الجمهورية الجزائرية الديمقراطية الشعبية
People's Democratic Republic of Algeria

وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research

جامعة غرداية
University of Ghardaia

كلية العلوم و التكنولوجيا
Faculty of Science and Technology

قسم الرياضيات و الإعلام الآلي
Department of Mathematics and Computer Science

Presented to obtain the Master Degree in Computer Science

Master Thesis

Field: Mathematics and Computer Science

Option: Computer Science

Specialty : Intelligent Systems for Knowledge Extraction

Theme

Deep Learning for Human Activity Recognition in Smart Environments

Presented by :

Imane AMEUR

Jury members

Mr. Slimane BELLAOUAR	MCA	Univ.Ghardaia	- President
Mr. Ahmed SAIDI	MAA	Univ.Ghardaia	- Examiner
Mr. Houssem Eddine DEGHA	MCB	Univ.Ghardaia	- Supervisor

University Year : 2021/2022

Acknowledgements

First of all, I must thank almighty Allah , who has guided me and given me the ability and the perseverance necessary for the accomplishment of this work. All praise and thanks is due to almighty Allah, the Lord of the World.

We would like to appreciate and express our utmost gratitude to our respected supervisor Doctor M.Houssem Eddine DEGHA for his counsel, support and valuable guidance in our research.

To the examiners who kindly accepted to allocate time and find energy to read and evaluate our dissertation.

Special thanks go to all teachers of the Computer Science department at Ghardaia University.

Finally, special regards go to our colleagues and friends, for their moral support and permanent presence.

Dedication

This thesis is particularly dedicated to the memory of my dear father, Professor Messaoud AMEUR , at Laghouat University, who passed away this year before completing this work. He has always pushed and motivated me in my studies. Without him, this work probably would never have been completed. This thesis therefore represents the culmination of the support and encouragements he has given me throughout my life.

To my dear mother, that no dedication can express my sincere feelings, for her love, support, patience and her sacrifice throughout my studies, that almighty Allah always protects her.

To my dear brothers for their endless love and unwavering support.

And to all my family.

I dedicate this humble work.

Imane

ABSTRACT

Over the last twenty years, many advancements within the field of technology have occurred, and new ways of smart digital sensors have emerged. The growth of IoT-based services in smart homes, buildings, cities, factories, and smart environments, in general, creates value for individuals, industries, and public organizations. In general, a smart home environment is a typical habitation that was improved and equipped with all types of sensors and actuators in order to offer services to its residents. In fact, among the most important topics and inputs for many smart home applications is identifying residents' routine living activities. The ability to automate the Human Activity Recognition system from human behavior patterns is challenging due to the human life complexity inside the home environment by one or many inhabitants and residents. Hence, to overcome this complexity, several algorithms of deep learning, which have recently proven their efficiency in many areas, were studied to increase the human activity recognition accuracy. This research seeks to investigate and design the Human Activity Recognition system in smart environments by implementing the CNN-LSTM hybrid deep learning approach.

Keywords: Smart environment, Smart home, HAR, IoT, Sensor, Deep learning.

RÉSUMÉ

Au cours des vingt dernières années, de nombreux progrès dans le domaine de la technologie ont eu lieu et de nouvelles façons de capteurs numériques intelligents ont émergé. La croissance des services basés sur l'Internet des objets dans les maisons intelligentes, les bâtiments, les villes, les usines et les environnements intelligents en général crée de la valeur pour les particuliers, les industries et les organisations publiques. En général, un environnement de maison intelligente est une habitation typique qui a été améliorée et équipée de tous types de capteurs et d'actionneurs afin d'offrir des services à ses résidents. En effet, parmi les sujets et les entrées les plus importants pour de nombreuses applications de maison intelligente est l'identification des activités de vie courantes des résidents. La capacité d'automatiser le système de reconnaissance de l'activité humaine à partir des modèles de comportement humain est difficile en raison de la complexité de la vie humaine à l'intérieur de l'environnement domestique par un ou plusieurs habitants et résidents. Ainsi, pour pallier cette complexité, plusieurs algorithmes d'apprentissage profond, qui ont récemment prouvé leur efficacité dans de nombreux domaines, ont été étudiés pour augmenter la précision de la reconnaissance de l'activité humaine. Cette recherche vise à étudier et à concevoir le système de reconnaissance de l'activité humaine dans des environnements intelligents en mettant en oeuvre l'approche d'apprentissage en profondeur hybride CNN-LSTM.

Mots clés : Environnement intelligent, Maison intelligente, reconnaissance de l'activité humaine, Internet des objets, Capteur, Apprentissage profond.

ملخص

على مدار العشرين عامًا الماضية ، حدثت العديد من التطورات في مجال التكنولوجيا وظهرت طرق جديدة لأجهزة الاستشعار الرقمية الذكية. إن نمو الخدمات القائمة على إنترنت الأشياء في المنازل الذكية والمباني والمدن والمصانع والبيئات الذكية بشكل عام يخلق قيمة للأفراد والصناعات والمؤسسات العامة. بشكل عام ، تعد بيئة المنزل الذكي مسكنًا نموذجيًا تم تحسينه وتجهيزه بجميع أنواع أجهزة الاستشعار والمحركات من أجل تقديم الخدمات لسكانها. في الواقع، من بين أهم الموضوعات والمدخلات للعديد من تطبيقات المنزل الذكي تحديد الأنشطة المعيشية الروتينية للمقيمين. تعد القدرة على أتمتة نظام التعرف على النشاط البشري من أنماط السلوك البشري أمرًا صعبًا بسبب تعقيد الحياة البشرية داخل بيئة المنزل من قبل واحد أو العديد من السكان والمقيمين. ومن ثم، للتغلب على هذا التعقيد ، تمت دراسة العديد من خوارزميات التعلم العميق ، والتي أثبتت مؤخرًا كفاءتها في العديد من المجالات ، لزيادة دقة التعرف على النشاط البشري. يسعى هذا البحث إلى التحقيق في نظام التعرف على النشاط البشري وتصميمه في البيئات الذكية من خلال تطبيق نهج التعلم العميق الهجينCNN-LSTM.

الكلمات المفتاحية: بيئة ذكية، منزل ذكي، التعرف على النشاط البشري، انترنت الاشياء، لاقط،التعلم العميق.

CONTENTS

LIST OF FIGURES

ACRONYMS

PDA Personal Digital Assistant

ADL Activity of Daily Living

AE Automatic Encoders

API Application Programming Interface

BA Office automation

CA Automation of communication

CNN Convolutional Neural Networks

DL Deep Learning

DNN Deep Neural Networks

DRL Deep Reinforcement Learning

FCN Fully Convolutional Network

GAN Generative Adversary Networks

GRU Gated Recurrent Units

HAR Human Activity Recognition

ICT Information and Communication Technology

ID Integration Device

IoT Internet of Things

LSTM Long Short Term Memory

ML Machine Learning

NLP Natural Language Processing

RBM Restricted Boltzmann Machines

RFID Radio Frequency Identification

RNN Recurrent Neural Networks

$Seq2Seq$ Sequence-to-Sequence model

TCP/IP Transmission Control Protocol/Internet Protocol

TSC Time Series Classification

UDP/IP User Datagram Protocol/Internet Protocol

$WISDM$ Wireless Sensor for Data Mining

Chapter1

GENERAL INTRODUCTION

1.1 Context, problem and research objectives

Recent improvements in wireless communications and sensor-embedded devices have blazed the trail to the activity identification systems. The growth of IoT-based services in smart homes, buildings, cities, factories, and smart environments, in general, creates value for individuals, industries, and public organizations. For example, in homes, IoT devices are embedded in the environment or attached to human bodies to support health-monitoring systems, optimize energy usage, and secure systems. Hence, in smart homes, older adults or children could be supported by developing Human Activity Recognition or briefly (HAR) techniques based on IoT systems, including Activity of Daily Living applications (ADL) (Ali, 2022).

Advanced IoT devices and off-the-shelf sensors make it less expensive to collect sensor data. However, full deployment of such systems is challenging due to scalability, big data management, and maintenance of large-scale deployments (Szt, 2019). Due to the emergence of big data-based IoT and various analysis tools in the last decade, AI/ML techniques utilization increased.

Nowadays, the research of human activity detection is becoming extremely important. Recognizing the activity of humans is a crucial task in planning, controlling, security, and operating energy consumption systems. To assist people in keeping track of their daily activity movements, the recognition of activity is introduced in different technologies.The main problem to resolve is to describe an algorithm labeling the activity performed by residents from collected sensor data in a smart environment. Recently, human activity recognition and discovery have acquired a lot of interest due to their huge potential in context-aware computing systems, including smart home environments. Researchers have commonly tested machine learning techniques to resolve this problem (Zah et al., 2021).

Nevertheless, notable ML algorithms frequently have some flaws. Traditional machine learning techniques (Gradient Regression, Artificial Neural Networks, and Support Vector Machine) fail to learn sequential data patterns for accurate activity recognition. They are imperfect for complex real-world scenarios. Unlike machine learning-based techniques, deep learning models yield better accuracy (Sar, 2021), which is widely studied in different fields of data science, such as video synthesis, image classification, and recognition of human activities.

The main goal of using the smart devices in this research is to collect data from a user and classify the data using Deep Neural Networks algorithms. There are many types of body activities. Mobile types of activities (walking, sitting, and standing) are our focus of classifications, and we recognize human activities according to those categories.

To handle the human activity recognition problem, we designed and implemented a hybrid classification model that uses Convolutional Neural Networks (CNN) in addition to LongShort Term Memory Networks (LSTM). We used a database from the literature to test our recognition activity model.

1.2 Structure of the Thesis

The current study embodies four chapters. The second chapter is theoretical, providing a historical background of the smart environments, IoT devices, and deep learning algorithms. Furthermore, the third chapter discloses a synthesis of the related work and the currently available solutions to the presented Human Activity Recognition problem, with a qualitative comparison of the prominent works. Accordingly, the fourth chapter provides a brief description of our research design and the method used for the system design, while the last chapter shows the evaluation and the results. Finally, this thesis ends with a general conclusion and some perspectives for future work.

Chapter2

THEORETICAL BACKGROUND

2.1 Introduction

In smart environments, including smart cities and homes, it is essential to automate tasks to improve the lifestyle of the inhabitants and meet their needs, such as comfort, security, and entertainment. Managing environments through smart living means being able to perform system functions in a simple way, whenever and wherever we are, because sensors are all around us. Today, the variety of sensor types is increasing rapidly. The smart homes domain has had one of the fastest growth rates in sensor deployment. Miniaturized sensing devices are widely used to create an invisible wireless network that connects everything (Nam and Par, 2011).

