imperfection of available healthcare infrastructure have driven the establishment of assisted living systems that monitor the activities of daily living (ADLs) [3, 4].

HAR uses various types of sensors to collect data and recognise human motions. [5]. HAR can prevent potentially unsafe movements and behaviours such as falling or forgetting to take essential medication. Many practical applications of HAR have emerged in recent years, such as AAL systems and smart home monitoring [6].

Research interest in HAR (especially in smart home environments) has increased alongside the growth in elderly populations. Smart homes require the development of various types of sensors and communication technologies. Different sensor devices worn on residents or installed in smart environments collect the data of the residents and determine their movements through advanced recognition technology. This behavioural information is then sent to families or caregivers to improve the residents' care in-house [7].

Recognition techniques for multi-resident activity recognition are based on sensor-deployment strategies. From a macroscopic aspect, sensors can be divided into two types. The first (stationary) type is deployed in smart environments, usually by attachment to stationary objects. The second (nomadic) type are carried by humans, such as smartphones and other wearable devices such as wristbands and smartwatches. From a microscopic aspect, the sensors that discern human activities and anomalies in the smart environment are divisible into five types: vision-based sensors, wearable sensors, smartphone sensors, acoustic sensors, and ambient sensors [7].

Video surveillance is usually solved by installing a camera that captures the human-body movement data within a specific area. The captured movements are then recognised by image analysis technology [8]. However, this technique is disadvantaged by the high cost of cameras, video storage and analysis equipment, and by intrusion of personal privacy under constant video monitoring [9]. The other solution is wearable-based sensors for activity recognition, which reduce the equipment cost and lower the privacy intrusiveness. However, wearable sensors-based technology has other limitations. Some people, especially elderly individuals, are reluctant to wear the devices or sometimes forget to wear them. Therefore, we believe that interactive sensors such as motion sensors, temperature sensors, and humidity sensors that assist activity recognition will become a future trend [10].

Most of the research has focused on activity recognition in single-resident living conditions. Some simple activities involve a single action such as drinking water. However, even a single person performs more complex activities composed of many sub-activities in the home. For example, the activity "take medicine" can be regarded as a sequence of non-overlapping atomic (single) activities [11]: taking a cup, pouring water into the cup, picking up the medicine, and taking the medicine.

In addition to sequential activities, there are interleaving activities and concurrent activities. All three activity types must be detected by single-resident activity recognition [12]. Interleaving activity refers to activities that are shifted by a single person. For example, a resident might be shifting between cooking food and boiling water in the kitchen. Concurrent activity refers to more than one activity performed by a single person simultaneously (e.g., watching TV and drinking tea).

In real life, the living environment is usually occupied by the resident, their family and/or guests [13]. Multi-resident activity recognition is even more challenging, particularly when complex activities are being performed by two or more people. Such activities can be parallel or collaborative [7]. Naturally, when the number of residents increases in the living space, the complexity of activity recognition increases. Parallel activity refers to many activities performed by different people in the living space at the same time. For example, while one person is making tea in the kitchen, another resident might be making a phone call in the living room. Collaborative activity refers to synergic cooperation among multiple residents either performing the same activity (e.g., two people moving a dining-table) or working separately towards the same objective [7] (e.g., two people preparing dinner in the kitchen).

The above-mentioned activities are typical activities of daily life. Many studies have addressed sequential activity [3] and interleaving activity [14], but parallel and collaborative activities (especially complex activities in multi-resident situations) have been relatively neglected.

1.2 Problems/ Research Questions

With the rapid development of sensor technology and artificial intelligence, the HAR field has entered a new phase. However, the recognition of multi-resident activities requires further study. This thesis separately investigates three questions of multi-resident activity recognition, which are elaborated below.

Research Question (1): By what means a sensor location feature can be incorporated into sensor event segmentation, and to what degree can it improve the performance of multi-resident activity recognition?

It has been proven that sensor event segmentation improves the performance of HAR. In most current studies, sensor event segmentation is divided into static time-based, static sensor-based and sensor-time correlation methods [15]. However, these methods have limited ability to recognise the activities in multi-resident settings. Static time-based and static sensor-based methods obviously cannot distinguish between sensor events in multi-resident situations. Although dynamic sensor-time correlation methods outperform the static models in this scenario, the position features would improve the partitioning [16]. In this research, we add location as another parameter to the dynamic segmentation of sensor events to improve the recognition accuracy. To evaluate our proposed solution, we aim to compare the results obtained before and after segmentation. We will also compare our work with other segmentation approaches.

Research Question (2): By what means can deep learning methods be used for classification, and to what degree can deep learning improve multi-resident activity recognition performance over traditional machine learning classification techniques?

Deep learning techniques have tremendously progressed in recent years and have been applied to various fields, such as speech and image recognition with excellent results [17]. Unlike traditional machine learning methods, deep learning can automatically perform tedious feature selection, thus improving the accuracy of classification. A number of deep learning classification models have been applied to the sensor-based HAR domain and have outperformed traditional machine learning methods [18]. Therefore, we will apply deep learning approaches to complex multi-resident activity recognition. The activity recognition performances of different deep-learning methods will be compared with those of traditional machine learning methods.

Research Question (3): By what means can online learning methods be used for classification and implemented for activity recognition, and which model achieves the best performance among the online learning classification models?

Almost all current HAR studies of multi-residents are performed offline. The offline approach generally consists of five steps: data collection, data preprocessing, segmentation, feature extraction, and classification. Classification models are first trained offline and later on online events coming from different sensors. However, the offline classification method is ineffective for long-term monitoring in multi-resident environments because the movement habits differ among the residents and new guests may arrive in the home. As the initially trained classifier cannot identify these changes and new residents [19], the classification model must be constantly updated to adapt to newly generated changes and thus identify different inhabitants and actions. In this research, we will evaluate various online learning models on different multi-resident datasets.

1.3 Contributions

Multi-resident activity recognition is still in its infancy and many challenges remain unsolved. This thesis tackles three of these challenges: segmentation, classification, and online recognition. It also provides a comprehensive review of this field. Specifically, our thesis makes the following contributions:

We conduct an extensive review and a comprehensive analysis of modern technologies and related HAR techniques that address the challenges of monitoring multi-resident environments using non-intrusive sensors. First, we extensively review and classify research publications on multi-resident HAR. We then categorize and compare the advantages and disadvantages of the existing activity segmentation techniques. Finally, we summarise and compare the existing classification models for multi-resident HAR, including traditional machine learning methods and the state-of-the-art deep learning models.

To answer the first research question, we propose a novel hybrid fuzzy C-means CPD-based segmentation method that clusters the sensors using the fuzzy C-means clustering method [20]. To improve the performance of multi-resident recognition, we segment each activity of the diverse residents using the Change Point Detection (CPD) method. We also evaluate our data-driven segmentation technique and compare its performance with those of baseline data-driven segmentation techniques. Our proposed method significantly improved the performances of all baseline and state-of-the-art classification models used in our study.

To answer the second research question, we proposed a Transformer with a Bidirectional GRU deep learning method (TRANS-BiGRU), which efficiently learns and recognizes the complex activities of multi residents. We evaluate our proposed model by comparing its performance with those of state-of-the-art models, traditional ML methods, and other DL models proposed for multi-resident HAR. Our TRANS-BiGRU achieved the highest performance amongst the existing models.

To answer the third research question, we proposed a novel Locally-weighted Ensemble Detection-based Adaptive Random Forest Classifier (LED-ARF) classifier and applied it to online analysis of multi-resident identification. We comprehensively compared the performances of eight popular online-learning classification algorithms. In the experimental evaluations, the LED-ARF model achieved the highest performance on online multi-resident HAR.

1.4 Thesis Structure

This thesis consists of six chapters.

Chapter 2 overviews non-intrusive sensor based HAR in multi-resident environments. Section 2.2 discusses the challenges and general framework of non-intrusive sensor-based multi-resident HAR. Section 2.3 presents activity segmentation methods in HAR. Section 2.4 reviews and discusses the different classification techniques of HAR in multi-resident settings. Section 2.5 introduces non-intrusive sensors, public datasets, and evaluation metrics. Section 2.6 summarizes the chapter.

Chapter 3 tackles the segmentation problem to improve the performance of multi-resident activity recognition. Section 3.2 introduces related works in sensor event segmentation and HAR.

Section 3.3 introduces our Hybrid Fuzzy C-means CPD-based Segmentation algorithm. Section 3.4 presents the experiments and their results. Section 3.5 concludes our findings and suggests ideas for future work.

Chapter 4 studies the use of deep learning models to recognize the complex activities of multiple residents. Section 4.2 reviews the related works on multi-residents and various classification models. Section 4.3 introduces our TRANS-BiGRU model. Section 4.4 presents the experiments and their results, and Section 4.5 concludes the chapter.

Chapter 5 investigates real-time recognition of multi-resident activities using online learning classification models. Section 5.2 reviews the work related to online learning in HAR. Section 5.3 introduces the proposed LED-ARF for sensor-based online multi-resident activity recognition. Section 5.4 discusses the experimental results, and Section 5.5 concludes the chapter.

Chapter 6 presents our conclusions (Sections 6.1) and proposes ideas for future work (Section 6.2).

Chapter 2 A Survey on Non-intrusive Sensor-based Human Activity Recognition in a Multi-Resident Environment

Non-intrusive sensor-based Human Activity Recognition (HAR) has become a popular research field in smart environments such as classrooms, office, and homes with a primary purpose to recognize human activities from the multifarious of low-level sensor readings. Numerous studies have made tremendous progress in a single-resident situation during the past several years. However, there are usually multiple residents present in the living environment, such as their family or friends. Therefore, HAR under the circumstance of multi-resident is more challenging and intricate due to complexity in activities' identification and recognition. The recent advancement of sensor technologies and machine learning makes it possible to collect and analyze multi-level information of activities, thus achieving outstanding performance. With recent concerns in privacy and practicality, several non-intrusive sensor-based approaches have been proposed for multi-resident HAR. This chapter provides a comprehensive survey of multi-resident HAR in non-intrusive sensor-based environments. We extensively reviewed and analyzed existing work related to activity segmentation techniques and classification approaches with highlights of their advantages and limitations. Furthermore, we discuss evaluation methods and present publicly available datasets for non-intrusive sensor-based multi-resident HAR. Finally, we summarize research challenges and open issues that require further study in the field.

Chapter 2 has been submitted to IEEE Internet of Things Journal

2.1 Introduction

Human Activity Recognition (HAR) is a challenging research area that deal with human motions and activities monitoring and analysis based on data that can be obtained from different types of instruments or sensors [21]. Activity recognition can be used for recognition of daily activities but also potentially dangerous and accidental activities, such as falls [22]. There are several real-world applications of HAR including healthcare home monitoring and ambient assisted living systems [23-26]. HAR became a popular field of study due to an increasing number of elderly people with disabilities, especially those who require personalized, monitored care. The advance of communication technology and sensor has led to the realization of smart homes with various sensing and monitoring technologies that could be used to collect various physical information to learn and recognize the activities of each person. The activities learned can be further used to infer personal behavior for their family members or caregivers to understand their living situation to provide better care to them [6]. Current research has addressed many action recognition problems in single-resident environments, [27-34], but in real-world situations, residential environments often have multiple residents present, such as their relatives or friends. Therefore, their activities will also be more complex, bringing in more technical challenges beyond traditional problems in a single resident environment [35]. Research in multi-resident HAR is still in infancy because several challenges for a single resident environment still need to be resolved before those for multi-resident can be addressed [7, 36, 37].

HAR in a multi-resident environment has acquired increasing attention during the last several years, and recently with the speedy advancement of sensors and machine learning technologies, it becomes easier to collect and analyze multi-level information of activities, thus achieving outstanding performance [35, 38-42]. We can group and analyze existing techniques for HAR based on sensor deployment strategies [43]. From a macroscopic aspect, sensors can be divided into two types: stationary and nomadic. First type includes environmental (or ambient) sensors, usually attaching to the object and making them mostly stationary while the latter includes smartphones and other wearable devices such as wristbands and smartwatches [44]. From a technology aspect, there are five groups of sensors: vision-based sensing, wearable sensing, smartphone sensing, acoustic sensing, and ambient sensing [5]. Most of the existing solutions for

HAR are based on video surveillance which usually requires installing a camera to record human body movement within a specific area and rely on image recognition techniques to recognize human movements and activities [8]. However, these solutions have the following disadvantages: (1) Surveillance equipment such as cameras and storage is expensive, and (2) Personal privacy concerns [9]. Besides, most wearable-based sensors have critical characteristics of low cost and low privacy intrusiveness, which can replace video surveillance systems. However, this type of technology still has some drawbacks such as the need to always carry or attach to a body. Some people, especially elderlies, may be reluctant to wear a device, or sometimes they forget to wear it when needed [7]. As we can see in Figure 2-1, which illustrates the number of proposed solutions based on different sensor types, there is a growing number of research works use ambient and acoustic sensors along with line with smart phones. Based on this trend, we believe that non-intrusive sensing technology is one of the promising areas of research in HAR due to its being privacy-aware, non-wearable, cheaper, and easy to install while providing sufficient information for HAR [9]. In this chapter, we scope our survey on non-intrusive ambient sensors and related multi-resident HAR techniques.

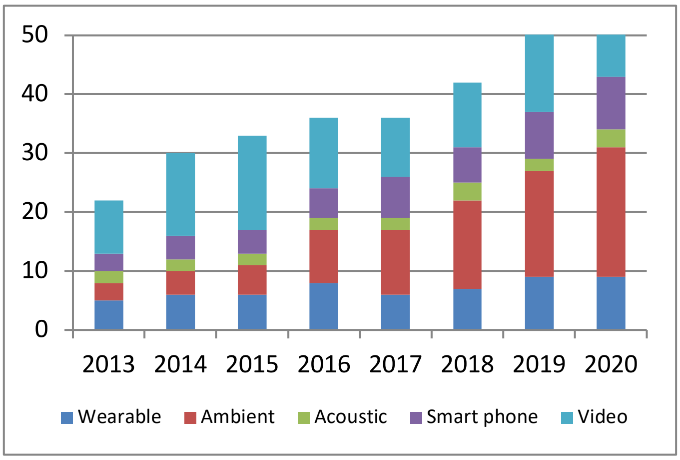

Figure 2-1 Classification of HAR work based on sensor types

Among the existing surveys on HAR, Chen et al. [45] provided an in-depth review of sensor-based HAR focusing on deep learning techniques. Raval et al. [46] analyzed HAR based on surveillance videos. Ramamurthy et al. [47] investigated recent trends in machine learning for HAR. These surveys are based on single-person activity recognition while there is one survey work related to multi-resident HAR which was conducted by Benmansour et al. [44] in 2015. However, this study has some limitations, including the absence of activity segmentation techniques and state-of-the-art machine learning methods, and advanced deep learning approaches.

With the recent momentum and advances of deep learning, there have been more significant achievements for the research in the multi-resident HAR domain and hence a further review and analysis are needed to help both academic and industry communities understand the latest developments and trends. In our survey, we conducted an extensive review and provide a comprehensive analysis of modern technologies and related HAR techniques to address challenges that exist in a multi-resident environment using non-intrusive sensors. Our main contributions are as follows.

- We extensively review and classify research publications in the field of multi-resident HAR.
- We categorize and compare the advantages and disadvantages of the existing activity segmentation techniques.
- We summarize and compare the existing classification models for multi-resident HAR, including the state-of-the-art machine learning approaches and advanced deep learning models.
- We highlight key limitations and proposes ideas for future work.

The paper is organized as follows. Section 2.2 introduces the fundamentals of multi-resident HAR. Section 2.3 discusses activity segmentation techniques. Section 2.4 discusses various classification models proposed in a multi-resident environment and evaluation matric commonly used in this field. Section 2.5 discusses different kinds of non-intrusive sensors and publicly available datasets for benchmarking. Section 2.6 presents limitations and summarizes key challenges and open issues that require further study. Finally, Section 2.7 concludes the survey.

2.2 An Overview of Multi-Resident HAR

In this section, we first discuss existing challenges in multi-resident HAR. Then we present a general framework for non-intrusive multi-resident HAR.

2.2.1 Challenges in Multi-Resident HAR

A living environment often has more than one inhabitant in the real world, and human activities are also more intricate [13]. Thus, multi-resident HAR is more challenging, particularly for recognizing complex activities involving two or more people who may perform parallel activities and collaborative activities [7].

With the increasing number of residents in a living space, the complexity of activity recognition increases. A parallel activity contains two or more distinct activities performed by different people at the same time. For example, one person is making tea while another person is making a phone call. On the other hand, a collaborative activity is a kind of complex activity that is performed by two or more persons in a synergic manner for one objective, e.g., two people moving a dining table or preparing dinner together, [7]. These two types of activities are quite common in our daily life. Many studies have addressed problems related to sequential activity [48] and interleaving activity [14], yet very few studies address parallel activities and collaborative activities. Interaction is the main challenge in recognizing these two activity types [44]. Different residents may trigger different sensors in different places or even the same sensor in the same place. Therefore, we should identify the state of sensor event triggering and determine who triggered the sensor while identifying activities. Especially, information collected by non-intrusive sensors is more coarse-grained and not easily used for machine learning to learn and distinguish such two kinds of activities performed by multiple residents [7].

2.2.2 A General Framework of Non-Intrusive Sensor-based Multi-Resident HAR

Advances in sensor technology have led to an increasing number of applications for HAR. More and more scholars apply non-intrusive sensors for activity recognition since they can protect residents' privacy [9, 49]. Data-driven-based non-intrusive sensor-based multi-resident HAR usually shares a similar framework, as shown in Figure 2-2, several steps are involved in the multi-resident HAR process, from human behaviour data collection based on diversiform sensors to final confirmation of performed activities. A general structure of non-intrusive sensor-based multi-resident HAR can be divided as follows: (1) ***data collection***: different types of non-intrusive ambient sensors can be installed in a multi-residential home environment to collect sensor trigger information. These ambient sensors may include door sensors, Passive Infrared (PIR) sensors, light sensors, humidity sensors, and temperature sensors. (2) ***data preprocessing and segmentation***: preprocessing raw sensor data for eliminating redundancy and data noise, handing incomplete data, and conducting data normalization and aggregation. Segmentation is the splitting of a sequence of sensor events into multiple overlapping or non-overlapping segments. A superior segmentation algorithm can largely improve the performance of activity recognition [50]. (3) ***activity classification***: identifying activities using various classification models. Classification approaches could be data-driven or knowledge-driven. In this chapter, we mainly focus on data-driven classification techniques, mainly traditional machine learning (ML) classification algorithms, such as Hidden Markov model (HMM), Decision Tree (DT) and Deep learning (DL) algorithms, such as CNN, DNN, LSTM, and GRU. (4) ***activity inference and evaluation***: evaluating classification models and recognizing multi-resident activities [51]. It is standard that the recognition performance should be measured based on accuracy, precision, recall, and F-measure.

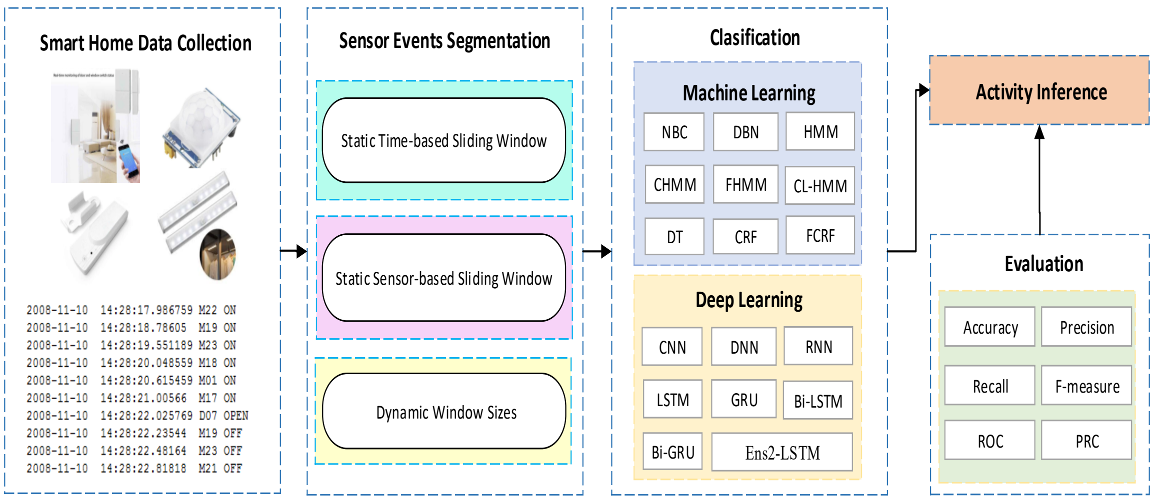

Figure 2-2 General framework of non-intrusive sensor-based multi-resident HAR

2.3 Activity Segmentation

Activity segmentation is partitioning sensor events into a discrete chunk, aims at improving the classification performance of HAR [21]. As persons can perform parallel activities or interleaved activities for maximum efficiency, these activities may not have a clear boundary [52]. Therefore, continuous sensor events need to be partitioned into smaller sections to help a classifier learn and distinguish different activities. As various approaches have been proposed to address challenges in sensor event segmentation for HAR, in this section, we provide and discuss their taxonomy.

Sliding window is one of the most used methods to segment sensor events, and many studies have adopted this technique. [35, 53, 54]. The most typical Sliding window segmentation is to partition the entire sensor sequence into different groups using the same time interval, as shown in Figure 2-3B [10]. Timestamps play a key role in segmenting the sensor events and generating activity segments. Second way of segmentation is to divide the whole sequence using the same number of sensor events as a measure, as illustrated in Figure 2-3C.

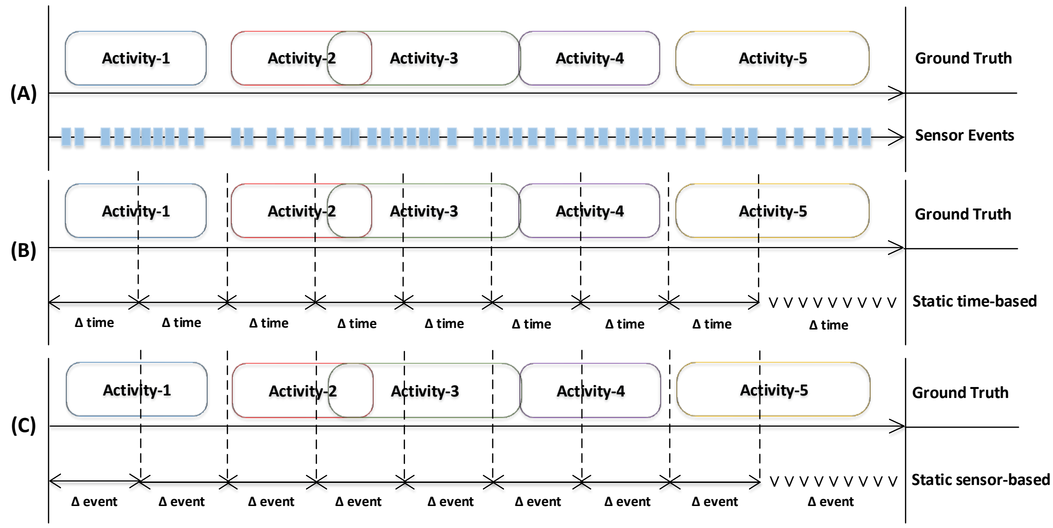

Figure 2-3 Sliding window segmentation methods

1) Static Time-based Sliding Window

A static time-based window method relies on timestamps of sensor events to indicate the boundary of activity and generates subsequence of sensor events [54]. In this method, each window contains a fixed time interval but the optimal time interval is difficult to be decided, as a result significantly impacting the performance of classification [55]. If the time interval is in a too-small range, a window may not contain enough events of one activity. On the other hand, multiple activities may be embedded into one chunk if a window is too large.

Wang et al. [56] proposed a static time-based window for segmenting a sensor events sequence. Sensor events were segmented using the same time interval. A wireless body network is used to collect sensor data at a low cost for the processing which is suitable for situations with regular and periodic sensor events.

Chen et al. [57] evaluated the effect of static time-based segmentation on multi-resident activity recognition for different parameters ranging from a few seconds to several hundred seconds, and the experimental results show a significant improvement in multi-resident activity recognition for segmentation times of 60 seconds and 90 seconds. However, even with the great improvement achieved in the multi-resident recognition performance under the parameters of a time interval of 60 and 90 seconds, it is still difficult for us to determine what the optimal time interval is, which is the biggest drawback of the static time-based segmentation method.

Nevertheless, the static-time-based window approach does not work well with ambient sensors where they can be installed in multiple areas for recording different activities that may be performed by multiple persons at arbitrary time lengths. For example, taking a medicine may finish in less than a minute, sweeping the floor may take several minutes, and sleeping can last for several hours. A too-long window selection may not be able to recognize short-term activities while a too-short window could cause a single activity to be divided into different partition groups.

2) Static Sensor-based Sliding Window

A segmentation method is used to segments a long event sequence into multiple fixed-length series of events. This approach is commonly used when sensor data are transmitted and recorded asynchronously [58]. The number of sensor events is a key parameter in this approach. Although a window's size is fixed based on the number of sensor events, the time durations may because, in high-density activities, a sensor can be triggered frequently. In contrast, during a silent period, the sensor may not be triggered at all or just only once for a long period, thus having very sparse sensor events. This segmentation approach has several inherent shortcomings [59].

For example, Figure 2-3C illustrates the sequence according to the endless constant number of sensor events, and the size of the window does not synchronize well with the occurrence of an activity. The last sensor event in the subsequence is likely to have a close relationship with the first sensor event in the next sequence. However, the static sensor-based segmentation approach may also divide an action into two different groups, which reduces the recognition performance. [60]. In a multi-resident environment, various residents perform different activities, and their sensor events can be put into a single window, thus affecting the classification performance [50].

Singla et al. [61] proposed a static sensor-based sliding window method to improve the classification performance with the assumption that a set of sensor events in a single-window represents only an activity. However, this approach is not good at recognizing short-term activities because the optimal static window parameters are difficult to determine. For example, one large sensor window is suitable to recognize long-term activities, but they will suffer to recognize short-term activities. Similarly, a small partition window may not cope well with longstanding activities.

Like the static time-based method, the static sensor-based method faces challenges in determining the optimal number of sensor events concerning all different types of activities. For example, it may take only a few seconds to leave home with only a few other sensors triggered, while many people may trigger sensors in the kitchen while cooking repeatedly [21, 62].

3) Dynamic Window Sizes

Both static time and static sensor-based segmentation methods are flawed because the actual activity of residents is not constant [15]. For instance, sensor events related to action can be divided into various windows, or a window may include multiple activities. Furthermore, partitioning sensor events are even more challenging when concurrent activities (performed by multiple people) occur simultaneously. The static window methods utilize a fixed-time interval or a quantifiable number of sensor events for segmenting the sequence and the most significant challenge with these approaches is finding the optimal parameters. In contrast, a dynamic window size method aims to automatically find the suitable single window size from sensor events [54].

Wan et al. [15] proposed a dynamic segmentation method for recognizing single resident activities by using sensor correlation and time correlation. Person product-moment correlation (PMC) was used to calculate the dependency between any two sensors. For example, sensor A and sensor B is absolute positive correlated if the value is +1, and 0 means no relationship between sensor A and B, the value of -1 indicates total negative correlation between the sensors. The PMC value between two sensors is positively reflected in that they are adjacent to each other in geography and more likely to trigger the sensors together or in a sequential way. In their work, the threshold value of PMC is at 0.5, and it can be considered that sensor S_i and S_{i-1} are correlated if the PMC value between them is greater than 0.5 or smaller than -0.5, which can be calculated by Eq. (2.1):

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\big((X-\mu_X)(Y-\mu_Y)\big)}{\sigma_X \sigma_Y} \tag{2.1}$$

The time correlation method is applied to evaluate whether two sensor events at times T_i and T_{i-1} are correlated or not. The first timestamp is expressed as T_{first} and the last timestamp is indicated as T_{last}. By calculating T_{first} and T_{last} based on Equations (2.2) and (2.3), respectively, sensor events can be segmented accordingly. However, different threshold values are set due to various activities having a great difference in their durations.

$$T\frac{first}{cor}(T_i, T_{i-1}) = \frac{Time_{Distance}}{Maximum_{Time_{Inteval}}} = \frac{T_i - T_{i-1}}{T_{max}} \tag{2.2}$$

$$T\frac{last}{cor}(T_i, T_{i-1}) = \frac{Time_{Distance}}{Maximum_{Time_{Inteval}}} = \frac{T_i - T_{i-1}}{T_{max}} \tag{2.3}$$

Noor et al. [54] presented a segmentation approach based on a gordian technique where the window's size can be adjusted appropriately according to the probability of transmitted signals to acquire the most efficient segmented sensor events. Moreover, an activity transition structure of AR is proposed to evaluate transition activities and improve the overall performance. The proposed adaptive sliding window approach generates a new window size if a transitional activity is found. The proposed transitional activity classifier first evaluates the extracted features and then computes the probability density function (PDF) of each activity to decide an optimal sliding window size, which can be calculated by Eq. (2.4).

$$\mathrm{p}(\mathbb{x}|A_j) \propto \mathrm{p}(\mathbb{x}, \mu_j, \Sigma_j) = \frac{1}{2\pi^{n/2}\left|\Sigma_j\right|^{1/2}} e^{-\frac{1}{2}(\mathbb{x}-\mu_j)^T \Sigma^{-1}(\mathbb{x}-\mu_j)} \tag{2.4}$$

where Σ_j is the covariance matrix related to the extracted features from the windows, and μ_j is the mean matrix. The parameters Σ_j and μ_j are estimated by Eq. (2.5) and (2.6).

$$\Sigma_j = \frac{1}{N_j} \sum_{x \epsilon A_j} x x^T - \mu_j {\mu_j}^T \tag{2.5}$$

$$\mu_j = \frac{1}{N_j} \sum_{x \epsilon A_j} x \tag{2.6}$$

where N_j is the total quantity of observation sequence in an activity A_j. Based on this, the optimal window size can be defined to improve the classification accuracy. Their results show that the proposed method achieved an overall accuracy of 96%, although some activity accuracies such as standing and sitting are slightly decreased by 0.7% and 1.3%, respectively.

Aminikhanghahi et al. [63] introduced an online activity segmentation method based on change-point detection (CPD) using various unscripted data, so-called SEParation change point detection (SEP), which is an unsupervised online detection method for adjacent sensor events. By calculating a conversion value between two sequential sensor events, it is possible to determine whether the activity's boundary is reached and if so, then the method cuts the sequence into the subsequence. A separation distance S is used to calculate the probability distribution ratio and we use the following Eq. (2.7) to calculate the S between two consecutive change point events.

$$S = \mathrm{Max}_i \left(1 - \frac{f_{t-1}^{i}\,(x)}{f_t^{i}\,(x)}\right) \tag{2.7}$$

where $f_{t-1}^{i}\,(x)$ and $f_t^{i}\,(x)$ are the estimated probability densities of the previous and current views, each with a length m. A variable i iterates over all data points in the change point's view. Besides, A separation distance is calculated by using a Gaussian kernel function $g_i(x)$ to estimate two probability densities, and a change point score based on Eq. (2.8).

$$\widehat{SEP} = Max\left(0, \left(1 - \frac{1}{m}\sum_{i=1}^{m} g_i(x)\right)\right) \tag{2.8}$$

A larger SEP value means a higher probability of change point. If their SEP scores are below the specified threshold, all candidate points are rejected. The local peak score is the only parameter we consider reducing false alarms and avoid repetitive change points.

Recently, Chen et al. [57] proposed a hybrid fuzzy C-means CPD-based segmentation method for improving HAR performance in the multi-resident smart home environments. They first applied fuzzy C-means to divide the whole sensor events into different subgroups based on sensor locations. Then they adopted a CPD-based segmentation method [63] for further segmenting sensor events. Similarly, various other activity segmentation methods have been proposed. Cho et al. [64] proposed a contextual relationship-based segmentation approach using a long short-term memory (LSTM) to segment the sensor events by capturing the sequence's intrinsic contextual relationships. Their LSTM model is customized into three layers with equal size, which addresses the bias problem by detecting advisable time intervals. Their input data is augmented with other status information to solve sensor events problems using the optimal parameters. Then they proposed a time interval evaluation algorithm to inspect the second layer's boundaries to confirm the final borders. This study sufficiently improves the performance of both single resident and multi-resident HAR. Wang et al. [65] proposed a three-layer segmentation method based on combined association rule mining and Support Vector Machine (SVM). The first layer in their architecture employs an SVM model to recognize residents' activities. Their second layer determines an activity's boundary, and the third layer verifies AR’s boundary using association mining rules to reduce the recognition errors. However, whether this method is applicable to the identification of multiple residents needs to be verified.

2.4 Activity Classification Models

In this section, we review and discuss different classification techniques used for HAR in a multi-resident setting. We discuss existing traditional machine learning models followed by deep learning models.

2.4.1 Traditional Machine Learning Models

Traditional machine learning models are the most popular data-driven methods in general [48] as they can be used to explain the interrelation between different random variables and probability distributions and incorporate them into the event model. The models give the probability distribution as a solution [66].

1) Naive Bayes Classifier (NB)

Naive Bayesian classifier (NB) is one of the most commonly used and simplest probabilistic models for HAR [9]. It depends on the Bayesian theory to establish decision boundaries using all assumptions that input the features independently which makes the classification easy to handle. Observations and labels in joint probability can be decomposed as shown in Eq. (2.9).

$$p\left(y_{j,t}|X_t\right) \propto p(y_{j,t}) \prod_{i=1}^{N} p\left(x_{i,t}|y_{j,t}\right) \tag{2.9}$$

where $p\left(y_{j,t}\right)$ is an activity-based prior probability on resident j. We assume that all the input data are independent, and then we can directly calculate the conditional probability of the labeled data $(y_{j,t}|X_t)$ [9].

In the settings, set $\mathrm{X} = \{\mathrm{X}_1 . \mathrm{X}_2 \cdots \mathrm{X}_\mathrm{t}\}$ can be expressed for sensors data and y represents different activities, as shown in Figure 2-4.

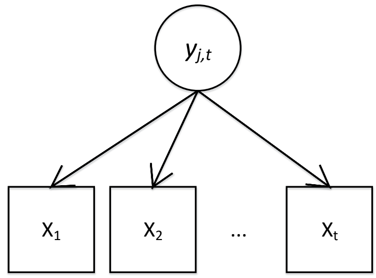

Figure 2-4 Naïve Bayes

Yin et al. [9] proposed a Naive Bayes (NB) model for multi-resident's tracking and activities recognition. Various non-intrusive ambient sensors such as power sensors, temperature sensors, and motion sensors were installed in the smart environment. The NB model achieves the best performance with 73.01% of accuracy. However, the basic NB model is not suitable for application to time series data. Therefore, Dynamic Bayesian Networks (DBNs), which is an upgrade of NB, have been proposed to solve the limitation [44].

2) **Dynamic Bayesian Network (DBN)**

Dynamic Bayesian Network (DBN) is an upgrade version of Bayesian Network, also known as Probabilistic Network or Belief Network. DBN associates different variables with adjacent temporal steps (so-called a Bayesian network of "two-time slices" at any point in time). DBN is also proven to produce Kalman Filters and Hidden Markov Models equivalent solutions [12]. Figure 2-5 shows a general structure of the Dynamic Bayesian network.

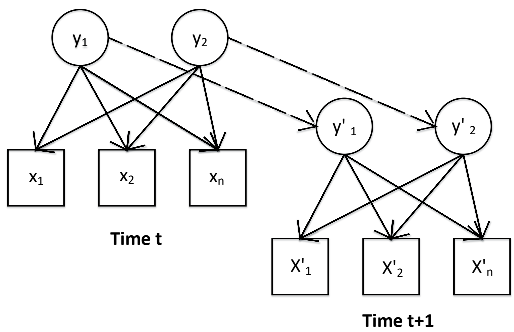

Figure 2-5 The structure of Dynamic Bayesian network

Alam et al. [12] proposed a Hierarchical Dynamic Bayesian Network (HDBN) for recognizing multi-resident activities. The CACE and CASAS datasets were used to evaluate the HDBN model, and the results show that the coupled HDBN model achieves 95% of accuracy in recognizing multi-resident activities. The proposed method outperformed other models such as HMM, FCRF, and CHMM.

3) Decision Trees (DT)

A decision tree (DT) is based on a generated top-down structure where each event or decision can lead to two or more events, leading to various outcomes, as shown in Figure 2-6. A classification decision tree consists of some nodes representing branches and features and defines the features' values. A class label can be reprsented by each leaf node. [6]. A decision tree is also used for regression when the output data is continuous

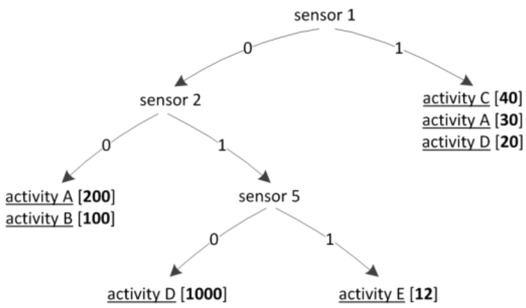

Figure 2-6 Example of decision tree [67]

Prossegger et al. [67] proposed a DT model to recognize daily life activities in a multi-resident environment. A DT algorithm named E-ID5R was proposed, and leaf nodes were added to the generated tree. ARAS datasets were applied to evaluate the performance, and the results show that E-ID5R achieved excellent performance in House B, with an accuracy rate of 84.45%, only 49.28% in House A.