On the other side, artificial intelligence represented in deep learning is very valuable for large datasets such as big data generated from smart devices (Sar, 2021). In this chapter, we introduce the two pillars that form the foundation around which our work revolves: smart environments and deep learning algorithms. This chapter is divided into two sections. In the first section, we present the definition, the characteristics, and the global architecture of the smart environments presented in smart cities, buildings, and homes. The second section is dedicated to machine learning and deep learning.

The present chapter will focus on the theoretical background about the smart environments and the new trends of technology, namely, IoT devices and wireless sensors. In addition, it will highlight the notions related to the deep learning. This chapter is split into two sections: section one is devoted to the concept of smart environments, and the other section is devoted to the deep learning, with an emphasis on its two most largely used approaches.

2.2 Smart Environments and IoT

This section discusses the main technological components of a smart environment, including IoT, sensor technologies and data processing.

2.2.1 Smart Environments

Smart environments have the capacity to make users interact easily with their immediate surroundings due to the invention of intelligent technologies attached to software-based services. Technological advancements have ushered in a new age for sensing technology and computational processing to assist the smart environments' vision. Despite the number of challenges in their deployment, various large-scale programs strive to accelerate their adoption (Xu et al., 2021).

In particular, we describe the home, building, and city-based IoT smart devices within the realms of smart environments.

2.2.2 Smart Environments Segments

We live in a society that is becoming more connected and digitized. Smart environments exemplify this trend by connecting computers and other devices to everyday settings and activities. Although the desire to develop smart environments has existed for decades, study on this multidisciplinary topic has gotten more rigorous in recent years. Smart devices, sensor networks, robotics, agent technologies, machine learning, and human-machine interfaces have all made the idea of smart environments a fact.

A smart environment, according to Das and Cook, is an undersized world in which sensor-enabled and networked devices operate continually and collaboratively to improve the lives of its residents (Das and Coo, 2005). Accordingly, the word "Smart" was coined and expanded throughout all segments of the built environment [1], including homes, buildings, and cities.

Smart Home

Definition

A Smart House denotes the logical evolution of a house with many connections. According to (Sep et al. 2019), a "Smart Home" is a home that is combined with a "Controller" to manage the various automation systems. It is a high-performance concept that puts into action all the electronic, computer, and telecommunications techniques and technologies to automate and optimize tasks within a house without any human intervention and to centralize the control of the various systems of the home (heating, electrical outlets, garage gate, etc.) [2].

The Smart House pursuits to offer technical answers to satisfy the needs of comfort (energy management, optimization of lighting and heating), security (alarm), and communication (visual signals, sounds, remote controls, etc.) [1][3].

Undoubtedly, the number of sensors, actuators, and rules placed will determine the level of "intelligence" of the home. The result is not a Smart Home but multiple levels of Smart Homes, starting from the installation and managing the essential functionalities (heating, intrusion, fire safety). The current revolution is driven by the multitude of new products, which allow, for a much more affordable cost, to benefit from features that were once reserved for very high-end homes (Bou, 2014).

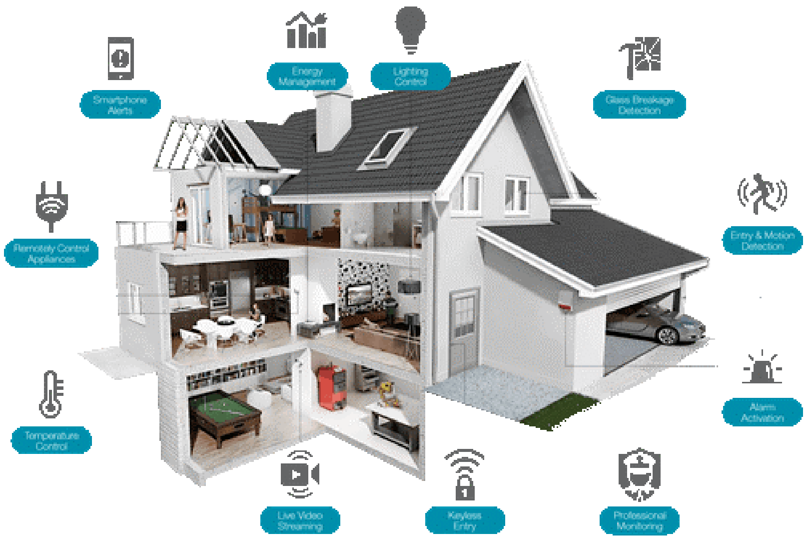

Figure 2.1: Smart House [1]

It seems worth mentioning that home automation is not only intended for new homes. In fact, many people have in mind that these techniques are only applied to houses under construction. Homes undergoing renovation are also affected by home automation and can therefore evolve to improve comfort and safety.

Smart Home Applications

Smart homes work in the following areas [4]:

- **Comfort:** automation, customization, scenarios, system intelligence.
- **Energy savings:** measurement and recovery of consumption data.
- **Security:** fire alarm, flood, intrusion, video surveillance, etc.
- **Home care:** loss of autonomy in the residential sector (homes).

Smart Building

Definition

The term "Smart Building" refers to the different technologies that are integrated into buildings (Hoy, 2016) [1]. The notion of a "smart building" is still up for debate. The smart building takes several definitions in different parts of the world.

In the United States of America, a smart building is a system that creates a productive and profitable environment by improving its main components: structure, systems, and services, and by managing the interrelationships between these elements. In Europe, the UK-based Smart Buildings Group defines a smart building as one that creates an environment that increases the efficiency of building occupants while enabling efficient management of resources and minimizing material cost of living, and facilities." Whereas in Asia, a smart building must meet three conditions (Meg and Dje, 2019):

- Advanced control systems should be installed automatically in the building to control various amenities such as temperature, air conditioning, lighting, fire, and security.
- To allow data transfer between floors, the building needs to have a solid network infrastructure.
- The structure should have suitable telecommunications facilities.

In other terms, a smart building should contain three essential functions:

- Automation of communication (CA).
- Office automation (BA).
- Building management automation (Meg and Dje, 2019).

Figure 2.2: An Overview of Smart Building (Bel and Que, 2017)

Smart Buildings Challenges

Given the current economic and energy situations, smart buildings must meet a number of needs to improve them. In general, smart buildings should meet the following expectations (Bel and Que, 2017) :

- Improvement of building security;
- Adaptation of the operation of the equipment to the presence of the occupants and their activities;
- Improvement of the comfort of the inhabitants, such as improved heating and cooling systems;
- Development and reinforcement of ventilation systems;
- Generation of energy, for instance, using photovoltaic panels;
- Information and awareness: measurement and monitoring of energy consumption for each type of user, occupant, operator, maintainer, and owner.

Smart Cities

Definition

The concept of "Smart City" initially emerged in the 1990s to highlight the influence of new ICT on contemporary city infrastructures. In general, there are many keywords in the majority of the existing definitions of smart city, including infrastructure, efficiency, resources, technology, and data [1].

A smart city represents a modern concept in the currently developed urban area. The aim is to improve city residents' conditions of living by making the city more flexible and efficient through the use of new technologies that rely on an ecosystem of objects and services. This new mode of city management encompasses public infrastructure (buildings, street furniture, home automation, etc.), networks (water, electricity, gas, telecoms); transport (public transport, smart roads, smart cars, mobility by bicycle or on foot, etc.); e-services and e-administrations [5]. According to (Hal et al., 2000), A smart city makes use of all technology and resources to support urban centers efficiently and collaboratively.

The concept of a "smart city" thus covers a large number of areas, all converging towards one of these three ambitions: efficiency, good living, and sustainability. The next figure presents a typical smart city.

Figure 2.3: A Typical Smart City Network

In fact, ICT is operated within a smart city to increase operational effectiveness, disseminate information, and deliver better state services and public welfare. By utilizing smart technologies and data processing, a smart city strives to enhance residents' quality of life, improve city management, and boost the economy [7]. No matter how much technology is available, the value resides in how this technology is used.

Smart Cities Characteristics

The smartness of a city is affected by a variety of characteristics, which include [7]:

- A technological infrastructure.
- Environmental initiatives.
- Highly efficient and functional public transportation.
- Reliable and forward-thinking city planning.
- Use of the resources of a smart city by its citizens.

Importance of Smart Cities

Urbanization is a never-ending process. By 2050, cities are predicted to hold 85% of the world's population, posing supply, resource, and environmental issues if immediate and efficient action is not adopted [6]. Undoubtedly, we will have built many new cities in the following 40 years. Cities have built on outdated models and are now in critical need of preventing the collapse caused by overpopulation [8]. Humanity's future is unavoidably urban, and digitization is a necessary, unstoppable transformation to provide future-proof cities centered on residents and the environment.

Smart Cities emerged to create more sustainable cities. There is a potential for a more inclusive and sustainable society at the intersection of technology and cities, and smart city technology is crucial to achieving success and meeting these goals [9]. Citizens and local government officials can collaborate to create initiatives and employ smart technologies to manage resources and assets in the growing urban environment in smart cities [7].

Examples of Smart Cities

World Context

Cities all over the world are at various levels of developing and implementing smart technologies. Nonetheless, there are others who are in the lead and are paving the way for truly smart cities. Barcelona, Columbus, Dubai, Hong Kong, Kansas City, London, Melbourne, New York City, Reykjavik, San Diego, California, Singapore, Tokyo, Toronto, and Vienna are among them [7].

Algerian Context

The Algerian city is no exception to this new concept because technology imposes itself and conditions the way of life of our citizens. A first experience has already been performed with the new Sidi Abdallah city (inaugurated in 2011 and located west of Algiers). This city was described as "intelligent and interconnected," with "high-tech" infrastructure and vast parks and green spaces. Unfortunately, this project has yet to be completed. Other new smart city projects have been launched, but they are also taking their time to be concrete (Mha, 2019).

Figure 2.4: Algiers Smart City project logo (Mha, 2019)

The "Algiers smart city," whose strategic intent was set by the master program for development and urban planning for 2035, intends to overcome the fundamental difficulties that Algiers currently has to convert it into a more contemporary, accessible, safer, cleaner, and more appealing city (Mha, 2019).

Main Components and Smart IoT Devices

One usually needs several components to implement a system in a smart environment, which are (Rua et al., 2018):

1. **Sensors or Cameras:** A set of sensors or cameras (or both) with the purpose of collecting raw data from the environment. These sensors must be connected and attached to the house itself or the inhabitants and connected to the overall system. Cameras and heat sensors, for example, may monitor temperature and humidity in the home.