4) Hidden Markov Model (HMM)

Hidden Markov Model (HMM) is a probabilistic model that describes unobservable random sequences from a hidden Markov chain and random observation sequences based on various states [6]. From a given input sequence $(x_1, x_2, \cdots x_T)$, a hidden state sequence $(y_1, y_2, \cdots y_T)$ can be estimated, as illustrated in Figure 2-7 and related joint probability of $p(x, y)$ of the hidden states and the observations can be expressed as follow Eq. (2.10).

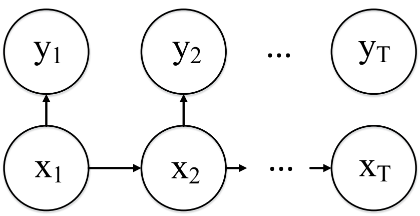

Figure 2-7 Hidden Markov Model

$$p(x,y) = \prod_{t-1}^{T} p(y_t|y_{t-1})p(x_t|y_t) \tag{2.10}$$

HMM is one of the most popular probabilistic models deployed for HAR. For example, as shown in Figure 2-8, the activities can be defined as hidden states, while sensor event numbers (in the rectangles) mean observation sequence based on data collected from different sensors. The horizontal arrow represents a transition probability and the downward arrow represents the probability of emission for a corresponding observed state.

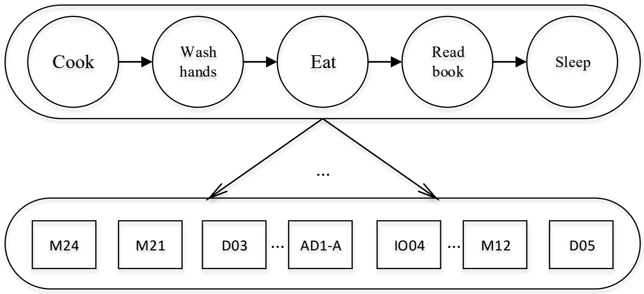

Figure 2-8 Activity modeling using HMM.

In general, we can use an HMM model to represent the action when an action is sequential. However, HMM may not be best suitable for interleaved or collaborative activities because the interlacement of activity may be ignored. In some conditions, some sub-activities cannot be observed directly when the complex activity is disintegrated [68]. For example, an activity “tidy the room” can be divided into two activities, “mop the floor” and “take out the trash,” and each of them also includes some sub-activities. Therefore, different activities trained using HMMs individually could be integrated to establish a global HMM to form a hierarchical HMM. Some variants of HMMs, such as CHMM, FHMM, CL-HMM have been proposed to recognize multi-resident activities to address these limitations [44].

5) Coupled-Hidden Markov Models (CHMM)

Coupled-Hidden Markov Models (CHMM) can be described as a group of HMMs interacting in different manners where the previous states in one HMM affect the current state of another HMM. [44]. Figure 2-9 illustrates a general structure of fully connected chains of two HMMs.

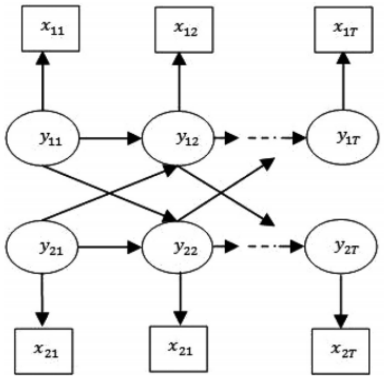

Figure 2-9 CHMM model [44]

Roy et al. [69] proposed a CHMM to recognize daily activities based on combined ambient sensor data and smartphone data. In this research, motion sensors are used to confirm the residents' location in the room. Then this location information is combined with smartphone data to recognize the residents' activity more accurately. They used CHMM because there are two different acceleration data streams connected with varying sequences of observations. The concurrent activities of occupants are imposed spatiotemporal constraints by using ambient motion sensor data. Two separate activity datasets are used in the experiments and the results show that their model can increases complex activity recognition accuracy by more than 30% compared with only smartphone-based solutions.

6) Factorial Hidden Markov Model (FHMM)

Factorial Hidden Markov Model (FHMM) has different individual Markov chains in different states and at each time stage, observation states are related to all the corresponding states [5]. The observation states of the standard variable bring more complicacy, although these chains are prior independent. Figure 2-10 shows the general structure of the FHMM.

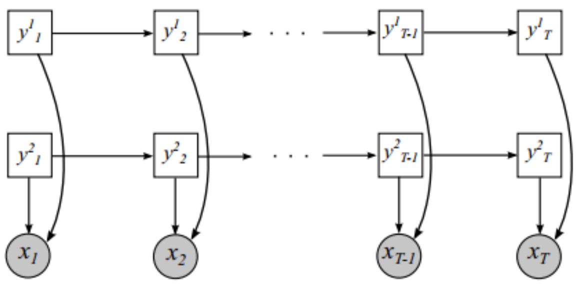

Figure 2-10 FHMM model [44]

Alemdar et al. [5] used an FHMM to model hidden observations using multiple activity sequence chains. They focus on concurrent activities based on data collected from ambient sensors in a multi-resident environment. The experimental results show that the proposed FHMM model achieved 78.7% of f-measure, superior to HMM and CRF.

7) Hidden Markov model-based combined label (CL-HMM)

Each occupant has its sequence of activities in a smart environment. Therefore, as a part of activities, each occupant has an observation state at each time point. Benmansour et al. [7] utilized various HMM methods to recognize parallel activities and collaborative activities in a multi-resident environment They proposed a CL-HMM model to combine all the resident activity labels and generate a new activity label. If there are *k* occupants in the smart environment, the vector of length for observation states is *k* and the same length for activities at each stage. CL-HMM is used to transform and integrate the activities $(a_{1j,} a_{2j,} \cdots a_{kj,})$ and the relevant data $(d_{1j,} d_{2j,} \cdots d_{kj,})$ of individual occupants into a single combined activity label (L) and a single observation (d). They compared CL-HMM with Coupled HMM (CHMM) and Parallel HMM (PHMM), and the results show that CL-HMM achieved an accuracy of 91.91% and outperforms the other models (best for collaborative activities). Mohamed et al. [70] evaluated the random forest model on the Label Combination method in a smart home environment. The experiment has been applied to ADL multi-resident from CASAS and achieved an average accuracy of 94.6%.

2.4.2 Deep Learning Models

In recent years, Deep Learning (DL) methods have been proven with outstanding performance results in multiple application domains including image recognition, speech recognition, and HAR due to computing power improvement [71, 72]. Compared with traditional ML models for HAR, DL concentrates on processing high-order essential features through a sophisticated neural network, rather than manual feature engineering. Also, deep neural network structures are considered faster in executing unsupervised learning and online learning [41]. Quite several deep learning models have been proposed for a single-resident HAR. However, it is still in the infancy stage for multi-resident HAR.

1) Long Short-term Memory (LSTM)

Long short-term memory (LSTM) is a particular Recurrent Neural Network (RNN) designed to remember meaningful information for a long period. It can efficiently learn long dependencies and sequential data [72]. The LSTM's structure is shown in Figure 2-11.

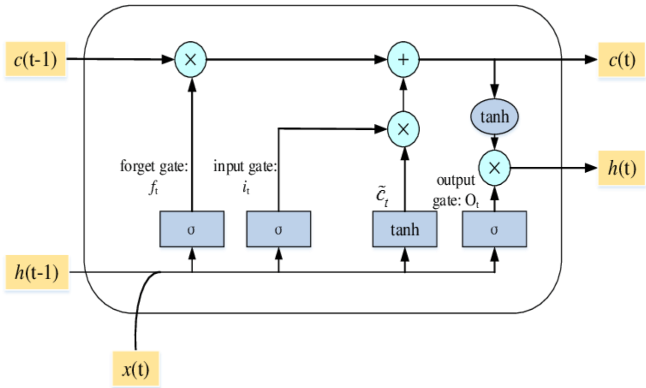

Figure 2-11 The structure of an LSTM

The output layer can represent a multi-resident scenario. For example, the probability of a combined activity at time t (at) is:

$$\mathrm{p}(a^t|o^{1:t}) = softmax(h^t c^t U + b) \tag{2.11}$$

Where U is the weight matrix linking the output and hidden layer, h^t and c^t (Cell State) are two hidden states. In several works, each resident's activities are modeled separately, and the output layer generates multiple layers. Supposing that some activities are performing by N residents, the probability of a resident n performs an activity at time t ($a^{n,t}$) is: $\mathrm{p}(a^{n,t}) = softmax(h^t c^t U_n + b_n)$ using a weight matrix U_n to connect each output layer with a shared hidden layer.

Liciotti et al. [72] evaluated the performance of various ML and DL models on CASAS datasets. The experimental results show that LSTM model achieved best performance compare with other ML models, such as CRF, HMM, and NB. Anubhav [41] employed an LSTM model on the ARAS multi-resident dataset and their results show that the performance achieved best 86.55% of accuracy. However, one of the problems with LSTM models is that there are too many parameters during training, and the computational complexity will be increased. Therefore, a fewer parameters model Gated Recurrent Unit (GRU) is proposed to make the training faster and less complicated. Besides, they proposed two LSTM-based models namely Ens2-LSTM and CascEns-LSTM. The Ens2-LSTM combines a basic LSTM and a Bi-LSTM with a combined softmax function to predict activities. On the other hand, CascEns-LSTM combines a basis LSTM and an Ens2-LSTM. Liciotti et al. [72] compared basic LSTM, Ens2-LSTM, Casc-LSTM, and CascEns-LSTM models on various multi-resident datasets published by CASAS. The results show that their proposed Ens2-LSTM and CascEns-LSTM obtained the best performance. However, with the increase of model complexity, the computation time of the dataset also increases, which is not suitable for online learning.

2) Gated Recurrent Unit (GRU)

Like LSTM, GRU is proposed to address the vanishing gradient and long-term memory issues in backpropagation, and it can largely improve training performance with fewer parameters. Compared with LSTM, GRU can also achieve better results, e.g., the recent performance comparison between LSTM and GRU shows that GRU has achieved higher performance in most machine learning applications. [73].

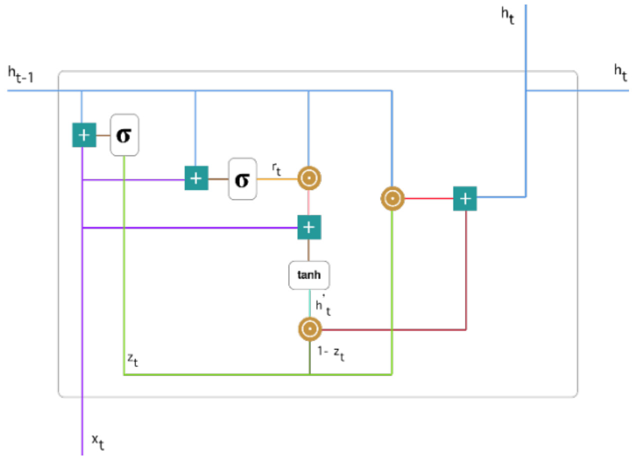

Figure 2-12 The structure of a Gated Recurrent Unit (GRU)

The structure of GRU is shown in Figure 2-12. LSTM and GRU are different in updating the next hidden state and content exposure mechanism. While LSTM is updated by summation, GRU associates it according to the amount of time required to save it in memory to update the next hidden state. In this structure, GRU has reset and update gates to enable each unit to pass information across multiple time windows for better classification or prediction [17]. Specifically, weights and sensor event data are stored in memory to be used with a given state to update future values. For an update gate, GRU computes z_t at a given time t to address the vanishing gradient problem by using the following formula:

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \quad (2.12)$$

Whereas for a reset gate, z_t is calculated at a given time t to decide how much the past information it needs to forget based on the following calculation:

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \quad (2.13)$$

The current storage content can be calculated using the following formula:

$$\tilde{h} = tanh(W_r[r_t h_{t-1}, x_t]) \quad (2.14)$$

Finally, h_t can be calculated in the final memory of the current time step to store the current unit information for calculating the output vector o_t, as follows:

$$h_t = (1 - z_t)h_{t-1} + z_t\tilde{h}_t \quad (2.15)$$

$$o_t = \sigma_0(W_0 h_t + b_0) \quad (2.16)$$

Tran et al. [42] evaluated the multi-resident HAR performance of GRU and LSTM based on ARAS and CASAS datasets. The experimental results show that GRU is slightly better than LSTM in multi-resident HAR. However, the performance of recognition is unsatisfactory, and data association issues have not been addressed in multi-resident HAR. Likewise, Natani et al. [41] evaluated the GRU model on the ARAS datasets. The results show that the GRU model outperforms the LSTM model.

3) Bi-directional LSTM (Bi-LSTM)

A Bi-directional LSTM (Bi-LSTM) consists of two LSTM layers (a forward and a backward LSTM layer). [74]. A traditional LSTM network can only learn one way, so it ignores future information. In Bi-LSTM, the input at the current time depends not only on the previous events but also on the subsequent events. Before and after the event, timing information is fully considered in combining the two units. The model structure is shown in Figure 2-13.

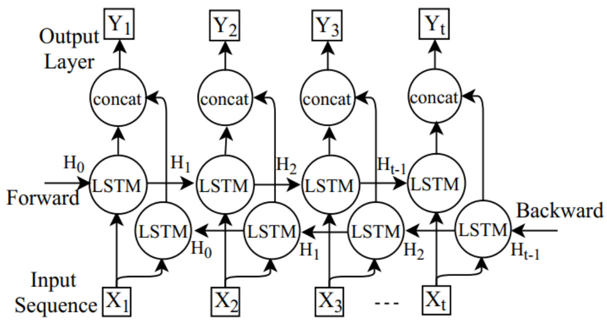

Figure 2-13 A structure of Bi-LSTM

The hidden states of the forward and backward layers can be expressed as h'_t and h''_t, respectively, where h'_t refer to a forward LSTM process and h''_t is a backward LSTM process, respectively. Then at a time step t, h_t is a concatenation of the forward and backward hidden states, shown as follows:

$$h_t = h'_t \oplus h''_t \tag{2.17}$$

Liciotti et al. [72] proposed a Bi-LSTM model and evaluated it using Cairo, Kyoto2, Kyoto3, and Kyoto4 datasets published by CASAS. Their results show that the optimal performance is obtained when the number of units is 64 and achieve the accuracy of 86.90%, 74.37%, 93.25%, and 85.82% in the four datasets, respectively, all of which are higher than an LSTM model.

The following Table 2-1 summarizes the advantages and limitations of various models in multi-resident HAR.

Table 2-1 Advantages and limitations for different models in multi-resident AR

References	Approach	Advantages	Limitations
[9] [69] [75] [48]	Naive Bayes (NB)	1. Stable classification performance 2. It achieves good results when the data size is small, can handle multiple classification tasks, and can be trained in increments, especially when the amount of data exceeds the memory. 3. The algorithm is simple and insensitive to missing data.	1. Performance perform not well when the number of data attributes is large. 2. Since the posterior probabilities are determined by the data and the prior, there is a certain amount of error in the classification. 3. Sensitive to the representation of input data
[12] [69] [48]	Dynamic Bayesian Network (DBN)	1. It is suitable for modeling temporal processes (time series). 2. It can monitor and update the system as time proceeds, even predict further behavior.	1. It is difficult for complex online learning HAR. 2. In changing, relationship pattern remains an open challenge.
[67] [13] [69] [75] [48]	Decision Trees (DT)	1. Tree structure visualization is easy to understand and explain. 2. Fewer data needed for training 3. Be able to process numerical data and classified data	1. It is easy to produce an overly complex model with poor generalization performance to the data. 2. Small changes in data may result in completely different trees being produced
[76] [12] [42]	Conditional Random	1. Able to relax strong independence assumptions.	The convergence speed of model training is slow.

	Field (CRF)	2. Ability to Incorporate arbitrary overlapping features	
[76] [72] [42] [48] [12] [9] [42]	Hidden Markov Model (HMM)	1. Based on strong statistical theory 2. Classification can be performed from the original data sequence. 3. Flexible generalization of sequence capabilities	1. There are many unstructured parameters 2. First-order HMMs are restricted by their first-order Markov property.
[12] [7] [69]	CHMM	Compare with HMM, CHMM is effective in multi-resident collaborative activities.	The complexity of the model is increased, and the processing time of the training set is also increased
[7] [5]	FHMM	FHMM performs better for some activities such as taking a shower, being outside, and relaxing.	1. Need more extensive training data because of the complex probabilistic model. 2. The computation time increase when increasing the number of residents
[7]	CL-HMM	1. Solved the problem of data association. Turn the multi-label problem into a single label problem. 2. Parallel and collaborative activities are well recognized.	1. The complexity of action recognition is significantly increased. 2. All tags need to be relabelled when new residents are added.
[41] [42] [72]	LSTM	1. It is very suitable for processing, classifying, and predicting time series data events 2. Compared with the traditional RNN model, LSTM can achieve better performance.	1. Suffer some exploding and vanishing gradients problems. 2. Require a large number of parameters. 3. Difficult to apply to online activity learning.
[41] [42]	GRU	GRU has fewer parameters and easy to train.	The standard GRU network processes the sequence in chronological order, ignoring the future context.
[72]	Bi-LSTM	Bi-LSTM networks have two LSTM layers and can store past and future information.	1. Require many parameters. 2. Difficult to apply to online activity learning.

2.5 Non-Intrusive Sensors, Public Datasets, and Evaluation Metrics

In this section, we first review the non-intrusive ambient sensors that have been used for

HAR in recent years. Then, we discuss the datasets for multi-resident HAR followed by evaluation metrics used in this field.

2.5.1 Non-Intrusive Ambient Sensors in HAR

Sensors are a kind of equipment that captures physical environment parameters, such as temperature and humidity, in a form of analog or digital signals, reported as sensor events. Recently, with the mushroom growth of sensor technology, sensors' cost gets lower while having a better functionality, which accelerates the development of artificial intelligence applications, including HAR [46].

For HAR, multiform non-invasive sensors have been widely welcomed and applied because people do not feel that their life is being monitored by others. Non-invasive sensors are not attached to an individual but can be seamlessly installed in any environment. Although this type of sensor can relieve the discomfort of a person as they do not need to be carried or worn, the complexity for HAR is greater than wearable sensors and VDO surveillance systems. Next, we introduce non-intrusive sensors that are commonly used in a smart indoor environment [44].

1) Passive Infrared (PIR) Sensors

PIR sensors (i.e., motion sensors) are applied to detect objects' movements through infrared radiation. If a PIR sensor detects a difference in infrared radiation (e.g., from the human body) higher than a predefined value, a sensor event is generated. This type of sensor is commonly used for security purposes, where alarms may sound, or lights may turn on when objects' movement is detected [77]. PIR sensors are obliged to operate in a dusky lighting environment because they are sensitive to thermal motions and they can sense the movement of any object generating heat, even if an object is inanimate. [10].

Figure 2-14 shows a diagrammatic diagram of a PIR sensor with two sensor elements placed in a motion plane. The output signals it captures in distinct directions, distinct distances, and distinct speed levels when walking. It can be seen from Figure 2-14a, by changing the polarity of the sensing element in a PIR sensor, the captured two signals can be distinguished based on the direction of walking [78]. Similarly, Figure 2-14b and c show that the amplitudes of the two signals and the time to reach the peak can also be distinguished. We can use output signals to differentiate movements and activities of multiple persons based on the distance of each person from the PIR sensor and their walking speeds.

2) Magnetic Door Sensors

A door sensor consists of a magnet and a reed switch, and the magnet component pulls the metal switch in the optical switch to close when a door is closed, thus completing the circuit and changing the state of the sensor, which is marked as a sensor event [33]. This sensor feature can be used to detect if windows, doors, or drawers are shut or open states [79]. For example, the Magnetic Door Sensors sensor is used in both the CASAS and ARAS datasets. The triggering of this sensor can tell us if a resident has entered or left the area, which is important for activity recognition [80].

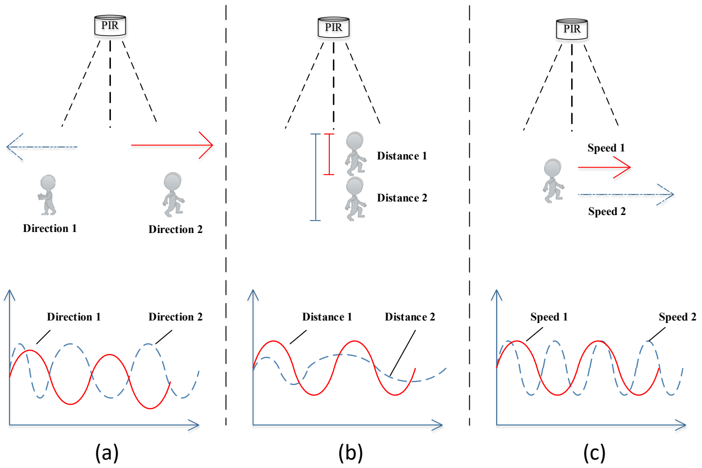

Figure 2-14 (a) walking in different directions; (b) walking at different distances; (c) walking at different speed levels. [81]

3) Light, Humidity, and Temperature Sensors

Other types of sensors including light, humidity, and temperature sensors are frequently integrated into one package, placed in a smart home environment to measure ambient information. This kind of sensor can read the surrounding environment's data according to the pre-set period of the current environment [12]. The sensors can report information when their status vastly differs from their previous state. For example, when residents are cooking in the kitchen, the temperature sensor comes into play and can detect the increase in kitchen temperature due to cooking, and we can know whether the user is cooking by the change in temperature [62].

4) Pressure Sensors

Various sensors can be applied to measure the pressure, such as a tactile sensor, which is sensitive to force, touch, or pressure. This kind of sensor is used to measure the interactivity between a person and a contact surface (e.g., floors, beds, sofa, or chairs) for monitoring the location and weight information of a person [6]. Many pressure sensors are used in the ARAS dataset for data collection. We summarize the advantages and disadvantages of these sensors, as shown in Table 2-2.

Table 2-2 Various Sensors Used in Multi-Resident HAR

Sensor type	Examples	Advantage	Disadvantage
Wearable sensors [17, 82, 83]	Smartphone Smartwatch	Ability to collect accurate data information and easily distinguish between residents.	With battery life limitations, many residents are reluctant to wear wearable devices.
Non-intrusive ambient sensors [7, 12, 13, 35, 39, 42]	PIR, Door, Light, Humidity, Temperature, Pressure	Easy to install and high privacy, residents do not feel be monitored.	Noise data can be generated, and it is difficult to distinguish between residents who trigger the sensor in a multi-resident environment.
Video Sensors [8, 84-86]	Camera	Ability to provide clear, rich video data information	Equipment can be expensive and undermines user privacy, constant monitoring has extensively intruded personal privacy.
Acoustic sensors [22, 87, 88]	Microphone	Ability to improve the accuracy of action recognition through acoustic information.	Producing noise signals and complex algorithms applied.

2.5.2 Datasets for Multi-Resident HAR

Publicly available benchmarking datasets are necessary for evaluating and comparing different HAR methods. Some researchers put great effort into developing sensor hardware, setting up scenarios, recruiting participants, and collecting data in real life. In contrast, most other works rely on some public datasets to evaluate their proposed method. A variety of single-resident datasets are collected and made publicly available (e.g., in[18, 49, 89]). However, seldom datasets based on non-intrusive ambient sensors are collected in a multi-resident environment. The CASAS (Centre for Advanced Studies in Adaptive Systems) group has published some multi-resident datasets, which are collected based on an in-lab environment, e.g., "MultiResident ADLs", "Kyoto spring 2009", "Kyoto summer 2009", "Kyoto 2010", "Tulum 2009-2010", "Tulum 2009", and "Cairo 2009". Similarly, Alemdar et al. [80] have published a multi-resident dataset named ARAS (Activity Recognition with Ambient Sensing), which is collected from a real-world environment. The datasets include two separate collections: House A and House B. These datasets are only the ones collected based on non-intrusive ambient sensors for multi-resident HAR.

1) CASAS Datasets

CASAS was established at Washington State University in 2007. Up to now, there are more than 20 datasets were collected by CASAS, and some of them are related to multi-resident datasets [36]. For example, one dataset named "Multiresident ADL Activities" presents two people in the apartment simultaneously performing fifteen Activities of Daily Living (ADL). The types of activities, based on how they are performed, include sequential, interleaving, parallel, and collaborative [44]. The details of the most widely used multi-resident benchmark datasets published by CASAS are summarized in Table 2-3 and Table 2-4. As far as we know, no research uses Kyoto 2008, Tulum 2009-2010, and Paris 2010 for multi-resident HAR due to their limitation, which can be seen in Table 2-3.

2) ARAS Datasets

ARAS is one of the famous datasets collected for multi-resident HAR. This dataset can be divided into two collections: House A and House B where all the same twenty-seven activities were performed in House A and House B, and both were equipped with various non-intrusive ambient sensors [44]. Each dataset's first 20 columns denote a binary sensor value, 0/1, where 0 means sensor fired, and one means not fired. The last two columns, 21 and 22, refer to the labels of activities for Resident 1 (R1) and Resident 2 (R2), respectively. The details of the ARAS dataset are summarized in Table 2-4 and Table 2-5.

In general, researchers often use more than one dataset to evaluate their proposed method. For example, Ifat et al. [36] used the ARAS dataset and CASAS to evaluate their SARRIMA system. Alam et al. [12] collected a real-world dataset named CACE to evaluate their method but also used CASAS for comparison. The CASAS and ARAS datasets have become well-known benchmarking datasets for multi-resident HAR in a non-intrusive environment.

Table 2-3 Non-intrusive sensor-based datasets in the multi-resident setting published by CASAS

Dataset	# of residents	Duration	# of Sensors	# of ADL	# of sensor events	Environment	Annotated	Serial number	Last Updated	Limitation
Kyoto MultiResident ADLs	2	2 months	37	15	17258	Lab	Yes	4	7/7/2009	
Kyoto 2008	2	1 month	21	4	20952	Lab	Yes	6	5/24/2010	- Only four activities performed - The label does not distinguish the actions of R1 and R2
Kyoto spring 2009	2	2 months	71	9	137789	Lab	Yes	7	7/8/2014	
Kyoto summer 2009	2	2 months	86	8	772544	Lab	Yes	8	5/13/2010	
Tulum 2009	2	4 months	20	9	486912	Lab	Yes	9	1/18/2011	
Tulum 2009-2010	2	1 year	36	15	1 085 902	Lab	Yes	10	1/18/2011	- The label does not distinguish the actions of R1 and R2
Kyoto 2010	2	1 year	87	13	2 804 813	Lab	Yes	11	7/10/2014	
Cairo 2009	2+pet	2 months	32	11	726 534	Lab	Yes	14	9/3/2009	
Paris 2010	2+pet	1 year	NA	NA	1269265	Lab	No	16	11/29/2010	- No label

Table 2-4 Activities of the CASAS Multi-Resident Datasets

Kyoto MultiResident ADLs	Kyoto 2008	Kyoto spring 2009	Kyoto summer 2009	Tulum 2009	Tulum 2009-2010	Kyoto 2010	Cairo 2009
- Pack supplies in the picnic basket -Fill medication dispenser -Water plants - Pack food in the picnic basket -Help move the couch and coffee table - Play a game of	- Eat - Relax - Sleeping - Cook	-Watch TV -Wash bathtub -Study -Work -Sleep -Personal hygiene -Bed to toilet -Meal	-Wake up -Sleep -Shower -Grooming -Cooking -Cleaning -Bed to toilet -Work	-Wash dishes -Snack -Leave home -Eat breakfast -Group meeting -Enter home	-Worktable -Work living room -Work bedroom 2 -Work bedroom 1 -Watch TV -Wash dishes -Sleeping in bed	-Watch TV -Wandering in room -Sleeping not in bed -Sleep -Personal hygiene -Meal preparation -Leave home	-Take medicine -Night wandering -Lunch -Leave home -Laundry -Dinner -Work in office

checkers -Set out ingredients for dinner - Read a magazine-gather food for a picnic - Hang up clothes-sweep the kitchen floor - Set the dining room table for dinner - Paying an electric bill - Retrieve dishes		preparation -Clean		-Cook lunch -Cook breakfast -Watch TV	-Personal hygiene -Meal preparation -Leave home -Enter home -Eating -Bathing -Yoga -Bed to toilet	-House-keeping -Enter home -Eating -Bed to toilet -Bathing -Work	-Wake -Sleep -Breakfast -Bed to toilet

Table 2-5 Setting and Activities in the ARAS dataset

	House A	House B
# of Ambient Sensors	20 of 7 different types	20 of 6 different types
Size of the House	50m2	90m2
House Information	One bathroom, one kitchen, one living room, one bedroom	One kitchen, one bathroom, one living room, two bedrooms,
Residents	2 males, age 25	Married couple. Average age 34
Duration	30 full days	30 full days
# of Activities	27	27
Activities	Other, watching TV, listening to music, preparing lunch, having lunch, having a conversation, napping, preparing dinner, having dinner, talking on the phone, having a snack, sleeping, brushing teeth, studying, preparing breakfast, having breakfast, washing dishes, having a shower, toileting, using the Internet, changing clothes, reading a book, laundry, shaving, cleaning, having guest, going out	

Table 2-6 Sensors Used in the ARAS Datasets

House A	House B
Kitchen temperature sensor	Kitchen sonar distance
Water closet proximity sensor	Bathroom door sonar distance
Tap proximity sensor	Armchair pressure mat
Kitchen sonar distance	Bed pressure mat
Hall sonar distance	Couch pressure mat

Shower cabinet DCS	Kitchen drawer photocell
Bathroom DCS	Fridge photocell
House DCS	Chair force sensors
Bathroom cabinet photocell	Tap distance sensor
Wardrobe photocell	Shower cabinet DCS
Kitchen drawer photocell	Wardrobe DCSs
Fridge photocell	House DCS
Chair proximity sensors	Kitchen cupboards CSs
Couch force sensors	Closet sonar distance
TV infrared receiver	
Convertible couch photocell	
(Resident 2's bed)	
Wardrobe photocell	
Bed force sensor	

2.5.3 Performance Evaluation Metrics

The performance of HAR can be evaluated with commonly used evaluation metrics including accuracy, f-measure, precision, recall, complexity, computation time and data size, robustness, diversity, scalability, types of sensors, storage requirements, and users [90].

The evaluation approach ought to illustrate the methods of using data, training data, and testing data. [51]. The recognition result is used in the confusion matrix $M_{n\times n}$ for the n-type classification problem. In a binary classification problem, the following values can be acquired from the confusion matrix.

- True Positives (TP): The number of positive samples predicted to be positive by the model.
- True Negatives (TN): The number of negative samples predicted to be negative by the model.
- False Positives (FP): The number of negative samples predicted to be positive by the model.
- False Negatives (FN): The number of positive samples predicted to be negative by the model.

Accuracy is the most used evaluation to measure the overall classification performance and the formula is defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.17}$$

Precision is a measure of accuracy that indicates the proportion of actual positive cases among the examples categorized as positive.:

$$Precision = \frac{TP}{TP + FP} \tag{2.18}$$

The recall is a measure of coverage, and it measures the ratio of positive simples are classified as positive to the total number of positive instances:

$$Recall = \frac{TP}{TP + FN} \quad (2.19)$$

The f-measure is the weighted harmonic mean of precision and recall:

$$f - measure = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (2.20)$$

We can extend these metrics for n classes and it is possible that, of one class, its instance can be positive or negative [91]. For example, positives could be walking, while negatives could be all instances other than walking. Performance evaluation metrics, datasets, and results from the existing studies are presented in Table 2-7.

Table 2-7 Evaluation Metrics for Multi-Resident AR

References	Dataset	Approach	Validation	Performance
(Alemdar et al 2017) [5]	ARAS	FHMM	leave-one-out	F-measure House A- R1: 78.8%; R2: 69.2% House B- R1: 80.1%; R2: 77.3%
(Alam et al 2016) [12]	CASAS	DBN	10-fold CV	Accuracy: 94.5% Precision: 96.5% Recall: 94.5% Weighted ROC: 98.6%
(Benmansour et al 2017) [7]	CASAS Multi-resident ADLs	CL-HMM	leave-one-out	Accuracy: 86.04% Precision: 72.26% Recall: 76.75% F-measure: 74.32%
(Chen et al 2014) [76]	CASAS Multi-resident ADLs	HMM CRF	3-fold CV	Accuracy HMM:84.19% CRF: 82.88%
(Emi et al 2015) [36]	CASAS Spring 2009 CASAS Summer 2010 ARAS	SARRIMA	10-fold CV	Accuracy CASAS Spring 2009: 97.34% CASAS Summer 2010: 98.15% ARAS House A: 87% ARAS House B: 95.3%
(Yin et al 2016) [9]	Real-world data	NB HMM	leave-one-out	Accuracy NB: 73.01% HMM: 73.70%
(Wang et al 2017) [31]	Real-world data	HMM	10-fold CV	Accuracy HMM: 96.5%
(Tran et al 2018) [42]	CASAS ARAS	LSTM GRU	leave-one-out	Accuracy CASAS- LSTM: 82.14%; GRU: 83.66%

				ARAS House A- LSTM: 56.44%; GRU:56.65% ARAS House B- LSTM: 76.72%; GRU:76.83%
(Liciotti et al 2020) [72]	CASAS	Ens2-LSTM CascEns-LSTM	3-fold CV	Accuracy: Ens2-LSTM:94.24%; CascEns-LSTM: 93.60% Precision: Ens2-LSTM:94.33%; CascEns-LSTM: 93.33% Recall: Ens2-LSTM:94.33%; CascEns-LSTM: 93.33% F-measure: Ens2-LSTM:94.00%; CascEns-LSTM: 93.33%
(Natani et al 2019) [41]	ARAS	LSTM GRU	10-fold CV	Average accuracy Resident A- LSTM: 71.61%; GRU: 86.55% Resident B- LSTM: 72.79%; GRU: 88.21%
(Malazi et al 2018) [48]	CASAS	EP score + random forest	10-fold CV	F-measure: 98.15%
(Prossegger et al 2014) [67]	ARAS	DT	leave-one-out	Accuracy- House A: 49.28%; House B: 84.45%
(Zhao et al 2019) [53]	Real-world data	DT	leave-one-out	Accuracy: 85.49%
(Li et al 2020) [35]	Real-world data	Markov Logic Network	leave-one-out	Precision: 99.4%
(Yuan et al 2020) [40]	CASAS	Frequent Itemset Mining	10-fold CV	Precision: 81% F-measure: 66.8%

2.6 Open Issues and Future Directions

Although more and more researchers have studied multi-resident HAR in recent years, it is still in its infancy stage. Here, we highlight key open issues and future directions that should be considered for significant advances in the field of multi-resident HAR.

2.6.1 Complex Activity Recognition in a Large Number of Residents' Situation

Human activities are often executed in real-life situations in a more complicated manner, sometimes with more than two residents performing cooperative activities together [9]. Interleaved or concurrent activities are common in a multi-resident environment, but the existing datasets only contain two persons. Specifically, there are two types of activities in multi-resident HAR that should be studied in the future. Firstly, concurrent or interleaved activities performed by more than two residents are needed [85]. We believe that the difficulty and challenges of HAR will be a geometric increase when there are more than two residents in an environment. Secondly, recognizing collaborative activities performed by multiple people is also challenging because collaborative activities generate more, and more complex, sensor events, increasing the difficulty of recognition [69].

2.6.2 Online Learning for Multi-Resident HAR

Compared with offline HAR, online learning is also still in infancy, even for a single-resident situation. Although some proposed solutions based on online learning (e.g., Meng et al. [30] created an online daily habit modeling and anomaly detection (ODHMAD) model for real-time activity recognition, abnormal detection, and habit modeling), their work is only applied in a single-resident scenario.

Often, surveillance systems are expected to make immediate inferences in situations that could put people at risk. Occasionally, there could be some emergencies, such as falling suddenly or forgetting to take medicine, and we should be able to learn and recognize these situations as early as possible. Thus, it is necessary and important to have an online solution in place to recognize and distinguish all the residents (including their guests) and their activities in real-time [30]. Since online inference and online learning are relevant to activity monitoring, more effort should be put into this area [19].

2.6.3 Datasets for Non-Intrusive Complex Activities of Multi Residents

Quantitative comparisons of multi-resident HAR methods are not straightforward due to the various datasets [40]. The lack of standardized benchmarks makes it challenging to evaluate multi-resident HAR models exhaustively and fairly. Most existing studies rely on the CASAS and

ARAS datasets [7, 42] which are limited. These two datasets contain only a portion of ADL performed by a maximum of two residents. More datasets that contain more than two people performing a variety of complex activities are needed. Also, the existing datasets use simple environmental sensors which are rather impractical and inconvenient for setting up. As sensor technology evolves, further research is needed on how effective it is when using different sensor technology and if the number of sensors can be minimized to improve the ease of deployment while obtaining a similar recognition performance.

2.6.4 Intelligent Solutions for a Multi-Resident Environment

Current solutions can only recognize past and current activities, but in a real situation, there is a necessary for more intelligent solutions which can recognize more kinds of activities, abnormal activities, and potential future activities [43]. Current HAR systems can recognize many actions and bring benefits to many real-life situations, it is interesting and challenging if HAR can be used to predict future activities. This function is critical, especially like prevention and fall detection. A possible research direction would be studying how we build a HAR system that can predict future activities based on past activities [92].

2.7 Conclusion

Multi-resident HAR has gained attention since this research field can be used and applied for people's daily living and healthcare monitoring. In this survey, we conducted an extensive review of different HAR techniques and models that can be applied to multi-resident environments. While much of the effort has been put in a single-resident setting, multi-resident HAR has recently become the central focus of many studies. Yet many unresolved technical issues and challenges are introduced and needed to be addressed in the future. This paper also reviews different types of sensors, public datasets, and performance evaluation metrics. Finally, we highlight and discuss open issues and future directions of the field.

Chapter 3 Hybrid Fuzzy C-means CPD-based Segmentation for Improving Sensor-based Multi-resident Activity Recognition

Multi-resident Activity Recognition (AR), which has become a popular research field in smart environments, aims to recognize the activities of multiple residents based on data collected from various types of sensors, and sensor event segmentation is an important technique for enhancing the performance of activity recognition. While quite some segmentation methods have been proposed for the single person setting, few studies have been done for the multi-resident setting. In this chapter, we first evaluate the baseline and the state-of-the-art segmentation methods using the popular multi-resident dataset CASAS, to confirm that the performance of multi-resident AR can be improved by applying segmentation techniques; we then propose a novel Hybrid Fuzzy C-means CPD-based segmentation method that can further enhance the performance of multi-resident AR. We combine a Fuzzy C-means method with a CPD-based method for sensor event segmentation. The Fuzzy C-means method is used to classify the sensor events in terms of sensor locations, and then the Change Point Detection (CPD) technique is used to probe the transition actions to determine the segmentation sequence. Our experimental results show that the proposed method significantly improves the performance of multi-resident AR in comparison with the baseline and state-of-the-art classification methods.

Chapter 3 has been published by IEEE Internet of Things Journal.

Chen, D., Yongchareon, S., Lai, E. M., Yu, J., & Sheng, Q. Z. (2021). Hybrid Fuzzy C-means CPD-based Segmentation for Improving Sensor-based Multi-resident Activity Recognition. IEEE Internet of Things Journal. DOI: 10.1109/JIOT.2021.3051574.

3.1 Introduction

Human activity recognition (HAR) aims to learn and recognize different types of human activities from various formats of data including images and sensor signals [93, 94]. HAR has been widely applied in some real-world applications such as Ambient Assisted Living (AAL) systems and home monitoring [63] as it assists in the detection of potentially unexpected activities, such as sudden falling or dangerous situations [9, 22]. With the increasing elderly population and advances in smart environments, HAR has become a popular research field [7]. The development of sensor and communication technology enables the realization of smart homes. By installing different kinds of devices in a smart home, the activities of residents can be monitored and recognized by many well-known traditional machine learning (ML) models such as naïve Bayes (NB), Dynamic Bayesian Networks (DBNs), Support Vector Machines (SVMs), Hidden Markov Models (HMMs) and Conditional Random Field (CRF) [7]. Meanwhile, recent years have witnessed the fast advancement and development of deep learning (DL) methods, such as Convolutional Neural Network (CNN), Deep Neural Network (DNN), and Recurrent Neural Network (RNN) [18], that have been developed to increase the recognition performance the field of HAR.

Although many HAR solutions have been proposed for a single resident [32, 34, 95], real-life environments often involve more than one person, with intricate interactions between the resident and their guests or family members. Thus, multi-resident AR is a significant research area but is still in its infancy [5, 44].

Before addressing the challenges in multi-resident settings, we must address the many unsolved problems in single-resident environments [67]. In recent years, multi-resident situations in smart-home environments and complex AR in multi-interactive settings have attracted increasing attention [5, 7, 9, 12]. Sensor event segmentation partitions a continuous sequence of sensor events into discrete and significant chunks which can help to isolate different activities [21]. The segmentation techniques for sensor events have been proved to be able to improve the accuracy of classification models in a single-resident environment by using sliding windows because it offers the advantage that the start and endpoint of each activity occurrence are known

[50, 63]. The identification of activity boundaries enables important features such as start time, end time and duration to be extracted in sensor data. Such information can improve the performance of activity recognition [60].

Several approaches for partitioning sensor events have been proposed in recent years, however, most of them are developed based on pre-segmented sensor streams [52, 56]. Besides, the situation of multi-resident AR is more complex, the composition of sensor events is composed of multi-resident triggering multiple sensors concurrently, so for a time-series sensor data, general sliding windows methods cannot well demarcate the boundaries of multi-resident activities. Besides, the multi-resident AR faces unique challenges of complex activities such as parallel and collaborative activities. Parallel activities refer to multiple residents perform activities simultaneously (e.g., one resident is cooking in the kitchen, and the other is sweeping the floor in the living room). Collaborate activities refer to multiple residents work together, and each resident performs the same activity (e.g., two residents play a video game in the living room) [44]. Traditional data-driven segmentation methods do not effectively distinguish between multi-resident triggering sensor events simultaneously. Pre-segmented offline segmentation techniques are performed by inspecting all dataset and thus are not suitable for the real-time application. Furthermore, whether data-driven segmentation techniques can provide desirable performance efficiency for multi-resident AR considering its complex activities remains to be investigated. We argue that one of the most important and prerequisite processes for identifying complex activities is the segmentation of a continuous sequence of sensors events into segments, each corresponding to a single ongoing activity. By doing so, parallel and collaborative activities of multiple persons can be transformed into single-resident sequential activities [96]. In this paper, we first evaluate well-known data-driven segmentation techniques using a series of experiments and analyze their influence on the performance of various classification models in a multi-resident setting. The segmentation methods include static time-based windows, static sensor-based windows, and dynamic sensor–time correlation method. Then, we propose a novel hybrid fuzzy C-means CPD-based segmentation method that puts sensors into different clusters by using the fuzzy C-means clustering method [20], and then a change point detection method is used to segment each activity of diverse residents, to improve the overall activity classification accuracy. The main contributions of this paper are as follows:

- We evaluate the baseline and the state-of-the-art data-driven segmentation techniques and discover that they can improve the performance of multi-resident AR based on traditional machine learning and deep learning classification models.
- We propose a novel online, hybrid Fuzzy C-means CPD-based segmentation technique to provide an optimal time-window size for recognizing multi-resident complex parallel and collaborative activities.
- We evaluate our data-driven segmentation technique and compare it with the other baseline data-driven segmentation techniques and apply to classification models. The results show that our proposed method can significantly improve the performance of all the baseline and state-of-the-art classification models used in our study.

The remainder of this paper is organized as follows. Section 3.2 reviews the related works and Section 3.3 describes the methodologies. Section 3.4 reports the experimental and evaluation results. The study concludes with Section 3.5.

3.2 Related Works

In general, existing HAR studies can be divided into two categories: segmentation of sensor sequences and identification of activities. This section discusses the existing works in both categories of HAR sub-problems.

3.2.1 Data-Driven Segmentation Methods

The most crucial technique for segmenting sensor stream data is the sliding window technique, which has been employed in many studies. Panel A of Figure 3-1 illustrates the actual activity of the sensor events during actions 1–4, and panels B and C demonstrate the segmentation in the time-based and event-based approaches, respectively [15].

As shown in Figure 3-1B, the time-based sliding windows approach partitions the whole sequence of sensor events into a series of segments with equal time intervals. The timestamp is a key parameter for generating activity segments in a sequence of sensor events. In the parallel setting, the entire sequence of sensor events is partitioned into a series of sliding windows with equal sensor numbers. For example, every 30 sensor events can be grouped, as demonstrated in

Figure 3-1C. A critical question in the static window size approach is identifying the right window size. Some works calculate the window size by averaging the times of the various activities and sampling the sensor's frequency [20].

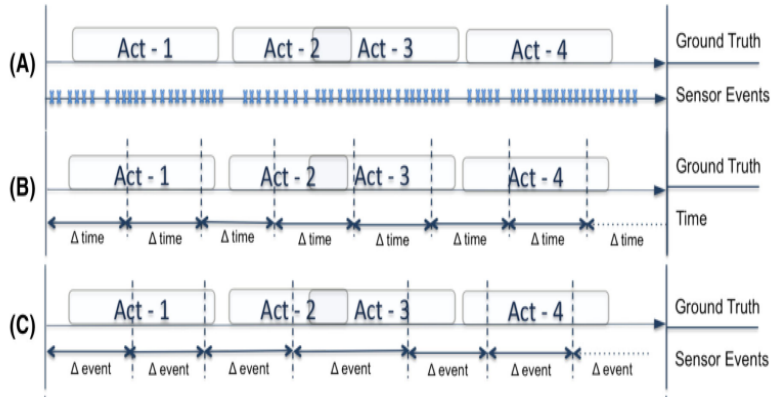

Figure 3-1 Segmentation methods [15]

Singla et al. [97] employed a static sensor-based sliding window covering an endless number of sensor events. This approach improves the detection performance by partitioning the whole sequence of sensor events into small groups, each containing the information of one activity. However, this method does not always recognize short-lived actions because (as mentioned above) the optimal static window is hard to define.

Wang et al. [56] partitioned a sequence of sensor events by a static time-based window approach. This method segments the sequence into small temporal chunks with the same period. To reduce the processing cost, the sensor data were collected by a wireless body network. This approach is suitable for classifying continuous and periodic sensor events. Still, its performance deteriorates when applied to motion sensors, door sensors, temperature sensors, and other ambient sensors installed in the environment. Defining the optimal window size is again challenging because various activities are completed in different timeframes. For example, drinking water is a short-term activity; a working activity may consume several minutes or hours; and sleep may continue for several hours. A large sliding window size tends to miss the short-term activities, whereas a small, segmented window size can divide one action into two or more partitions.

Wan et al [15] proposed a dynamic segmentation approach that identifies simple actions using both sensor and time correlations. They used the Pearson product-moment correlation (PMC) to calculate the relationship between two sensors. To evaluate the sensor dependency on the classification result, they estimated the sensor and time correlations separately. For example, when two sensors A and B are positively correlated, uncorrelated, and negatively correlated, the correlation coefficients are +1, 0, and -1, respectively. The time correlation was determined by deciding whether different action timeframes belonged to the same time division. Finally, the sensor and time correlations were combined to determine the sensor event segmentation.

Aminikhanghahi and Cook [63] introduced a change point detection (CPD) based segmentation method, which splits residents' sensor events into different non-overlapping portion in real-time. Specifically, they applied a separation (SEP) algorithm to confirm whether the activity transition has occurred between two sequential sensor events. SEP is a method of estimating the probability distribution ratio by using separation distance as a dissimilarity measure. The experiments on real-life smart home datasets provide evidence that detecting activity transformation and utilizing segment features in activity recognition can improve recognition performance while providing activity boundaries and transformation details.

3.2.2 Knowledge-Driven Segmentation Methods

Meanwhile, there are some knowledge-driven segmentation approaches relying on pattern recognition applied in HAR and other fields.

Ye et al. [96] proposed a novel knowledge-driven concurrent activity recognition (KCAR) approach. In KCAR, the underlying semantics of each sensor event is explored, and semantic similarities are exploited to partition successive sensor sequences into fragments, each of which corresponds to an ongoing activity. At the same time, the Pyramid Match Kernel is used, because it is a hierarchical conceptually approximate matching advantages to identify activities by obtaining constraints at different granularities from potentially noisy sensor sequences.

Triboan et al. [98] proposed a semantics-based method to sensor data segmentation in real-time activity recognition. The system captured generic knowledge and resident-specific preferences for recognizing Activities of Daily Living (ADL) to support the segmentation process. They developed a system prototype and multi-threaded decision algorithm and applied in 30 case scenarios. The results illustrated that all sensor events could be appropriately segmented with 100% accuracy for a single ADL environment and received 97.8% accuracy for the composite ADL scenario.

Sfar and Bouzeghoub [50] proposed a method that combines a statistical learning and ontology interpretation to select the best time-window size. The method includes two subparts. The first part is offline training where a clustering method is applied offline and updates the ontology model. The second part is online processing where the updated ontology model is utilized to determine the optimal time-window, and a machine learning model Support Vector Machine (SVM) is implemented on the CASAS datasets and enhancing the performance of activity recognition.

Pan et al. [99] proposed a swimming analysis scheme to count and identify swimming strokes by using accelerometers. The proposed method includes a data processing stage and a swimming stroke identification stage. The data processing stage includes sensor data collection, floating data elimination, and feature points detection. The swimming stroke analysis stage identifies swimming stroke and counts strokes by applying a correlation coefficient and some rules to segment the sensor events, which can be considered swimmers' stroke. The experiment results proved that the proposed scheme could effectively count strokes and identify swimming stroke with more than 94% and 87% accuracies on average, respectively.

Malawski and Kwolek [100] presented a method dedicated to the real-time analysis of continuous step training routines in fencing. The authors proposed a model-based adaptive filtering algorithm for the accurate selection of the segmentation of interest from velocity signals collected from the Kinect motion sensor and removing false positives from the selected segments by extracting dedicated features and applying an SVM classifier. Finally, the parameters of the identified bowing movements, which constitute the fencer's feedback, are computed. The proposed method was evaluated on a dedicated dataset consisting of eight fencers' movements and achieved an accuracy of 99.51%.

Reily et al. [84] introduced a novel system for automatically analyzing a gymnast's performance on the pommel horse by using the Microsoft Kinect portable 3D camera. A revised depth of interest segmentation was applied to locate depth ranges in the image which are probably to include some objects or people of interest. The Kinect can effectively segment the scene, identify the depth of interest, locate the gymnasts in the area, identify what gymnasts are doing during their performance, and then analyze their performance accurately.

3.2.3 Multi-Resident Activities Identification

In recent years, a number of classification models have been studied for the identification of multi-resident activities.

Yin et al. [9] tracked the locations of multiple residents by a naïve Bayes approach, which recognized their daily activities such as making meals and watching television. The data were collected by ambient sensors (motion sensors, power sensors, temperature sensors, humidity sensors, and accelerometers) installed in different places. In various settings, naïve Bayes achieved higher accuracy (73.01%) than other methods. However, traditional naïve Bayes is inappropriate for modeling temporal processes. A more suitable approach is DBN, an upgrade of NBC [44].

Benmansour et al. [7] evaluated cooperative and parallel activities between two residents in a smart environment were removed by different methods based on HMM variants. The observation and activity labels of the different occupants were combined by a proposed CL-HMM method, where activity labels, as well as observation labels of different residents are combined to generate the corresponding sequence of activities, as well as the corresponding sequence of observations on which a conventional HMM is applied, producing a related sequence of activities and observations. The authors competed their approach against two other baseline models: parallel HMM (PHMM) and coupled HMM (CHMM). CL-HMM outperformed both PHMM and CHMM in predicting collaborative activity.

Liciotti et al. [72] applied Long Short-Term Memory (LSTM) for evaluating the multi-resident HAR based on the four multi-resident CASAS datasets. The results show that the LSTM based model outperforms traditional ML methods, such as NB, HMM, and CRF. Natain et al. [41] employed LSTM on ARAS multi-resident dataset and evaluated the LSTM model on resident A and resident B. The results show that HAR’ performance acquired the best accuracy of 86.55% when training 50 days. However, the LSTM model's problem is that there are too many parameters to be set during training the model, which increases the computational complexity. Therefore, A Gated recurrent unit (GRU) with fewer parameters is introduced to make the implantation less complicated and faster.

Recently, Tran et al. [42] evaluated the LSTM and GRU models’ performance on the multi-resident AR by using two published benchmark datasets, CASAS , and ARAS [80]. The experiments recognize resident A and resident B's activities and calculate the average accuracy as performance. The results show that the GRU model is more efficient in multi-resident AR. However, the average identification accuracy was only 77.2%, and the data association problems in multi-resident AR have not been solved. Similarly, Natain et al. [41] applied the GRU model on the ARAS dataset. GRU outperforms LSTM on both resident A and resident B.

3.3 Proposed Segmentation Approach

Segmentation of sensor events can bring various benefits to the analysis of human behaviour [50]. The existing research has been able to prove that the detection of behavior transformation and the segmentation of sensor activity data can improve the performance of activity recognition [63]. Besides, the segmentation based on sensor events provides extra meritorious insights for behaviour patterns by identifying the beginning and end of activities and generating some new features such as activity duration and overall activity behaviour parameters [60].

For multi-resident AR, we need to not only recognize the start and endpoints of the activity but also distinguish which resident triggered the sensor event. Figure 3-2 illustrates an overview of our proposed multi-resident activity segmentation structure, which contains four steps. The first step is to use a variety of non-invasive ambient sensors to collect the sensor data in the smart home of a multiple residential environment. In the second step, we cluster all triggered sensors according to their locations and divide the whole data into different subsets by applying the Fuzzy C-means method neighbouring triggered sensors. The third step is to recognize the activity transition on different subsets and generate the activity segments. The final step is to identify multi-resident activities by using various baseline and state-of-the-art machine learning and deep learning classification models as shown in Figure 3-2. In particular, we evaluate four segmentation techniques for multi-resident AR based on the collected and segmented data. The first approach is the static time-based sliding window that partitions the whole sensor event by offering a fixed length of time. The second approach is the static sensor-based sliding window that applies a fixed number of sensor events to divide the sensor data. The third approach uses dynamic time and sensor correlation to partition the sensor events into subsequence by calculating

Pearson product-moment correlation (PMC) between pairs of sensor events. The fourth approach is our proposed method that divides sensor events into different subset by clustering the sensor location and then a change point detection method is applied in the subgroup to labels the sensor events between two transitions as a single activity.

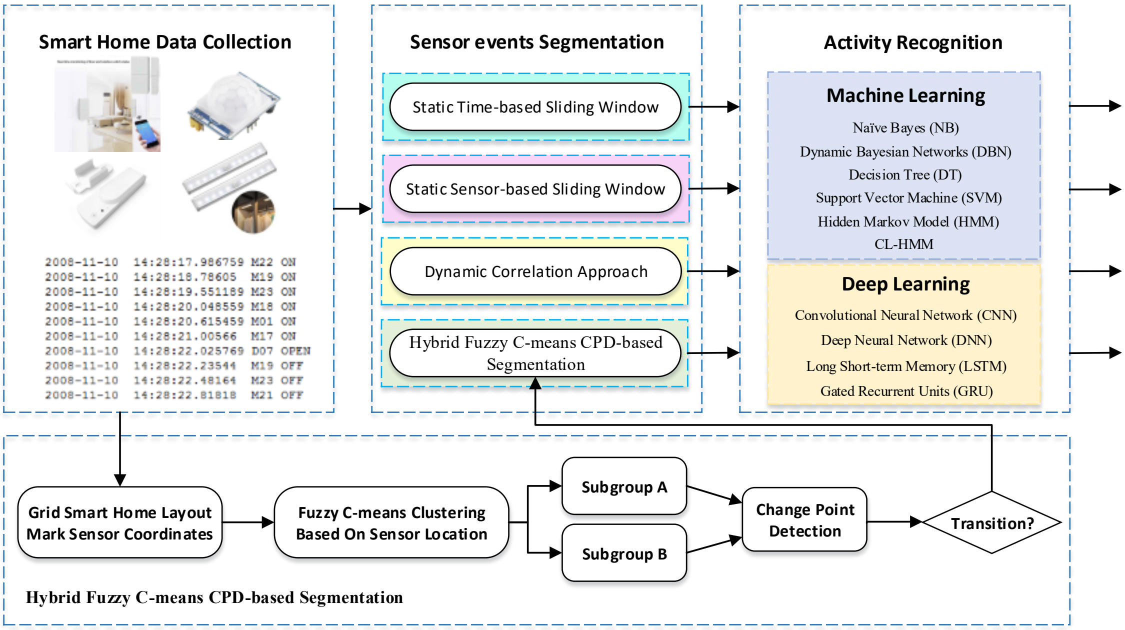

Figure 3-2 The architecture of the proposed segmentation approach for HAR

3.3.1 Fuzzy C-means (FCM)

The FCM algorithm is an unsupervised data clustering method based on the optimization of the objective function using the Fuzzy theory [20]. The clustering result is the membership degree of each data point to the clustering center, which is represented by a number. It obtains the membership degree of each sample point to all the class centers by optimizing the objective function, to determine the class of sample points and achieve the purpose of automatic clustering of sample data. An object can belong to multiple categories at different levels [20].

The Fuzzy C-means clustering method provides more flexible results compared with traditional K-means hard clustering as it can belong to a different cluster. In most cases, objects in a dataset cannot be divided into distinct separate clusters [101]. Therefore, each object and each cluster are given a weight indicating the degree to which the object belongs to the cluster. The membership function $\mu_A(X)$ is a function that represents the degree to which an object X belongs to set C. Its independent variable range is all objects that may belong to set N (that is, all points in the space where set *N* is located), and the value range of $\mu_A(X)$ is [0,1] [101].

The FCM algorithm divides a finite set of N elements $X = \{X_1, X_2, \dots X_N\}$ into a set of C clusters according to some given criteria. Each element $X_i \in X, i = 1,2, \dots, N$ is a vector with d dimensions. We define an approach to partitions X into C clusters centers $V_1, V_2, \dots V_C$ in the centroid set V.

In the FCM algorithm, *U* is a representative matrix for membership of each cluster. The matrix *U* has some characteristics:

- $U(i,j)$ is the membership value of the element X_i in a cluster with center $V_j, 1 \leq i \leq N; 1 \leq j \leq C$.
- $0 \leq U(i,j) \leq 1, 1 \leq i \leq N; 1 \leq j \leq C$ and $\sum_{j=1}^{C} U(i,j) = 1$ for each X_i.
- The larger $U(i,j)$ is, the higher the degree of confidence that the element X_i belongs to the cluster *j*. An objective function is defined such that the clustering algorithm minimizes the objective function using Eq. (3.1):

$$J(U,V) = \sum_{i=1}^{N} \sum_{j=1}^{C} U(i,j)^m D(i,j)^2 \tag{3.1}$$

where $D(i,j)^2 = \| X_i - V_j \|^2$ is the Euclidean Distance between the element X_i and the cluster center V_k. *m* is the fuzzification coefficient of the algorithm. Fuzzy *C* is an iterative process of computing the affiliation degree $U(i,j)$ and cluster center V_j until they reach an optimum, $U(i,j)$ and V_j can be calculated using Eq. (3.2) and (3.3):

$$U(i,j) = \frac{1}{\sum_{k=1}^{C} \left(\frac{\| X_i - V_j \|}{\| X_i - V_k \|} \right)^{\frac{2}{m-1}}} \tag{3.2}$$

$$V_j = \frac{\sum_{i=1}^{N} U(i,j) \cdot X_i}{\sum_{i=1}^{N} U(i,j)} \tag{3.3}$$

The termination conditions for the iteration using Eq. (4):

$$max_{ij} = \left\{ \left| U_{ij}^{k+1} - U_{ij}^{k} \right| \right\} < \varepsilon \tag{3.4}$$

where *k* is the number of iterative steps and ε is the error threshold. The implication of the above equation is that the level of affiliation will not change significantly as the iterations continue. That is, it is assumed that the degree of affiliation is unchanged, and the comparative optimal (local or global optimal) state has been reached.

A simple threshold on a distance approach might not apply in multi-resident environments because such classification is too rigid. The Fuzzy C-means method can help to determine the sensor partitioning by the percentage of sensors belonging to a certain cluster. This percentage clustering allows for more flexibility in determining whether the two sensor events before and after can be triggered by the same resident, and thus whether to place them into a sequence. In multi-resident AR, we define the input data collected from different kinds of ambient sensors and generate a sequence of sensor events. Each sensor event takes the form $e = < d, t, s, v >$ where *d* is the date when the sensor data is collected, *t* is the time stamp in this date, *s* is the type of the sensor, and *v* is the value of the sensor.

We presume that such activities can be detected from these sensor events when each resident's action is performed, and for parallel activities, we assume that activities occur in different locations as one place can only be occupied by one resident, and collaborative activities are performed in the same place. To recognize these activities from continuously sensor events, we generate some subsequences of sensor events, and each activity is treated as a sequence of *n* sensor events $<e_1\ e_2 \cdots, e_n>$. To divide the actions performed by different residents into different subsequences, we introduce sensor coordinates (Figure 3-3) as a feature and use the Fuzzy C-means method for unsupervised clustering to realize online learning, and the newly generated sensor event $e = < d, t, s, v, c >$ where *c* is the cluster area of sensors. We then compare the two sensor events at times t_i and t_{i+1}.

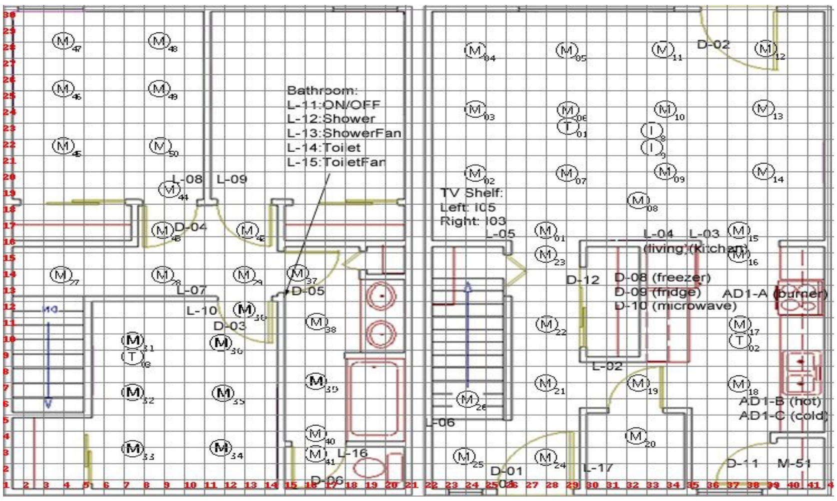

Figure 3-3 Sensor coordinates of CASAS datasets [44]

We allocate two sensor events into different subsequences if the locations of the two sensors belong exactly to two different clusters, otherwise we treat the upper and lower sensor events as the same sequence. For instance, in time *ti* to *ti*+5, the sensor triggers M20, M19, M21, M19, M18 respectively. Our Fuzzy C-mean clustering results reveal that all five sensor locations belonged entirely to one cluster, a situation in which it is likely that a single resident has triggered these sensors or that multiple residents have performed collaborative activities at the same location, so we allocate these sensor events into one subsequence. However, when the time is ti+6, sensor M05 is triggered and compared to the previous sensor position, we find that M18 and M5 are assigned to different clusters, which means that it is not possible that the same inhabitant just triggers this sensor, so we allocate this sensor event into another subsequence. In this way, we can divide the entire sensor event into different subsequences in the case of two residents. Therefore, for any incoming sensor events $\mathrm{S} = \{S_1, S_2 \dots S_n\}$, if sensors S_i and S_{i+1} are not in the same cluster, the sensor S_i and sensor S_{i+1} are segmented to different subgroups.

3.3.2 Change Point Detection (CPD)

After completing location-based subsequences, we apply a CPD technique called SEP, proposed in [63] to detect further whether there is an activity transition between two continuous windows on different subsequences, respectively, then analyze the performance based on average accuracy.

Change point detection (CPD) is a problem of identifying time points when the probability distribution of a time series event changes [63]. Separation change point detection (SEP) is a non-parametric sequential change detection that calculates the change score between two consecutive sensor events to represent the amount of change from the first sensor event sequence to the next sensor event. SEP employs separation distance as a dissimilarity measure to estimate the probability distribution ratio. The separation distance S between two continuous change point events can be calculated as shown in Eq. (3.5):

$$S = \mathrm{Max}_i \left(1 - \frac{f_{t-1}^i\,(x)}{f_t^i\,(x)}\right) \tag{3.5}$$

where $f_{t-1}^i\,(x)$ and $f_t^i\,(x)$ are the estimated probability density of the previous and current view, each with length m. Variable i iterates over all data points in the change point of view. Meanwhile, a Gaussian kernel function $g_i(x)$ is applied to estimate two probability densities by calculating separation distance as a change point score without estimating the densities $f_{t-1}^i\,(x)$

and $f_t^i\ (x)$ using Eq. (3.6):

$$g(x) = \frac{f_{t-1}\ (x)}{f_t\ (x)} = \sum_{i=1}^{m} \theta_i \prod_{j=1}^{m} K\left(x_t^i, x_{t-1}^j\right) \tag{3.6}$$

K is a non-negative basis function in the model and θ will be determined from data samples. [102] Then change points can be detected by using the score calculated in Eq. (3.7):

$$\widehat{SEP} = Max\left(0, \left(1 - \frac{1}{m}\sum_{i=1}^{m} g_i(x)\right)\right) \tag{3.7}$$

The larger score signifies a greater change point probability. All the candidate points are rejected when SEP scores are less than a threshold value. Local peak scores are the only parameter we take into account to reduce the false warning and avert reduplicative change points. The threshold value is selected according to optimal performance for a specific time series. In our experiment setting, a threshold value is chosen by optimizing a balanced value between True Positive Rate (TPR) and False Positive Rate (FPR) measures based on the training data. Another significant parameter m, the length of the view is identified that a threshold value of 0.3 is efficient and effective for non-intrusive ambient sensor events [63].

It is essential to identify resident' activity labels in real-time in evaluating recognition algorithms. The transition detection we use needs to consider the state at t_{i-1} as well as at t_{i+1} to determine whether a transition has occurred at t. The accuracy of transition detection has a considerable impact on the segmentation of sensor events and thus on multi-resident activity recognition. A larger false positive rate will split the time series sensor events into more segments and thus shorter activities.

```
Input
    A sequence of incoming Sensor Event: S={S_1,S_2···S_n}
    Addj =: initial status of the sensor enent S_j
Output
    A set of Sensor Segmentation: Seg={Seg_1,Seg_2···Seg_m}
Method
    Fuzzy C mean Check: FCM(S_i,S_j)>(True:False)
    CPD Check: CPD(S_k,S_p)>(True:False)
Algorithm
    WHILE S_j && Add_j = = false
            IF Seg not exist THEN
               Initial Seg_1
               Add S_j to Seg_1
               Addj=true
            ELSE
                  FOR Seg_i i=1 to m
                      IF Seg_i not Empty THEN
                            Read the last sensor event of Seg_i: Seg_i.last();
                            Read the first sensor event of Seg_i: Seg_i.first();
                            fcm_1=FCM(S_j, Seg_i.last())
                            cpd_1=CPD(S_j, Seg_i.last())
                            IF fcm_1 && cpd_1 THEN
                                        Add S_j to Seg_i
```

```
                                    Add_j=true
                        ELSE IF fcm_1 && !cpd_1 THEN
                            Process Seg_i
                            clear Seg_i
                            Add S_j to Seg_i
                            Add_j=true
                        END IF
                END FOR
                IF Add_j == false THEN
                    Initial Seg_m+1
                    Add S_j to Seg_m+1
                END IF
            END IF
        END WHILE
```

Figure 3-4 Proposed sensor event segmentation algorithm

As a summary, Figure 3-4 describes the pseudo code of the proposed segmentation algorithm. The proposed algorithm disposes real-time streaming sensor events. The proposed algorithm includes two methods: Fuzzy C-means check (FCM) and change-point detection check (CPD). Giving an incoming sensor event S_j, both FCM and CPD will be used with all the existing segmentations. Afterwards, S_j will be added into the segmentation based on the FCM and CPD results. However, a new segmentation S_j will be initialized if the existing segmentations cannot be identified.

The typical segmentation methods in existing works can be grouped into two categories: static-based windows and dynamic-based windows [103]. We can determine the best performer among these segmentation and classification models. Firstly, we segment the data by the time-based and event-based window methods and evaluate the effect of segmentation on the classification performance. Then, we examine the extent to which segmenting the dynamic sensor events improves performance. Finally, we evaluate our proposed segmentation method on different ML and DL classification models.

3.4 Experiments and Results

This section provides details on our experiments and analyzes the experimental results. Section 3.4.1 describes our experimental setup. Sections 3.4.2 and 3.4.3 report the outcomes of the different models with the static sliding windows approach, and with the dynamic sensor event segmentation method, and compares our proposed segmentation technique with other state-of-the-art approaches. Section 3.4.4 compares our proposed segmentation technique with other state-of-the-art approaches in a statistical method. Section 3.4.5 illustrates the limitations of the approach.

3.4.1 Experiment Setup

1) Datasets

The performance of the segmentation and classification methods were evaluated on the CASAS multi-resident public datasets [44]. These datasets include the data from ambient sensors (motion sensors, temperature sensors, door sensors, and other sensors placed on objects). The CASAS group collected several multi-resident activity datasets: "twor.2009", "twor.summer.2009", "twor.2010", "tulum", "tulum2", “cairo10” and "Multi-resident ADLs” [7], from the smart apartment testbed at Washington State University. The streaming sensor data store several features, and each sensor event is formatted as {date, time, sensorID, sensorValue, Resident, Activity}, which can be seen in Figure 3-5. The characteristics, sensors, and activities in the chosen CASAS multi-resident datasets are listed in Table 3-1, Table 3-2, and Table 3-3, respectively. The chosen datasets were “towr.2009”, “towr.summer.2009”, “towr.2010” and “Multiresident ADLs”, hereafter expressed as Dataset 1, Dataset 2, Dataset 3 and Dataset 4, respectively.

Table 3-1 CASAS Multi-Resident Datasets [7]

Dataset	# of Residents	Duration	# of Sensors	# of Activities	Sensor events
"twor.2009"	1 pair	2 months	71	9	137789
"twor. summer. 2009"	1 pair	2 months	86	8	772544
"twor.2010"	1 pair	2009 -2010 academic year	87	13	2804813
"Multi-resident ADLs"	26 pairs	Spread over 2 months	37	15	17258

Table 3-2 Sensors of the CASAS Multi-Resident Datasets [7]

"twor.2009" Dataset 1	"twor.summer 2009" Dataset 2	"twor.2010" Dataset 3	"Multi-resident ADLs" Dataset 4
- Motion Sensor - Item Sensor - Door Sensor - Burner Sensor - Hot Water Sensor			- Motion Sensors - Item Sensors - Cabinet Sensor - Burner Sensor - Phone Sensor

- Cold Water Sensor - Temperature Sensors - Electricity Usage	- Temperature

Table 3-3 Activities in the CASAS Multi-Resident Dataset [44]

"twor.2009" **Dataset 1**	**"twor.sum-mer. 2009"** **Dataset 2**	**"twor.2010"** **Dataset 3**	**"Multi-resident ADLs"** **Dataset 4**
-Clean -Meal preparation -Bed to toilet -Personal hygiene -Sleep -Work -Study -Wash bathtub -Watch TV	-Bed to toilet -Cleaning -Cooking -Grooming -Shower -Sleep -Wake up -Work.	-Bathing -Bed to toilet -Eating -Enter home -Housekeeping -Leave home -Meal preparation -Personal hygiene -Sleep -Not sleeping in bed -Wandering in room -Watch TV -Work	-Filling medication dispenser -Hanging up clothes -Reading magazine -Sweeping floor -Setting the table -Watering plants -Preparing dinner -Moving furniture -Playing checkers -Paying bills -Gathering and packing -picnic food

Date	Timestamp	Sensor	Value	Resident	Activity
2008-11-10	14:28:17.986759	M22	ON	2	2
2008-11-10	14:28:18.78605	M19	ON	1	1
2008-11-10	14:28:19.551189	M23	ON	2	2
2008-11-10	14:28:20.048559	M18	ON	1	1
2008-11-10	14:28:20.615459	M01	ON	2	2
2008-11-10	14:28:22.23544	M19	OFF	1	1

Figure 3-5 An example of a dataset that consists of time-stamped sensor events with activity label[7]

2) Experimental Settings and Implementation

As different actions have different durations, ranging from a few seconds to several hours, in our evaluation, we varied the time parameter from 10 to 1800 seconds, and the number of sensor events from 10 to 100, in the AR results. The parameters used in the static-sliding window approach are listed in Table 3-4.

Table 3-4 Parameters of the Static Sliding Window

Time-based	Sensor-based

10 seconds	10 sensor events
30 seconds	20 sensor events
60 seconds	30 sensor events
300 seconds	50 sensor events
600 seconds	100 sensor events
900 seconds	
1800 seconds	

A suitable window-size can be obtained from the sensor and time correlations using a dynamic sensor event segmentation approach. Different segments can be established by computing the Pearson product-moment correlation (PMC) between pairs of sensor events. When the PMC value is +1, 0, and -1, the sensor pair X and Y are entirely positively correlated, uncorrelated, and wholly negatively correlated, respectively. The PMC is calculated by Eq. (3.8) [21]:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\big((X-\mu_X)(Y-\mu_Y)\big)}{\sigma_X \sigma_Y} \tag{3.8}$$

where $cov(X,Y)$ refers to the covariance of X and Y, μ_X (σ_X) and μ_Y (σ_Y) denote the mean (standard deviation) of X and Y, respectively, and E is the expectation value.