2. **Main computer:** Represents the set of software and hardware that allow the appropriate functioning of the smart environment by controlling all the sensors and actuators. This computer contains one or many processors that allow the calculation of the collected data. Also contains a set of software in the form of an API allowing the calculation of data. It also needs memory to store and manage the collected data. The brain will allow access to the internet to control the various components or be notified remotely.

3. **A network:** represents the connections between the different components, either through a network of cables or wireless (wi-fi).

4. **Actuator:** component allowing the execution and control of commands and actions in the server. These commands can launch a device (washing machine for example) or change its state (change the temperature of an air conditioner.

5. **Interface between the human and the smart environment:** a tool allowing communication with the system. The interface could be a graphical interface on a smartphone, computer, or stationary tablet. It can either be vocal-based (through voice instructions) or gestural in nature (heads, hand gestures, etc.).

Sensor types are becoming increasingly diverse, and they are all around us. In reality, the smart home domain has seen a surge in sensor adoption. The scattered sensors form an invisible wireless network that connects everything. According to predictions, about 50 billion smart devices (IoT) will be in use worldwide by 2030 (Fu, 2021).

The top four IoT application examples, which range from simple devices to big applications, are as follows (Fu, 2021):

- **Smart home gadgets:** They are designed to give homeowners a safer and more convenient living environment. A smart lock is a gadget that helps to eliminate the problem of misplacing a physical key. IoT thermostats enable temperature management based on user preferences, allowing for more granular control and energy savings. To be informed every morning, use the Smart Mirror to show the current weather, time, date, and other information from the smartphone or smartwatch.
- **Smart manufacturing:** The IoT is expected to reduce costs for the manufacturing industry. Enhancing networking, automation, and data analytics can help prevent and identify potential problems in the process chain early on. For instance, Google Glass' augmented reality capabilities can show written instructions right in the user's range of vision to hasten construction.
- **Smart farming:** Precision farming is a different term for smart farming. This attempts to use sensors and image recognition to distinguish between plants and weeds, as well as to use digitization and enhanced automation to directly influence how the plants are nursed and produced. Wild animals are driven away using drones supplied with infrared and visible cameras.
- **Futuristic driverless cars:** Due to the abundance of sensors, cloud architecture, the internet, and other technological advancements, the data gathered can be employed to create clever algorithms that will aid the automobile in understanding its surroundings and making the best control choices. As autonomous vehicles are predicted to minimize the population's need for automobile ownership, such sustainable transportation options should result in a drop in the number of cars in urban areas [7].

The next part of this chapter is devoted to ML and DL techniques and algorithms.

2.3 Deep Leaning Algorithms

This section will concentrate on deep learning, examining its models, which includes the two primary deep learning algorithms used for human activity recognition.

2.3.1 Introduction to Machine Learning

Machine learning, abbreviated as ML, is a form of artificial intelligence, which permits machines such as computers to learn with no explicit programming. Computers, however, require data to analyze and train in order to learn and evolve. It is the technology that enables full utilization of big data's potential. The ability of conventional machine learning methods to handle natural data in its raw form has been limited, as mentioned in (Lec et al., 2015).

Actually, deep learning, abbreviated as DL, represents a feature of AI. It is just a subfield of the realm of machine learning, using artificial neural networks inspired by the human brain's function and structure [10]. Data processing for object detection, money laundering or fraud, speech or activity recognition, language translation, and decision-making can all benefit from the adoption of DL. DL-based applications are also able to learn with no human supervision, relying on unlabeled and unstructured data (Goo et al., 2016).

We present in this part the approaches of deep learning as well as their architectures.

2.3.2 Deep Learning Approaches

ML and DL methods can be classified into supervised, semi-supervised, and unsupervised approaches. Furthermore, there is another type of learning known as Reinforcement Learning (RL) or Deep RL (DRL), which is frequently addressed using unsupervised or semi-supervised learning approaches (Alo et al., 2019).

1. **Supervised learning:** refers to the learning approach where the data used are labeled. The environment contains a collection of inputs and outputs for the supervised DL method. The network parameters will be adjusted to provide a better fit to the intended outputs. The deep learning method includes CNN "convolutional

neural networks", DNN "deep neural networks", and RNN "recurrent neural networks" techniques. The latter incorporates GRU "gated recurrent units networks" and LSTM "longshort term memory networks" (Alo et al., 2019).

2. **Semisupervised learning:** is a learning approach that makes use of partially labeled data sets (also called reinforcement learning). Semisupervised learning techniques such as GAN "generative adversarial networks" and DRL are applied in some circumstances (Alo et al., 2019).

3. **Unsupervised learning:** This approach of learning is based on data that has not been labeled. In order to determine the structure or discover the unknown relationships in the incoming data, the network will learn the key features and their internal representation. Techniques like dimensionality reduction, clustering, and generative algorithms are frequently deemed unsupervised learning techniques. Many deep learning techniques, including RBM "Restricted Boltzmann Machines", Automatic Encoders (AE), and the recently created GAN, are effective for clustering and nonlinear dimensionality reduction (Aru et al., 2017).

4. **Transfer learning:** enables data scientists to gain knowledge from a machine-learning model that has previously been applied to a comparable problem. Numerous pretrained models can be employed as the initial point for training a neural network rather than beginning from scratch. The pretrained model (weights and parameters trained on a large old database) is reused and refined with the new database. Our own classifier will take the place of the network's last layer. After that, we train the network while freezing the weights of all additional layers. These already-trained models provide a more robust architecture, saving time and resources. When there is not enough data for training or when we need better results rapidly, transfer learning is employed (Hon and Kha, 2017).

Deep Learning focuses mainly on three basic neural networks, including (Hon and Kha, 2017):

- **RNN** "Recurrent Neural Networks";
- **GAN** "Generative Adversarial Networks";
- **CNN** "Convolutional Neural Networks";

2.3.3 Deep Learning Models

In this section, we will discuss the deep learning models that were used in our design (namely CNN and LSTM).

Convolutional Neural Network (CNN)

The term "Convolutional Neural Network" refers to the network's usage of a mathematical procedure known as convolution. Convolutional networks are a kind of neural network that replaces the general multiplication matrix in at least single layer with convolution. CNN is one of the best learning algorithms for performing the convolution operation, which aids in the extraction of relevant features from locally connected data points. The output of the convolutional kernels is then given to the activation function (a nonlinear processing unit) that supports both learning abstractions and the introduction of nonlinearity in the space of features. This nonlinearity generates various activation patterns, which facilitates the learning of meaning differences in images. The CNN topology is split into many training stages that include convolutional layers, nonlinear processing units, and down-sampling layers (Kha et al., 2020) (Jar et al., 2009). The general structure of a CNN network is depicted in figure 2.5.

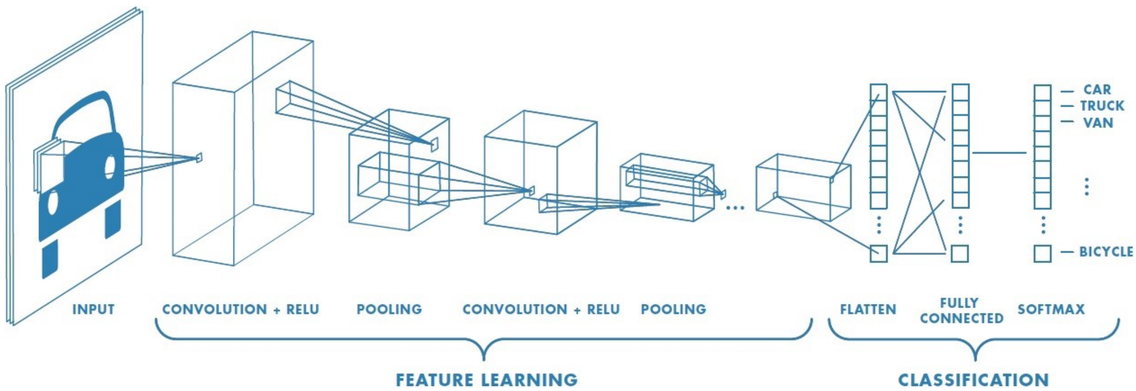

Figure 2.5: Neural Network with Multiple Layers (Kha et al., 2020)

- **Convolution Layer:** to extract features out of an input image. Convolution maintains the association among pixels by learning the image's features via the input data of small squares. This mathematical operation has two inputs, namely a kernel or filter and an image matrix (Ind et al., 2018). Figure 2.6 illustrates a simple filter operation for a convolution step.

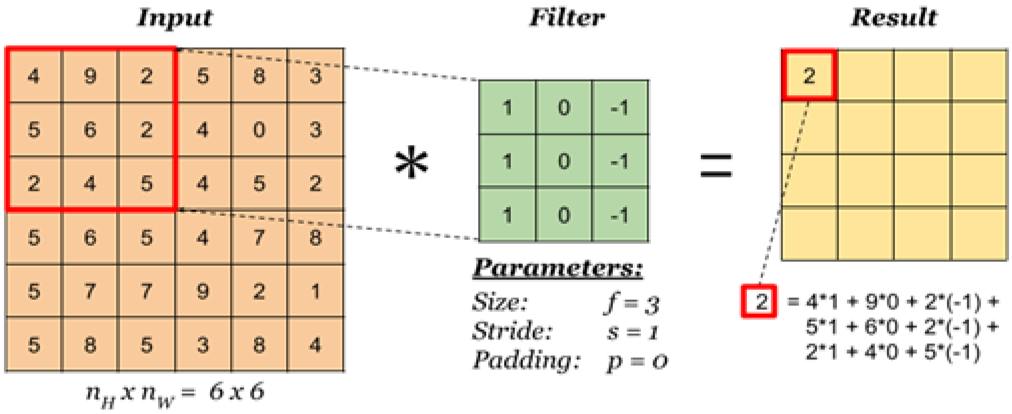

Figure 2.6: Convolution Operation [11]

- **Pooling layer:** refers to the method of subsampling input that is typically positioned between two convolution layers. Pooling layers differ from convolutional layers by having no weighted values. Subsampling the image helps alleviate the computational load of the CNN. The goal is to reduce the dimensionality of an input representation. The pooling acts only to aggregate values with varying aggregation functions. There are different kinds of pooling (Ind et al., 2018): (1) Maximum pooling which takes the pixel, which has the maximum value among all the pixels of the selection. Figure 2.7. (a), (2) Average pooling which takes the pixels, which has the average value of all the pixels of the selection. See figure 2.8. (b).