Time correlation detects the time dependence between two sensor events. Sensor events can be described as $s = < s_1 \, s_2 \cdots s_n >$. Each sensor event includes the information T_i, S_i, and V_i, referring to the i_{th} time stamp of the sensor events, the i_{th} sensor in the sequence, and the value of Si, respectively. The first and last timestamps are denoted as T_{first} and T_{last}, respectively. The sensor correlation is thus calculated by Eq. (2.1), and the time correlation is calculated using Eq. (2.9) and (2.10) [15]:

$$T\frac{first}{cor}(T_i, T_{i-1}) = \frac{Time_{Distance}}{Maximum_{Time_{Span}}} = \frac{T_i - T_{i-1}}{T_{max}} \tag{3.9}$$

$$T\frac{last}{cor}(T_i, T_{i-1}) = \frac{Time_{Distance}}{Maximum_{Time_{Interval}}} = \frac{T_i - T_{i-1}}{T_{max}} \tag{3.10}$$

For feature extraction, we used One-Hot encoding approach because the features in the CASAS datasets can be treated as discrete values. One-Hot encoding, also known as one-bit valid encoding, mainly uses N status registers to encode N states, each state by independent register bits, and only one valid at any time [104]. For example, for the sensor values, we have only two states, ON/OPEN or OFF/CLOSE, so we start with 1 for ON/OPEN and 0 for OFF/CLOSE. Following the principle of the N-bit state register to encode N states, the result of the process is (there are only two features here, so N = 2): ON/OPEN = 10 and OFF/CLOSE = 01.

As mentioned above, numerous classification models are available for multi-resident AR [7, 41]. To evaluate whether segmentation methods affect the accuracy of AR, we classified the activities in the CASAS datasets using various ML and DL models including NB, DBN, DT, SVM, HMM, CL-HMM, CNN, DNN, LSTM and GRU. These classification models were chosen because multi-resident recognition has been improved by modern hybrid models and other sophisticated models such as CL-HMM, PHMM, and DBN [44], fueled by the recent popularity of deep learning methods such as LSTM, GRU [105]. If segmentation alters the performances of the chosen baseline and the state-of-the-art models, it will exert similar impacts on other multi-resident recognition models.

For the traditional machine learning classification models, the results were evaluated by 10-fold cross-validation on all the datasets, and test different parameters and kernels to get the best performance. For example, for HMM, we selected the best models based on the Laplacian smoothing factor. The parameters are selected from 10-6 to 10-2 in log-space [76]. For deep learning models, we used a grid-like search on hidden units within {10, 50, 100, 500, 1000}, and learning rate from 0.0001 to 1 in log-space [16]. We applied a stochastic gradient descent with early stopping on training the deep learning models and added a dropout layer to avoid overfitting problems [72]. Meanwhile, we repeated each experiment on deep learning models 100 epochs.

3.4.2 Static Sliding-Window Segmentation

To evaluate the influence of the segmentation method, we performed additional experiments using the time-based and sensor-based static sliding window techniques. As mentioned above, the time-based method divides a sequence of sensor events into different groups with equal time intervals, and the sensor-based approach partitions the sequence into several chunks with the same number of sensors [56]. The essential task of static segmentation is optimizing the time interval or sensor number. For this purpose, we set up parameter ranges for classifying the activities in the selected datasets.

Our selected datasets included many daily activities (see Table 3-3). These actions range from momentary activities that are completed in a few seconds or minutes, such as rising from bed to long-term activities that consume several hours or even longer. Thus, in the time-based segmentation, the partition interval t was ranged from 10 to 1,800 seconds. The effect of time-based segmentation was illustrated on the same classification models including NB, DBN, DT, SVM, HMM, CL-HMM, CNN, DNN, LSTM and GRU. The results of the static time-based segmentation technique for each t are shown in Table 3-5. This table reports the average accuracy of each evaluation classification under the specified parameter, determined by cycling all training sequences in 10-fold cross-validation.

Table 3-5 Parameters and Results of the Static Time-based Sliding Window

Results		Overall accuracy (%)						
Dataset	Classifier	t = 10	t = 30	t = 60	t = 300	t = 600	t = 900	t = 1800
		t = period of time in seconds						
Dataset 1	NB	83.31	89.65	91.74	**91.77**	91.09	91.13	88.14
	DBN	88.01	92.85	**93.56**	92.81	92.10	91.65	88.68
	DT	78.58	84.84	86.38	90.03	**91.18**	90.68	88.43
	SVM	80.43	85.45	90.43	**91.32**	90.74	89.86	87.44
	HMM	86.53	90.83	**92.47**	91.45	91.56	90.89	88.34
	CL-HMM	84.63	89.46	91.53	**93.55**	90.34	88.60	83.64
	CNN	91.24	92.34	**96.73**	96.69	95.78	94.77	92.38
	DNN	93.42	94.98	97.12	**97.67**	96.15	95.89	93.67
	LSTM	94.65	95.54	**97.82**	97.42	97.62	96.89	94.76
	GRU	94.34	94.78	**96.87**	95.34	95.34	92.54	92.65
Dataset 2	NB	90.68	92.93	93.93	**94.16**	93.87	93.81	92.68
	DBN	91.70	93.93	**94.54**	**94.54**	94.04	93.93	92.98
	DT	80.91	81.01	83.23	94.50	95.19	**95.70**	95.00
	SVM	86.88	88.53	**92.42**	92.31	91.84	90.52	90.78
	HMM	91.85	94.28	94.86	**94.89**	94.53	94.23	93.64
	CL-HMM	90.74	92.94	**95.32**	94.65	92.12	90.15	89.84
	CNN	95.55	96.76	97.56	**97.89**	96.77	95.66	93.35
	DNN	96.43	95.66	**97.94**	96.65	96.75	95.76	92.19
	LSTM	96.22	96.76	**98.53**	98.22	97.13	96.75	94.66
	GRU	94.64	95.73	97.54	**98.51**	95.76	95.42	95.64
Dataset 3	NB	90.73	94.38	95.81	**95.99**	94.94	94.46	93.39
	DBN	93.46	96.24	**96.62**	96.28	95.22	94.73	93.71
	DT	78.49	85.27	90.25	93.24	**95.76**	93.79	92.09
	SVM	86.84	90.53	92.32	**93.94**	93.42	92.13	92.10
	HMM	90.89	95.34	95.58	**96.23**	95.03	94.56	93.51
	CL-HMM	88.54	92.43	95.63	**96.14**	95.56	94.53	92.93
	CNN	95.12	95.23	96.14	**96.17**	95.79	93.44	91.65
	DNN	95.78	95.82	**96.58**	96.56	95.96	94.55	92.90
	LSTM	96.56	96.66	97.02	**97.26**	96.88	95.22	93.10
	GRU	95.74	**97.64**	96.43	96.86	95.61	92.20	91.18

Dataset 4	NB	53.46	54.50	**52.91**	45.65	43.02	41.38	35.38
	DBN	73.16	**73.70**	72.07	66.18	64.66	62.31	56.89
	DT	29.98	29.98	29.98	29.98	43.91	**43.54**	39.02
	SVM	65.14	67.63	**68.83**	67.33	63.24	63.10	61.49
	HMM	65.15	65.29	**63.98**	58.45	57.64	54.38	49.56
	CL-HMM	79.57	81.32	**82.14**	81.15	81.85	81.00	80.52
	CNN	80.14	80.15	80.32	**80.76**	80.12	79.89	79.86
	DNN	82.62	82.66	83.05	**83.12**	82.89	81.19	80.80
	LSTM	83.67	83.80	**83.76**	83.18	83.08	82.55	82.13
	GRU	78.94	80.12	**81.43**	80.53	81.21	80.80	80.45

As shown in Table 3-5, the static time-based window segmentation usually improved the accuracy of the multi-resident activities, but the accuracy depended on the partition interval. In most cases, the accuracy was higher at t = 60 and t = 300 seconds. For example, on Dataset 1, LSTM achieved 97.82% accuracy at t = 60 seconds, whereas DNN achieved 97.67% accuracy at t = 300 seconds. The timeframes of the activities varied widely: the short-term actions were well recognized within a short interval, but the long-term activities were more accurately recognized in a 900- or 1,800-second range.

Each dataset includes both immediate and long-term actions. Thus, moderate settings improved the accuracy of the AR, indicating that the optimal window of the static time-based window approach lay in this vicinity. The exception was the Decision Tree model, which achieved the best performance at t = 600 seconds on three datasets, and at t = 900 seconds on the remaining dataset. Therefore, we speculate that the parameters of the optimal window depend not only on the length of the time-division but also on the selected algorithm.

From the above experimental results, we concluded that dividing the time into static-time periods primarily affects the accuracy of the multi-resident AR. Specifically, we see that on Dataset 1, the LSTM algorithm achieves an excellent accuracy of 98.23% at t = 60 seconds. The DT, which has the lowest accuracy, also achieved 91.18% at t = 600 seconds. At the same time, we also found that different time intervals have a significant impact on the accuracy of behavior recognition. Using DT on Dataset 1 as an example, when t = 10 seconds, the accuracy is only 78.58%, while when t = 600 seconds, the accuracy improves to 91.18%, with a difference of 12.6%. On Dataset 2 as well as on Dataset 3, we found that all models achieved quite excellent results. The LSTM achieved 98.53% accuracy on Dataset 2 and the GRU achieved 97.64%

accuracy on Dataset 3. The high accuracies on Dataset 2 and 3 can be explained by the type of activities in Dataset 2 and 3. As seen in Table 3-3, these are simple everyday-life actions not performed concurrently by single individuals, nor collaboratively in a group. Instead, the simple actions of one individual did not affect another person. However, we found that on Dataset 4, all models underperformed. The highest accuracy was achieved by the LSTM algorithm with 83.76% accuracy, while the lowest accuracy was achieved by the NB algorithm with only 52.91% accuracy.

Table 3-6 Parameters and Results of the Static Sensor-Based Sliding Window

Results		Overall accuracy (%)				
Dataset	Classifier	s = 10	s = 20	s = 30	s = 50	s = 100
		s = the number of sensor events				
Dataset 1	NB	76.17	**76.26**	**76.26**	76.17	75.94
	DBN	**84.01**	83.87	83.36	83.03	82.06
	DT	**81.06**	80.75	80.42	79.63	78.59
	SVM	**80.65**	80.21	80.10	79.56	79.21
	HMM	**83.26**	81.67	81.89	82.13	80.56
	CL-HMM	81.32	**84.24**	84.00	83.13	82.42
	CNN	**88.45**	88.21	87.89	87.12	86.89
	DNN	**87.57**	87.62	87.32	87.02	86.91
	LSTM	89.32	**89.69**	88.92	88.32	88.17
	GRU	87.43	**88.58**	87.25	86.74	86.89
Dataset 2	NB	87.22	**87.23**	87.18	87.21	87.15
	DBN	**95.20**	95.09	95.04	94.81	94.42
	DT	96.22	**96.32**	96.12	95.23	94.97
	SVM	94.45	**94.84**	94.36	93.78	92.42
	HMM	**94.73**	94.32	94.66	95.12	95.10
	CL-HMM	93.65	**94.45**	93.14	93.64	93.54
	CNN	**95.78**	95.20	95.18	95.08	94.92
	DNN	**96.76**	96.60	96.59	96.52	96.38
	LSTM	**97.24**	97.04	97.11	97.15	96.94
	GRU	96.85	**97.53**	96.35	96.53	95.64
Dataset 3	NB	83.96	**83.98**	83.93	83.95	83.86
	DBN	**96.06**	95.87	95.75	95.48	94.56
	DT	94.26	**94.45**	94.14	93.99	93.58
	SVM	92.84	**93.29**	93.03	92.84	92.21
	HMM	**95.64**	94.37	94.58	95.31	94.52
	CL-HMM	**96.23**	96.20	95.52	96.02	93.92
	CNN	96.12	**96.43**	96.29	95.21	94.98
	DNN	**96.56**	96.52	96.34	96.22	95.92
	LSTM	**97.43**	97.39	96.66	96.35	95.89

	GRU	95.64	**96.26**	94.67	95.00	94.10
Dataset 4	NB	32.94	32.92	**32.95**	32.83	32.80
	DBN	**70.61**	70.39	70.19	69.99	66.13
	DT	**57.19**	55.44	55.79	56.60	50.78
	SVM	**63.84**	62.14	61.95	61.29	61.06
	HMM	**64.59**	64.12	64.09	62.15	58.47
	CL-HMM	**76.89**	75.75	74.67	74.23	73.43
	CNN	**80.53**	80.32	80.16	80.04	79.99
	DNN	**84.86**	83.05	83.54	82.89	82.23
	LSTM	**84.93**	84.25	84.16	84.06	83.53
	GRU	**83.57**	82.19	83.56	83.10	82.59

The second part of our static sliding-window segmentation experiment explored the static sensor-based approach. The performance of the sensor-based method was evaluated on the same datasets using the same classification methods as the static time-based approach [39]. The number of sensor events was ranged from 10 to 100. The results are shown in Table 3-6. The classification methods delivered their best performance for 10 or 20 sensor events. These results show that the parameter range of the optimal window, providing the highest performance on each dataset, was similar among the algorithms. The sensor-based window segmentation significantly improved the accuracy in most cases. The optimally segmented LSTM achieved 97.43% accuracy on Dataset 3. However, the results of Dataset 4 remained poor. The performance of NB was catastrophic; after segmentation, the accuracy of this algorithm was only 32.95% on Dataset 4.

Both data segmentation methods improved the accuracy of activity recognition. To determine a more practical approach, we compared the results of the time-based and sensor-based methods. The average accuracy of the two segmentation methods were compared on the different datasets, and the results are shown in Figure 3-6. Observing this figure, the static time-based window approach performed slightly better than the static sensor-based window approach, and the results on Datasets 1, 2, 3 and 4 were 9.92%, 0.98%, 1.6%, and 1.42%, respectively, illustrating that the different segmentation method yields different results on different datasets. Although the time-based segmentation method outperformed the sensor-based method in the present experiments, we cannot judge that the time-based approach is better than the sensor-based approach, because we tested specific parameters in the four datasets. Whether these parameters define the optimal windows for each dataset is unconfirmed. We can only confirm that both methods improve the accuracy of multi-resident AR.

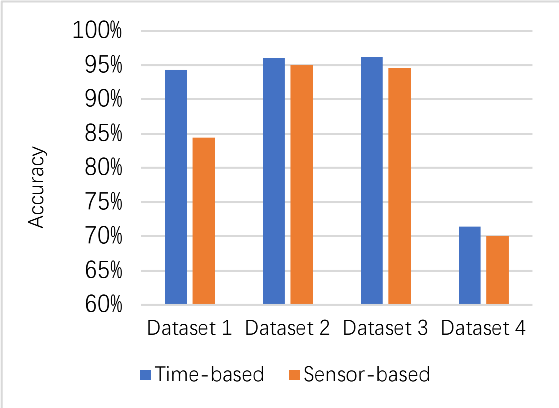

Figure 3-6 Comparison of average accuracy improvements after time-based and sensor-based static sliding window segmentations

3.4.3 Dynamic Correlation-based Window Segmentation

As shown in the previous two experiments, segmentation can significantly improve the accuracy of multi-resident AR, but choosing the optimal parameters is a difficult task. Therefore, we dynamically segmented the sensor events based on the sensor-correlation and time-correlation algorithm proposed in [15]. In this experiment, the sensor and time correlations were calculated simultaneously, and the sensor events were partitioned into different groups. The approach was verified by cycling the training series through 10-fold cross-validation [48]. The accuracies of the four classification algorithms after dynamic correlation segmentation are presented in Table 3-7.

Table 3-7 Dynamic Correlation Segmentation

Datasets	Approach	Accuracy	F-measure
Dataset 1	NB	93.33	92.49
	DBN	94.50	93.81
	DT	91.45	90.80
	SVM	92.42	92.32
	HMM	93.89	92.72
	CL-HMM	94.32	93.94
	CNN	97.13	96.78
	DNN	97.76	96.87
	LSTM	97.82	97.26
	GRU	**98.53**	**97.75**
Dataset 2	NB	93.19	89.16
	DBN	96.53	91.13
	DT	97.64	92.22
	SVM	95.75	95.43
	HMM	96.89	91.96
	CL-HMM	97.32	96.86

	CNN	98.02	97.76
	DNN	98.09	97.67
	LSTM	**98.23**	**98.05**
	GRU	98.14	98.02
Dataset 3	NB	94.57	94.32
	DBN	96.58	96.47
	DT	93.34	94.30
	SVM	94.24	94.02
	HMM	95.37	95.30
	CL-HMM	95.87	95.32
	CNN	96.89	95.89
	DNN	96.92	96.05
	LSTM	97.64	96.97
	GRU	**98.24**	**97.96**
Dataset 4	NB	61.59	62.17
	DBN	77.46	76.40
	DT	60.02	60.59
	SVM	69.85	65.28
	HMM	70.45	70.85
	CL-HMM	83.53	83.21
	CNN	85.89	85.82
	DNN	87.78	86.87
	LSTM	89.53	89.12
	GRU	**89.84**	**88.83**

As revealed in Table 3-7, all segmented algorithms achieved high accuracies and F-measures on Datasets 1 to 3. These classification models accurately identified multiple activities, and their algorithms were relatively stable (as evidenced by F-measures). The accuracies of all classification models exceeded 90% on the first three datasets, ranging from 91.45% (DT on dataset 1) to 98.53% (GRU on Dataset 1). However, the accuracies of all methods were much lower on Dataset 4 than on the other three datasets. The highest and lowest accuracies were 89.84% (GRU) and 60.02% (DT), respectively. Figure 3-7 shows the accuracy improvements of all the algorithms on the four datasets after the dynamic correlation-based window segmentation comparing with static time-based segmentation. The improvement after dynamic data segmentation was especially apparent on Datasets 1 and 4. The dynamic method achieved higher accuracies than either of the static methods and conserved the runtime by reducing the number of configurations of different parameters.

Through the experiments described above, we found that even when using the general segmentation methods, we were able to achieve incredible results on Datasets 1 ,2, and 3, which is nearly impossible for recognizing multiple complex actions, especially parallel and collaborative activities. So, we also investigated the issues by examining the dataset itself. After studying the dataset, we found that although Datasets 1 to 3 are multi-resident datasets, the activities in them are relatively simple, and most of the time they are sequential actions of a single person, in which one person finishes one job, and another person does another job. There are rarely parallel actions and much fewer collaborative activities, so they are closer to the identification of sequential actions of a single person.

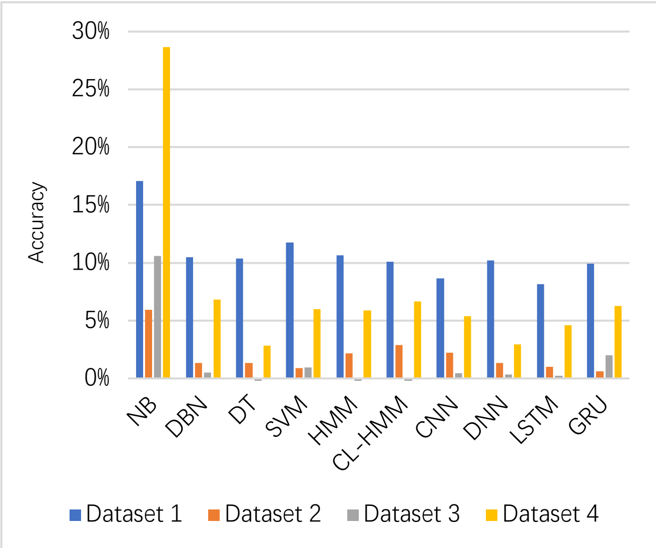

Figure 3-7 Accuracy improvement in the dynamic correlation-based window segmentation (relative to static time-based segmentation)

Table 3-8 Comparison of the Proposed Method with Other Segmentation Methods

Results		Overall accuracy (%)			
Dataset	Model	Time-Based Sliding Window	Sensor-Based Sliding Window	Dynamic Correlation Segmentation	Proposed Segmentation
Dataset 1	NB	91.77	76.26	**93.33**	92.78
	DBN	93.56	84.01	94.50	**97.39**
	DT	91.18	81.06	91.45	**95.19**
	SVM	91.32	80.65	92.42	**96.32**
	HMM	92.47	83.26	93.89	**97.63**
	CL-HMM	93.55	84.24	94.32	**96.42**
	CNN	96.73	88.45	97.13	**98.58**
	DNN	97.67	87.57	97.76	**98.41**
	LSTM	97.82	89.69	97.82	**98.22**

	GRU	96.87	88.58	98.53	**98.89**
Dataset 2	NB	94.16	87.23	93.19	**95.64**
	DBN	94.54	95.2	**96.53**	96.48
	DT	95.7	96.32	97.64	**98.09**
	SVM	92.42	94.84	95.75	**96.89**
	HMM	94.89	94.73	96.89	**97.34**
	CL-HMM	95.32	94.45	97.32	**98.53**
	CNN	97.89	95.78	**98.02**	**98.02**
	DNN	97.94	96.76	98.09	**98.54**
	LSTM	98.53	97.24	98.23	**98.98**
	GRU	98.51	97.53	98.14	**98.87**
Dataset 3	NB	95.99	83.98	**94.57**	94.39
	DBN	96.62	96.06	96.58	**98.53**
	DT	95.76	94.45	**93.34**	93.29
	SVM	93.94	93.29	94.24	**97.32**
	HMM	96.23	95.64	95.37	**97.65**
	CL-HMM	96.14	96.23	95.87	**97.85**
	CNN	96.17	96.43	96.89	**97.01**
	DNN	96.58	96.56	96.92	**97.04**
	LSTM	97.26	97.43	97.64	**98.65**
	GRU	97.64	96.26	**98.24**	**98.24**
Dataset 4	NB	52.91	32.95	61.59	**67.23**
	DBN	73.70	70.61	77.46	**82.00**
	DT	43.54	57.19	60.02	**63.56**
	SVM	68.83	63.84	69.85	**75.99**
	HMM	63.98	64.59	70.45	**76.45**
	CL-HMM	82.14	76.89	83.53	**87.75**
	CNN	80.76	80.53	85.89	**89.65**
	DNN	83.12	84.86	87.78	**92.65**
	LSTM	83.76	84.93	89.53	**93.62**
	GRU	81.43	83.57	89.84	**94.53**

Despite the significant accuracy advances after dynamic data segmentation, the performance on Dataset 4 remained deficient. The activities are more involved in this dataset than in the other datasets. Some of the 15 actions in Dataset 4 are interleaved; a single resident concurrently performs others. More importantly, some activities (such as packing picnic food) are collaborative and involve multiple people. Therefore, the performance of the algorithms on this dataset are relatively low. Next, we use our proposed segmentation model to evaluate the multi-resident AR to judge whether our proposed model is effective, and compare with other state-of-the-art segmentation techniques with the best parameter to observe the performance. The results are shown in Table 3-8.

From the table, we can find that our proposed segmentation model is superior to any other segmentation techniques, regardless of which multi-resident dataset or which classification model. In Dataset 1 to Dataset 3, the accuracies of the model we proposed are higher than that of other models, but it is relatively insifigicant. The reason is that although the first three datasets are also multi-resident AR, the actions taken are relatively simple, and in most cases, there are no collaborative activities between two people, which makes them less difficult in recognition. We also see that the proposed segmentation technique combines with LSTM model acquired the best performance in Dataset 2, which is 98.98%

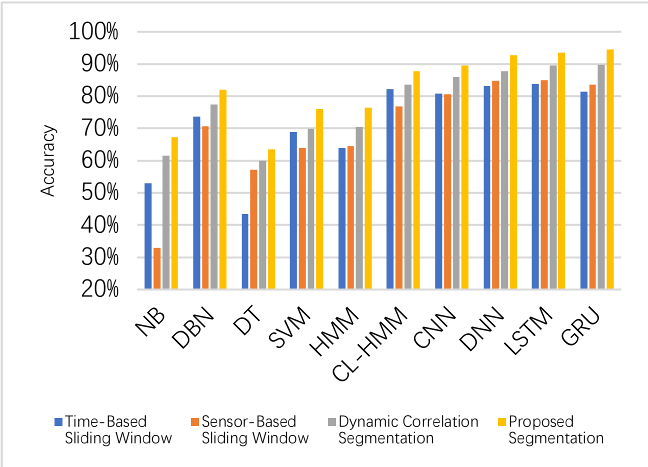

Figure 3-8 Accuracy of Segmentation Methods on Dataset 4

At the same time, we pay more attention to Dataset 4 (Figure 3-8), because in this dataset, residents' activities are more complex, including a variety of single-person activities and collaborative multi-resident activities, and the proposed model also greatly improves the accuracy of identification. Compared with the traditional machine learning models, the performance of the deep learning models is higher. CNN, DNN, LSTM, and GRU achieved 89.65%, 92.65%, 93.62%, and 94.53% respectively on this dataset. From the technical analysis, the proposed model can separate the activities of different residents according to the location of sensors, which makes it impossible to separate the activities that occur together into two separate sensor events. Although in the process of division, there will also be the activities of two residents who are very close together, such as collaborative activities, the change point detection algorithm can help us detect the occurrence of transition as well, which can improve the overall recognition performance in the multi-resident environment. What is more, we observe from Figure 3-9 that the F-measure results also illustrate the importance of segmentation techniques for multi-resident recognition and the stability of these segmentation techniques as evidenced by their relatively stable performance.

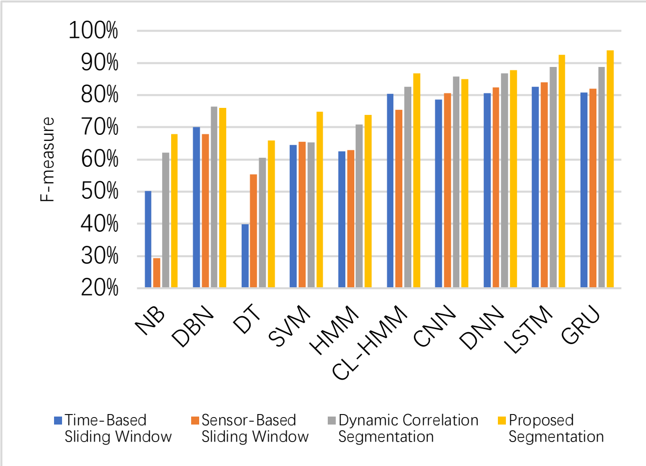

Figure 3-9 F-measure of Segmentation Methods on Dataset 4

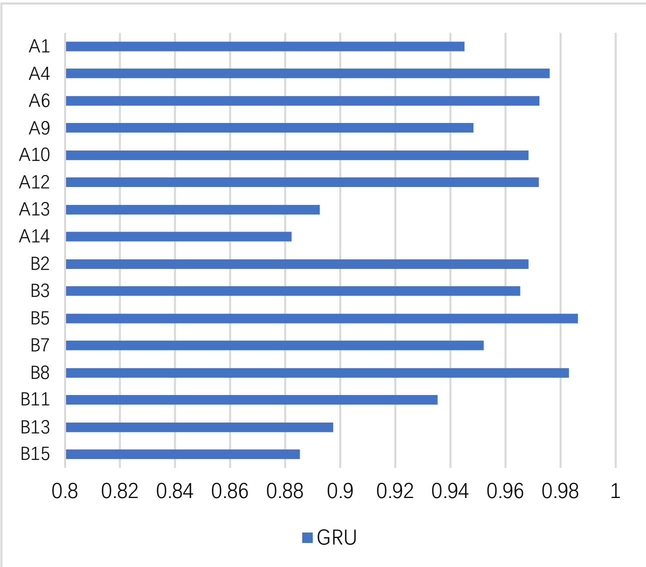

Figure 3-10 Results of accuracy per label by using GRU

Table 3-9 List of Activities Performed by the Multi-Resident

Resident A		Resident B	
1	Filling medication dispenser	2	Hanging clothes
4	Reading magazine 1	3	Moving furniture
6	Sweeping floor	5	Watering plants
9	Setting Table for Dinner	7	Playing checkers
10	Reading magazine 2	8	Preparing dinner

12	Gathering picnic food	11	Paying bills
13	Getting dishes from a cabinet	13	Getting dishes from a cabinet
14	Packing Supplies for picnic food	15	Packing picnic food

For multi-resident AR, the most important are parallel as well as collaborative activities. For Dataset 4, Table 3-9 lists the activities performed by resident A and resident B, respectively Figure 3-10 depicts the accuracy of each action identified using our proposed segmentation technique, and the GRU model was used as the classification model as this model achieved the highest accuracy. The results show that our proposed segmentation method can accurately segment the multi-resident sensor events and improve the performance of AR. For example, watering plants, preparing dinner, reading magazines, sweeping floor activities staying in a different location that is exclusive from other activities, so they are very well recognized. Conversely, activities like retrieving dishes from the cabinet, packing supplies for picnic food, if the actions occurring are in the same location or very close together, and the recognition will be slightly less accurate. A fundamental reason may be that when multiple people perform collaborative activities, the probability case is that the actions they perform are geographically the same or similar, and it is difficult for the segmentation algorithm to delineate precisely which inhabitant triggered the sensor, so we find that the accuracy of collaborative activities is lower than the accuracy of parallel activities.

3.4.4 Statistical comparison

Through the results, we can easily see that dynamic correlation segmentation and our proposed hybrid fuzzy C-means CPD-based method perform better comparing with the other two traditional static sliding windows approaches. In this section, we applied a Wilcoxon signed-rank test, which is a famous nonparametric test for paired or matched data to prove that our proposed method is superior in multi-resident AR than the dynamic correlation method [106]. The different scores are taken into account by analyzing the signed differences.

First of all, we compute the difference d_i between the accuracy in the segmentation of two algorithms on the four datasets. Secondly, these differences are ranked based on their absolute values. Table 3-10 displays various values of segmentation and ranks corresponding to the two methods. Next, we calculate R^+ and R^-, where R^+ is the sum of the ranks for positive and R^- indicates negative. The values of R^+ and R^- are 799 and 21, respectively. Based on critical values for the *Wilcoxon signed-rank* test with a confidence level of α = 0.05, the difference between the two algorithms is significant if the smaller of the sums is less than or equal to 218. Therefore, we can conclude that our proposed method is superior to the dynamic correlation method.

Table 3-10 Comparison of the Proposed Method with the Correlation Segmentation Method

	Model	Proposed	Correlation	Difference	Rank
Dataset 1	NB	92.78	93.33	-0.55	1
	DBN	97.39	94.5	2.89	26
	DT	95.19	91.45	3.74	29.5
	SVM	96.32	92.42	3.9	32
	HMM	97.63	93.89	3.74	29.5
	CL-HMM	96.42	94.32	2.1	23
	CNN	98.58	97.13	1.45	20
	DNN	98.41	97.76	0.65	14
	LSTM	98.22	97.82	0.4	10
	GRU	98.89	98.53	0.36	9
Dataset 2	NB	95.64	93.19	2.45	25
	DBN	96.48	96.53	-0.05	3.5
	DT	98.09	97.64	0.45	12
	SVM	96.89	95.75	1.14	18
	HMM	97.34	96.89	0.45	12
	CL-HMM	98.53	97.32	1.21	19
	CNN	98.02	98.02	0	5.5
	DNN	98.54	98.09	0.45	12
	LSTM	98.98	98.23	0.75	16
	GRU	98.87	98.14	0.73	15
Dataset 3	NB	94.39	94.57	-0.18	2
	DBN	98.53	96.58	1.95	21
	DT	93.29	93.34	-0.05	3.5
	SVM	97.32	94.24	3.08	27
	HMM	97.65	95.37	2.28	24
	CL-HMM	97.85	95.87	1.98	22
	CNN	97.01	96.89	0.12	7.5
	DNN	97.04	96.92	0.12	7.5
	LSTM	98.65	97.64	1.01	17
	GRU	98.24	98.24	0	5.5
Dataset 4	NB	67.23	61.59	5.64	38
	DBN	82.00	77.46	4.54	35
	DT	63.56	60.02	3.54	28
	SVM	75.99	69.85	6.14	40
	HMM	76.45	70.45	6	39
	CL-HMM	87.75	83.53	4.22	34
	CNN	89.65	85.89	3.76	31

DNN	92.65	87.78	4.87	37
LSTM	93.62	89.53	4.09	33
GRU	94.53	89.84	4.69	36

3.5 Conclusions and Future Work

This paper comprehensively evaluates the segmentation approaches that improve the performance of multi-resident AR in smart home environments. We evaluate and compare our proposed Hybrid Fuzzy C-means CPD-based Segmentation method with the well-known segmentation methods including static time-based, static sensor-based, and dynamic correlation-based segmentation. Our method can accurately segment multi-resident sensor events and significantly improve the overall performance (in terms of both accuracy and F-measure) of complex activities. At the same time, we also observe that the recognition performance of the parallel activities is higher than that of the collaborative activities. In the future, we plan to improve the dynamic sensor-events segmentation algorithm to evaluate our method in a more complex setting and a larger number of people involved by doing our own data collection.

Chapter 4 Transformer with Bidirectional GRU for Non-intrusive, Sensor-based Activity Recognition in a Multi-Resident Environment

Several techniques for Human Activity Recognition (HAR) in a smart indoor environment have been developed and improved along with the rapid advancement of sensor technologies. However, recognizing multiple people's activities is still challenging due to the complexity of their activities, such as parallel and collaborative activities. To address these challenges, we propose a Transformer with a Bi-directional GRU deep learning method, called TRANS-BiGRU, to efficiently learn and recognize different types of activities performed by multiple residents. We compare the proposed model with the state-of-the-art models and various deep learning (DL) models, such as Ensemble2LSTM (Ens2-LSTM), Bidirectional Gated Recurrent Units (Bi-GRU), and traditional machine learning (ML) models, such as Support Vector Machine (SVM). Our experimental results based on the CASAS and ARAS public datasets show that our model significantly outperforms the existing models for complex activity recognition of multiple residents.

Chapter 4 has been submitted to IEEE Internet of Things Journal.

4.1 Introduction

Our global aging trend is increasing, with the number and proportion of older adults increasing significantly in most countries around the world. By 2050, the number of people aged 60 will reach 2.1 billion [1]. As they grow older, many of them are likely to suffer from some geriatric diseases, such as dementia or Alzheimer's, which will make them more challenging to live independently [2]. Due to the shortage of caregivers and the imperfection of available healthcare infrastructure, many studies have begun to establish the assisted living system to help monitor activities of daily living (ADLs) [3, 4]. Especially, Human Activity Recognition (HAR) is a vital research area that focuses on identifying human actions based on the information obtained from various kinds of sensors used for activity monitoring [107].

There are two main types of methods for HAR: sensor-based and video-based [43]. Although video-based activity recognition is instrumental and well-studied in the field of human activity and behavior recognition, it is deemed not preferable for a smart home environment due to intrusiveness and privacy concerns [108, 109]. Therefore, we focus on non-intrusive sensor-based HAR, which can be divided into two sub-categories: wearable sensors (e.g., mobile phones and smart bracelets) and infrastructure sensors (e.g., motion sensors including Passive Infrared Sensors (PIRs), contact and temperature sensors) [110, 111]. Many studies used wearable sensors, which provide straightforward access to individual, personal details [112]. However, their shortcomings are apparent. Some people, especially elderlies, are unwilling to wear a body-attached sensor or often forget to do so, thus causing data loss and inaccuracy [69]. In contrast, infrastructure sensors can provide an advantage that individuals' privacy is not violated, thus achieving non-intrusive monitoring. For example, some studies placed PIR sensors on the ceiling or other sensors attached to an object in the environment (e.g., door sensors) [44]. With these non-intrusive ambient sensors, residents can live more safely, and their daily life is unaffected by the surrounding technology [45].

Most of the existing HAR studies have focused on single resident recognition in a smart home [4, 89, 113]. However, in a real-life situation, more than one person may stay in a home simultaneously, even if it is usually occupied by a single resident (for example, visited by family members or caregivers) [114]. HAR in a multiple residents (multi-resident) environment is more challenging than in a single-resident situation as a single person usually only performs sequential, interleaving, and concurrent activities [44]. In contrast, a multi-resident situation not only gets involved with these types of activities performed by the individuals, but they can also conduct parallel and collaborative activities [35]. A parallel activity refers to many activities performed by different people in the same living space at the same time. For example, one person is making tea in the kitchen while the other resident is making a phone call [7]. Collaborative activities are another type of activity in multi-resident recognition. It refers to multiple persons joining an effort together in a synergic manner so that each resident performs one same activity together (e.g., two people moving a dining-table together), or they work separately but to meet the same objective (e.g., two people are preparing dinner together in the kitchen [7].

The main challenges of multi-resident HAR include accurate recognition of every action and identifying a correct person who performed the activity. Several works have solved the problems of multi-resident HAR using data collected from non-intrusive, sensor-based technology. For example, in [7], a variant of a combined label approach based on a Hidden Markov Model (CL-HMM) and a linked version of HMMs (LHMM) were proposed for multi-resident HAR. In [12], a coupled Hierarchical Dynamic Bayesian Network (HDBN) was proposed. In [115], the latest deep learning-based technology CascadeEnsemble (CascEns-LSTM) and Ensemble2LSTM (Ens2-LSTM) were proposed to model multi-resident activities. Ens2-LSTM is the best among the others as it can achieve 94.24% accuracy on the Milan dataset.

Multi-resident activity recognition is still in infancy due to its complexity. Many challenges still exist in single residents and need to be addressed before concentrating on the multi-resident situation. In addition to this, there are no studies on complex multi-resident HAR using advanced deep learning models. Our research further advances the multi-resident HAR research focusing on recognizing multi-resident complex activities, which are the vital components in a multi-resident environment. Specifically, our paper makes the following contributions:

- We propose a new Transformer with a Bidirectional GRU deep learning method, called TRANS-BiGRU, to efficiently learn and recognize complex activities of multi residents.
- We evaluate our proposed model by comparing it with the state-of-the-art models, traditional machine learning (ML) methods, and other deep learning (DL) models proposed for multi-resident HAR. The results show that our TRANS-BiGRU can significantly achieve the better performance than the existing models.

The remainder of this paper is organized as follows. Section 4.2 introduces related works in recent years. Section 4.3 describes our proposed HAR method. Section 4.4 discusses different DL and ML models' evaluation performance based on the CASAS and ARAS public datasets. Section 4.5 concludes this chapter.

4.2 Related Work

In general, a typical multi-resident HAR framework includes data collection, data pre-processing, and classification. Most of the studies focus on various classification models to improve the performance of identification. This section discusses the existing works on sensor-based multi-resident classification models.

4.2.1 Traditional Machine Learning Models

Recently, many studies on multi-resident HAR in the non-intrusive sensor-based environment have been conducted. Yin et al. [9] applied the Naive Bayes (NB) classifier to recognize some simple daily activities of multi-resident such as watching TV and making meals. Some non-intrusive ambient sensors such as motion sensors, temperature sensors, power sensors, and humidity sensors were installed in a smart home environment. The experimental results show that NB has achieved the best accuracy of 73.01%. However, basic NB can be used to model some temporal processes. Therefore, Dynamic Bayesian Networks (DBN) is proposed to address the limitation [44].

Alam et al. [12] proposed a coupled Hierarchical Dynamic Bayesian Network (HDBN) for recognizing multi-resident activities. Some ambient sensors such as mobile phones and passive infrared sensors were installed in the smart home environments to collect daily activities. A real-world dataset CACE and CASAS dataset were used for evaluating the performance of the coupled HDBN model. The experimental results show that the proposed model can achieve 95% of accuracy and outperforms the recognition performance.

Prossegger et al. [67] proposed a decision tree-based algorithm named E-ID5R to recognize multi-residents' daily activities. Leaf nodes are added to the algorithm and can be multi-gagged. The algorithm E-ID5R gradually introduces decision trees to adapt to new instances and new collaborative activities over time. They evaluated the performance based on a real-world dataset ARAS and the experimental results show that the proposed E-ID5R model performs better and achieved 82% of accuracy.

Roy et al. [69] combined ambient sensor data and smartphone data to recognize daily activities. In this research, motion sensors were used to confirm the residents' location in the room. This location information was combined with smartphone data to recognize the residents' activity more accurately. They used CHMM because two different acceleration data streams were connected with different observation sequences. The concurrent activities of occupants were imposed spatiotemporal constraints by using ambient motion sensor data. They used two separate activity datasets in the experiment. The results prove that this modeling method increases complex activity recognition accuracy by more than 30% compared with only smartphone-based solutions.

Alemdar et al. [5] proposed a Factorial Hidden Markov Model (FHMM) to model the hidden observations using multiple activities sequence chains. In this research, the primary purpose is to recognize multi-residents' concurrent activities. Many non-intrusive ambient sensors collect the residents' daily activities information. The experimental results show that FHMM achieved a better result compare with HMM and CRF in some specific activities such as taking a shower, relaxing, and going outside.

Benmansour et al. [7] proposed a CL-HMM method to recognize multi-residents' collaborative activities and parallel activities in a smart home environment. The CL-HMM was proposed based on HMM theory and combined activity labels of each resident to produce the new labels. They compared the CL-HMM model to the other two models, CHMM and Parallel HMM (PHMM). The results show that CL-HMM achieved the best accuracy (88.03%) in the collaborative activities.

Tran et al. [116] proposed an ensemble of HMMs mixed-dependency models (MDM) for recognizing multi-resident activities. This hybrid model is a combination of six variants of the HMM model. The authors evaluated six models and a hybrid MEM model using CASAS dataset, and the experimental results show that the MEM was superior to the other HMM models. The authors also compared the MEM model with the deep learning models Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). The deep learning model achieved better performance than MEM.

Recently, Li et al. [35] utilized the Markov Logic Network classification approach to recognize residents' types in a multi-resident environment. The activity preference mainly includes duration and period, time sequence, and action events location preference. Based on the resident type (e.g., gender, age bracket, job), this approach performed further reasoning on the family role (e.g., father, mother, daughter, and so on). This combined data-driven and knowledge-driven method is effective in identifying user identification, especially in cases where the number of users is uncertain.

4.2.2 Deep Learning Models

Liciotti et al. [115] applied Long Short-Term Memory (LSTM) for modeling spatiotemporal sequences acquired by smart home sensors. They proposed two new deep learning-based models called CascadeEnsemble (CascEns-LSTM) and Ensemble2LSTM (Ens2-LSTM). Experimental results performed on the Milan dataset (one of the CASAS datasets) show that the proposed LSTM-based approaches outperformed existing DL and ML methods, which achieved accuracy 94.24% in the Milan dataset, giving superior results compared to the existing literature. However, the proposed model does not consider the data association, which means that it cannot recognize parallel and collaborative activities.

Ignatov et al. [117] proposed a solution for individual HAR problems based on the Convolutional Neural Networks (CNN). Some statistical features were used to evaluate the performance using the UCI and WISDM datasets. The results show that CNN based method can significantly improve the performance of recognizing simple individual activities such as jogging, walking, upstairs, downstairs, sitting, and standing. Especially in the UCI dataset, the overall accuracy reached 97.63%. However, this research only focuses on the individual and specific activities scenario. Whether the proposed model can be applied in a complicated multi-resident situation still need to be investigated.

Munkhjargal et al. [49] used some binary sensors and a Deep Convolutional Neural Network (DCNN) unobtrusive HAR classifier to recognizing the elderly's individual activities. They employed Aruba (CASAS) dataset to evaluate some daily activities recognition. In their experiments, eight activities were segmented using different sliding windows parameters, then the data were converted into activities' images, and the DCNN model was applied to classify the activities. The results show that the proposed DCNN model outperformed the existing HAR models with an F1-score of 95.1% (excluding Leave_Home and Wash_Dishes). Nevertheless, there are ten daily activities in the Aruba dataset, and for all activity recognition, they only achieved an F1-score of 79% in total.

Tran et al. [42] applied the LSTM and Gated Recurrent Unit (GRU) models to CASAS and ARAS multi-resident datasets for experimental comparison. The experiment identified resident A and resident B's activities separately, and the average accuracy was calculated. The experimental results show that the GRU model outperformed the LSTM in multi-resident HAR. However, the accuracy of the recognition is only 77.2%.

In summary, the existing works on multi-resident HAR utilize traditional machine learning techniques, while the application of deep learning techniques has been relatively sparse. We have summarized the advantages and disadvantages of these models, as shown in Table 4-1. Our proposed hybrid model leverages the strengths of both Transformer and Bi-GRU models to significantly improve recognition performance. In the following subsection, we will describe our proposed model in detail.

Table 4-1 Advantages and Limitations of Various Classification Models

Model	Dataset	Advantages	Limitations
NB [9]	Real-world data	The model is simple and runs data fast.	1. Only some of the simple activities of multiple residents were recognized. 2. Low performance in multi-resident activity recognition.
HDBN [12]	CASAS CACE (Real-world data)	It has a high performance in identifying shared activities (such as Move Furniture and Play Checker).	The problem of data association in multi-resident identification is not solved.
E-ID5R [67]	ARAS	E-ID5R can predict parallel and cooperative activities of multiple people	House A presents a quite challenging task. The classification rate is insignificantly as much low as 40%.
CHMM [69]	CASAS PUCK OPPORTUNITY	CHMM is effective in multi-resident collaborative activities.	The complexity of the model is increased, and the processing time of the training set is also increased.
FHMM [5]	ARAS	FHMM performs better for some activities such as taking a shower, being outside, and relaxing.	The computation time increase when increasing the number of residents.
CL-HMM [7]	CASAS	1. Solved the problem of data association. Turn the multi-label problem into a	The model is poorly scalable, and labels need to be re-labeled when the number of

		single-label problem. 2. CL-HMM performed the best in the case of collaborative activities.	residents is added.
LSTM [42]	CASAS ARAS	It has a long-time memory function, so it has a greater advantage for time series data like sensor events with multiple residents' activities and is easy to implement.	The excessive number of parameters increases the complexity of the model, and it takes longer to recognize multiple residents' activities than other models.
GRU [42]	CASAS ARAS	1. It has fewer parameters and faster activity recognition compared to LSTM. 2. The recognition performance of multiple residents is superior compared to LSTM.	The standard GRU network processes the sequence in chronological order, ignoring the future context. So there is still room for improvement in recognition performance.

4.3 Proposed Deep Learning Approach

In this section, we first illustrate the Transformer architecture. Then, we introduce the Bidirectional GRU network. Besides, we present our proposed Transformer with a Bidirectional GRU (TRANS-BiGRU) approach for HAR using non-intrusive sensors in a smart home environment.

4.3.1 Transformer Architecture (TRANS)

The Transformer mechanism was first proposed by Vaswani [118] in 2017. The traditional CNN and Recurrent Neural Network (RNN) are discarded in Transformer, and the whole network structure is entirely composed of an attention mechanism. More precisely, Transformer only consists of feed-forward and self-attention neural network. A Transformer-based trainable neural network can be built by stacking the Transformer. The authors' experiments were conducted by building Encoder-Decoder with six layers each of encoder and decoder for 12 layers and achieved a new high bilingual evaluation understudy (BLEU) value in machine translation. The reason why the authors adopted the Attention mechanism is that the computation of RNN (or LSTM, GRU, etc.) is restricted to be sequential, which means that the RNN-related algorithms can only be computed sequentially from left to right or from right to left, and this mechanism poses two problems. First, the computation of time slice t depends on the computation results at time $t-1$, limiting the parallelism capability of the model. Besides, the structure of gate mechanisms such as LSTM can only alleviate the problem of long-term dependence to a certain extent, but it does not solve the problem fundamentally.

The proposed Transformer solves the above two problems by first using the Attention mechanism to reduce the distance between any two positions in the sequence to a constant. Secondly, it is not an RNN-like sequential structure, so it has better parallelism and is compatible with existing GPU frameworks. A Transformer architecture has been used in many applications and has shown its advantages such as machine translation, sentence classification, and question answering. LSTM, GRU, and other RNN-based models have been shown to work well with temporal sequences of sensor data, and with Transformer is optimized on top of this, it combines the advantages of self-attention and RNN, and the attention mechanism has been shown to assist well in processing sensor data, so we believe that the Transformer helps improve the overall performance via its multi-head attention mechanism. Figure 4-1 shows the framework structure of the Transformer.

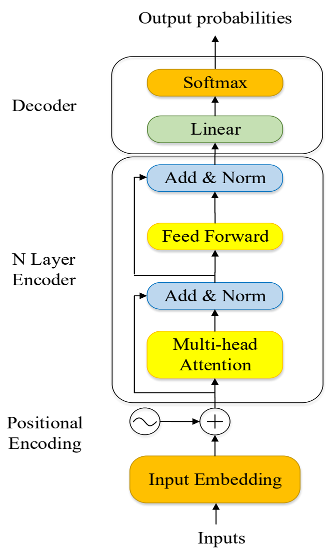

Figure 4-1 The Transformer architecture

The Transformer is essentially an Encoder-Decoder structure. An encoder is a stack of multiple identical layers. There are two sub-layers in each layer. First through a multi-headed self-focus mechanism, followed by a simple, positional, fully connected feedforward network. The decoder is also composed of multiple layers of the same stack [118, 119]. In contrast to the encoder, the decoder inserts a third sub-layer used to perform multi-headed attention on an encoder stack's output.

Self-attention is the core element of the Transformer using the weight of each sensor event of the input vector, expressed by Eq. (4.1):

$$\text{Attention}(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)\mathrm{V} \tag{4.1}$$

In self-attention, it first transforms input sensor events into an embedding vector, and the three vectors, Query vector (*Q*), Key vector (*K*), and Value vector (*V*) are obtained from the embedding vectors. Then, a score is calculated for each vector: $score = q \cdot k$. For gradient stability, the Transformer uses score normalization, i.e., dividing by $\sqrt{d_k}$. Next, the softmax activation function is applied to the score. Softmax dot product value *v*, to obtain the weighted score *v* for each input vector. The final output is obtained after summing.

Multi-Head Attention is equivalent to the integration of *n* different self-attention (ensemble). It can efficiently process the three vectors Q, K, V in parallel and get the final value by concatenated and computing. The formula can be seen from Eq. (4.2):

$$\text{MultiHead}(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_\mathrm{n})W^\mathrm{O}$$
$$\text{where head}_1 = \text{Attention}\left(\mathrm{QW_i^Q}, \mathrm{KW_i^K}, \mathrm{VW_i^V}\right) \tag{4.2}$$

where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W^O \in \mathbb{R}^{n d_\mathrm{v} \times d_{\text{model}}}$.

However, the abandonment of RNN and CNN in the architecture also deprives the model of capturing local features. Therefore, we believe that the Transformer+RNN-based model will achieve superior results.

4.3.2 Bidirectional Gated Recurrent Units (Bi-GRU)

Simple Recurrent Unit (SRU) is a simple type of recurrent unit employed to establish an RNN [73]. However, the RNN model is challenging for training and may suffer from exploding or vanishing gradients, limiting its application in simulating long-term activity sequence and time correlation in sensor data [17]. Therefore, variations of RNN such as GRU and LSTM were proposed. Both models integrate varieties of memory cells and gates to capture temporal activity sequences. LSTM combines with a memory unit to store context information, to control information flow into the network. With the addition of input gate, function gate, output gate, and other storage units and learnable weights, LSTM can build time-correlated models in time series data to improve recognition performance through specific feature extraction [110].

Despite the inherent advantages of LSTM, they observed that too many parameters need to be updated in the training process, which increases the calculation complexity of LSTM. To reduce the parameter update, they introduced a recursive gating unit with fewer parameters, making the implementation faster and simpler [115]. LSTM and GRU are different in updating the next hidden state and content exposure mechanism. While LSTM is updated by summation, GRU associates it according to the amount of time required to save it in memory to update the next hidden state. Besides, the recent performance comparison between LSTM and GRU shows that GRU is slightly better than LSTM in most ML applications [73].

In this structure, GRU has a rest and update gates. Both gates enable GRU to pass information across multiple time windows for better classification or prediction. More specifically, weights and data are stored in memory for use with a given state to update future values. The typical structure of GRU is shown in Figure 4-2. As mentioned earlier, GRU consists of an update gate and reset gate [17]. In the update gate, GRU computes z_t at a given time t to solve the vanishing gradient problem using the following formula:

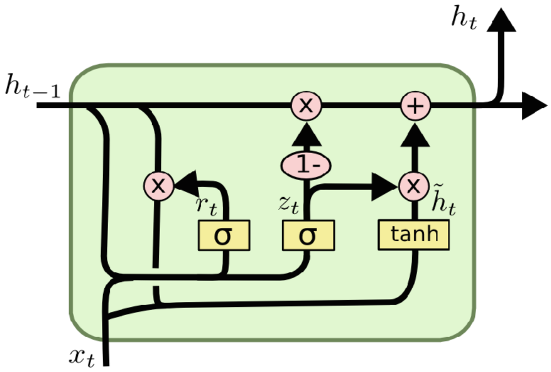

Figure 4-2 Gated Recurrent Unit (GRU) [73]

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \tag{4.3}$$

Whereas, in the reset gate, GRU calculates z_t at a given time t to illustrate how much past information to forget. The gate executes the following calculation:

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \tag{4.4}$$

The current storage content stage is calculated according to the following formula:

$$\tilde{h} = tanh(W[r_t h_{t-1}, x_t]) \tag{4.5}$$

Finally, h_t is calculated in the final memory of the current time step to store the current unit information for calculating the output vector o_t, as follows:

$$h_t = (1 - z_t)h_{t-1} + z_t\tilde{h}_t \tag{4.6}$$

$$o_t = \sigma_0(W_0 h_t + b_0) \tag{4.7}$$

For many sequence modeling tasks, it is helpful to access future and past contexts. However, the standard GRU network processes the sequence in chronological order, ignoring the future context. Bidirectional GRU networks extend the unidirectional GRU network by introducing a second layer, in which the hidden connections flow in reverse chronological order [120]. As a result, the model can take advantage of past and future information, as shown in Figure 4-3.

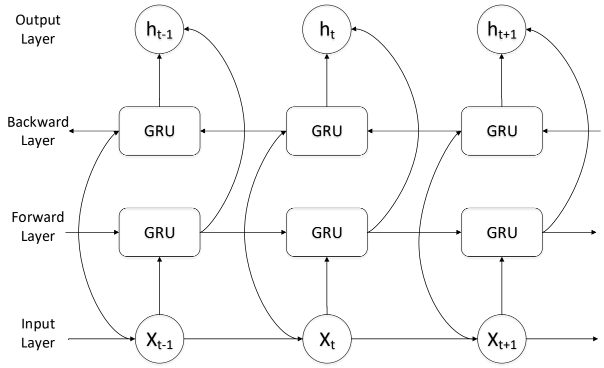

Figure 4-3 Bidirectional Gated Recurrent Unit (Bi-GRU)

4.3.3 Our Proposed HAR Framework

The proposed multi-resident HAR approach includes raw sensor data input, data preprocessing, deep feature extraction & model building, and activity inference stages. The input data are collected from different kinds of ambient sensors, such as temperature sensors (T), motion sensors (M), door sensors (D), and Passive Infrared Sensors (P). Our preprocessing stage comprises data filtering and segmentation from the CASAS and ARAS datasets. Data filtering is a significant task of reducing the noise or errors from collected data. In HAR, data filtering includes invalid activities cleaning, error label handing, and so on. All sensor events have two states, OPEN/CLOSE or ON/OFF in the CASAS dataset. We use 1 for OPEN/ON and 0 for OFF/CLOSE. In our framework, the combined label (CL) approach is used for multi-resident identifications, so the action labels of resident 1 (R1) and resident 2 (R2) need to be merged to produce a new label to represent each of the activities they perform. Segmentation is a method to segment the whole sensor events into a small chunk which can improve the performance of HAR. We use a static time-based segmentation method which divides the sensor data into smaller groups of every 60 seconds that will be fed into our TRANS-BiGRU model. We add a Bi-GRU in parallel to the encoder in the Transformer structure, the Bi-GRU and Transformer's encoder layer are merged by using concatenation operation. The concatenation operation is used as it can learn a different set of parameters from each of the two models to improve the model's capabilities. By adding a parallel Bi-GRU model, self-attention and Bi-GRU can be integrated more effectively to create a better framework for joint models [121]. Figure 4-4 shows our proposed multi-resident HAR architecture, and the pseudocode of our proposed architecture is shown in Algorithm 1.

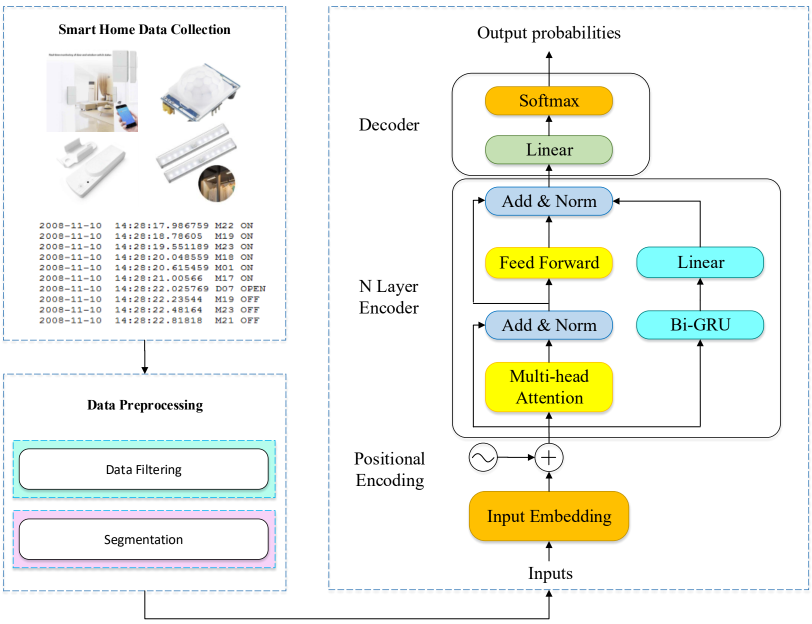

Figure 4-4 Our proposed multi-resident HAR architecture

Algorithm 1 Pseudocode for TRANS-BiGRU

Input : D_{train}: dataset training, Y: target class,
n_enc_layers: N encoder layers, n_dec_layers: N decoder layers,
epoch: total maximum epoch, dp: dropout probability,
Output: model

```
for I=1 to epoch
    for data in D_train
        for i in range (n_enc_layers):
            attn1 = multi_head_attention(Q,K,V)
            attn1 = dropout(dp)
            attn1 = layer_normalization(dp + attn1)
            attn2 = Bi-GRU (D_train)
            attn = merge (attn1,attn2)
        end for
        for i in range (n_dec_layers):
            X_out = attn1_out + attn2_out
            output = softmax (X_out)
            loss = compute_loss(output, Y)
        end for
    end for
end for
```

4.4 Experiments and Results

In this section, we discuss smart home datasets, our experimental setup, and the evaluation results. We evaluate our proposed TRANS-BiGRU model and compare it with other well-known deep learning models such as Deep Belief Networks (DBN), CNN, LSTM, and GRU to evaluate the recognition performance of multi-resident activities. Apart from that, we also compare TRANS-BiGRU with traditional ML models proposed in the HAR research, such as Decision Tree (DT), Support Vector Machine (SVM), Naïve Bayes (NB), and Factorial Hidden Markov Model (FHMM), to evaluate the performance of multi-resident HAR.

4.4.1 Home Datasets

There are standard benchmarking datasets used by researchers who work in the field of multi-resident HAR. We introduce some of the datasets collected from various kinds of non-intrusive sensors installed in our experiment's smart homes. Although there are many other datasets in a multi-resident environment, we choose these datasets because they collected more complicated residents' activities, which can be seen from Table 4-2. Our purpose is to identify the residents' complex activities closer to real life.

Table 4-2 Non-Intrusive Sensor-based Datasets

Dataset	# of Residents	Duration	# of Sensors	# of Activities
Multi-resident ADLs (CASAS)	2	2 months	37	15
ARAS (House A)	2	30 days	20	27
ARAS (House B)	2	30 days	20	27

Table 4-3 Setting and Activities in CASAS Dataset

	Multi-resident ADLs (CASAS)
# of Ambient Sensors	37 of 7 different types

House Information	Two bedrooms, one living room, one kitchen, one bathroom
Residents	2
Duration	2 months
# of Activities	15
Activities	Filling medication dispenser, Moving furniture, Watering plants, Hanging up clothes, Playing checkers, Preparing dinner, Reading magazine, Gathering and packing picnic food, Sweeping floor, Setting the table, Paying bills

This research first uses the Center for Advanced Studies in Adaptive Systems (CASAS) dataset established at Washington State University in 2007. Up to now, there are more than 20 datasets published by CASAS, and some of them are related to multi-resident datasets. For example, one dataset named "Multiresident ADL Activities" represents two people in the apartment while performing fifteen ADL activities in the apartment. The activities include sequential activity, interleaving activity, parallel activity, and collaborative activity. The details of activities are shown in Table 4-3. Besides, seven types of sensors are applied to this scenario: motion sensors, item sensors, cabinet sensors, water sensors, burner sensors, phone sensors, and temperature sensors. Figure 4-5 depicts the specific deployment of sensors in this room. For the second experiment, we use public benchmark ARAS HAR datasets to collect data from real-life scenarios in two different houses. ARAS contains two famous datasets used in the multi-resident HAR research and was released by Hande Alemdar in 2013 [80].

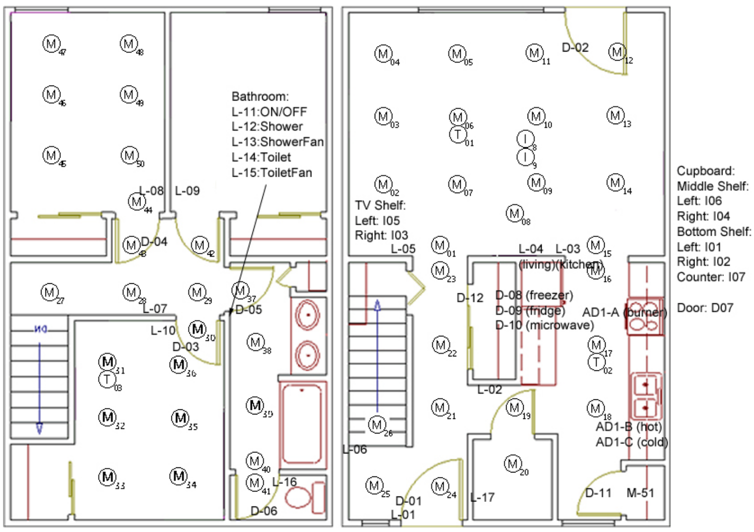

Figure 4-5 Specific deployment of sensors in the CASAS dataset

Compared with the CASAS dataset, ARAS datasets perform more complex actions, so in our scenario, ARAS is also chosen for evaluation. ARAS can be divided into two separate datasets, House A and House B. Each house is equipped with 20 different types of interactive binary sensors. A month's worth of sensor-reading data was collected from each house, together with ground activity labels for two residents [80]. The residents themselves provide ground truth for activity labels during data collection. Each dataset contains a comprehensive list of 27 different activities. Because the data is based on real-life activities, there are no restrictions on the residents' activity selection, and some of the planned activities are not performed. For instance, there is a dishwashing activity on the list, House B has a washing machine, but the residents didn't prefer to annotate the activity during data collection. Basic attributes of housing, demographic information, sensor infrastructure details, and activities are shown in Table 4-4, and the layout of the homes and their sensor placement are shown in Figure 4-6 [80]. Unlike other similar multi-resident HAR studies, each ARAS dataset consists of two sensor events readings that share the same room's furnish and decorate. We believe that such no movement restrictions environment can better reflect real life because it can explain the majority of people living with spouses, family, and friends, and it will also provide an opportunity to study the social interaction between couples.

Table 4-4 Settings and Activities in ARAS Datasets

	House A	**House B**
# of Ambient Sensors	20 of 7 different types	20 of 6 different types
Size of the House	$50m^2$	$90m^2$
House Information	One bedroom, one living room, one kitchen, one bathroom	Two bedrooms, one living room, one kitchen, one bathroom
Residents	2 males both aged 25	Married couple. Age average 34
Duration	30 full days	30 full days
# of Activities	27	27
Activities	Other, having breakfast, having lunch, toileting napping, having a shower, brushing teeth, preparing dinner, having dinner, having a snack, sleeping, watching TV, preparing lunch, using	

	the Internet, washing dishes, reading a book, laundry, talking on the phone, listening to music, shaving, preparing breakfast, studying, cleaning, going out, having a conversation, having guest, changing clothes

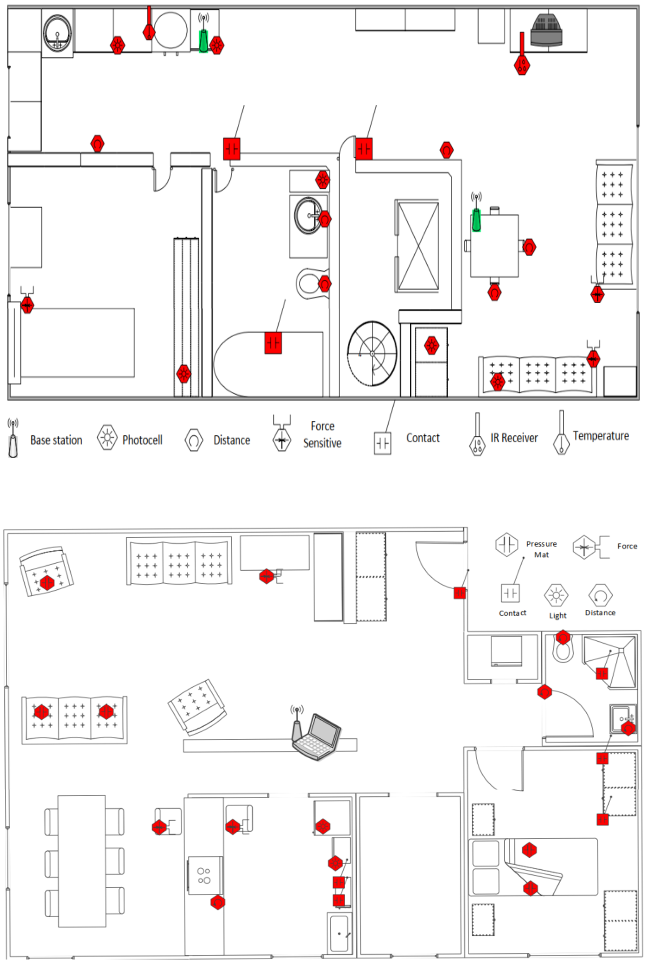

Figure 4-6 Layouts of Houses A (up) and B (down) in the ARAS datasets [80]

4.4.2 Experimental Setup

Our experiments are conducted based on the CASAS and ARAS datasets. For the CASAS dataset, 17,258 sensor events are collected. For the ARAS dataset, sensor-reading data are recorded in units of seconds, so a total of 86,400 sensor sequence data are captured per day and continue for 30 days. We also apply a sliding window method to segment sensor event sequences and $\Delta t = 60\ s$ intervals in discretizing sensor data, thus $T = 1{,}440$ data points generated for each day. The 10-fold cross-validation method is used in our experiments. We conduct two sets of evaluations. First, we compare our model with the existing DL models. Second, we compare it with the traditional ML models used for HAR. We use a standard evaluation matrix, which includes accuracy, recall, precision, and F-measure.

In our proposed TRANS-BiGRU model, for the Transformer architecture, we choose the same experimental settings as [118]. The encoder and decoder are composed of a stack of N=6 identical layers. Two hidden layers of GRU cells are applied for the Bi-GRU model, and all the GRU cells were fully connected with these inputs and were repeatedly connected with other GRU cells. We choose a dense output layer to classify the labels with Dropout = 0.5, Epochs = 200, Batch_Size = 40, Units = 64, the RMSProp optimiser and Softmax function are chosen for training the network. For DBN, we apply three fully connected dense layers with 64 units in the hidden layers and a dense output layer with the Softmax function to perform classification, and the RMSProp optimizer is selected in our evaluation. For CNN, the 1D Convolutional Neural Network with two layers is used for training, and the RMSProp optimizer is also used. For traditional ML models applied in our experiment, DT uses an entropy function to perform training, SVM employs the Gaussian kernel, and the Viterbi algorithm is used in FHMM.

4.4.3 Results of Various DL Models

We first evaluate our model against commonly used DL models for HAR, including DBN, CNN, LSTM, GRU, Ens2-LSTM, LSTM+CNN, CascEns-LSTM, Bi-LSTM, Bi-GRU, and the Transformer architecture. Figure 4-7 and Table 4-5 show the performance results obtained from all the DL models using 10-fold cross-validation.

Table 4-5 Different DL Models on Three Multi-Resident Datasets

Metric (%)	Model	Dataset									
		CASAS			ARAS House A			ARAS House B			
		R1	R2	CL	R1	R2	CL	R1	R2	CL	Average
Accuracy	CNN	84.21	83.45	80.32	80.84	84.64	78.76	80.95	77.54	75.23	80.66
	DBN	85.56	85.76	83.05	75.78	79.33	73.45	75.80	72.93	70.28	77.99
	LSTM	89.42	87.42	82.16	78.23	85.59	80.34	84.91	80.74	79.40	83.13
	GRU	92.25	87.75	83.66	79.79	86.93	80.32	86.85	81.02	78.94	84.17
	Ens2-LSTM	89.23	87.84	85.43	80.92	87.94	80.65	87.30	82.95	79.44	84.63
	LSTM+CNN	85.56	83.45	82.92	78.06	80.83	78.81	81.06	79.25	78.30	80.92
	CascEns-LSTM	88.84	88.41	85.65	80.20	85.93	80.29	86.25	83.49	80.26	84.37
	Bi-LSTM	91.53	88.94	85.33	84.95	83.12	79.93	85.93	83.04	80.85	84.85

	Bi-GRU	**93.34**	90.12	87.92	85.29	90.29	85.30	88.05	84.73	84.03	87.67
	Transformer	89.15	90.44	86.85	85.76	**92.21**	84.25	89.34	84.89	83.14	87.34
	TRANS-BiGRU	93.28	**92.73**	**92.86**	**89.70**	91.63	**89.48**	**91.76**	**90.93**	**90.59**	**91.44**
Precision	CNN	84.19	82.94	79.94	80.34	84.65	77.34	79.86	76.45	75.98	80.19
	DBN	83.58	84.94	81.82	74.65	78.85	72.54	74.71	71.84	69.86	76.98
	LSTM	89.40	87.35	82.12	78.20	85.62	80.31	84.81	80.78	79.30	83.10
	GRU	90.25	88.75	82.97	75.95	85.94	80.15	85.76	79.93	76.85	82.95
	Ens2-LSTM	90.23	85.84	87.12	82.74	85.48	80.43	86.21	81.86	79.15	84.34
	LSTM+CNN	85.40	82.75	82.64	79.76	80.42	79.86	80.36	79.65	78.56	81.04
	CascEns-LSTM	87.95	88.23	85.45	80.02	85.52	79.36	85.52	83.15	80.10	83.92
	Bi-LSTM	91.47	88.91	85.30	84.91	83.02	79.73	85.53	83.10	80.65	84.74
	Bi-GRU	**93.32**	90.10	87.87	85.28	90.19	85.10	88.15	84.56	84.13	87.63
	Transformer	87.65	88.44	85.94	84.36	**92.06**	83.39	88.25	83.80	82.66	86.28
	TRANS-BiGRU	93.20	**91.73**	**91.08**	**88.84**	90.85	**89.75**	**90.67**	**89.84**	**89.79**	**90.64**
Recall	CNN	84.20	83.43	80.56	80.68	84.33	78.45	80.08	76.67	75.12	80.39
	DBN	85.46	85.56	83.23	75.62	79.02	73.19	74.93	72.06	70.09	77.68
	LSTM	89.34	87.25	82.11	78.27	85.54	80.32	84.95	80.72	79.46	83.11
	GRU	92.55	87.45	84.75	79.63	86.62	80.22	85.98	80.15	78.84	84.02
	Ens2-LSTM	89.53	87.23	85.75	80.76	87.63	80.34	86.43	82.08	79.33	84.34
	LSTM+CNN	85.42	82.77	82.62	79.71	80.40	79.83	80.32	79.61	78.52	81.02
	CascEns-LSTM	87.92	88.21	85.40	79.97	85.50	79.31	85.49	83.11	80.02	83.88
	Bi-LSTM	91.50	88.87	85.31	84.92	83.11	79.92	85.78	83.14	80.75	84.81
	Bi-GRU	**93.26**	90.12	87.90	85.24	90.23	85.31	88.04	84.70	84.12	87.66
	Transformer	88.74	89.84	87.06	85.60	**92.20**	84.43	88.47	84.02	83.13	87.05
	TRANS-BiGRU	93.16	**92.12**	**92.58**	**89.54**	91.32	**90.14**	**90.89**	**90.06**	**90.53**	**91.15**
F-measure	CNN	84.19	83.18	80.25	80.51	84.49	77.89	79.97	76.56	75.55	80.29
	DBN	84.51	85.25	82.52	75.13	78.93	72.86	74.82	71.95	69.97	77.33
	LSTM	89.37	87.30	82.11	78.23	85.58	80.31	84.88	80.75	79.38	83.10
	GRU	91.39	88.10	83.85	77.75	86.28	80.18	85.87	80.04	77.83	83.48
	Ens2-LSTM	89.88	86.53	86.43	81.74	86.54	80.38	86.32	81.97	79.24	84.34
	LSTM+CNN	85.41	82.76	82.63	79.73	80.41	79.84	80.34	79.63	78.54	81.03
	CascEns-LSTM	87.93	88.22	85.42	79.99	85.51	79.33	85.50	83.13	80.06	83.90
	Bi-LSTM	91.48	88.89	85.30	84.91	83.06	79.82	85.65	83.12	80.70	84.77
	Bi-GRU	**93.29**	90.11	87.88	85.26	90.21	85.20	88.09	84.63	84.12	87.64
	Transformer	88.19	89.13	86.50	84.98	**92.13**	83.91	88.36	83.91	82.89	86.67
	TRANS-BiGRU	93.18	**91.92**	**91.82**	**89.19**	91.08	**89.94**	**90.78**	**89.95**	**90.16**	**90.89**

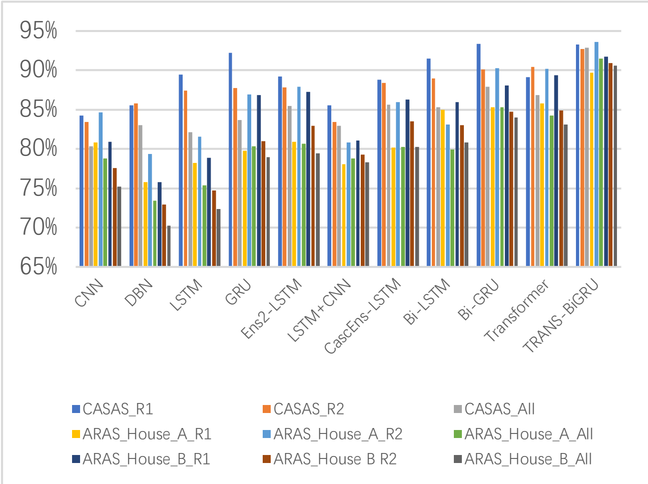

Figure 4-7 Accuracy of various DL models

Firstly, from averaging the three dataset's performance, our proposed model outperforms the other DL models on all four metrics, accuracy, precision, recall, and F-measure. Specifically, our proposed model achieves average accuracy of 91.44%, which outperforms CNN, DBN, LSTM, GRU, Ens2-LSTM, LSTM+CNN, CascEns-LSTM, Bi-LSTM, Bi-GRU and Transformer by approximately 10.78%, 13.45%, 8.31%, 7.27%, 6.81%, 10.52%, 7,07%, 6.59%, 3.77% and 4.10% respectively. The performance of DBN is worse than that of the other DL models because it is the most basic DL model, which is a full connection between layers, and there is no specific consideration of time dependency in time series. CNN achieves an average accuracy of 80.66% in three datasets. CNN has been proved to be one of the best models for image activity recognition, but for the time-series data, compared with the RNN-based model, it cannot predict the action very well.