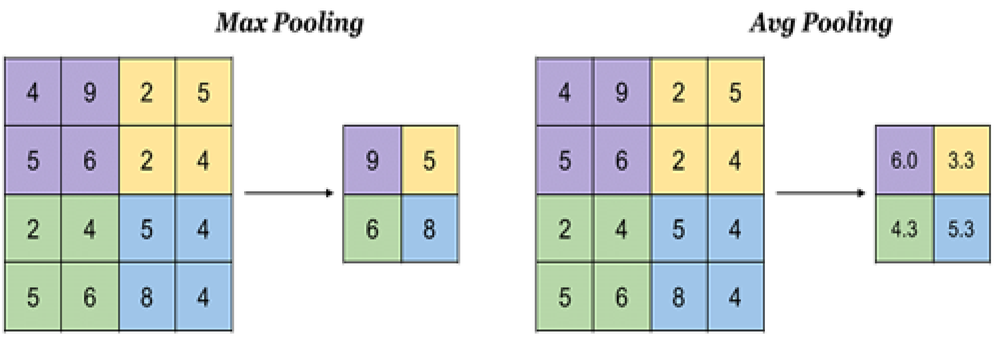

Figure 2.7: (a) Maximal pooling (b) Average pooling [11]

- **Fully connected layer:** In conventional models, the FC "fully connected layer" is equivalent to the fully connected network. As shown in figure 2.8, the output of the first phase (containing convolution and repetitive pooling) is given to the FC layer, and the product operation of the weight vector and the input vector is calculated to yield the final output (Ind et al., 2018).

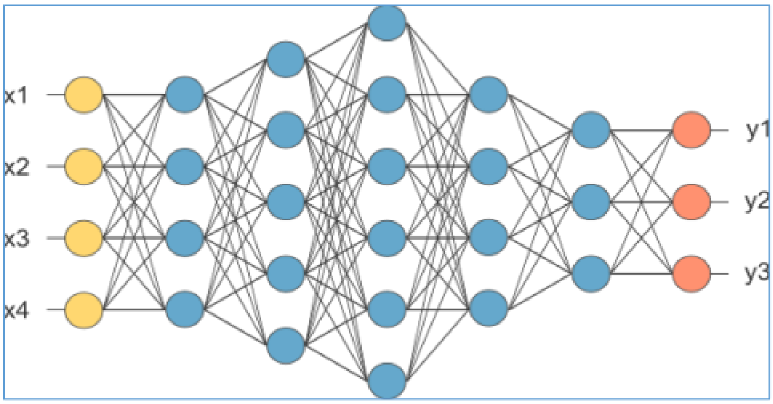

Figure 2.8: Fully Connected Layer (Ind et al., 2018)

The convolutional neural network presupposes that the model's inputs as well as outputs are independent of one another. However, because the acquired data is time-dependent, time information must be included in the input data in some applications. LSTM, an RNN extension, has been proposed as a solution to this problem. It stores and outputs data using memory cells rather than loop units.

Recurrent Neural Networks (RNN) RNN are a critical variation of NN, which are already widely employed in NLP "Natural Language Processing". They are referred to as recurrent since they do the same action on each component of a sequence, and the output relies on past calculations. In other words, RNN have a "memory" of calculation of what has been captured of information. Theoretically, RNN can use information in infinitely long sequences, but practically, they can only look a limited number of steps back. RNN are a sort of neural network with hidden states, allowing past predictions to serve as inputs (She, 2020). Figure 2.9 displays the form of this variety of networks.

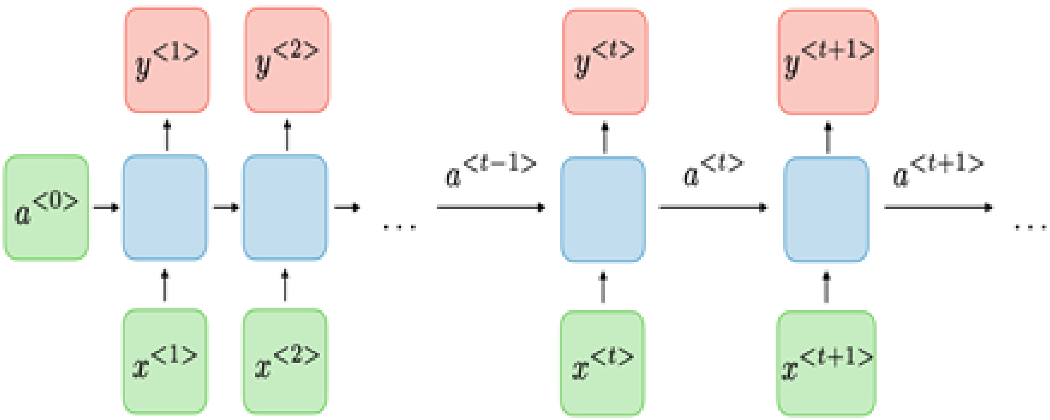

Figure 2.9: RNN Architecture [12]

Long Short Term Memory (LSTM) Networks

They are a form of RNN that can learn dependencies over the long term. Actually, they were coined by (Hoc and Sch, 1997), and they have been improved and popularized by numerous other researchers in the works that follow. They are currently frequently utilized and function incredibly well in a diverse range of situations. LSTM are expressly designed to prevent the long-term problem of dependency. Their default behavior is to remember information for extended periods. As a result, RNN are made up of repeated neural network modules linked together in a chain with a simple topology and a tanh layer (She, 2020) as shown in the following figure:

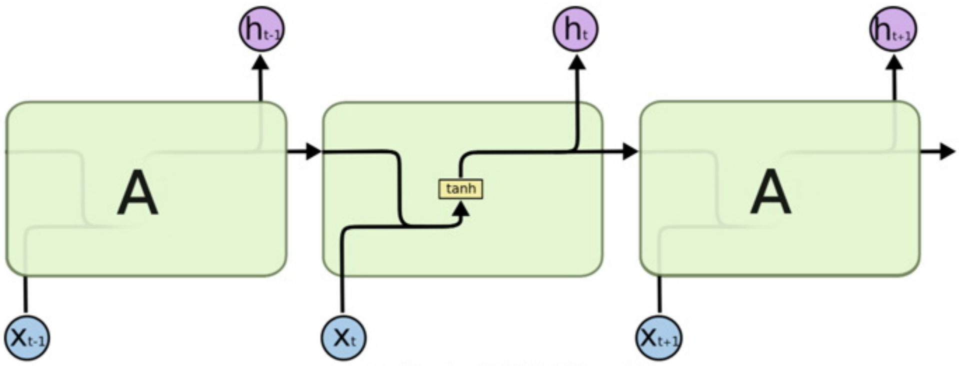

Figure 2.10: RNN Module Architecture (She, 2020)

While LSTM have a chain of modules structure that differs in the repeating modules structure. Each repeated module has four interacting neural network layers (She, 2020), as depicted in the next Figure:

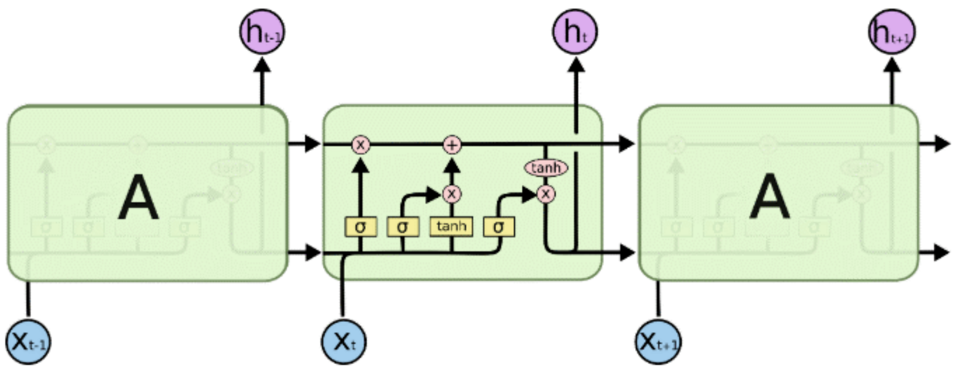

Figure 2.11: Recurrent Module in LSTM (She, 2020)

A more detailed vision of one module from the chain is presented as follows:

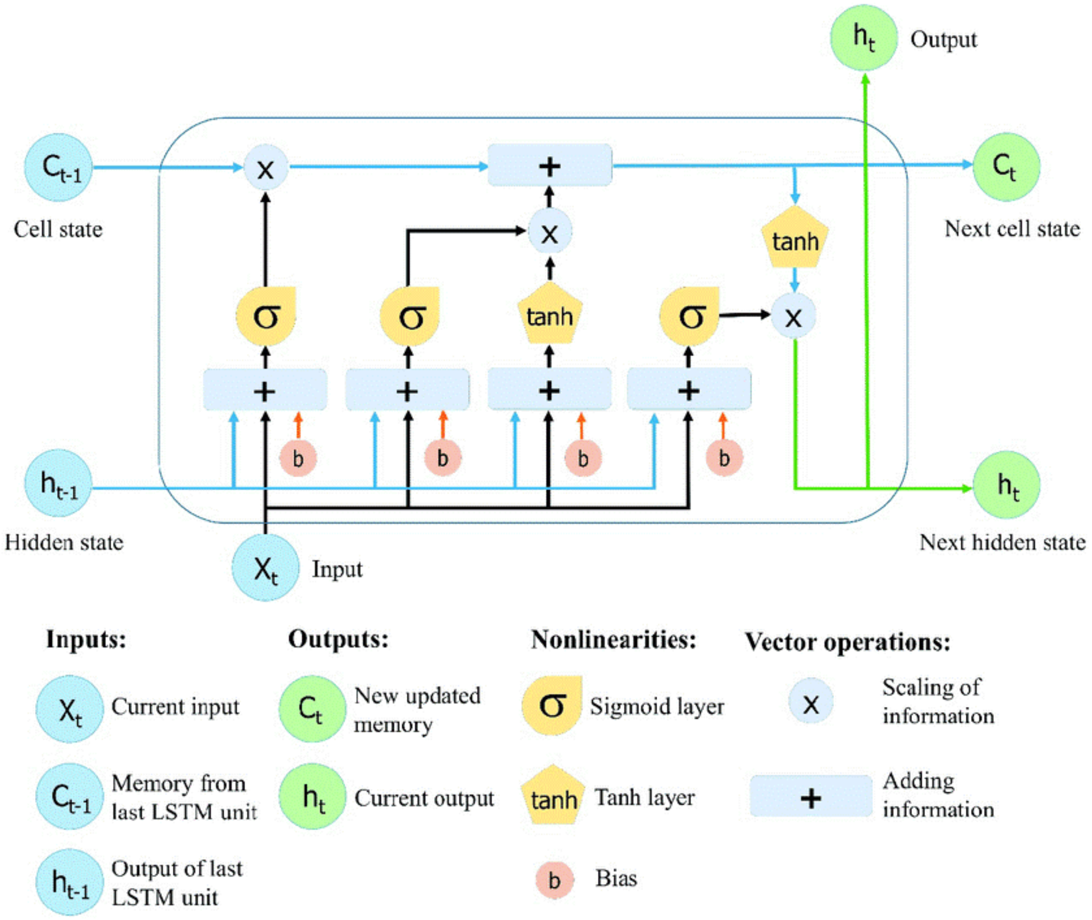

Figure 2.12: Detailed Recurrent Module in LSTM [12]

A typical LSTM network consists of memory cells blocks. The cell and the hidden states are both transferred to the following cell. The cell state is the fundamental data flow chain that permits data to pass forward basically unchanged. Nonetheless, the sigmoid gates can cause some linear changes from the hidden state, where data can be stored or erased from the cell state. A gate comprises various distinct weights (She, 2020). Because the use of gates to manage the operation of memorization, LSTMs are intended to prevent the long-term problem of dependency.