From the detailed analysis of the three datasets CASAS, House-A, and House-B, we found that our proposed model achieves better performance in most cases, and the performance is more stable regardless of the individual identification of the R1, R2, and CL method [7, 39, 70]. In the CASAS dataset, Bi-GRU achieves the highest accuracy of 93.34% in the identifying R1, which is slightly higher than our proposed model. However, for R2 recognition, our model is more accurate. Besides, our proposed model is much higher than Bi-GRU when using the CL approach, so it can be said that our model is more stable in the CASAS dataset.

As for the ARAS dataset's performance, the Transformer framework achieves the highest accuracy of 92.21% for R2 on the House A dataset, but it is only 0.58% more accurate than our proposed model. However, our proposed model outperforms the Transformer framework by 3.94%

and 5.23% for R1 recognition and using the CL approach respectively. Besides, our proposed model outperforms all other DL models in terms of accuracy, achieving 91.76%, 90.93%, and 90.59% for recognition under R1, R2 on the House B dataset and the CL method, respectively. Therefore, we can conclude that our proposed model achieves better and relatively stable performance in different multi-resident recognition datasets.

We also find that our proposed model does not always achieve the best performance. Specifically, the Bi-GRU's recognition accuracy for R1 in CASAS and the Transformer's recognition accuracy for R2 in ARAS House A are better than that of our model. Further examination of the datasets does not reveal any significant differences of the activities performed by the residents, and the performance of our proposed model is very marginally lower (less than 1%) than the other two models, while our model achieves better performance in the majority of cases. Therefore, we believe that our proposed model has better stability.

Meanwhile, our experiments show that the RNN-based models and hybrid deep learning models (LSTM, GRU, Ens2-LSTM, LSTM+CNN, CascEns-LSTM Bi-LSTM, and Bi-GRU) can better predict time-series data than CNN, DBN models. In these models, Bi-LSTM, Bi-GRU and Ens2-LSTM perform better than LSTM and GRU. Technically, Ens2-LSTM combines the output of a Bi-LSTM and Uni-LSTM to generate a new LSTM-based model. In this way, the layer of the model is increased. A basic LSTM can be used to identify simple information. Also, Bi-LSTM can be applied to store past and future information, improve identification accuracy, and be superior to a single LSTM model. Bi-GRU has an additional backpropagation layer than GRU, so the model can utilize past and future information to improve recognition accuracy. However, not all hybrid DL models are better than individual DL models, and we see that hybrid models of LSTM+CNN have lower performance than the LSTM model. Therefore, we found that not simply stacking the number of layers of deep models can improve the performance. For the Transformer framework, it breaks the limitation that RNN models cannot be computed in parallel. Moreover, compared to CNN, the number of operations required to compute the association between two locations does not grow with distance. The Transformer framework's disadvantage is also apparent, and the local information acquisition is not as substantial as RNN and CNN-based models. Therefore, our proposed TRANS-BiGRU model can solve this problem well, and our experiments also show that our proposed model can achieve better performance.

4.4.4 Results of Various ML Models

Figure 4-8 and Table 4-6 illustrate the comparison between our TRANS-BiGRU method and other existing ML approaches proposed for multi-resident HAR. We averaged the recognition performance of the R1, R2, and the CL method separately for each dataset. The results show that our proposed TRANS-BiGRU model outperforms all those ML methods when applying to three datasets. Besides, we observe that SVM achieves the highest performance in traditional machine learning with an accuracy of 83.83% in CASAS, 84.26% in House A, and 84.51% in House B, which are even better than CNN and DBN, but it is still unpromising from the model we proposed. FHMM has also achieved relatively poor performance, with an accuracy of 78.98% in CASAS, 79.25% in House A, and 79.65% in House B. The performance of DT and NB is not stable, DT can only get 63.54% of accuracy in CASAS but 73.50% of accuracy in House B. On the contrary, NB performs much better in CASAS and House A than in House B.

Table 4-6 Different ML Models on Three Multi-Resident Datasets

Metric (%)	**Model**	**Dataset**		
		CASAS	**ARAS House A**	**ARAS House B**
Accuracy	DT	63.54	69.92	73.50
	NB	72.91	75.54	61.46
	SVM	83.83	84.26	84.51
	FCRF	75.22	70.43	71.86
	FHMM	78.98	79.25	79.65
	TRANS-BiGRU	**92.96**	**90.27**	**91.09**
Precision	DT	63.34	54.57	73.12
	NB	72.77	75.96	73.19
	SVM	83.67	84.22	84.47
	FCRF	75.13	70.32	71.79
	FHMM	78.76	79.04	79.76
	TRANS-BiGRU	**92.00**	**89.81**	**90.10**
Recall	DT	63.31	69.92	83.50
	NB	72.56	75.54	61.45
	SVM	83.74	84.26	84.51
	FCRF	75.04	70.19	71.65
	FHMM	78.56	79.25	79.65
	TRANS-BiGRU	**92.62**	**90.33**	**90.49**
F-measure	DT	62.82	60.18	77.27
	NB	72.66	72.45	65.78
	SVM	83.70	84.23	84.48

FCRF	75.08	70.25	71.72
FHMM	78.66	79.14	79.20
TRANS-BiGRU	**92.31**	**90.07**	**90.30**

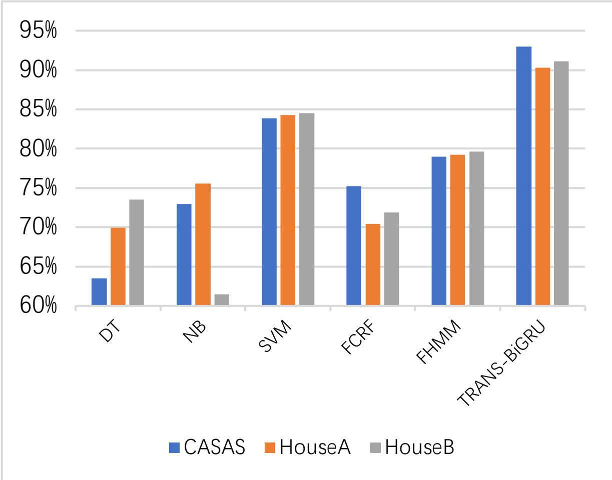

Figure 4-8 Accuracy of various traditional ML models and our proposed model

4.4.5 Statistical Comparison

According to the classification results, it can be easily seen that the Bi-GRU model and our proposed TRANS-BiGRU model are outstanding comparing with other DL and ML models. We apply a famous nonparametric test, Wilcoxon singed-rank test, to prove that the proposed DL model is superior in a multi-resident environment to other classification models. We analyze the singed of differences by calculating the different scores.

In the test, we calculate the difference d_i between the accuracy in TRANS-BiGRU and Bi-GRU models on three datasets first. Then, absolute values are utilized for differences ranking. Table 4-7 presents various accuracy of classification models and ranks corresponding to the two approaches. Afterward, R^+ and R^- are calculated, where R^+ is the sum of the ranks for positive and R^- refers to negative. In this case, the values of R^+ and R^- are 44 and 1, respectively. According to the Wilcoxon signed-rank test's critical values with a confidence level of α = 0.05, the difference between the two models is considered significant if the smaller sums are equal to or less than 8. Therefore, we conclude that our proposed TRANS-BiGRU model is statistically superior to the state-of-the-art Bi-GRU model.

Table 4-7 Comparison of Our Proposed Method with Transformer Architecture

Dataset	Resident	TRANS-BiGRU	Bi-GRU	Difference	Rank
	R1	93.28	93.34	-0.06	1

CAS	AS	R2	92.73	90.12	2.61	3
		CL	92.86	87.92	4.94	7
ARAS	House A	R1	89.70	85.29	4.41	6
		R2	91.63	90.29	1.34	2
		CL	89.48	85.30	4.18	5
ARAS	House A	R1	91.76	88.05	3.71	4
		R2	90.93	84.73	6.2	8
		CL	90.59	84.03	6.56	9

4.5 Discussion and Conclusion

This paper proposes a Transformer with a Bidirectional GRU deep learning method, TRANS-BiGRU, which leads to an apparent improvement in a multi-resident environment Human Activity Recognition (HAR). In particular, after a comprehensive comparison of various HAR models, RNN-based models (LSTM, GRU, Ens2-LSTM, Bi-LSTM, and Bi-GRU) outperforms other DL models (DNN, CNN), and the most commonly used ML models (FHMM, NB, DT) in the CASAS and ARAS datasets, which also indicate that RNN-based models are propitious to the classification of time series data.

We compared these RNN-based models, LSTM, GRU Ens2-LSTM, Bi-LSTM, Bi-GRU, Transformer architecture, and our proposed TRANS-BiGRU. The results show that Bi-GRU and Transformer slightly outperform LSTM and GRU. Ens2-LSTM model combined a basic LSTM and a Bi-LSTM model, where the LSTM model is applied to study the simple activities, and Bi-LSTM is utilized to analyze the complicated activities and improve the performance. Our proposed model integrated Transformer architecture and Bi-GRU to recognize the involved activities and achieved the best performance comparing with other HAR models.

In the future, we will explore multi-resident HAR in a more complex real-world situation. Instead of using an extensive benchmark dataset only, we will apply and enhance our proposed DL model on real-life, long-term activity monitoring.

Chapter 5 Locally-weighted Ensemble Detection-based Adaptive Random Forest Classifier for Sensor-based Online Activity Recognition for Multiple Residents

In recent years, various approaches for multi-resident Human Activity Recognition (HAR) in a smart indoor environment have been developed and improved along with the rapid development of sensors and AI technologies. Research in data stream-based Online Learning (OL) for multi-resident HAR is relatively new and a majority of the existing works have been developed based on training batches of data which cannot recognize real-time activities. To address the challenges of online learning for multi-resident HAR, we propose a novel online learning architecture based on a locally-weighted ensemble detection-based adaptive random forest classifier (LED-ARF). We conduct a comprehensive comparison for the performance of eight famous online learning classification techniques and our LED-ARF method. The comparison is evaluated based on the two benchmarking CASAS and ARAS datasets. Our experimental results show that the LED-ARF achieves the best performance with the highest robustness for online multi-resident HAR.

Chapter 5 has been submitted to IEEE Internet of Things Journal.

5.1 Introduction

The global population has entered the aging phase and the proportion and number of older people has been increasing around world. As they age, many older adults may suffer from a variety of diseases, such as Alzheimer's or coronary heart disease. Due to the lack of caregivers and the high cost of care, not all seniors have access to specialty care services. Recently, with the rapid development of the Internet of Things (IoT) and various sensors, there has been a lot of research into the creation of intelligent assisted living systems to help monitor activities of daily living (ADLs) so that caregivers or their families can be quickly notified when danger occurs [36, 69]. Particularly, Human Activity Recognition (HAR) covers the field of using various sensor data and machine learning techniques to identify human activities and has achieved excellent recognition performance in a smart home environment.

With the rise of the IoT, many advanced sensors have sprung up, such as video sensors, wearable sensors, and infrastructure sensors [34]. Although video-based sensors can accurately identify behavioral movements of multiple people, they are not suitable for use in a home environment due to their invasive nature and privacy concerns. Wearable sensors also have a great advantage of having direct access to personal data, which can be used to make behavioral judgments [108]. For HAR in a multi-resident environment, wearable sensors can be used to distinguish between different residents and to solve data association problems. However, many older adults are reluctant to carry wearable sensors or they may forget to carry them, leading to missing out from being monitored. On the contrary, infrastructure type of sensors, which can attached to the surface of an object, can address both the privacy concern and the applications for sensitive people such as elderlies not feeling under surveillance from surrounding sensors, so they can live in such a home-only environment without interference [43].

In reality, an elderly or person under care in the family usually do not live by themselves, there may be their spouse, or pets. Sometimes, their relatives or caregivers come to visit them at home. Therefore, HAR in an multi-resident environment considers real-world problems. In recent years, there have been many studies using machine learning (ML) and deep learning (DL) techniques to address some of the existing multi-resident HAR issues, especially around recognizing parallel and collaborative activities [7, 42]. Almost all current multi-resident HAR studies use offline learning methods. Offline approaches generally consist of five steps: data collection, data processing, segmentation, feature extraction, and classification. Classifiers are first to be trained offline and then they can be used to classify streaming data coming from various sensors. However, offline trained classifiers are not suitable for a long-term use of multi-resident HAR because a person's activities and habits may change and vary from one person to another, and new guests may arrive at home, making the originally trained classifier unable to identify the changes and new people [19]. Therefore, classifiers need to be constantly updated to adapt to newly generated changes in order to accurately identify different inhabitants and their activities.

Multi-resident HAR is still in the initial stage of research due to the complexity brought about by the different habits and activities of different residents. There are still many challenges in the field of multi-resident recognition, such as data association and recognition of collaborative activities [10, 12]. In addition, there are also many challenges in the study of online learning (OL). One of the main challenges in online learning for multi-resident HAR is about recording of activity labels of new sensor event to update a classifier. First method is to ask all the residents to give constant feedback when they perform new activities, but this is not a wise solution because it would take too much energy from the residents. Another solution is to use surveillance cameras for real activity labeling. However, this method tends to consume a lot of energy and memory, and it invades the privacy of the residents in a smart home environment, which can make them feel very uncomfortable [30, 45]. Another key challenge is to ensure that a classifier is continuously updated to improve activity recognition performance rather than degrade it when new sensor data arrives [122].Most importantly, sensor data triggered by multiple people at the same time increases the complexity on this challenge.

To address the second challenge mentioned above, this paper focuses on processing multi-resident activity sensor events in real-time and classifying the activities. We combine the online classification model Adaptive Random Forest Classifier (ARF) with the Locally-weighted Ensemble Detection (LED) algorithm to process sensor events in real-time and continuously update the parameters of the classification model based on the LED results, thus improving the performance of online recognition. Specifically, our paper makes the following contributions:

- We proposed a novel Locally-weighted Ensemble Detection-based Adaptive Random Forest

Classifier (LED-ARF) for online learning of multi-resident activity recognition.

- We conduct a comprehensive evaluation and comparison for the performance of eight popular online learning classifiers and the experimental result shows that our LED-ARF model achieves the best performance on online multi-resident HAR based on the two well-known benchmarking CASAS and ARAS datasets.

The remainder of this paper is organized as follows. Section 5.2 introduces related works in the recent years. Section 5.3 discusses our proposed online HAR method. Section 5.4 evaluates and discusses different state-of-the-art and online learning model's' performance based on the CASAS and ARAS datasets. Section 5.5 concludes the paper.

5.2 Related Work

In general, a multi-resident HAR framework includes data collection, data pre-processing, and classification. Most of the studies focus on developing various offline classification models to improve the performance of activity recognition. There have not been any studies applying online learning to multi-resident recognition, although several studies have proposed online learning for single-resident HAR. This section discusses the existing works on sensor-based multi-resident classification models and online learning for HAR.

Fatlawi et al. [123] evaluated various stream data processing models to find the optimal model for Biomedical Data. These evaluated models include two categories, one is individual stream data processing models, such as Naive Bayesian, Hoeffding Tree, and KNN classifiers. the other is Ensemble model classifier, including Oza Boosting, Adaptive Random Forest, and Oza Bagging classifiers. Their experimental results show that Adaptive Random Forest achieves superior performance.

Abdallah et al. [124] proposed a phone-based online recognition framework to process the relevant data streams to recognize resident activities. Their framework combines active learning and incremental learning for recognizing real-time activities, and adaptation in streaming environments. In this framework, there are two phases, offline and online. The offline phase model is trained on the labeled data and recognizes different activities. The online phase updates the model to recognize unlabelled stream data. However, this framework still requires a large amount of labeled data for training.

Mohamad et al. [125] proposed active learning approaches to query resident-specific activity tags to label newly generated stream data and improve the accuracy of recognition. They propose a semi-supervised classifier combined with an active learning approach based on Bayesian stream data for online activity recognition. The proposed model is also compared with the traditional machine learning models (support vector machines and decision trees) and the performance of the proposed model is superior. However, their model is based on a continuous analog signal based on wearable sensors, and whether the proposed model can be used under non-intrusive environment-based sensor signals still needs to be investigated.

Asghari et al. [19] proposed an online activity recognition model based on Hidden Markov Model to recognize current activity on the sensor events stream. The model consists of two phases: in the first phase, a related data stream is segmented based on the activity patterns, and HMM is implemented to receive the ongoing activity observation. The second phase is an update phase to correct the provided label for the sensor stream based on a statistical method. Besides, the proposed model can be used to recognize interrupted activities.

All the existing multi-resident recognition techniques have been developed for offline recognition. Benmansour et al. [7] proposed a Combine Label approach to recognize the parallel and collaborative activities For every possible combination of two activities,they (labels) are merged into one. By using this method, the problem of data association between residents can be addressed and different activitites of different residents can be recognized simultaneously. They evaluated their proposed model with other state-of-the-art models such as Parallel HMM (PHMM) and CHMM. The results illustrate that the CL-HMM model achieved the best accuracy (88.03%) in the collaborative activities.

Tran et al. [42] evaluated basic Recurrent Neural Network (RNN), Long-Short-Term-Memory (LSTM), and Gated Recurrent Unit (GRU) models based on the CASAS and ARAS multi-residents datasets. The experiments recognize resident A and resident B’s activities separately and the Combine Label approach is performed for comparison. The experimental results show that the GRU model performs best compare with other models such as RNN, LSTM, HMM, and CRF. Besides, the Combine Label method is more effective than a separate label method. However, the average recognition accuracy is only 77.2%.

Recently, several works have investigated Group Activity Recognition (GAR) [126-129]. For example, Yan et al. [126] proposed a Participation-Contributed Temporal Dynamic Model (PC-TDM) to recognize group activity, a Bidirectional LSTM (Bi-LSTM) was applied to improve GAR performance and the proposed method is superior to the state-of-the-art methods. Tang et al. [127] proposed a coherent constraint graph LSTM (CCG- LSTM) with the functions of Spatio-Temporal Context Coherence (STCC) and Global Context Coherence (GCC), which can effectively identify group activities by modeling the relevant movements of individuals while suppressing irrelevant movements.

5.3 Proposed Online Multi-Resident HAR Approach

In this section, we discuss our online learning method for multi-resident HAR which consists of Adaptive Random Forest (ARF) and Locally-weighted Ensemble Detector (LED).

5.3.1 Proposed Multi-Resident Online HAR

Our online multi-resident HAR architecture proposed in this paper, as show in Figure 5-1, consists of four components: data input, data pre-processing, online recognition model building, and activity inference. The input data can come different ambient sensors, such as door sensors (D), Motion Sensors (M), and temperature sensors (T). The data pre-processing phase includes data filtering and data transformation. Data filtering is very essential for HAR recognition because noisy data and missing data can significantly affect the recognition performance [54]. The purpose of data transformations is to convert the incoming raw data from the sensor into a data stream type that can be used for online analysis. The stream data is then passed into our online learning model. If the LED finds a drift, we re-tune the model and adjust the parameters to ensure the accuracy of the recognition.

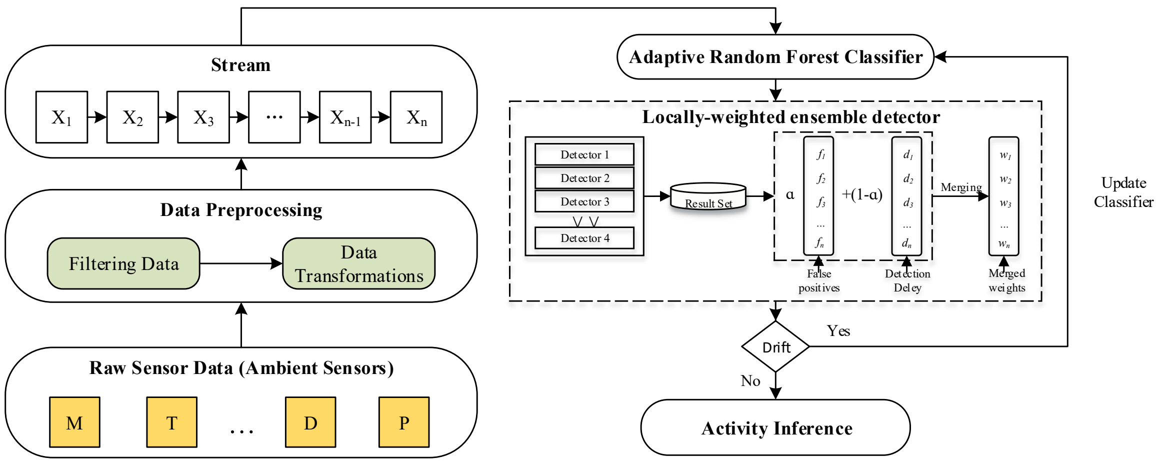

Figure 5-1 Proposed Online Multi-resident HAR architecture.

5.3.2 Adaptive Random Forest (ARF)

Random forests (RF) are part of the Bagging method in ensemble learning. Many decision trees form a random forest, and the number of different decisions is not necessarily related to each other [32]. After the new sample input, we let each decision tree in the random forest perform the judgment and classification separately when we perform the classification task. Each decision tree will get its own classification result, and the random forest will give the final result to the one with the most classification results [130]. We draw N times from a sample size of N, one at a time, so that N samples are generated. These selected samples are used for decision tree training and as the root node of the decision tree.

Random forests have many advantages. For example, it can provide high-dimensional data without downscaling and feature selection, as well as determine the importance of features and the interaction between different features, is not prone to overfitting, is faster to train, and is easy to make into a parallel method. [123]. For unbalanced datasets, they can balance the error and still maintain performance if a large portion of features are missing. However, the RFs also has its drawbacks, overfitting in some noisy regression or classification problems. When the attributes with more value divisions, the random forest receives more influence, so the attribute weights of the random forest on such data are not accurate and cannot be used for real-time analysis of data streams.

The upper bound on the generalization error (GE) of the random forest converges to the following expression Eq. (5.1) when the number of decision trees is sufficiently large.

$$GE \leq \frac{\bar{\rho}(1 - S^2)}{S^2} \tag{5.1}$$

where $\bar{\rho}$ is the average correlation coefficient between trees, S is a measure of the "strength" of the tree classifier, the strength of a set of classifiers is the average performance of the classifier, and the performance is measured in terms of the residual of the classifier M using the probabilistic algorithm, seen in Eq. (5.2).

$$M(X, Y) = P(\hat{Y}_\theta = Y) - maxP(\hat{Y}_\theta = Z) \tag{5.2}$$

where $\hat{Y}_\theta$ is the prediction class of X made by the classifier constructed from the modal random vector θ, and Y and Z are not equal. The larger the margin, the higher the probability that the classifier correctly predicts the given sample X. From the formula defining the upper bound of the generalization error, it is clear that the upper bound of the generalization error tends to increase as the relevance of the tree increases or the strength of the combined classifier decreases. Therefore, a way to improve the generalization error of the combined classifier is to increase the randomness, as it can help to reduce the correlation between decision trees.

5.3.3 Locally-weighted Ensemble Detector (LED)

Many machine learning models are rely on stable probability distribution data samples, and as the probability distribution of the sample data changes, the accuracy of the model decreases. In contrast, Drift Detection Method (DDM) is a method for detecting events changes in the probability distribution of sample data [106]. The idea of DDM is to control an (online) error rate of a machine learning model. If sample data are uninformedly distributed, the error rate of the model will gradually decrease and when the probability distribution changes, the error rate of the model will increase [131].

The DDM uses two threshold values for the error rate, one is *warning* and the other one is *drift*. When the error rate reaches the *warning* value at the w_{th} data input in the sample data, it indicates a precursor of a change in the sample probability distribution. If the error rate is not reduced by successive data inputs and it reaches the *drift* value at the d_{th} data input, it is determined that the sample's probability distribution has been changed [123]. In order to adapt to the new sample data, the model will then learn with the data after w; and if the successive input data makes the error rate decrease, it is a false alarm, which can be seen in Figure 5-2.

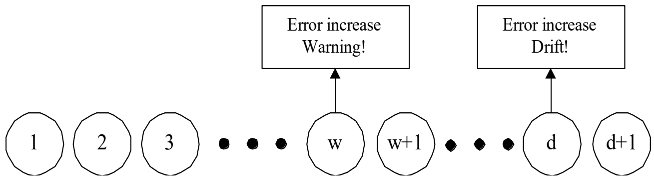

Figure 5-2 Drift Detection Method

The locally-weighted ensemble detector (LED) was proposed by Zhang et al [132], the detection of its relative performance is a semi-supervised method by assigning different weights. In each training session, the goal of the training is to learn various weights and assign them to each base detector through a local weighting mechanism. The local weighting mechanism works as follows:

From Figure 5-1, we can see that the false positives and detection delays are calculated firstly for each basic detector and then the related results are stored in the dataset. After that we can normalize the detection delays and false positives separately and then calculate the vectors' weights separately. In this way, for each basic detector, two sets of weights are generated separately, with false positives denoted as $\overrightarrow{FP}$ and a detection delay denoted as $\overrightarrow{DD}$. For any detector that does not detect a correct drift, the vectors' weights in both $\overrightarrow{FP}$ and $\overrightarrow{DD}$ are set to 0. Next, we use the following formula to combine the two sets of weights $\overrightarrow{FP}$ and $\overrightarrow{DD}$ to obtain a single weight.

$$\overrightarrow{W} = \alpha\,\overrightarrow{FP} + (1-\alpha)\overrightarrow{DD} \tag{5.3}$$

where α is a fusion parameter (the parameter between 0 and 1), a larger value of α indicates that fewer false positives are more significant compared to shorter detection delays. There is a trade-off between reduced detection delay and increased false positives. If α is set to 0, it indicates that we focus entirely on the shorter detection delays, and the number of false positives is ignored. If α is set to 1, we care more about false positives, and the detection delays may be ignored. In our experiments, we set the value of α to 0.5 to balance the detection delays and false positives. Then the weights learned in the current training course are merged with the set of weights learned in the previous training course in order to obtain a final weights of the ensemble, based on the following formula:

$$\overrightarrow{W_f} = \beta\,\overrightarrow{W_c} + (1-\beta)\overrightarrow{W_f} \tag{5.4}$$

where β is a smoothing parameter (the user-defined parameter ranging from 0 to 1) that is used for balancing the significance of history information with current information. The more the value of β tends to 1, the higher the importance of the current information, and the more it tends to 0, the higher the importance of the historical information. Our model assumes that current information is as important as historical information, so the value of β is set to 0.5.

5.4 Experiments and Results

In this section, we discuss the two well-known smart home CASAS and ARAS datasets, our experiment setup, and evaluation results. We evaluate the LED-ARF model and compare it with other well-known online learning models such as Hoeffding Tree (HT), Hoeffding Adaptive Tree (HAT), KNN, Naive Bayes (NB), Online Boosting (OBo), Oza Bagging (OzBa), and Adaptive Random Forest (ARF) to evaluate the performance of multi-resident HAR.

5.4.1 Smart Home Datasets

In the field of multi-resident activity recognition, there have been many non-intrusive sensor-based datasets released. Our research is dedicated to solving complex activity recognition among multiple inhabitants in order to be more relevant to real life. The datasets used in this study can be seen in Table 5-1.

Table 5-1 Non-intrusive Sensor-based Datasets

Dataset	# of Residents	Duration	# of Sensors	# of Activities
Multi-resident ADLs (CASAS)	2	2 months	37	15
ARAS (House A)	2	30 days	20	27
ARAS (House B)	2	30 days	20	27

Table 5-2 Settings and Activities in CASAS Datasets

	Multi-resident ADLs (CASAS)
# of Ambient Sensors	37 of 7 different types
House Information	Two bedrooms, one living room, one kitchen, one bathroom

Residents	2
Duration	2 months
# of Activities	15
Activities	Filling medication dispenser, Moving furniture, Watering plants, Playing checkers, Preparing dinner, Reading magazine, Gathering and packing picnic food, Hanging up clothes, Sweeping floor, Setting the table, Paying bills

In this study, we used two datasets, one is the Multiresident ADL dataset published by the Centre for Advanced Studies in Adaptive Systems (CASAS), and the other is the ARAS (Activity Recognition with Ambient Sensing) dataset published by Alemdar et al [80]. The CASAS dataset consists of 15 types of activities of two residents being monitored for two months via 37 environmental sensors. The activities collected include included sequential, interactive, parallel, and collaborative activities, which are shown in Table 5-2 . In addition, the CASAS's data contains seven different ambient sensors, such as motion sensors, water sensors, and temperature sensors, which are listed in Table 5-3.

Table 5-3 Sensors in CASAS Datasets

Sensors	**Number**
Motion sensors	27
Door sensors	8
Item sensors	2

The ARAS dataset is divided into two sub-datasets, House A and House B. Twenty different types of environmental sensors were installed inside each house, and multi-resident activity data were collected over a period of one month. The data collection process was completely free of any external interference, and the residents performed their activities exactly as they wish. There were 27 pre-designed activities, but during the actual data collection phase, some activities did not occur. For example, there is a dishwashing activity on the list, House B has a washing machine, but the residents did not prefer to annotate the activity during data collection. The activities in the ARAS datasets are shown in Table 5-4. We believe that the data collected from such unrestricted multi-resident activities can give a realistic representation of the situations in real-life and are therefore suitable for the study of multi-resident complex HAR. Table 5-5 lists the sensors used in different houses.

Table 5-4 Activities in ARAS Datasets

House A	House B
-Watching TV	-Napping
-Going out	-Toileting
-Other	-Shaving
-Sleeping	-Having shower
-Having breakfast	-Cleaning
-Preparing breakfast	-Reading book
-Studying	-Changing clothes
-Having lunch	-Brushing teeth
-Preparing lunch	-Having conversation
-Washing dishes	-Laundry
-Having dinner	-Listening to music
-Having snack	-Using Internet
-Preparing dinner	-Talking on the phone
	-Having guest

Table 5-5 Sensor Infrastructure Used by ARAS

House A	House B
1 Wardrobe photocell	2 Kitchen cupboards CSs
1 Convertible couch photocell	1 House DCS
(Resident 2's bed)	2 Wardrobe DCSs
1 TV infrared receiver	1 Shower cabinet DCS
2 Couch force sensors	1 Tap distance sensor
2 Chair proximity sensors	3 Chair force sensors
1 Fridge photocell	11 Fridge photocell
1 Kitchen drawer photocell	2 Kitchen drawer photocell
1 Wardrobe photocell	2 Couch pressure mat
1 Bathroom cabinet photocell	1 Bed pressure mat
1 House DCS	1 Armchair pressure mat
1 Bathroom DCS	1 Bathroom door sonar distance
1 Shower cabinet DCS	1 Kitchen sonar distance
1 Hall sonar distance	1 Closet sonar distance
1 Kitchen sonar distance	
1 Tap proximity sensor	
1 Water closet proximity sensor	
1 Kitchen temperature sensor	
1 Bed force sensor	

5.4.2 Experiment Setup

Our experiments are conducted based on the CASAS and ARAS datasets. For the CASAS dataset, 17258 sensor events are collected. For the ARAS dataset, sensor-reading data is recorded in units of seconds, so a total of 86400 sensor sequence data are captured per day and continue for 30 days. We treat each sensor time as a data stream and continuously update the model by analyzing each data stream. We evaluate and compare the performance of different online learning models based on the accuracy, recall, precision, and f-measure.

For parameter settings, we use the default parameters of various online models in the Massive Online Analysis (MOA) platform. Specifically, for Hoeffding Tree Classifier, grace_peried = 200, split_criterion = Information Gain, leaf_prediction = Naïve Bayes Adaptive. For Hoeffding Adaptive Tree, we apply the same settings with Hoeffding Tree, and additional ADWIN drift detection is used for detecting the performance decreases. For KNN, n_neighbors = 5 and leaf_size = 30. For Online Boosting Classifier, KNN with ADWIN concept drift is applied and the size of the ensemble = 10. Adaptive Random Forest (ARF) use ADWIN (delta = 0.001) for drift detection and n_estimators = 10, split_criterion = Information Gain. We used the same parameters as [132] in Locally-weighted Ensemble Detection (LED).

5.4.3 Results of Various OL Models on CASAS Dataset

We evaluate our model against commonly used OL models for HAR, including HT, HAT, KNN, NB, OBo, OzBa, and ARF. Figure 5-3 and Table 5-6 show the performance results obtained from all the OL models using single label and combine label (CL) methods. Figure 5-4 shows an example of online recognition by using the LED-ARF model on the CASAS's R1 data as in CASAS, R1, R2 and CL methods have the similar performance.

Table 5-6 Models Comparison on CASAS Dataset

Metric (%)	Model	R1	R2	CL
	HT	80.58	79.54	82.13
	HAT	81.25	80.36	83.66
	KNN	70.22	70.34	71.85
Accuracy	NB	75.25	78.21	78.12
	OBo	81.32	80.45	82.98
	OzBa	78.78	79.48	80.69
	ARF	82.68	83.47	83.27

	LED-ARF	**83.54**	**84.11**	**85.82**
Precision	HT	75.22	74.18	76.77
	HAT	77.55	76.66	79.96
	KNN	67.32	67.44	68.95
	NB	71.57	74.53	74.44
	OBo	79.19	78.32	80.85
	OzBa	75.93	76.63	77.84
	ARF	80.00	80.79	80.59
	LED-ARF	**81.40**	**81.97**	**83.68**
Recall	HT	75.55	74.51	77.10
	HAT	77.91	77.02	80.32
	KNN	67.86	67.98	69.49
	NB	71.91	74.87	74.78
	OBo	79.84	78.97	81.50
	OzBa	76.51	77.21	78.42
	ARF	80.23	80.02	80.82
	LED-ARF	**81.83**	**82.40**	**84.11**
F-measure	HT	75.38	74.34	76.93
	HAT	77.73	76.84	80.14
	KNN	67.59	67.71	69.22
	NB	71.74	74.70	74.61
	OBo	79.51	78.64	81.17
	OzBa	76.22	76.92	78.13
	ARF	80.11	80.90	80.70
	LED-ARF	**81.61**	**82.18**	**83.89**

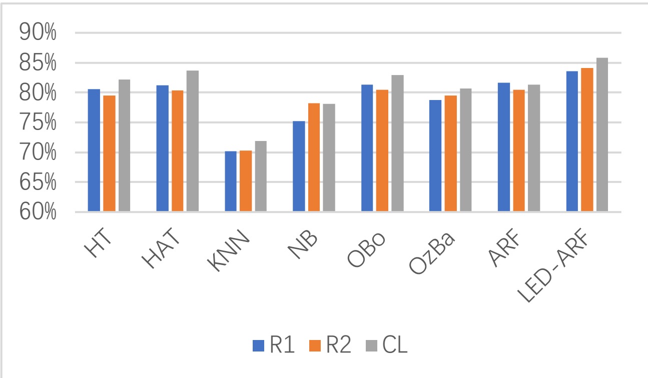

Figure 5-3 Accuracy of multi-resident HAR in CASAS dataset

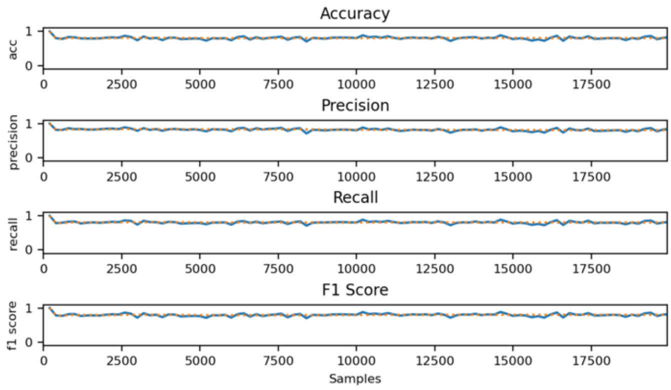

Figure 5-4 Online recognition results by using LED-ARF (CASAS R1). The blue realization indicates current performance and the orange dashed line indicates tied performance.

We can see that our proposed model outperforms the other online recognition models. Specifically, our recognition accuracy reached 83.54% for R1, 84.11% for R2, and 85.82% using the CL method. The CL method is slightly higher than identifying R1 and R2 separately. The KNN model achieved the worst performance, with the recognition accuracy of 70.22%, 70.34%, and 71.85% for R1, R2, and CL, respectively. Because the KNN model is particularly dependent on the training data, it is too error-tolerant to the training data. If there is data in the training dataset that is wrong, this will directly lead to inaccurate prediction data if the next data is to be classified. In addition, HAT and Obo achieve relatively good performance, with the accuracy exceeding 80% for both the individual recognition of each resident and the overall recognition. From the technical point of view, Hoeffding tree is a decision tree-based stream classification method, which can guarantee the mining efficiency while achieving the requirements of some necessary operations on data streams when processing them. The algorithm only needs to check each sample in the data stream once and generate the decision tree step by step as these samples are updated in the decision tree without storage. Only the information about the decision tree needs to be kept in memory, because the information needed to extend the decision tree is stored in the leaf nodes, the information in the decision tree can be used to make predictions when processing the training data. The drawback of the Hoeffding Tree model is that there is no design in terms of decision tree adjustment.

From our experiments, we can see that our proposed model can accurately recognize the activities of multiple residents. Next, we evaluate our proposed model with a more complex ARAS dataset.

5.4.4 Results of Various OL Models on ARAS Datasets

As reported in Figure 5-5 and Table 5-7 for the ARAS dataset's performance, we can see that ARF achieves the highest accuracy of 84.88% for R1 on House A and 83.43% on House B, but it is marginally 0.12% and 0.13% more accurate than our LED-ARF model. However, the LED-ARF model outperforms the ARF by 1.88% and 0.77% for R1 recognition and using the CL approach, respectively. Besides, LED-ARF outperforms all other OL models in terms of accuracy, achieving 83.40%, 84.12%, and 85.60% for recognition of R1 and R2 on House B and the CL method, respectively. Therefore, we can conclude that our LED-ARF classifier achieves better and relatively stable performance in different multi-resident recognition datasets.

KNN and NB have achieved bad performance, with lower accuracy compared to the other models. Interestingly, although the ARAS dataset includes more activities and is more complex than CASAS, the performance obtained on ARAS is similar to that of CASAS. Our observation is that the ARAS dataset has more types of sensors used on different objects, which enables a high probability of better and accurate activity segmentation thus improving the recognition accuracy.

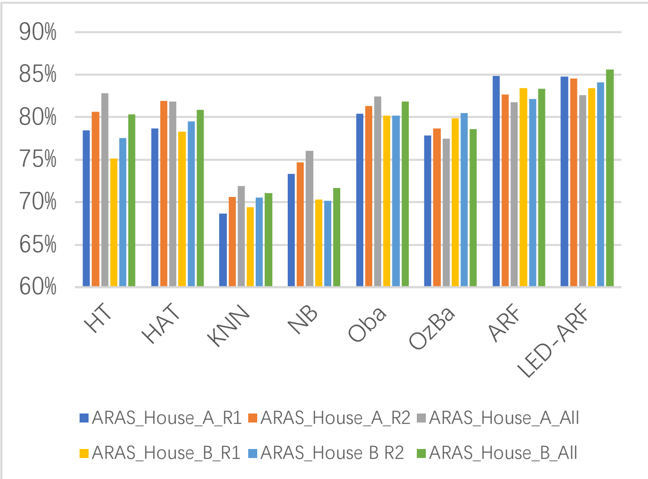

Figure 5-5 Accuracy of various OL models on ARAS datasets

Table 5-7 Different OL Models on ARAS Datasets

Metric (%)	Model	ARAS House A			ARAS House B		
		R1	R2	CL	R1	R2	CL
	HT	78.45	80.64	82.81	75.13	77.54	80.35
	HAT	78.68	81.93	81.84	78.32	79.48	80.85
Accuracy	KNN	68.65	70.62	71.90	69.41	70.55	71.03
	NB	73.31	74.65	76.02	70.32	70.18	71.68
	OBo	80.41	81.33	82.46	80.15	80.14	81.86

	OzBa	77.82	78.67	77.44	79.89	80.48	78.60
	ARF	**84.88**	82.63	81.79	**83.43**	82.10	83.32
	LED-ARF	84.76	**84.51**	**82.56**	83.40	**84.12**	**85.60**
Precision	HT	73.09	75.28	77.45	69.77	72.18	75.22
	HAT	74.98	78.23	78.14	74.62	75.78	77.55
	KNN	65.75	67.72	69.00	66.51	67.65	67.32
	NB	69.63	70.97	72.34	66.64	66.50	71.57
	OBo	78.28	79.20	80.33	78.02	78.01	79.19
	OzBa	74.97	75.82	74.59	77.04	77.63	75.93
	ARF	**83.20**	80.95	80.11	**81.75**	80.42	81.00
	LED-ARF	82.62	**82.37**	**80.42**	81.26	**81.98**	**81.40**
Recall	HT	73.42	75.61	77.78	70.10	72.51	75.55
	HAT	75.34	78.59	78.50	74.98	76.14	77.91
	KNN	66.29	68.26	69.54	67.05	68.19	67.86
	NB	69.97	71.31	72.68	66.98	66.84	71.91
	OBo	78.93	79.85	80.98	78.67	78.66	79.84
	OzBa	75.55	76.40	75.17	77.62	78.21	76.51
	ARF	**83.43**	81.18	80.34	**81.98**	80.65	81.23
	LED-ARF	83.05	**82.80**	**80.85**	81.69	**82.41**	**81.83**
F-measure	HT	73.25	75.44	77.61	69.93	72.34	75.38
	HAT	75.16	78.41	78.32	74.80	75.96	77.73
	KNN	66.02	67.99	69.27	66.78	67.92	67.59
	NB	69.80	71.14	72.51	66.81	66.67	71.74
	OBo	78.60	79.52	80.65	78.34	78.33	79.51
	OzBa	75.26	76.11	74.88	77.33	77.92	76.22
	ARF	**83.31**	81.06	80.22	**81.86**	80.53	81.11
	LED-ARF	82.83	**82.58**	**80.63**	81.47	**82.19**	**81.61**

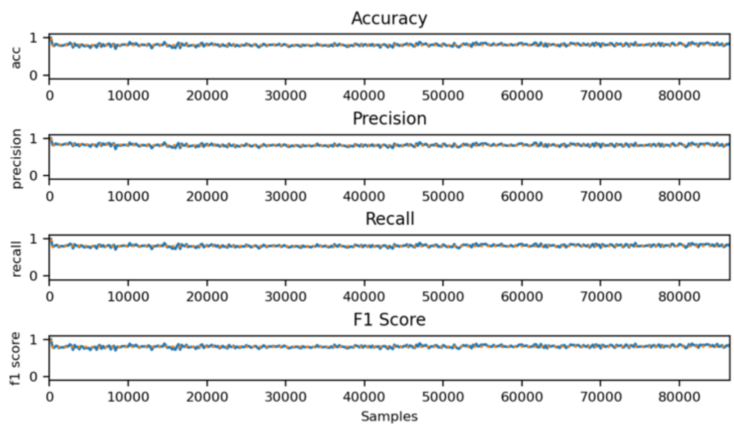

Figure 5-6 LED-ARF model on ARAS_A_R1

5.4.5 Statistical Comparison

According to the classification results, the ARF model and the LED-ARF model are outstanding compare with the other OL models. We apply a famous nonparametric Wilcoxon singed-rank test, to evaluate which OL model is significantly better. We analyze the singed of differences by calculating the different scores of each model.

Table 5-8 Comparison of Our Proposed Method with ARF Model

Dataset	Resident	LED-ARF	ARF	Difference	Rank
CASAS	R1	83.54	82.68	0.86	5
	R2	84.11	83.47	0.64	3
	CL	85.82	83.27	2.55	9
ARAS House A	R1	84.76	84.88	-0.12	1
	R2	84.51	82.63	1.88	6
	CL	82.56	81.79	0.77	4
ARAS House A	R1	83.40	83.43	-0.03	2
	R2	84.12	82.10	2.02	7
	CL	85.60	83.32	2.28	8

In the test, we calculate the difference d_i between the accuracy of the LED-ARF and ARF models on three datasets first. Then, absolute values are utilized for differences ranking. Table 5-8 presents various accuracy of classification models and ranks corresponding to the two approaches. Afterward, R^+ and R^- are calculated, where R^+ is the sum of the ranks for positive and R^- refers to negative. In this case, the values of R^+ and R^- are 42 and 3, respectively. According to the Wilcoxon signed-rank test's critical values with a confidence level of $\alpha = 0.05$, the difference between the two models is considered significant if the smaller sums are equal or less than 10. Therefore, we conclude that the LED-ARF model is statistically superior to the ARF model.

5.5 Discussion and Conclusion

Data analytics systems contribute effectively to behavior recognition and smart homes, driving the development of these technologies. Since data streams are an important source for these analyses, developing techniques to handle online data streams rather than batch data sets is a logical choice for current data science research interests. This paper attempts to investigate popular online data stream classification techniques for the recognition of multi-resident activities, especially adaptive random forest classifiers, achieving advantages including in handling large amounts of data. It also uses the technique of concept drift to improve the performance of recognition.

In the future, we will explore the use of active learning techniques to label multiple labels in multi-resident recognition and more complex online multi-resident HAR problems.

Chapter 6 Conclusion and Future Work

In this thesis, we solve three subproblems of multi-resident activity recognition, namely, segmentation, classification, and online recognition. In this chapter, we first summarise the contributions of this thesis and answer the research questions in detail. Based on the current research, we then propose future work.

6.1 Summary of Contributions

This thesis focused on improving the performance of multi-resident HAR. Specifically, we studied three subproblems: 1) improving the performance by segmenting the sensor events method; 2) improving the performance using an advanced deep learning model; 3) recognising multi-resident activities in real-time.

Chapter 2 comprehensively surveyed multi-resident activity recognition in non-intrusive sensor-based environments. We extensively reviewed and analysed existing work related to activity segmentation techniques and classification approaches, highlighting their advantages and limitations. We then discussed evaluation methods and presented publicly available datasets for non-intrusive sensor-based multi-resident HAR. Finally, we summarised the research challenges and open issues requiring further study in this field.

Chapter 3 studied the problem of segmenting the sensor events. We comprehensively evaluated the segmentation approaches that improve the performance of multi-resident AR in smart-home environments. We evaluated and compared our proposed Hybrid Fuzzy C-means CPD-based Segmentation method with the well-known segmentation methods, including static time-based, static sensor-based, and dynamic correlation-based segmentation. Our method can accurately segment multi-resident sensor events and significantly improves the overall performance (accuracy and F-measure) of complex activities. At this time, we observed that parallel activities gained higher recognition than collaborative activities. In future work, we plan to improve the dynamic sensor-events segmentation algorithm and evaluate our method in a more complex setting with a larger number of involved people. For this purpose, we will perform our own data collection.

Chapter 4 studied the problem of applying an advanced deep learning model. We proposed a Transformer with a Bidirectional GRU deep learning method, TRANS-BiGRU, which apparently improves HAR in a multi-resident environment. After a comprehensive comparison of various HAR models, it was found that RNN-based models (LSTM, GRU, Ens2-LSTM, Bi-LSTM, and Bi-GRU) outperformed other DL models (DNN, CNN) and the most commonly used ML models (FHMM, NB, DT) on the CASAS and ARAS datasets, indicating that RNN-based models profit the classification of time series data.

The performances of the RNN-based models (LSTM, GRU Ens2-LSTM, Bi-LSTM, Bi-GRU, and Transformer architecture) were then compared with that of our proposed TRANS-BiGRU. The Bi-GRU and Transformer models slightly outperformed LSTM and GRU. The Ens2-LSTM model combines a basic LSTM and a Bi-LSTM model. The LSTM model is applied to simple activities, and Bi-LSTM analyses more complicated activities and improves the performance. Our proposed model integrates the Transformer architecture with Bi-GRU to recognise the involved activities and boost the HAR performance.

Chapter 5 studied the problem of recognising multi-resident activities in real-time. We investigated the ability of popular online data-stream classification techniques to recognise multi-resident activities. Adaptive random forest classifiers were advantageous for handling large amounts of data. The concept drift technique improves the recognition performance. We also proposed a novel Locally-weighted Ensemble Detection-based Adaptive Random Forest Classifier (LED-ARF) for online analysis of multi-resident identification. We comprehensively compared the performances of eight popular online learning classification algorithms. In the experiments, our LED-ARF model achieved the highest performance on online multi-resident HAR.

6.2 Limitations

In this research, we addressed three problems in multi-resident activity recognition. Although our proposed methods significantly improved the performance of multi-resident HAR, the following limitations should be rectified in future: (1) The datasets contain only some simple daily activities collected from non-intrusive ambient sensors, and only two residents were engaged in the activities; (2) The recognition performance requires improvement; (3) The proposed methods can only identify the current activity; they cannot predict the upcoming activity. Each of these limitations is discussed below.

(1) The datasets contain only some simple daily activities of two residents collected from non-intrusive ambient sensors.

Multi-resident activity recognition should be performed not only for simple activities (eating, working, sleeping) of two residents only but also for complex activities of three or more residents.

Whether our proposed model can effectively recognise the activities of three or more residents is not certain. The proposed models should be evaluated on more sophisticated datasets in the future.

(2) The recognition performance must be improved.

Although both our proposed segmentation method and classification model showed higher recognition performance than state-of-the-art methods, there is still much room for improvement. The HAR framework includes data collection, data processing, classification, and action prediction. Any inappropriate data processing or model parameter adjustment may degrade the performance. Specific data processing and model parameter tuning to improve the recognition performance will be attempted in future.

(3) The proposed methods can only identify the previous and current action; they cannot predict future activities.

Our study uses both offline and online learning models to identify the activity currently performed by the resident. However, in real life, activity recognition must also predict possible abnormal future activities that have not yet occurred.

6.3 Future Work

The limitations of the research were discussed in the previous section. Based on these limitations, this section presents some research avenues requiring further exploration. Open to discussion are evaluations on complex high-level activity datasets, improving the recognition performance, and predicting upcoming actions. The research directions on these critical themes are suggested below.

(1) Datasets for non-intrusive complex activities of multi-residents

Quantitative comparisons of multi-resident activity recognition approaches are not straightforward as different studies relay on different datasets [40]. The lack of standardised benchmarks hinders the fair and exhaustive evaluation of multi-resident HAR methods. Most of the existing studies are tested on the CASAS and ARAS datasets [7, 42] which are limited in scope. Both datasets contain only a portion of the daily activities performed by at most two residents. More datasets containing the activities of more than two people performing various complex activities are needed. Also, the existing datasets use simple environmental sensors, which are rather impractical and inconvenient to install. As sensor technology evolves, its effectiveness must be evaluated in further research. In particular, we must trial different sensor technologies and assess whether the number of sensors can be minimised to improve the ease of deployment without degrading the recognition performance.

(2) Improving the performance of multi-resident HAR

The study of multi-resident activity recognition is still in its infancy. Actions between different residents interfere and are much more complicated than single-resident actions, increasing the difficulty of recognition. Although our proposed segmentation method and classification method outperformed the state-of-the-art methods in recognition tasks, there is still massive room for improvement. In future work, we will build an ensemble recognition architecture that combines a segmentation algorithm and classification model to improve the recognition performance.

(3) Prediction of future activities

Current solutions can only recognise past and current activities, but real situations require more intelligent solutions that can recognise more kinds of activities, abnormal activities, and potential future activities [43]. Current HAR systems can recognise many actions and bring benefits to many real-life situations, but whether they can predict future activities is an interesting and challenging question. This function is critical, especially in applications such as fall detection and prevention. A possible research direction is building a HAR system that can predict future activities based on past activities [92].

References

[1] V. Soto-Mendoza, J. Beltr´an, E. Ch´avez, J. u. Hern´andez, and J. A. Garc´ıa-Mac´ıas, "Abnormal Behavioral Patterns Detection from Activity Records of Institutionalized Older Adults," presented at the 6th International Workshop, HBU 2015, Osaka,Japan, 2015.

[2] R. A. Hamad, A. S. Hidalgo, M.-R. Bouguelia, M. E. Estévez, and J. M. Quero, "Efficient

Activity Recognition in Smart Homes Using Delayed Fuzzy Temporal Windows on Binary Sensors," *IEEE J. Biomedical and Health Informatics,* vol. 24, no. 2, pp. 387-395, / 2020, doi: 10.1109/JBHI.2019.2918412.

[3] H. T. Malazi and M. Davari, "Combining emerging patterns with random forest for complex activity recognition in smart homes," *Appl Intell,* 2017, doi: 10.1007/s10489-017-0976-2.

[4] D. Riboni and M. Murtas, "Sensor-based activity recognition: One picture is worth a thousand words," *Future Gener. Comput. Syst.,* vol. 101, pp. 709-722, / 2019, doi: 10.1016/j.future.2019.07.020.

[5] H. Alemdar and C. Ersoy, "Multi-resident activity tracking and recognition in smart environments," (in English), *J Amb Intel Hum Comp,* vol. 8, no. 4, pp. 513-529, Aug 2017, doi: 10.1007/s12652-016-0440-x.

[6] M. Amiribesheli, A. Benmansour, and A. Bouchachia, "A review of smart homes in healthcare," *Journal of Ambient Intelligence and Humanized Computing,* vol. 6, no. 4, pp. 495-517, 2015/08/01 2015, doi: 10.1007/s12652-015-0270-2.

[7] A. Benmansour, A. Bouchachia, and M. Feham, "Modeling interaction in multi-resident activities," *Neurocomputing,* vol. 230, no. Supplement C, pp. 133-142, 2017/03/22/ 2017, doi: https://doi.org/10.1016/j.neucom.2016.05.110.

[8] M. Z. Uddin, "Human activity recognition using segmented body part and body joint features with hidden Markov models," *Multimedia Tools and Applications,* vol. 76, no. 11 %@ 1573-7721, pp. 13585-13614, 2017, doi: 10.1007/s11042-016-3742-2 %U https://doi.org/10.1007/s11042-016-3742-2.

[9] J. Yin, M. Fang, G. Mokhtari, and Q. Zhang, "Multi-resident Location Tracking in Smart Home through Non-wearable Unobtrusive Sensors," 2016. [Online]. Available: https://doi.org/10.1007/978-3-319-39601-9_1.

[10] Y. Kashimoto, M. Fujiwara, M. Fujimoto, H. Suwa, Y. Arakawa, and K. Yasumoto, "ALPAS: Analog-PIR-Sensor-Based Activity Recognition System in Smarthome," 2017. [Online]. Available: https://doi.org/10.1109/AINA.2017.33.

[11] Y. Song, Z. Wen, C.-Y. Lin, and R. Davis, "One-Class Conditional Random Fields for Sequential Anomaly Detection," 2013.

[12] M. A. U. Alam, N. Roy, A. Misra, and J. Taylor, "CACE: Exploiting Behavioral Interactions for Improved Activity Recognition in Multi-Inhabitant Smart Homes," (in English), *Int Con Distr Comp S,* pp. 539-548, 2016, doi: 10.1109/Icdcs.2016.61.

[13] A. Alaa, M. Vaidehi, B. Doreen, hnstedt, and S. Ralf, "Activity Recognition in Multi-User Environments Using Techniques of Multi-label Classification," in *Proceedings of the 6th International Conference on the Internet of Things %@ 978-1-4503-4814-0*, ed. Stuttgart, Germany: ACM, 2016, pp. 15-23.

[14] R. Daniele, S. Timo, C. Gabriele, and S. Heiner, "Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing %@ 978-1-4503-4461-6*, ed. Heidelberg, Germany: ACM, 2016, pp. 1-12.

[15] M. J. O. G. Jie Wan "Dynamic sensor event segmentation for real-time activity recognition in a smart home context," *Pers Ubiquit Comput,* vol. 19, pp. 287–301, 2015.

[16] M. H. M. Noor, Z. A. Salcic, and K. I. K. Wang, "Adaptive sliding window segmentation

for physical activity recognition using a single tri-axial accelerometer," *Pervasive and Mobile Computing,* vol. 38, pp. 41-59, / 2017, doi: 10.1016/j.pmcj.2016.09.009.
[17] H. F. Nweke, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," *Expert Systems With Applications,* pp. 233-261, 2018, doi: doi.org/10.1016/j.eswa.2018.03.056.
[18] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Generation Comp. Syst.,* vol. 81, pp. 307-313, / 2018, doi: 10.1016/j.future.2017.11.029.
[19] P. Asghari, E. Soleimani, and E. Nazerfard, "Online human activity recognition employing hierarchical hidden Markov models," *J. Ambient Intell. Humaniz. Comput.,* vol. 11, no. 3, pp. 1141-1152, / 2020, doi: 10.1007/s12652-019-01380-5.
[20] Y. Tang, F. Ren, and W. Pedrycz, "Fuzzy C-Means clustering through SSIM and patch for image segmentation," *Appl. Soft Comput.,* vol. 87, p. 105928, / 2020, doi: 10.1016/j.asoc.2019.105928.
[21] D. Triboan, L. Chen, F. Chen, and Z. Wang, "Semantic segmentation of real-time sensor data stream for complex activity recognition," *Personal and Ubiquitous Computing,* vol. 21, no. 3, pp. 411-425, 2017/06/01 2017, doi: 10.1007/s00779-017-1005-5.
[22] E. Garcia-Ceja, C. E. Galván-Tejada, and R. Brena, "Multi-view stacking for activity recognition with sound and accelerometer data," *Information Fusion,* vol. 40, no. Supplement C, pp. 45-56, 2018/03/01/ 2018, doi: https://doi.org/10.1016/j.inffus.2017.06.004.
[23] A. R. M. Forkan, I. Khalil, Z. Tari, S. Foufou, and A. Bouras, "A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living," *Pattern Recognition,* vol. 48, no. 3, pp. 628-641, / 2015, doi: 10.1016/j.patcog.2014.07.007.
[24] P. Lago, C. Jiménez-Guarín, and C. Roncancio, "Contextualized Behavior Patterns for Ambient Assisted Living," presented at the 6th International Workshop, HBU 2015, Osaka,Japan, 2015.
[25] W. Taylor, S. A. Shah, K. Dashtipour, A. Zahid, Q. H. Abbasi, and M. A. Imran, "An Intelligent Non-Invasive Real-Time Human Activity Recognition System for Next-Generation Healthcare," *Sensors,* vol. 20, no. 9, p. 2653, / 2020, doi: 10.3390/s20092653.
[26] G. v. Capelleveena, M. Poel, R. M. Mueller, D. Thornton, and J. v. Hillegersberg, "Outlier detection in healthcare fraud: A case study in the Medicaid dental domain," *International Journal of Accounting Information Systems,* pp. 18-31, 2016, doi: http://dx.doi.org/10.1016/j.accinf.2016.04.001.
[27] G. Xinze, R. Raviv, and W. Weng-Keen, "Efficient Multi-Instance Learning for Activity Recognition from Time Series Data Using an Auto-Regressive Hidden Markov Model," in *Proceedings of The 33rd International Conference on Machine Learning* vol. 48 %W PMLR, B. Maria Florina and Q. W. Kilian, Eds., ed. Proceedings of Machine Learning Research: PMLR, 2016, pp. 2330--2339 %U http://proceedings.mlr.press.
[28] J. Wen and Z. Wang, "Sensor-based adaptive activity recognition with dynamically available sensors," *Neurocomputing,* vol. 218, no. Supplement C, pp. 307-317, 2016/12/19/ 2016, doi: https://doi.org/10.1016/j.neucom.2016.08.077.
[29] J. Wen and Z. Wang, "Learning general model for activity recognition with limited

labelled data," *Expert Systems with Applications,* vol. 74, no. Supplement C, pp. 19-28, 2017/05/15/ 2017, doi: https://doi.org/10.1016/j.eswa.2017.01.002.

[30] L. Meng, C. Miao, and C. Leung, "Towards online and personalized daily activity recognition, habit modeling, and anomaly detection for the solitary elderly through unobtrusive sensing," *Multimedia Tools and Applications,* vol. 76, no. 8 %@ 1573-7721, pp. 10779-10799, 2017, doi: 10.1007/s11042-016-3267-8 %U https://doi.org/10.1007/s11042-016-3267-8.

[31] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Device-Free Human Activity Recognition Using Commercial WiFi Devices," *IEEE Journal on Selected Areas in Communications,* vol. 35, no. 5, pp. 1118-1131, / 2017, doi: 10.1109/JSAC.2017.2679658.

[32] C. E. Galván-Tejada *et al.*, "An Analysis of Audio Features to Develop a Human Activity Recognition Model Using Genetic Algorithms, Random Forests, and Neural Networks," *Mobile Information Systems,* vol. 2016, pp. 1784101:1-1784101:10, / 2016, doi: 10.1155/2016/1784101.

[33] Y. Kashimoto *et al.*, "Low-cost and Device-free Activity Recognition System with Energy Harvesting PIR and Door Sensors," 2016. [Online]. Available: http://doi.acm.org/10.1145/3004010.3006378.

[34] M. Á. Á. d. l. Concepción, L. M. Soria-Morillo, J. A. Álvarez-García, and L. G. Abril, "Mobile activity recognition and fall detection system for elderly people using Ameva algorithm," *Pervasive and Mobile Computing,* vol. 34, pp. 3-13, / 2017, doi: 10.1016/j.pmcj.2016.05.002.

[35] Q. Li *et al.*, "Multi-resident type recognition based on ambient sensors activity," *Future Gener. Comput. Syst.,* vol. 112, pp. 108-115, / 2020, doi: 10.1016/j.future.2020.04.039.

[36] I. A. Emi and J. A. Stankovic, "SARRIMA: smart ADL recognizer and resident identifier in multi-resident accommodations," 2015. [Online]. Available: http://doi.acm.org/10.1145/2811780.2811916.

[37] J. Ye, G.-J. Qi, N. Zhuang, H. Hu, and K. A. Hua, "Learning Compact Features for Human Activity Recognition Via Probabilistic First-Take-All," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 42, no. 1, pp. 126-139, / 2020, doi: 10.1109/TPAMI.2018.2874455.

[38] J. Zhao, X. Xie, X. Xu, and S. Sun, "Multi-view learning overview: Recent progress and new challenges," *Information Fusion,* vol. 38, pp. 43-54, / 2017, doi: 10.1016/j.inffus.2017.02.007.

[39] R. Mohamed, T. Perumal, M. N. Sulaiman, and N. Mustapha, "Multi-resident activity recognition using label combination approach in smart home environment," 2017. [Online]. Available: https://doi.org/10.1109/ISCE.2017.8355551.

[40] W. Yuan, J. Cao, Z. Jin, F. Xia, and R. Wang, "A Multi-resident Activity Recognition Approach based on Frequent Itemset Mining Features," 2019. [Online]. Available: https://doi.org/10.1109/INDIN41052.2019.8972016.

[41] A. Natani, A. Sharma, T. Perumal, and S. Sukhavasi, "Deep Learning for Multi-Resident Activity Recognition in Ambient Sensing Smart Homes," 2019. [Online]. Available: https://doi.org/10.1109/GCCE46687.2019.9015212.

[42] S. N. Tran, Q. Zhang, and M. Karunanithi, "On Multi-resident Activity Recognition in Ambient Smart-Homes," *CoRR,* vol. abs/1806.06611, / 2018. [Online]. Available: http://arxiv.org/abs/1806.06611.

[43] L. Minh Dang, K. Min, H. Wang, M. Jalil Piran, C. Hee Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognition,* vol. 108, p. 107561, 2020/12/01/ 2020, doi: https://doi.org/10.1016/j.patcog.2020.107561.

[44] A. Benmansour, A. Bouchachia, and M. Feham, "Multioccupant Activity Recognition in Pervasive Smart Home Environments," (in English), *Acm Comput Surv,* vol. 48, no. 3, Dec 2015, doi: Artn 3410.1145/2835372.

[45] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, and Y. Liu, "Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges and Opportunities," *CoRR,* vol. abs/2001.07416, / 2020. [Online]. Available: https://arxiv.org/abs/2001.07416.

[46] R. M. Raval, H. B. Prajapati, and V. K. Dabhi, "Survey and analysis of human activity recognition in surveillance videos," *Intell. Decis. Technol.,* vol. 13, no. 2, pp. 271-294, / 2019, doi: 10.3233/IDT-170035.

[47] S. R. Ramamurthy and N. Roy, "Recent trends in machine learning for human activity recognition - A survey," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.,* vol. 8, no. 4, / 2018, doi: 10.1002/widm.1254.

[48] H. T. Malazi and M. Davari, "Combining emerging patterns with random forest for complex activity recognition in smart homes," *Appl. Intell.,* vol. 48, no. 2, pp. 315-330, / 2018, doi: 10.1007/s10489-017-0976-2.

[49] T.-H. T. Munkhjargal Gochoo, "Unobtrusive Activity Recognition of Elderly People Living Alone Using Anonymous Binary Sensors and DCNN," *IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS,* vol. VOL. 23, NO. 2, pp. 693-702, 2019.

[50] H. Sfar and A. Bouzeghoub, "DataSeg: dynamic streaming sensor data segmentation for activity recognition," 2019. [Online]. Available: https://doi.org/10.1145/3297280.3297332.

[51] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors," *IEEE Communications Surveys and Tutorials,* vol. 15, no. 3, pp. 1192-1209, / 2013, doi: 10.1109/SURV.2012.110112.00192.

[52] J. Li, L. Tian, H. Wang, Y. An, K. Wang, and L. Yu, "Segmentation and Recognition of Basic and Transitional Activities for Continuous Physical Human Activity," *Ieee Access,* vol. 7, pp. 42565-42576, / 2019, doi: 10.1109/ACCESS.2019.2905575.

[53] Q. Zhao, C.-M. Tsai, R.-C. Chen, and C.-Y. Huang, "Resident activity recognition based on binary infrared sensors and soft computing," *Int. J. Machine Learning & Cybernetics,* vol. 10, no. 2, pp. 291-299, / 2019, doi: 10.1007/s13042-017-0714-4.

[54] M. H. M. Noor, Z. A. Salcic, and K. I. K. Wang, "Adaptive sliding window segmentation for physical activity recognition using a single tri-axial accelerometer," *Pervasive Mob. Comput.,* vol. 38, pp. 41-59, / 2017, doi: 10.1016/j.pmcj.2016.09.009.

[55] H. Kalantarian, C. Sideris, and M. Sarrafzadeh, "A Hierarchical Classification and Segmentation Scheme for Processing Sensor Data," *IEEE J. Biomed. Health Informatics,* vol. 21, no. 3, pp. 672-681, / 2017, doi: 10.1109/JBHI.2016.2526679.

[56] G. T. Wang L, Tao X, Lu J, "A hierarchical approach to realtime activity recognition in body sensor networks," *Pervasive Mob Computer Speech & Language,* vol. 8(1), pp. 115–130, 2012.

[57] D. Chen, Yongchareon, S., Lai, E. M., Yu, J., & Sheng, Q. Z. , "Hybrid Fuzzy C-means

CPD-based Segmentation for Improving Sensor-based Multi-resident Activity Recognition," *IEEE Internet of Things Journal,* 2021, doi: 10.1109//JIOT.2021.3051574.
[58] R. A. Nadi and M. G. H. a. Zamil, "A profile based data segmentation for in-home activity recognition," *IJSNet,* vol. 29, no. 1, pp. 28-37, / 2019, doi: 10.1504/IJSNET.2019.097553.
[59] L. Santos, K. Khoshhal, and J. Dias, "Trajectory-based human action segmentation," *Pattern Recognit.,* vol. 48, no. 2, pp. 568-579, / 2015, doi: 10.1016/j.patcog.2014.08.015.
[60] H. Sfar and A. Bouzeghoub, "Dynamic Streaming Sensor Data Segmentation for Smart Environment Applications," 2018. [Online]. Available: https://doi.org/10.1007/978-3-030-04224-0_7.
[61] G. Singla and D. J. Cook, "Interleaved Activity Recognition for Smart Home residents," 2009. [Online]. Available: https://doi.org/10.3233/978-1-60750-034-6-145.
[62] R. L. Granada, J. Monteiro, R. C. Barros, and F. R. Meneguzzi, "A Deep Neural Architecture for Kitchen Activity Recognition," 2017. [Online]. Available: https://aaai.org/ocs/index.php/FLAIRS/FLAIRS17/paper/view/15411.
[63] S. Aminikhanghahi and D. J. Cook, "Enhancing activity recognition using CPD-based activity segmentation," *Pervasive Mob. Comput.,* vol. 53, pp. 75-89, / 2019, doi: 10.1016/j.pmcj.2019.01.004.
[64] M. Cho, Y. Kim, and Y. Lee, "Contextual Relationship-based Activity Segmentation on an Event Stream in the IoT Environment with Multi-user Activities," 2016. [Online]. Available: https://doi.org/10.1145/3008631.3008633.
[65] Y. Wang, Z. Fan, and A. Bandara, "Identifying activity boundaries for activity recognition in smart environments," 2016. [Online]. Available: https://doi.org/10.1109/ICC.2016.7510732.
[66] P. C. Roy, S. R. Abidi, and S. S. R. Abidi, "Possibilistic activity recognition with uncertain observations to support medication adherence in an assisted ambient living setting," *Knowl.-Based Syst.,* vol. 133, pp. 156-173, / 2017, doi: 10.1016/j.knosys.2017.07.008.
[67] M. Prossegger and A. Bouchachia, "Multi-resident Activity Recognition Using Incremental Decision Trees," 2014. [Online]. Available: https://doi.org/10.1007/978-3-319-11298-5_19.
[68] S. S. Khan, M. E. Karg, J. Hoey, and D. Kulic, "Towards the detection of unusual temporal events during activities using HMMs," 2012. [Online]. Available: http://doi.acm.org/10.1145/2370216.2370444.
[69] N. Roy, A. Misra, and D. Cook, "Ambient and smartphone sensor assisted ADL recognition in multi-inhabitant smart environments," (in English), *Journal Of Ambient Intelligence And Humanized Computing,* vol. 7, no. 1, pp. 1-19, Feb 2016, doi: 10.1007/s12652-015-0294-7.
[70] R. Mohamed, T. Perumal, M. N. b. Sulaiman, N. Mustapha, and M. N. Razali, "Conflict resolution using enhanced label combination method for complex activity recognition in smart home environment," 2017. [Online]. Available: https://doi.org/10.1109/GCCE.2017.8229477.
[71] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, "Deep Learning Models for Real-time Human Activity Recognition with Smartphones," *Mob. Networks Appl.,* vol. 25, no. 2, pp. 743-755, / 2020, doi: 10.1007/s11036-019-01445-x.
[72] D. Liciotti, M. Bernardini, L. Romeo, and E. Frontoni, "A sequential deep learning

application for recognising human activities in smart homes," *Neurocomputing,* vol. 396, pp. 501-513, / 2020, doi: 10.1016/j.neucom.2018.10.104.

[73] J. Chen, D. M. Jiang, and Y. N. Zhang, "A Hierarchical Bidirectional GRU Model With Attention for EEG-Based Emotion Classification," *Ieee Access,* vol. 7, pp. 118530-118540, / 2019, doi: 10.1109/ACCESS.2019.2936817.

[74] A. H. Mirza, M. Kerpicci, and S. S. Kozat, "Efficient online learning with improved LSTM neural networks," *Digit. Signal Process.,* vol. 102, p. 102742, / 2020, doi: 10.1016/j.dsp.2020.102742.

[75] L. Liu, Y. Peng, S. Wang, M. Liu, and Z. Huang, "Complex activity recognition using time series pattern dictionary learned from ubiquitous sensors," *Information Sciences,* vol. 340-341, no. Supplement C, pp. 41-57, 2016/05/01/ 2016, doi: https://doi.org/10.1016/j.ins.2016.01.020.

[76] R. Chen and Y. Tong, "A Two-stage Method for Solving Multi-resident Activity Recognition in Smart Environments," *Entropy,* vol. 16, no. 4, pp. 2184-2203, 2014, doi: 10.3390/e16042184.