In essence, the removal of information designation from the cell is the initial stage in establishing an LSTM network. The sigmoid function determines data identification and exclusion by using the current input Xt at time t and the previous output of the LSTM unit h_{t-1} at time $t-1$. Furthermore, the sigmoid function determines how much of the old output must be discarded. This gate is termed as the gate of forget (or f_t), where f_t is a vector having values varying from 0 to 1, one for each number in the cell state,C_{t-1} (Le et al., 2019).

$$f_t = \sigma(W_f[h_{t-1}, X_t] + b_f) \tag{2.1}$$

Here, σ represents the sigmoid function, and W_f and b_f stand for the weight and bias matrices, respectively, f_t is the gate of forget.

The next stage is to save the new entry X_t information in the cell state and update it. This stage is split into two sections: the sigmoid layer and the tanh layer. First, the sigmoid layer decides whether to update or reject the information (0 or 1) and then the tanh function assigns weight to the passed data, determining their relevance level (-1 to 1). To update the new cell state, both values are multiplied. This new data memory will then be added to the existing one $C_t - 1$, yielding C_t (Le et al., 2019).

$$i_t = \sigma(W_i[h_{t-1}, X_t] + b_i), \tag{2.2}$$

$$N_t = \tanh(W_n[h_{t-1}, X_t] + b_n), \tag{2.3}$$

$$C_t = C_{t-1} f_t + N_t i_t \tag{2.4}$$

Here, $C_t - 1$ and C_t represent the cell states at time $t - 1$ and t while W and b stand for the weight and bias matrices, respectively, of the cell state.

In the last stage, the output h_t are determined by the output cell state O_t. The sigmoid layer first decides on the output cell state. The recent values provided by the layer tanh are therefore multiplied by the sigmoid output gate O_t from the state of the cell C_t with a value around -1 and 1 (Le et al., 2019).

$$O_t = \sigma(W_0[h_{t-1}, X_t] + b_0), \tag{2.5}$$

$$h_t = O_t \tanh(C_t), \tag{2.6}$$

Here W_0 and b_0 represent the weight and bias matrices, respectively, of the output gate.

Conclusion

The employment of CNN and RNN has become an optimal solution given the promising results obtained in several research studies. This chapter represents a theoretical background on the concepts related to the problem treated in this thesis and its design. It incorporates two major parts.

In the first, we went swiftly over some definitions and clarifications regarding smart environments and IoT devices. In the second, we approached deep learning in some detail, to show the architecture that we have chosen for our application. In the following chapter, we will review the recent works undertaken in the framework of the problem addressed, which is activity recognition of a human in a smart environment.

The convolutional neural network presupposes that the model's inputs as well as outputs are independent. However, because the acquired data is time-dependent, time information must be included in the input data in some applications. LSTM, an RNN extension, has been proposed as a solution to this problem. It stores and erases data using memory cells rather than loop units.

Chapter3

HUMAN ACTIVITY RECOGNITION BASED ON DEEP LEARNING

3.1 Introduction

Modern advancements in Internet of Things technologies (IoT), as well as sensor cost reductions, have urged the development of smart environments to enhance the quality of life, autonomy, health, energy, and a variety of other services. To deliver such services, a smart environment must be capable of comprehending an individual's daily activities within it (Le et al., 2019). Human activity recognition methods are being developed on a daily basis in smart environments. Nonetheless, new difficulties and new challenges arise continuously.

This chapter's objective is to better understand the notion of HAR "Human Activity Recognition" and the relationship between HAR and deep learning in smart environments. We will point out some related work on the recognition of human activity based deep learning, discussing their techniques, results, and putting into exactitude their current limits. Moreover, we will depict the eventual challenges of using deep learning in the HAR systems, as identified by many researchers. Finally, we will end this chapter with a comparison table of recent works.

3.2 Human Activity Recognition

Human Activity Recognition, abbreviated as HAR, is the tracking of a person's activities using a network of sensors and linked devices. This results in a time series of state changes or parameter values as data. RFID, accelerometers, detectors, noise sensors, and motion sensors are examples of sensors that can be placed directly on items, people, or the environment (Le et al., 2019).

HAR has become an attractive research field due to its increasing importance as well as the multiple challenges brought to the research community. These HAR systems are used by researchers to collect data on people's activities. The various HAR systems can indeed be restricted into three distinguishable categories, including sensor-based systems, vision-based systems, and multimodal-based systems (Fer et al., 2020) (see figure 3.1).

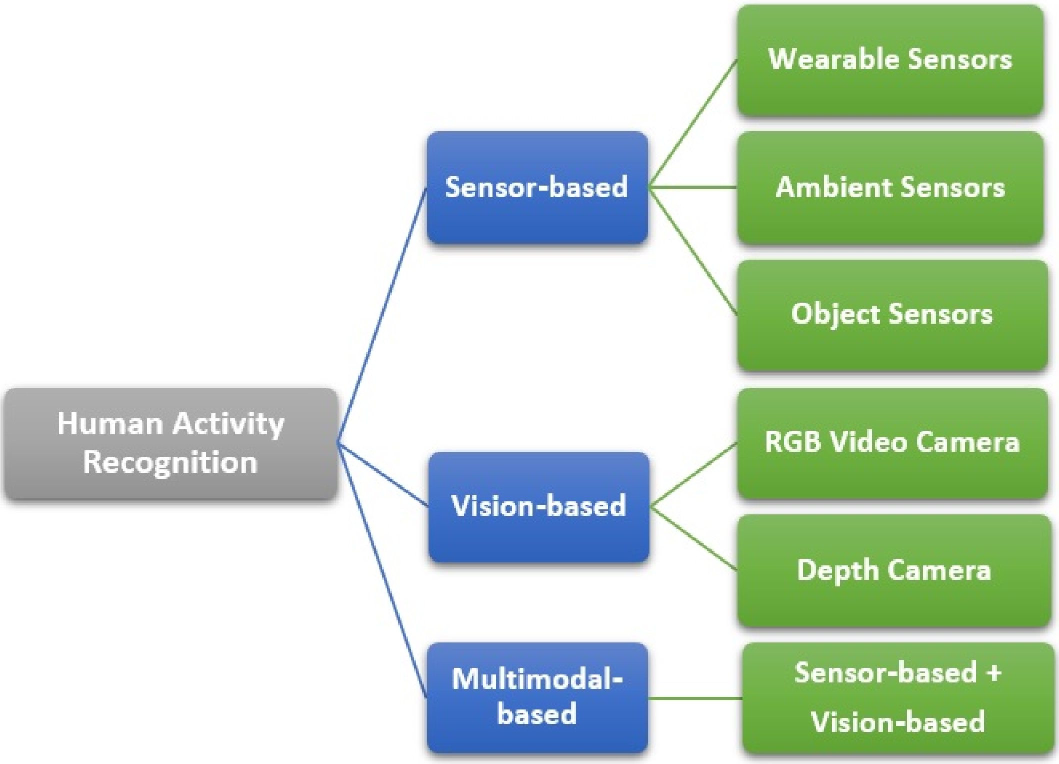

Figure 3.1: Various Systems for HAR (Fer et al., 2020)

As was already mentioned, there are three classes for HAR, sensor, vision, and multimodal-based HAR systems. First, the sensor-based comprises three types, wearable sensors, which are bodily-worn sensors that are attached to a person's limbs or clothing and record details of their activities. The Global Positioning GPS System, Inertial Measurement Unit (Gyroscope, Accelerometer, Magnetometer), smartphone sensors, smartwatch sensors, and biosensors are just a few examples of wearable sensors that can be involved in HAR solutions. Then, the ambient sensors use sensors like temperature, light, pressure, RFID, radar, Wi-Fi, and Bluetooth to provide environmental data. Finally, the sensors installed on vehicles like cars, trains, planes, etc. are the last type. The second is the vision-based HAR, where data is generally acquired by RGB video cameras and depth cameras. A vision-based modality may encounter difficulties because conventional cameras cannot function in total darkness (Fer et al., 2020).

In-depth innovative technology, like the Kinect camera, can act in total darkness to resolve this problem. They can generate a virtual skeleton in three dimensions and offer depth, RGB, and audio data. Third, multimodal HAR systems refer to combining sensor and vision-based techniques for activity detection (Fer et al., 2020) (Bou et al., 2021)a.

While using both visual and sensor data increases the robustness, accuracy, and reliability of activity recognition, it also increases the cost and complexity of the system. Compared to cameras installed in fixed locations, which have their own disadvantages such as high cost, high complexity, and privacy concerns, wearable sensors have the advantages of low cost, ease of handling, and placement. This makes wearable sensors extensively used in recent years. Cameras, Wearable sensors and inertial sensors work together to deliver information that single-modality are unable to do (Fer et al., 2020).

The data commonly collected from the signals is handled by ML algorithms to recognize the events. Hence, such HAR systems can be used in plenty of useful and practical applications in smart environments such as smart homes. For instance, a smart HAR system can constantly control patients for health diagnoses and medication. In addition, it can be placed to predict crimes that may take place in the near future through the automated monitoring of public places (Bou et al., 2021)a.

To understand the connection between HAR and deep learning in smart environments such as the smart home, we have to clarify the notion of the activity and the process of HAR.

3.2.1 Notion of Activity

Activity in our research domain is a group of a person's domestic physical behaviors that can be organized hierarchically into an action. For instance, the activity "sleep" can be fragmented into several different actions, such as "enter the bedroom" and "lie down in bed." They consist of actions that are the atomic steps of the activity, such as "push the door handle," "close the light," etc. Activities are therefore the sum of actions, which are composed of the sum of atomic operations (Ouk, 2019).

3.2.2 General Structure of HAR Systems

Four main stages constitute HAR Systems' general structure. The first step is to attach wearable sensors, for example, to a person's body to measure the desired characteristics, including motion, location, and temperature, among others. These sensors should be able to communicate with an Integration Device (ID), which could be a laptop, PDA, smartphone, or specially designed embedded system. The preprocessing of the data received from the sensors and, in some cases, the forwarding of the data to a server application for real-time monitoring, visualization, and/or analysis represents the ID's primary goal. Depending on the desired level of reliability, either UDP/IP or TCP/IP could be used as the communication protocol (Lar and Lab, 2012). The general data acquisition architecture for HAR systems is shown in the following figure.

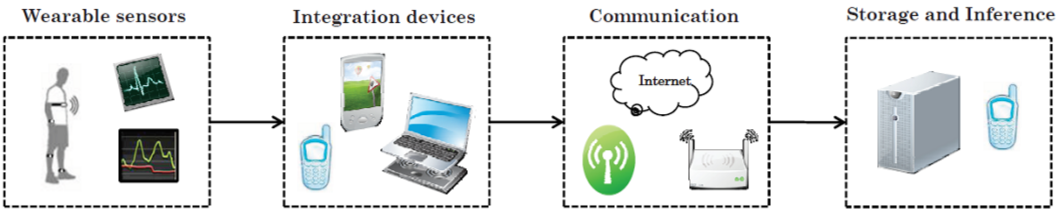

Figure 3.2: Generic Data Acquisition Architecture for HAR (Lar and Lab, 2012)

As stated in the prior section, tracking a person's activities at home can lead to serious privacy problems. Residents are generally reluctant to leave cameras and monitoring systems on when they are at home, even though camera installation can be a part of various security services. Since they are generally less intrusive, sensors have predominated in the applications of everyday activity recognition, especially in smart homes (Lar and Lab, 2012). Smart homes that are based on ambient sensors have emerged as a workable technical solution to deliver a plenty of services, thanks to the development of the IoT and the proliferation of affordable and potent smart devices. The ideal system also requires algorithms and solutions that can benefit from this potential behind the hardware.

3.2.3 Human Activity Recognition Process

HAR recently brought attention to a very difficult research topic. By using a variety of sensors, including pressure detectors, motion sensors, RFID, electrical power analyzers, etc., HAR aims to determine the activities conducted by one or more residents inside the home environment. The HAR procedure consists of multiple steps. The following are the key four steps (Bel et al. 2015):

- **Preprocessing:** removing the unprocessed data from sensor streams to handle incompleteness, get rid of noise and redundancy, and normalize the data;
- **Features extraction:** the process of extracting features from raw data to feed into machine learning;

- **Features selection:** reduce the features' number while improving their quality in order to lower the computational effort required for classification;

- **Classification:** is the process of identifying a given activity using machine learning and logic. HAR involves keeping an eye on and analyzing a person's movements to determine their activities. The HAR is the IoT device's network-based daily activity monitoring of the residents in a smart home. Personalized home assistance services can be offered by a smart home as a result of this monitoring to improve the autonomy, health, and quality of life of its residents, especially for kids, the elderly, and those who are dependent.

In general, the HAR process involves three critical steps: from gathering data on environmental conditions and human behavior, to deciding on the currently conducted activity. These steps are as follows: The first step is receiving sensor data from various sensor technologies. After that, redundancy and noise are removed, and data is gathered and normalized. Second, feature extraction is performed to extract from the data the most crucial activity features, such as spatial and temporal data. Finally, classification determines the activity by using ML and DL models to train data (Mag, 2020). Figure 3.3 illustrates those steps.

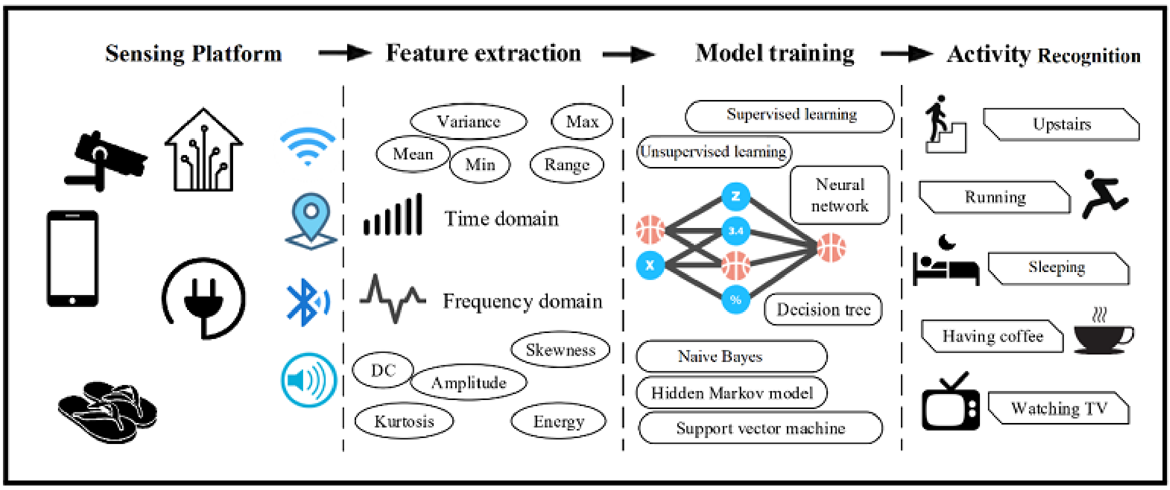

Figure 3.3: An Illustration of Sensor-based HAR Activity Recognition (Mag, 2020)

The ultimate main purpose of HAR systems is to partially or completely replace human operations inside homes, either by anticipating these operations and carrying them out when necessary, or by satisfying the needs and requirements that humans have already defined. For instance, a HAR system can use sensory devices to alert medical personnel in the event of an urgent need after monitoring a resident's health (Ser et al. 2022).

The difficulties with ambient as well as wearable sensors in HAR and the algorithms for HAR in smart environments can be categorized as problems with pattern classification. To deal with this challenge of activity classification, HAR-using DL methods is becoming popular. In the next section, we will overview the related work on the HAR-based DL.

3.3 Related Work

This section is split into three fundamental parts. First, we will summarize a set of related recent works to impart existing deep learning approaches, research directions, and open issues for the research domain of HAR within smart environments that are directly related to our research question. Secondly, we will compare in a table the related work according to the criteria that we set in advance, followed by a discussion. Lastly, we will discuss the challenges of the aforementioned research field.

Traditionally, to analyze and process the collected sensor data, machine-learning models were used to train the processed sensory data [10]. However, as stated in the survey (Wang et al. 2019), a constraint of this method is the signal processing and domain expertise necessary for interpreting raw data and changing the features needed to fit a model. This expertise is very much needed for each sensor modality or new dataset. Briefly, this method is expensive and not scalable.

In the last two decades, DNN models have started to convey their feature learning pledges and attain noteworthy results for HAR. They can proceed with feature learning automatically from raw sensor data and overcome models fit on handcrafted domain-specific features (Wang et al. 2019).

Currently, DL has prospered increasingly by modeling abstractions at a high level from complex raw data in many areas. DL models can learn automatic high-level features from raw signals without human direction. It is known as end-to-end learning (Bou et al., 2021)a. Nonetheless, the key point of DL algorithms is their direct capacity to learn features and classify tasks from raw data in a hierarchical fashion. (Rad et al., 2018) have argued the advantages of adopting DL algorithms to interpret the context and user activity as captured by multi-sensor devices. Moreover, in recent years, various studies were conducted and published on deep learning methods applied to HAR using a sensor-based system. In our research, only the methods based sensors are mentioned.

Two main techniques for neural networks are appropriate for time series classification: CNN models and RNN models. They are demonstrated to perform effectively on HAR with the use of sensor data from fitness tracking devices, smartwatches, and smartphones.

In fact, this part investigates some prominent studies that propose HAR models built using DL methodologies founded on CNN, LSTM, and ultimately hybrid models.

3.3.1 CNN

In recent years, as computational capabilities have increased, CNN has produced outstanding results on sensor-based HAR and surpassed other state-of-the-art methods that need advanced preprocessing or time-consuming handcrafting for feature extraction. For example (Zen et al., 2014), one of the first works using CNN for HAR, where a straightforward CNN model was developed for the data accelerometer. It extracts the acceleration time series' scale-invariant properties and local dependency, using different datasets that achieve a classification accuracy of 88.19% for Skoda, 76.83% for Opportunity, and 96.88% for WISDM, respectively. However, they employed a shallow model with only one accelerometer. (Hes and Tes, 2016) created a multi-sensor recognition framework in which a CNN model for dual accelerometers was provided. The architecture suggested in (Rav et al., 2016) operates on the aggregation of temporal convolutions of inputs, which is used in the spectral domain reserved for inertial signals, and is designed to foster real-time and accurate classification for low-power wearable devices. Yet, it implies the handcrafted features' extraction.

CNN is extensively used in image recognition tasks. It was able to foster the performance of previous works in numerous domains and has also been employed for HAR and Ubiquitous computing. This technique for HAR is extremely widespread, but it has thrived much better in image and video data than in sensor data. CNN is considered to be spatially deep, which aids in the extraction of signal spatial features (Mut and Han, 2020).

There are two varieties of CNN used for HAR: 2D CNN for image processing and 1D CNN for sequence processing (Bou et al., 2021)a. For the 2D CNN variety, studies such as (Jia and Yin, 2015), (Wen and Yin, 2015), and (Goc et al., 2018) have proposed an approach by converting the raw sensor information into an image signal of 2D and then using a two-layer CNN to categorize this signal image in the class of activity recognition. Their experiments confirm that the 2D-based structures could be adjusted to the HAR. However, for the 1D CNN variety, (Wan et al., 2019) and (Sin et al., 2017) have adopted a 1D-based structure due to its high feature extraction ability (See figure 3.4). Furthermore, the most attractive article for this variety was (Hee and Yoo, 2018). In this work, the authors divide activities into two classes: dynamic activities, denoting movement; and static activities, denoting stationary. Thereafter, they developed a CNN model known as the two-stage modeling approach to conquer the issue of individual activity recognition by differentiating between these two main classes using one dimension.