[77] Y. P. Raykov, E. Ozer, G. Dasika, A. Boukouvalas, and M. A. Little, "Predicting room occupancy with a single passive infrared (PIR) sensor through behavior extraction," presented at the Ubicomp' 16, Heigelberg, Germany, 2016.

[78] S. Narayana, R. V. Prasad, V. S. Rao, T. V. Prabhakar, S. S. Kowshik, and M. S. Iyer, "PIR sensors: characterization and novel localization technique," 2015. [Online]. Available: http://doi.acm.org/10.1145/2737095.2742561.

[79] K. Thapa, Z. M. A. Al, B. Lamichhane, and S.-H. Yang, "A Deep Machine Learning Method for Concurrent and Interleaved Human Activity Recognition," *Sensors,* vol. 20, no. 20, p. 5770, / 2020, doi: 10.3390/s20205770.

[80] H. Alemdar, "ARAS Human Activity Datasets In Multiple Homes with Multiple Residents," *2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops,* pp. 232-235, 2013.

[81] X. L. T. Liu, "Human Indoor Localization Based on Ceiling Mounted PIR Sensor Nodes," *13th IEEE Annual Consumer Communications & Networking Conference (CCNC),* pp. 289-296, 2016.

[82] J. Cao, W. Li, C. Ma, and Z. Tao, "Optimizing multi-sensor deployment via ensemble pruning for wearable activity recognition," *Information Fusion,* vol. 41, pp. 68-79, / 2018, doi: 10.1016/j.inffus.2017.08.002.

[83] W. Jiang and Z. Yin, "Human Activity Recognition Using Wearable Sensors by Deep Convolutional Neural Networks," 2015. [Online]. Available: https://doi.org/10.1145/2733373.2806333.

[84] B. Reily, H. Zhang, and W. A. Hoff, "Real-time gymnast detection and performance analysis with a portable 3D camera," *Comput. Vis. Image Underst.,* vol. 159, pp. 154-163, / 2017, doi: 10.1016/j.cviu.2016.11.006.

[85] C. Zhao, J. Wang, and H. Lu, "Learning discriminative context models for concurrent collective activity recognition," *Multimedia Tools and Applications,* vol. 76, no. 5 %@ 1573-7721, pp. 7401-7420, 2017, doi: 10.1007/s11042-016-3393-3 %U https://doi.org/10.1007/s11042-016-3393-3.

[86] K. K. Roudposhti, J. Dias, P. Peixoto, V. Metsis, and U. Nunes, "A Multilevel Body

Motion-Based Human Activity Analysis Methodology," *IEEE Trans. Cognitive and Developmental Systems,* vol. 9, no. 1, pp. 16-29, / 2017, doi: 10.1109/TCDS.2016.2607154.

[87] J. M. Sim, Y. Lee, and O. Kwon, "Acoustic Sensor Based Recognition of Human Activity in Everyday Life for Smart Home Services," *Int. J. Distributed Sens. Networks,* vol. 11, pp. 679123:1-679123:11, / 2015, doi: 10.1155/2015/679123.

[88] M. Nishida, N. Kitaoka, and K. Takeda, "Daily activity recognition based on acoustic signals and acceleration signals estimated with Gaussian process," 2015. [Online]. Available: https://doi.org/10.1109/APSIPA.2015.7415520.

[89] M. Gochoo, T.-H. Tan, S.-H. Liu, F.-R. Jean, F. S. Alnajjar, and S.-C. Huang, "Unobtrusive Activity Recognition of Elderly People Living Alone Using Anonymous Binary Sensors and DCNN," *IEEE J. Biomedical and Health Informatics,* vol. 23, no. 2, pp. 693-702, / 2019, doi: 10.1109/JBHI.2018.2833618.

[90] R. Singh, A. Sonawane, and R. Srivastava, "Recent evolution of modern datasets for human activity recognition: a deep survey," *Multim. Syst.,* vol. 26, no. 2, pp. 83-106, / 2020, doi: 10.1007/s00530-019-00635-7.

[91] I. Rodomagoulakis, A. Katsamanis, G. Potamianos, P. Giannoulis, A. Tsiami, and P. Maragos, "Room-localized spoken command recognition in multi-room, multi-microphone environments," *Computer Speech & Language,* vol. 46, pp. 419-443, / 2017, doi: 10.1016/j.csl.2017.02.004.

[92] Z. Hussain, Q. Z. Sheng, and W. E. Zhang, "A review and categorization of techniques on device-free human activity recognition," *Journal of Network and Computer Applications,* vol. 167, p. 102738, 2020/10/01/ 2020, doi: https://doi.org/10.1016/j.jnca.2020.102738.

[93] R. A. Hamad, A. S. Hidalgo, M.-R. Bouguelia, M. E. Estévez, and J. M. Quero, "Efficient Activity Recognition in Smart Homes Using Delayed Fuzzy Temporal Windows on Binary Sensors," *IEEE J. Biomed. Health Informatics,* vol. 24, no. 2, pp. 387-395, / 2020, doi: 10.1109/JBHI.2019.2918412.

[94] H. Li, G. D. Abowd, and T. Plötz, "On specialized window lengths and detector based human activity recognition," 2018. [Online]. Available: https://doi.org/10.1145/3267242.3267246.

[95] F. J. Quesada, F. Moya, J. Medina, L. Martínez-López, C. D. Nugent, and M. Espinilla, "Generation of a Partitioned Dataset with Single, Interleave and Multioccupancy Daily Living Activities," 2015. [Online]. Available: https://doi.org/10.1007/978-3-319-26401-1_6.

[96] J. Ye, G. Stevenson, and S. A. Dobson, "KCAR: A knowledge-driven approach for concurrent activity recognition," *Pervasive Mob. Comput.,* vol. 19, pp. 47-70, / 2015, doi: 10.1016/j.pmcj.2014.02.003.

[97] C. D. Singla G, "Interleaved activity recognition for smart home residents," *Intelligent Environments,* pp. 145–152, 2009.

[98] D. Triboan, L. Chen, F. Chen, and Z. Wang, "A semantics-based approach to sensor data segmentation in real-time Activity Recognition," *Future Gener. Comput. Syst.,* vol. 93, pp. 224-236, / 2019, doi: 10.1016/j.future.2018.09.055.

[99] M.-S. Pan, K.-C. Huang, T.-H. Lu, and Z.-Y. Lin, "Using accelerometer for counting and

identifying swimming strokes," *Pervasive and Mobile Computing,* vol. 31, pp. 37-49, 2016/09/01/ 2016, doi: https://doi.org/10.1016/j.pmcj.2016.01.011.

[100] F. Malawski and B. Kwolek, "Real-Time action detection and analysis in fencing footwork," 2017. [Online]. Available: https://doi.org/10.1109/TSP.2017.8076041.

[101] X. Li, B. Li, F. Liu, H. Yin, and F. Zhou, "Segmentation of Pulmonary Nodules Using a GMM Fuzzy C-Means Algorithm," *Ieee Access,* vol. 8, pp. 37541-37556, / 2020, doi: 10.1109/ACCESS.2020.2968936.

[102] S. Aminikhanghahi, T. Wang, and D. J. Cook, "Real-Time Change Point Detection with Application to Smart Home Time Series Data," *IEEE Trans. Knowl. Data Eng.,* vol. 31, no. 5, pp. 1010-1023, / 2019, doi: 10.1109/TKDE.2018.2850347.

[103] J. Wan, M. J. O'Grady, and G. M. P. O'Hare, "Dynamic sensor event segmentation for real-time activity recognition in a smart home context," *Personal and Ubiquitous Computing,* vol. 19, no. 2, pp. 287-301, / 2015, doi: 10.1007/s00779-014-0824-x.

[104] P. Rodríguez, M. Á. Bautista, J. Gonzàlez, and S. Escalera, "Beyond one-hot encoding: Lower dimensional target embedding," *Image Vis. Comput.,* vol. 75, pp. 21-31, / 2018, doi: 10.1016/j.imavis.2018.04.004.

[105] S. Münzner, P. Schmidt, A. Reiss, M. Hanselmann, R. Stiefelhagen, and R. Dürichen, "CNN-based sensor fusion techniques for multimodal human activity recognition," 2017. [Online]. Available: http://doi.acm.org/10.1145/3123021.3123046.

[106] Z. Liu, C. K. Loo, and M. Seera, "Meta-cognitive Recurrent Recursive Kernel OS-ELM for concept drift handling," *Appl. Soft Comput.,* vol. 75, pp. 494-507, / 2019, doi: 10.1016/j.asoc.2018.11.006.

[107] A. Jarraya, A. Bouzeghoub, A. Borgi, and K. Arour, "DCR: A new distributed model for human activity recognition in smart homes," *Expert Syst. Appl.,* vol. 140, / 2020, doi: 10.1016/j.eswa.2019.112849.

[108] M. A. R. Ahad, A. D. Antar, and M. Ahmed, *IoT Sensor-Based Activity Recognition - Human Activity Recognition* (Intelligent Systems Reference Library). Springer, 2021, pp. 1-189.

[109] Q. Li, R. Gravina, Y. Li, S. H. Alsamhi, F. Sun, and G. Fortino, "Multi-user activity recognition: Challenges and opportunities," *Inf. Fusion,* vol. 63, pp. 121-135, / 2020, doi: 10.1016/j.inffus.2020.06.004.

[110] J. Wang, "Deep learning for sensor-based activity recognition: A Survey," *Pattern Recognition Letters,* pp. 21-29, 2018, doi: doi.org/10.1016/j.patrec.2018.02.010.

[111] Y. Wang, S. Cang, and H. Yu, "A survey on wearable sensor modality centred human activity recognition in health care," *Expert Syst. Appl.,* vol. 137, pp. 167-190, / 2019, doi: 10.1016/j.eswa.2019.04.057.

[112] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Gener. Comput. Syst.,* vol. 81, pp. 307-313, / 2018, doi: 10.1016/j.future.2017.11.029.

[113] K. Chen, L. Yao, D. Zhang, X. Wang, X. Chang, and F. Nie, "A Semisupervised Recurrent Convolutional Attention Model for Human Activity Recognition," *IEEE Trans. Neural Networks Learn. Syst.,* vol. 31, no. 5, pp. 1747-1756, / 2020, doi: 10.1109/TNNLS.2019.2927224.

[114] Q. Ni, A. B. Garcia Hernando, and I. P. de la Cruz, "The Elderly's Independent Living in

Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development," *Sensors,* vol. 15, no. 5, pp. 11312-62, May 14 2015, doi: 10.3390/s150511312.

[115] D. Liciotti, M. Bernardini, L. Romeo, and E. Frontoni, "A Sequential Deep Learning Application for Recognising Human Activities in Smart Homes," *Neurocomputing,* 04/01 2019, doi: 10.1016/j.neucom.2018.10.104.

[116] S. N. Tran, N. T. Son, Q. Zhang, and M. Karunanithi, "Mixed-dependency models for multi-resident activity recognition in smart homes," *Multim. Tools Appl.,* vol. 79, no. 31-32, pp. 23445-23460, / 2020, doi: 10.1007/s11042-020-09093-0.

[117] A. Ignatov, "Real-time human activity recognition from accelerometer data usingConvolutional Neural Networks," *Applied Soft Computing,* pp. 915-922, 2018.

[118] A. Vaswani *et al.*, "Attention Is All You Need," *CoRR,* vol. abs/1706.03762, / 2017. [Online]. Available: http://arxiv.org/abs/1706.03762.

[119] K. Gavrilyuk, R. Sanford, M. Javan, and C. G. M. Snoek, "Actor-Transformers for Group Activity Recognition," *CoRR,* vol. abs/2003.12737, / 2020. [Online]. Available: https://arxiv.org/abs/2003.12737.

[120] C. Xu, Y. Zhou, Q. Wang, Z. Ma, and Y. Zhu, "Detecting Hypernymy Relations Between Medical Compound Entities Using a Hybrid-Attention Based Bi-GRU-CapsNet Model," *Ieee Access,* vol. 7, pp. 175693-175702, / 2019, doi: 10.1109/ACCESS.2019.2957827.

[121] Z. Huang, P. Xu, D. Liang, A. Mishra, and B. Xiang, "TRANS-BLSTM: Transformer with Bidirectional LSTM for Language Understanding," *CoRR,* vol. abs/2003.07000, / 2020. [Online]. Available: https://arxiv.org/abs/2003.07000.

[122] B. Krawczyk, "Active and adaptive ensemble learning for online activity recognition from data streams," *Knowl. Based Syst.,* vol. 138, pp. 69-78, / 2017, doi: 10.1016/j.knosys.2017.09.032.

[123] H. K. Fatlawi and A. Kiss, "On Robustness of Adaptive Random Forest Classifier on Biomedical Data Stream," 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-41964-6_29.

[124] Z. S. Abdallah, M. M. Gaber, B. Srinivasan, and S. Krishnaswamy, "Adaptive mobile activity recognition system with evolving data streams," *Neurocomputing,* vol. 150, pp. 304-317, / 2015, doi: 10.1016/j.neucom.2014.09.074.

[125] S. Mohamad, M. Sayed-Mouchaweh, and A. Bouchachia, "Online active learning for human activity recognition from sensory data streams," *Neurocomputing,* vol. 390, pp. 341-358, 2020/05/21/ 2020, doi: https://doi.org/10.1016/j.neucom.2019.08.092.

[126] R. Yan, Tang, J., Shu, X., Li, Z., & Tian, Q, "Participation-contributed temporal dynamic model for group activity recognition," *Proceedings of the 26th ACM international conference on Multimedia,* pp. 1292-1300, 2018.

[127] X. S. J. Tang, R. Yan and L. Zhang, "Coherence Constrained Graph LSTM for Group Activity Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2019.

[128] R. Yan, Xie, L., Tang, J., Shu, X., & Tian, Q, "Social adaptive module for weakly-supervised group activity recognition," *European Conference on Computer Vision,* pp. 208-224, 2020. Springer, Cham.

[129] R. Yan, Xie, L., Tang, J., Shu, X., & Tian, Q, "HiGCIN: hierarchical graph-based cross

inference network for group activity recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2020.

[130] H. M. Gomes *et al.*, "Adaptive random forests for evolving data stream classification," *Mach. Learn.,* vol. 106, no. 9-10, pp. 1469-1495, / 2017, doi: 10.1007/s10994-017-5642-8.

[131] X. Gao, Z. Chen, S. Tang, Y. Zhang, and J. Li, "Adaptive weighted imbalance learning with application to abnormal activity recognition," *Neurocomputing,* vol. 173, pp. 1927-1935, / 2016, doi: 10.1016/j.neucom.2015.09.064.

[132] S. Zhang, D. T. J. Huang, G. Dobbie, and Y. S. Koh, "SLED: Semi-supervised Locally-weighted Ensemble Detector," 2020. [Online]. Available: https://doi.org/10.1109/ICDE48307.2020.00183.

[133] H. Xue *et al.*, "DeepMV: Multi-View Deep Learning for Device-Free Human Activity Recognition," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.,* vol. 4, no. 1, pp. 34:1-34:26, / 2020, doi: 10.1145/3380980.

[134] K. Xia, J. Huang, and H. Wang, "LSTM-CNN Architecture for Human Activity Recognition," *Ieee Access,* vol. 8, pp. 56855-56866, / 2020, doi: 10.1109/ACCESS.2020.2982225.

[135] J. Suto, S. Oniga, C. Lung, and I. Orha, "Comparison of offline and real-time human activity recognition results using machine learning techniques," *Neural Comput. Appl.,* vol. 32, no. 20, pp. 15673-15686, / 2020, doi: 10.1007/s00521-018-3437-x.

[136] C. Singh and A. Sharma, "Online Learning Using Multiple Times Weight Updating," *Appl. Artif. Intell.,* vol. 34, no. 6, pp. 515-536, / 2020, doi: 10.1080/08839514.2020.1730623.

[137] Z. Qin, Y. Zhang, S. Meng, Z. Qin, and K.-K. R. Choo, "Imaging and fusing time series for wearable sensor-based human activity recognition," *Inf. Fusion,* vol. 53, pp. 80-87, / 2020, doi: 10.1016/j.inffus.2019.06.014.

[138] A. Lentzas and D. Vrakas, "Non-intrusive human activity recognition and abnormal behavior detection on elderly people: a review," *Artif. Intell. Rev.,* vol. 53, no. 3, pp. 1975-2021, / 2020, doi: 10.1007/s10462-019-09724-5.

[139] T. D. Khang, N. D. Vuong, M.-K. Tran, and M. W. Fowler, "Fuzzy C-Means Clustering Algorithm with Multiple Fuzzification Coefficients," *Algorithms,* vol. 13, no. 7, p. 158, / 2020, doi: 10.3390/a13070158.

[140] J. Huang, S. Lin, N. Wang, G. Dai, Y. Xie, and J. Zhou, "TSE-CNN: A Two-Stage End-to-End CNN for Human Activity Recognition," *IEEE J. Biomedical and Health Informatics,* vol. 24, no. 1, pp. 292-299, / 2020, doi: 10.1109/JBHI.2019.2909688.

[141] B. Fu, N. Damer, F. Kirchbuchner, and A. Kuijper, "Sensing Technology for Human Activity Recognition: A Comprehensive Survey," *Ieee Access,* vol. 8, pp. 83791-83820, / 2020, doi: 10.1109/ACCESS.2020.2991891.

[142] F. Cruciani *et al.*, "Feature learning for Human Activity Recognition using Convolutional Neural Networks," *CCF Trans. Pervasive Comput. Interact.,* vol. 2, no. 1, pp. 18-32, / 2020, doi: 10.1007/s42486-020-00026-2.

[143] P. A. Colpas, E. Vicario, E. D. l. Hoz-Franco, M. P. Melo, A. I. O. Carrascal, and F. Patara, "Unsupervised Human Activity Recognition Using the Clustering Approach: A Review," *Sensors,* vol. 20, no. 9, p. 2702, / 2020, doi: 10.3390/s20092702.

[144] S. Abbaspour, F. Fotouhi, A. Sedaghatbaf, H. Fotouhi, M. Vahabi, and M. Lindén, "A

Comparative Analysis of Hybrid Deep Learning Models for Human Activity Recognition," *Sensors,* vol. 20, no. 19, p. 5707, / 2020, doi: 10.3390/s20195707.

[145] H. Yang, Z. Pan, and Q. Tao, "Online Learning for Time Series Prediction of AR Model with Missing Data," *Neural Process. Lett.,* vol. 50, no. 3, pp. 2247-2263, / 2019, doi: 10.1007/s11063-019-10007-x.

[146] Q. Wang, X. Yin, J. Tan, T. Xing, J. Niu, and D. Fang, "DTransfer: extremely low cost localization irrelevant to targets and regions for activity recognition," *Personal and Ubiquitous Computing,* vol. 23, no. 1, pp. 3-16, / 2019, doi: 10.1007/s00779-018-1177-7.

[147] A. Mannini and S. S. Intille, "Classifier Personalization for Activity Recognition Using Wrist Accelerometers," *IEEE J. Biomedical and Health Informatics,* vol. 23, no. 4, pp. 1585-1594, / 2019, doi: 10.1109/JBHI.2018.2869779.

[148] S. Feng and M. F. Duarte, "Few-shot learning-based human activity recognition," *Expert Syst. Appl.,* vol. 138, / 2019, doi: 10.1016/j.eswa.2019.06.070.

[149] C. Dhiman and D. K. Vishwakarma, "A review of state-of-the-art techniques for abnormal human activity recognition," *Eng. Appl. Artif. Intell.,* vol. 77, pp. 21-45, / 2019, doi: 10.1016/j.engappai.2018.08.014.

[150] Z. Chen, L. Zhang, C. Jiang, Z. Cao, and W. Cui, "WiFi CSI Based Passive Human Activity Recognition Using Attention Based BLSTM," *IEEE Trans. Mob. Comput.,* vol. 18, no. 11, pp. 2714-2724, / 2019, doi: 10.1109/TMC.2018.2878233.

[151] A. Ignatov, "Real-time human activity recognition from accelerometer data using Convolutional Neural Networks," *Appl. Soft Comput.,* vol. 62, pp. 915-922, / 2018, doi: 10.1016/j.asoc.2017.09.027.

[152] T. Hayashi, M. Nishida, N. Kitaoka, T. Toda, and K. Takeda, "Daily Activity Recognition with Large-Scaled Real-Life Recording Datasets Based on Deep Neural Network Using Multi-Modal Signals," *IEICE Transactions,* vol. 101-A, no. 1, pp. 199-210, / 2018. [Online]. Available: http://search.ieice.org/bin/summary.php?id=e101-a_1_199.

[153] E. Garcia-Ceja, C. E. Galván-Tejada, and R. F. Brena, "Multi-view stacking for activity recognition with sound and accelerometer data," *Information Fusion,* vol. 40, pp. 45-56, / 2018, doi: 10.1016/j.inffus.2017.06.004.

[154] E. Garcia-Ceja and R. F. Brena, "An Improved Three-Stage Classifier for Activity Recognition," *IJPRAI,* vol. 32, no. 1, pp. 1-18, / 2018, doi: 10.1142/S0218001418600030.

[155] S. Ding *et al.*, "Kernel based online learning for imbalance multiclass classification," *Neurocomputing,* vol. 277, pp. 139-148, 2018/02/14/ 2018, doi: https://doi.org/10.1016/j.neucom.2017.02.102.

[156] E. ZDRAVEVSKI, P. LAMESKI, and V. TRAJKOVIK, "Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering," *Ieee Access,* vol. 5, pp. 5262-5280, 2017, doi: 10.1109/ACCESS.2017.2684913.

[157] J. Yu, Z. Chen, Y. Zhu, Y. J. Chen, L. Kong, and M. Li, "Fine-Grained Abnormal Driving Behaviors Detection and Identification with Smartphones," *IEEE Trans. Mob. Comput.,* vol. 16, no. 8, pp. 2198-2212, / 2017, doi: 10.1109/TMC.2016.2618873.

[158] J. Ye, H. Hu, G.-J. Qi, and K. A. Hua, "A Temporal Order Modeling Approach to Human Action Recognition from Multimodal Sensor Data," *TOMCCAP,* vol. 13, no. 2, pp. 14:1-14:22, / 2017, doi: 10.1145/3038917.

[159] N. Yala, B. Fergani, and A. Fleury, "Towards improving feature extraction and classification for activity recognition on streaming data," *J. Ambient Intelligence and Humanized Computing,* vol. 8, no. 2, pp. 177-189, / 2017, doi: 10.1007/s12652-016-0412-1.

[160] Q. Wang, H. Lv, J. Yue, and E. Mitchell, "Supervised multiview learning based on simultaneous learning of multiview intact and single view classifier," *Neural Computing and Applications,* vol. 28, no. 8, pp. 2293-2301, / 2017, doi: 10.1007/s00521-016-2189-8.

[161] J. Wang, X. Zhang, Q. Gao, H. Yue, and H. Wang, "Device-Free Wireless Localization and Activity Recognition: A Deep Learning Approach," *IEEE Trans. Vehicular Technology,* vol. 66, no. 7, pp. 6258-6267, / 2017, doi: 10.1109/TVT.2016.2635161.

[162] N. Twomey, T. Diethe, I. Craddock, and P. Flach, "Unsupervised learning of sensor topologies for improving activity recognition in smart environments," *Neurocomputing,* vol. 234, no. Supplement C, pp. 93-106, 2017/04/19/ 2017, doi: https://doi.org/10.1016/j.neucom.2016.12.049.

[163] V. Soto-Mendoza, J. A. García-Macías, E. Chávez, J. R. Gomez-Montalvo, and E. Quintana, "Detecting abnormal behaviours of institutionalized older adults through a hybrid-inference approach," *Pervasive and Mobile Computing,* 2017, doi: http://dx.doi.org/10.1016/j.pmcj.2017.06.019.

[164] F. Sikder and D. Sarkar, "Log-Sum Distance Measures and Its Application to Human-Activity Monitoring and Recognition Using Data From Motion Sensors," *IEEE SENSORS JOURNAL,* pp. 4520-4533, 2017.

[165] D. Shi, R. Wang, Y. Wu, X. Mo, and J. Wei, "A novel orientation- and location-independent activity recognition method," *Pers Ubiquit Comput,* pp. 427-441, 2017, doi: 10.1007/s00779-017-1007-3.

[166] S. Samarah, M. G. Al Zamil, A. F. Aleroud, M. Rawashdeh, M. F. Alhamid, and A. Alamri, "An Efficient Activity Recognition Framework: Toward Privacy-Sensitive Health Data Sensing," (in English), *Ieee Access,* vol. 5, pp. 3848-3859, 2017, doi: 10.1109/Access.2017.2685531.

[167] K. Murao, J. Imai, T. Terada, and M. Tsukamoto, "Activity Recognition and User Identification based on Tabletop Activities with Load Cells," *JIP,* vol. 25, pp. 59-66, / 2017, doi: 10.2197/ipsjjip.25.59.

[168] L. Liu, S. Wang, G. Su, Z. Huang, and M. Liu, "Towards complex activity recognition using a Bayesian network-based probabilistic generative framework," *Pattern Recognition,* vol. 68, pp. 295-309, / 2017, doi: 10.1016/j.patcog.2017.02.028.

[169] S. A. Khowaja, B. N. Yahya, and S.-L. Lee, "Hierarchical classification method based on selective learning of slacked hierarchy for activity recognition systems," *Expert Syst. Appl.,* vol. 88, pp. 165-177, / 2017, doi: 10.1016/j.eswa.2017.06.040.

[170] X. Huang and M. Dai, "Indoor Device-Free Activity Recognition Based on Radio Signal," *IEEE Trans. Vehicular Technology,* vol. 66, no. 6, pp. 5316-5329, / 2017, doi: 10.1109/TVT.2016.2616883.

[171] J. Hu, L. Xu, X. He, and W. Meng, "Abnormal Driving Detection Based on Normalized Driving Behavior," *IEEE Trans. Vehicular Technology,* vol. 66, no. 8, pp. 6645-6652, / 2017, doi: 10.1109/TVT.2017.2660497.

[172] K. S. Gayathri, K. S. Easwarakumar, and S. Elias, "Probabilistic ontology based activity recognition in smart homes using Markov Logic Network," *Knowl.-Based Syst.,* vol. 121, pp. 173-184, / 2017, doi: 10.1016/j.knosys.2017.01.025.

[173] M. Espinilla, J. Medina, A. Calzada, J. Liu, L. Martínez, and C. D. Nugent, "Optimizing the configuration of an heterogeneous architecture of sensors for activity recognition, using the extended belief rule-based inference methodology," *Microprocessors and Microsystems - Embedded Hardware Design,* vol. 52, pp. 381-390, / 2017, doi: 10.1016/j.micpro.2016.10.007.

[174] D. Arifoglu and A. Bouchachia, "Activity Recognition and Abnormal Behaviour Detection with Recurrent Neural Networks," 2017. [Online]. Available: https://doi.org/10.1016/j.procs.2017.06.121.

[175] M. A. U. Alam and N. Roy, "Unseen Activity Recognitions: A Hierarchical Active Transfer Learning Approach," 2017. [Online]. Available: https://doi.org/10.1109/ICDCS.2017.264.

[176] Q. G. X. Yin, "A Novel Infrared Motion Sensing System for Compressive Classification of Physical Activity," *IEEE SENSORS JOURNAL,* vol. 16, pp. 2251-2259, 2016, doi: 10.1109/JSEN.2016.2514606.

[177] J. Ye, G. Stevenson, and S. Dobson, "Detecting abnormal events on binary sensors in smart home environments," *Pervasive and Mobile Computing,* vol. 33, pp. 32-49, / 2016, doi: 10.1016/j.pmcj.2016.06.012.

[178] X. L. H. Tan, "Abnormal Activity Detection Using Pyroelectric Infrared Sensors," *Sensors,* pp. 822-839, 2016, doi: 10.3390/s16060822.

[179] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Syst. Appl.,* vol. 59, pp. 235-244, / 2016, doi: 10.1016/j.eswa.2016.04.032.

[180] D. Riboni, C. Bettini, G. Civitarese, Z. H. Janjua, and R. Helaoui, "SmartFABER: Recognizing fine-grained abnormal behaviors for early detection of mild cognitive impairment," *Artificial Intelligence in Medicine,* vol. 67, pp. 57-74, / 2016, doi: 10.1016/j.artmed.2015.12.001.

[181] J. Luo, J. Tang, and X. Xiao, "Abnormal Gait Behavior Detection for Elderly Based on Enhanced Wigner-Ville Analysis and Cloud Incremental SVM Learning," *Journal of Sensors,* vol. 2016, 2016, doi: http://dx.doi.org/10.1155/2016/5869238.

[182] L. Liu, S. Wang, Y. Peng, Z. Huang, M. Liu, and B. Hu, "Mining intricate temporal rules for recognizing complex activities of daily living under uncertainty," *Pattern Recognition,* vol. 60, pp. 1015-1028, / 2016, doi: 10.1016/j.patcog.2016.07.024.

[183] R. Khemchandani and S. Sharma, "Robust least squares twin support vector machine for human activity recognition," *Appl. Soft Comput.,* vol. 47, pp. 33-46, / 2016, doi: 10.1016/j.asoc.2016.05.025.

[184] M. H. Kabir, M. R. Hoque, K. Thapa, and S.-H. Yang, "Two-Layer Hidden Markov Model for Human Activity Recognition in Home Environments," *IJDSN,* vol. 12, no. 1, pp. 4560365:1-4560365:12, / 2016, doi: 10.1155/2016/4560365.

[185] A. Jalal, S. Kamal, and D. Kim, "Human Depth Sensors-Based Activity Recognition Using Spatiotemporal Features and Hidden Markov Model for Smart Environments," *Journal Comp. Netw. and Communic.,* vol. 2016, pp. 8087545:1-8087545:11, / 2016, doi:

10.1155/2016/8087545.

[186] P. N. Dawadi, D. J. Cook, and M. Schmitter-Edgecombe, "Modeling patterns of activities using activity curves," *Pervasive and Mobile Computing,* vol. 28, pp. 51-68, / 2016, doi: 10.1016/j.pmcj.2015.09.007.

[187] U. A. B. U. A. Bakar, H. Ghayvat, S. F. Hasanm, and S. C. Mukhopadhyay, "Activity and Anomaly Detection in Smart Home: A Survey," presented at the Springer International Publishing Switzerland 2016, 2016.

[188] M. Al-Nawashi, O. M. Al-Hazaimeh, and M. Saraee, "A novel framework for intelligent surveillance system based on abnormal human activity detection in academic environments," *Neural Comput & Applic,* 2016, doi: 10.1007/s00521-016-2363-z.

[189] B. M. h. Abidine and L. Fergani, "The joint use of sequence features combination and modified weighted SVM for improving daily activity recognition," *Pattern Anal Applic,* 2016, doi: 10.1007/s10044-016-0570-y.

[190] J. Saives, C. Pianon, and G. Faraut, "Activity Discovery and Detection of Behavioral Deviations of an Inhabitant From Binary Sensors," *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING,* pp. 1211-1224, 2015, doi: 10.1109/TASE.2015.2471842.

[191] D. Y. K. Rao, "PIR Sensors Deployment with the Accessible Priority in Smart Home Using Genetic Algorithm," *International Journal of Distributed Sensor Networks,* 2015, doi: 10.1155/2015/146270.

[192] F. J. Ordóñez, P. d. Toledo, and A. Sanchis, "Sensor-based Bayesian detection of anomalous living patterns in a home setting," *Personal and Ubiquitous Computing,* vol. 19, no. 2, pp. 259-270, / 2015, doi: 10.1007/s00779-014-0820-1.

[193] M. Moshtaghi, I. Zukerman, and R. A. Russell, "Statistical models for unobtrusively detecting abnormal periods of inactivity in older adults," *User Model. User-Adapt. Interact.,* vol. 25, no. 3, pp. 231-265, / 2015, doi: 10.1007/s11257-015-9162-6.

[194] G. Meditskos, E. Kontopoulos, and I. Kompatsiaris, "ReDef: Context-aware Recognition of Interleaved Activities using OWL 2 and Defeasible Reasoning," 2015. [Online]. Available: http://ceur-ws.org/Vol-1488/paper-04.pdf.

[195] M. A. A. H. Khan, H. M. S. Hossain, and N. Roy, "Infrastructure-less occupancy detection and semantic localization in smart environments," Coimbra, Portugal, 2015.

[196] P. Barsocchi, M. G. C. A. Cimino, E. Ferro, A. Lazzeri, F. Palumbo, and G. Vaglini, "Monitoring elderly behavior via indoor position-based stigmergy," *Pervasive and Mobile Computing,* vol. 23, pp. 26-42, / 2015, doi: 10.1016/j.pmcj.2015.04.003.

[197] H. Alemdar, C. Tunca, and C. Ersoy, "Daily life behaviour monitoring for health assessment using machine learning: bridging the gap between domains," *Pers Ubiquit Comput,* pp. 303-315, 2015, doi: 10.1007/s00779-014-0823-y.

[198] M. A. U. Alam, N. Pathak, and N. Roy, "An iBeacon-Assisted Smartphone-Based Real Time Activity Recognition Framework," presented at the Mobiquitous, 2015.

[199] T. L. J. Liu, "Design and implementation of a compressive infrared sampling for motion acquisition," *EURASIP Journal on Advances in Signal Processing,* pp. 85-99, 2014.

[200] B. Huang, G. Tian, H. Wu, and F. Zhou, "A method of abnormal habits recognition in intelligent space," *Eng. Appl. of AI,* vol. 29, pp. 125-133, / 2014, doi: 10.1016/j.engappai.2013.12.010.

[201] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier Detection for Temporal Data: A Survey," *IEEE Trans. Knowl. Data Eng.,* vol. 26, no. 9, pp. 2250-2267, / 2014, doi: 10.1109/TKDE.2013.184.

[202] J. L. C. Candás, V. Peláez, G. López, M. Á. Fernández, E. Alvarez, and G. Díaz, "An automatic data mining method to detect abnormal human behaviour using physical activity measurements," *Pervasive and Mobile Computing,* vol. 15, pp. 228-241, / 2014, doi: 10.1016/j.pmcj.2014.09.007.

[203] N. K. Suryadevara, S. C. Mukhopadhyay, R. Wang, and R. K. Rayudu, "Forecasting the behavior of an elderly using wireless sensors data in a smart home," *Eng. Appl. of AI,* vol. 26, no. 10, pp. 2641-2652, / 2013, doi: 10.1016/j.engappai.2013.08.004.

[204] N. D. Rodríguez, M. P. Cuéllar, J. Lilius, and M. D. Calvo-Flores, "A survey on ontologies for human behavior recognition," *ACM Comput. Surv.,* vol. 46, no. 4, pp. 43:1-43:33, / 2013, doi: 10.1145/2523819.

[205] S. M. Mahmoud, A. Lotfi, and C. S. Langensiepen, "Behavioural pattern identification and prediction in intelligent environments," *Appl. Soft Comput.,* vol. 13, no. 4, pp. 1813-1822, / 2013, doi: 10.1016/j.asoc.2012.12.012.

[206] Z. A. Khan and W. Sohn, "A hierarchical abnormal human activity recognition system based on R-transform and kernel discriminant analysis for elderly health care," *Computing,* vol. 95, no. 2, pp. 109-127, / 2013, doi: 10.1007/s00607-012-0216-x.

[207] H. Alerndar, "ARAS Human Activity Datasets In Multiple Homes with Multiple Residents," *2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops,* pp. 232-235, 2013.

[208] M. Usman, V. Muthukkumarasamy, X.-W. Wu, and S. Khanum, "Wireless Smart Home Sensor Networks: Mobile Agent Based Anomaly Detection," pp. 322-329, 2012, doi: 10.1109/uic-atc.2012.145.

[209] B. Marhic, L. Delahoche, C. Solau, A.-M. Jolly-Desodt, and V. Ricquebourg, "An evidential approach for detection of abnormal behaviour in the presence of unreliable sensors," *Information Fusion,* vol. 13, no. 2, pp. 146-160, / 2012.

[210] E. Hoque and J. A. Stankovic, "Semantic anomaly detection in daily activities," 2012. [Online]. Available: http://doi.acm.org/10.1145/2370216.2370340.

[211] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection for Discrete Sequences: A Survey," *IEEE Trans. Knowl. Data Eng.,* vol. 24, no. 5, pp. 823-839, / 2012, doi: 10.1109/TKDE.2010.235.

[212] J. H. Shin, B. Lee, and K. S. Park, "Detection of Abnormal Living Patterns for Elderly Living Alone Using Support Vector Data Description," *IEEE Trans. Information Technology in Biomedicine,* vol. 15, no. 3, pp. 438-448, / 2011, doi: 10.1109/TITB.2011.2113352.

[213] A. Lotfi, C. Langensiepen, S. M. Mahmoud, and M. J. Akhlaghinia, "Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour," *Journal of Ambient Intelligence and Humanized Computing,* vol. 3, no. 3, pp. 205-218, 2011, doi: 10.1007/s12652-010-0043-x.

[214] K. Park, Y. Lin, V. Metsis, Z. Le, and F. Makedon, "Abnormal human behavioral pattern detection in assisted living environments," 2010.