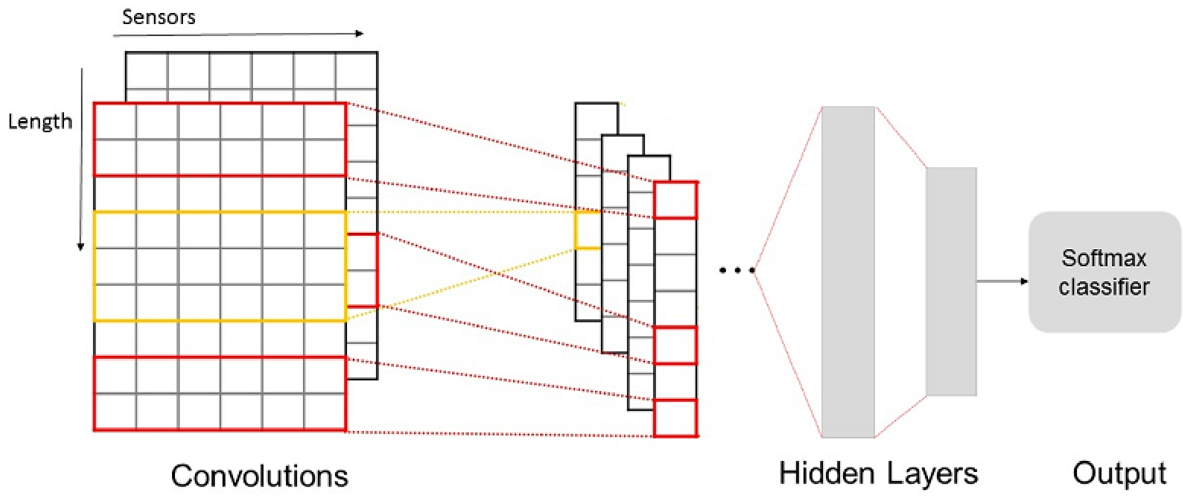

Figure 3.4: 1D CNN Overview for HAR (Sin et al., 2017)

Furthermore, other interesting articles for HAR using smartphone sensors tackle the filter size aspect on CNN. For the side of (Ron and Sun, 2016), the authors show the usefulness of larger filter sizes for signal data. This model attains an accuracy performance of 94.8%. Additionally, a full CNN configuration, which serves as a starting point for new HAR, is offered. However, in opposition, (Tan et al., 2020) investigate lower dimensional filters by employing a series of lightweight structures in the CNN model for HAR systems based on wearable devices. The standard filters could be replaced by a collection of smaller Lego filters that do not rely on any unique network topologies. The evaluation findings reveal that Lego filters and local loss within CNN can reduce computations and achieve 96.9% accuracy with UCI-HAR and 98.82% accuracy with WISDM datasets. In other words, the Lego CNN with local loss is faster, smaller, and more accurate.

As a result, CNN's effectiveness in HAR is owing to its ability to acquire discriminative and powerful features, as well as to use convolutions throughout a 1D temporal sequence to capture local relationships between neighboring input samples. CNN employs parameter sharing to detect local dependencies over time by using the same convolutional filter per time segment. Nevertheless, sharing parameters is ineffective for detecting all the connections and correlations across input samples (Mur and Jae, 2017). Generally, CNN assumes independence between the inputs and outputs of the model. Because the data collected is time-dependent, time information must be included in the input data. To tackle this issue, an LSTM, which is an RNN extension, has been proposed. It stores and outputs data using memory cells rather than loop units (Sun et al., 2018).

3.3.2 LSTM

It is worth mentioning that HAR in an environment of smart homes is an issue of pattern recognition in time series through irregular sampling. In this regard, RNN shows, currently, a stronger ability to represent time series or sequential multi-dimensional data. Nonetheless, as stated in [13], RNN suffers from the long-term dependency problem. To avoid this problem, an LSTM variation has been proposed.

(Ull et al., 2019) suggested a stacked LSTM network for detecting six human behaviors in smartphone data. The network comprises five LSTM cells that have been trained end-to-end on sensor data. A single-layer NN precedes the network and preprocesses the following stacked LSTM network's data. The network is tested using the public UCI dataset, and its performance is measured in terms of precision-recall and average accuracy. With no manual feature engineering, the suggested network improves average accuracy by 93%.

A deep residual bidirectional LSTM network (Res-Bidir-LSTM) is proposed by Zhao et al., with the advantage of being able to combine the forward state with the backward state (positive and negative time direction). Second, residual links between stacked cells operate as gradient shortcuts, thereby avoiding gradient vanishing. In general, the suggested network enhances temporal dimension by bidirectional cells, and spatial dimensions by stacked residual connections, to increase the recognition rate. When using the Opportunity and UCI datasets, the accuracy improves in a significant way (Zha et al., 2018).

In the same line and because of their success in extracting characteristics and producing predictions, BiLSTM techniques in HAR have gained popularity. The proposed model, BiLSTM, is founded on residual blocks as well as a bi-directional LSTM. The model first automatically extracts spatial attributes from multidimensional sensors with the residual block. Second, it uses BiLSTM to acquire bi-directional temporal feature relationships. Finally, the collected features are sent into the Softmax layer to perform the recognition of human actions. On the handmade, WISDM, and PAMAP2 datasets, the suggested HAR approach has an accuracy of 96.95%, 97.32%, and 97.15%, respectively. As a result, it has greater performance and fewer parameters than some existing models (Li and Wan, 2022).

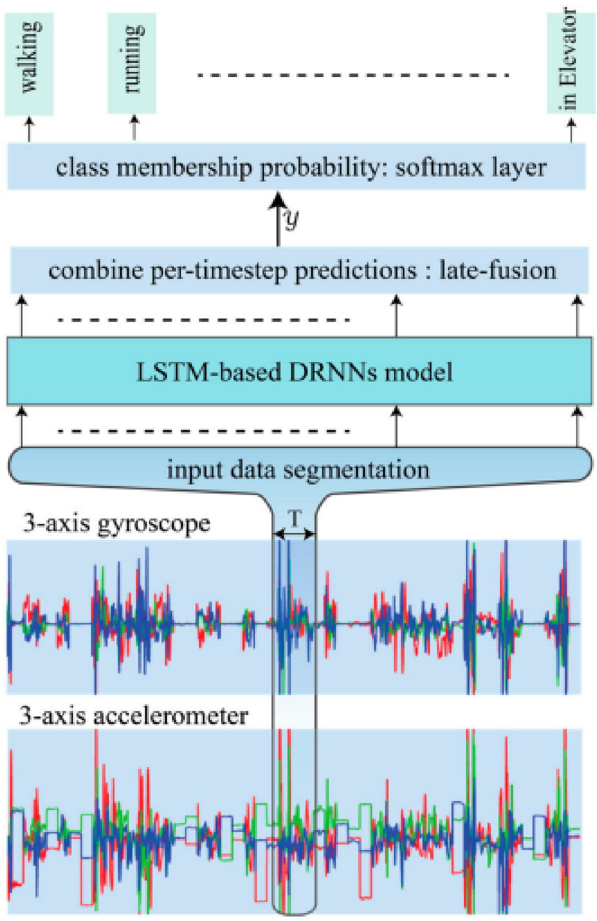

Figure 3.5: LSTM-based DRNN architecture (Mur and Jae, 2017)

LSTM-based unidirectional, bidirectional, and cascading architectures were proposed by (Mur and Jae, 2017). Those models can categorize variable-length periods of human activity by capturing long interdependencies in changing-length input window sequences. Several datasets were explored, but USC-HAD, achieved the best performance with the unidirectional LSTM architecture, with 97.8% of accuracy. Thus, the three aforementioned LSTM-based DRNN models are effective in classifying human activities. The previous figure represents their basic architecture.

Alawneh et al. have compared the HAR accuracy of the two unidirectional LSTM and bidirectional BiLSTM models in their paper. They test these two LSTM models on two separate datasets with different movement classes. The first is UniMiB SHAR dataset, which includes 17 different activities and fall states. The second is WISDM dataset, which contains six kinds of human motions gathered under controlled laboratory conditions. In terms of recognition accuracy, the results show that the BiLSTM outperforms the LSTM. However, the BiLSTM technique requires more time to train, which may limit its application on large datasets (Ala et al., 2020).

Nonetheless, sequential data analysis systems can only analyze primitive and simple actions; they cannot yet handle complicated ones. A single action or movement, such as walking, running, or turning on the light, constitutes a basic activity. A complicated activity, such as baking or typing, consists of a series of acts that may involve various interactions with things, equipment, or other people (Bou et al., 2021)a. As stated in the literature, the only extension for complex systems employing deep learning is the hierarchical LTSM, utilizing two layers that handle the complexity of actions in HAR video-based (Dev et al., 2019). Otherwise, as demonstrated in the research by Wan and Liu, 2020), the usage of two-hidden layers in the model of LSTM for HAR employing wearables can serve as inspiration for HAR in the application of smart homes.

3.3.3 Hybrid Models

Usually, for HAR problems, an LSTM is coupled with a CNN, as a CNN-LSTM model. Therefore, a CNN model is for extracting features from a raw data, and then an LSTM interprets the output features from the CNN. Recently, extensive HAR research has concentrated on the CNN and RNN hybrid models. (Naf et al., 2021) have created a model that uses CNN with kernel dimension variations in conjunction with BiLSTM to retrieve features at various resolutions. When compared to the UCI dataset (97.05%), the suggested technique obtains a greater accuracy of 98.53% in the WISDM dataset. The CNN-BiLSTM architecture is illustrated in the figure 3.6.

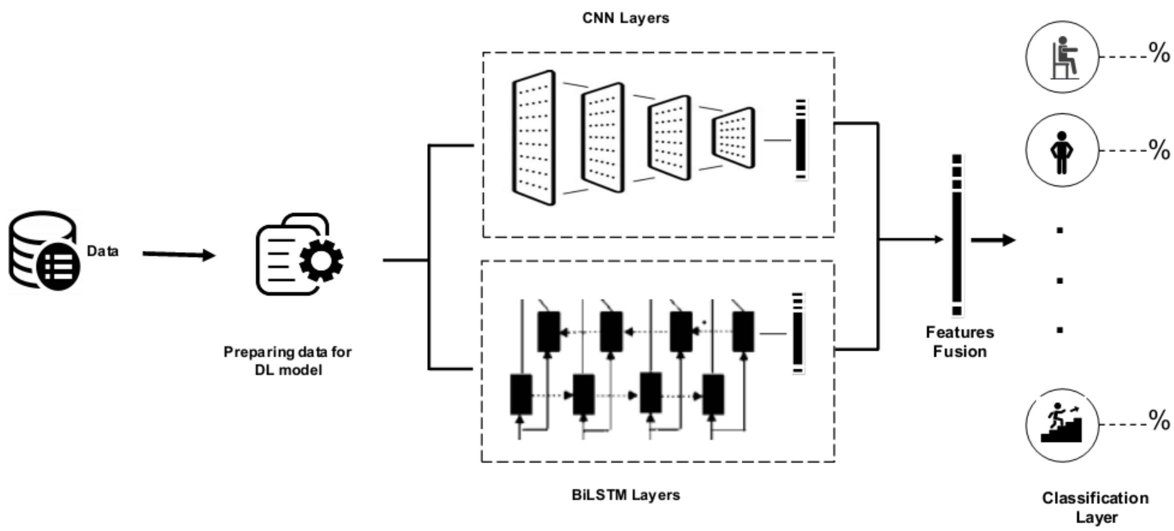

Figure 3.6: CNN-BiLSTM Architecture (Naf et al., 2021).

In the same vein, Nan et al. presented research on a CNN-LSTM multichannel-based HAR for the elderly using smartphones. The investigation included 53 senior participants. The results of tests on 1D CNN, CNN-LSTM, multichannel CNN, and multichannel CNN-LSTM models revealed that the CNN-LSTM with a multichannel model was the most effective (Nan et al., 2020). An example of the above-mentioned coupling models is the work (Ord and Rog, 2016), where they introduce a DeepConvLSTM by using convolutional layers in conjunction with RNN. According to the results, their technique outperforms rival non-recurrent deep networks.

To recognize complicated behaviors, four deep learning hybrid models built of CNN and RNN (LSTMs, GRUs, BiLSTMs, and BiGRUs) were examined in (Mek and Jit, 2021). Experiments on the UTwente dataset revealed that CNN-BiGRU outperforms numerous other models. Moreover, the study of (Pus and Shr, 2022) presents a method that can recognize and identify a change from one activity to another. A deep CNN using this method to extract the features is being built. Then, GRU captures the long-term dependency between the different actions, which helps to improve the identification of complex activities. Hence, the CNN-GRU model achieves a detection accuracy of 96.79%.

Furthermore, some papers incorporated self-attention into HAR. Abdel et al. presented a network of dual-channel that comprises a convolutional residual network, an LSTM, and an attention mechanism. On WISDM, the proposed architecture's accuracy reached 98.9% (Abd et al., 2020). In the same line of thought, (Ma et al., 2019) used self-attention-based multimodal NN to detect human actions, which combined three modules: a CNN, an attention mechanism, and a GRU. The experiments show that this method performs competitively in activity recognition across three public datasets when compared to other published methods.

Moreover, another paper has explored the AE in the direction of combining basic deep learning techniques. They have proposed an original deep learning architecture. It consists of three modules, which aim to purify the noise in raw data by AE, high-level features extraction by CNN, and disclose the temporal dependencies among data for precise HAR by LSTM, respectively. The proposed method overcomes existing methods and achieves 97.4% activity recognition accuracy without human intervention (Zou et al., 2018).

Overall, the former works have shown that deep learning algorithms, such as CNN, are capable of extracting features. Consequently, they are very rapid in the training phase and get accuracy levels near LSTM (Sin et al., 2017). Nevertheless, LSTM achieves better performance owing to their ability to use the dependencies of the long term. As a result, according to the literature, a hybrid conjunction of CNN 1D and LSTM structures is the best performing one (Bou et al., 2021)a.

3.3.4 Comparison of HAR Methods in Smart homes Environments

A number of methods and techniques have been studied for HAR in smart home environments. The following Table 3.1 presents a summary and comparison of the most recent and prominent works of the DL methods for HAR systems in smart environments.

Table 3.1. Summary and Comparison of HAR Methods in Smart Home Environments

Reference Approach	Deep Model	Feature Extraction	Activity	Dataset	Dataset Preprocessing	Performance Metric	Performance
(Abd et al., 2020) ST-DeepHAR	1D CNN + LSTM + Attention Model	No	Locomotion activity	WISDM, UCI HAR	No	Accuracy, Precision, Recall, F1 score	98.90 % 97.70 % Accuracy
(Bou et al., 2021)b LSTM + Embedding FCN- Embedding	LSTM + FCN	NLP + TSC	Daily life activity	ARUBA-CASAS, MILAN - CASAS	Segmentation + Word transformation + Sliding window Frequency based Encoding and Embedding	Accuracy, F1 score	92.44 % 90.86% Accuracy
(Che et al., 2016) LSTM	LSTM	Long short memory network	Locomotion activity	WISDM	No	Accuracy	92.1%
(Che et al., 2019) MASTAttn	LSTM + Attention Model	No	Locomotion activity	MHEALTH, PAMAP2, UCI HAR	No	Accuracy, Precision, Recall, F1 score	96.1% 89.9 % 85.5 % F1
(Fan et al., 2019) DeepTag	1D CNN +LSTM	CNN	Locomotion activity	Self-collected dataset	Multipath Periodogram + Pseudospectrum	Accuracy	94%
(Gho et al., 2019) activity2vec	LSTM	Seq2seq	Locomotion activity, Daily life activity	UCI-HAR, ARUBA-CASAS	Not mentioned	F1-score	0.923 0.476
(Goc et al., 2018) DCNN	2D CNN	CNN	Daily life activity	ARUBA-CASAS	Segmentation + Sliding window + Activity image	Accuracy, Precision, Recall, F1-score,	0.961, 0.949, 0.951, 99.23%
(Hee and Yoo, 2018) CNN+Sharpen	1D CNN	CNN	Locomotion activity	UCI-HAR, Opportunity	Denoising + Segmentation + Sliding window	Accuracy	97.62% 94.2%

(Ign, 2018) CNN	CNN	CNN+ Statistical features	Locomotion activity	UCI-HAR, WISDM	Data centering+ Normalization	Accuracy	90.42% 93.32%
(Ino et al., 2018) DRNN	LSTM	No	Locomotion activity	UCI-HAR,	No	Accuracy	95.42%
(Jia and Yin, 2015) DCNN	2D CNN	No	Daily life activity	UCI HAR , USC-HAD, SHO	No	Accuracy, Comput-cost	97.59% 97.83% 99.93 % Accuracy
(Li and Wan, 2022) BiLSTM	LSTM	Residual block + BiLSTM	Sport activity, Health activity	Homemade, WISDM, PAMAP2	Mean + Standard Deviation	Accuracy	96.95% 97.32% 97.15%
(Ma et al., 2019) AttnSense	1D CNN + GRU	CNN	Locomotion, Factory maintenance	STISEN, Skoda, PAMAP2	Data Augment + Fast Fourier Transform + Segmentation	F1 score	96.5% 93.1% 89.3%
(Mek and Jit, 2021) CNN-BiGRU	1D CNN + BiGRU	CNN	Simple + Complex activity	UTwente	Noise reduction + Missing+ Filling + Normalization	Accuracy, Precision, Recall, F1-score,	98.78% Accuracy
(Moh et al., 2020) DCNN AdaBoost	1D CNN	CNN	Daily life activity	ADL	Segmentation + Converting binary string data into greyscale images	Accuracy	99.5%
(Mur and Jae, 2017) DRNN	LSTM + RNN	No	Daily activity, Locomotion, health-related activity, factory maintenance	UCI-HAR, USC-HAD, Opportunity, FOG, Skoda	No	Precision, Recall, Accuracy, F1 score	96.7% 97.8% 92% 93% 92.6% Accuracy
(Naf et al., 2021) CNN-BiLSTM	1D CNN + BiLSTM	CNN	Locomotion activity	WISDM, UCI HAR	Time Series Data + Segmentation	Accuracy	98.53% 97.05%

(Nan et al., 2020) Multichannel CNN-LSTM	multichannel 1D CNN+LSTM	CNN	Free-living activity Older people	NRA	Segmentation + Sliding window	Accuracy, Comput-cost	81.1 % Accuracy
(Ord and Rog, 2016) DeepConvLSTM	Conv+LSTM	Conv	Locomotion activity, Factory maintenance,	Opportunity, Skoda	No	F1 score	95.8 % F1 score
(Par et al., 2018) Residual- RNN	RNN	No	Daily life activity	MIT	Attention module	Accuracy RMSE	90.85%
(Pus and Shr, 2022) CNN-GRU	1D CNN + GRU	CNN	Daily life activity	UCI-HAR	Fourier Transform	Accuracy	96.79%
(Rad et al., 2018) **MA-DNN** **MA-CNN**	1D CNN	CNN	Locomotion activity, Daily activity, indoor/outdoor detection	STISEN, GAIT, Sleep-Stage, Indoor/Outdoor detection	Fast Fourier Transformation (FFT) + Empirical Cumulative Distribution Function (ECDF)	F1 score	81.6% 89.5% 66.4% 82.3% for respective datasets
(Rav et al., 2016) CNN	1D CNN	handcrafted	Locomotion activity, Factory maintenance,	ActiveMiles, WISDM, Skoda, FoG	No	Accuracy	95.1 % 98.2% 95.9 % 91.5 %
(Ron and Sun, 2016) Convnet	1D CNN	CNN	Locomotion activity	Private data by smartphone	No	Accuracy	94.8%
(Sey et al., 2018) CAE	1D CNN + AE	CNN	Locomotion activity	Private data collected by Radar	Synthetic Minority Oversampling Technique (SMOTE)	Accuracy	94.2%
(Sin et al., 2017) 1D CNN	1D CNN	CNN	Three Houses activity	Kasteren	No	Accuracy	95.3% 86.8% 86.23% for 3 houses
(Tan et al., 2020) Lego CNN	1D CNN	Lego filters	Locomotion activity	UCI-HAR, WISDM, PAMAP2,	Sliding Window technique	Accuracy, F1 score	96.9 % 98.82% 92.97 %

				Opportunity, UNIMIB-SHAR			88.09 % 72.80% Accuracy
(Ull et al., 2019) **Stacked LSTM**	LSTM	No	Locomotion activity	UCI-HAR	Single layer NN Normalization	Accuracy, Precision, Recall	93% Accuracy
(Wan and Liu, 2020) **H-LSTM**	LSTM	Time–frequency-domain method	Locomotion activity	UCI-HAR	Smoothing + Denoising	Accuracy, F1-score, Precision, Recall	99.15%
(Wan et al., 2016) **SDAE**	LSTM+ AE	SAE	Daily life activity	ADL D1 ADL D2 ADL D3	Not mentioned	Accuracy	85.32%
(Wan et al., 2019) **C1D**	1D CNN	CNN	Gesture	ARIL	Manual Duration Splitting + Up Sampling	F1 score, Precision, Recall	88 % F1
(Wen and Yin, 2015) **DCNN**	2D CNN	CNN	Daily life activity	UCI HAR, USC-HAD, SHO	No	Accuracy	95.18% 97.01% 99.93%
(Zen et al., 2014) **CNN-Partial Weight Sharing**	1D CNN	CNN	Factory maintenance, Kitchen, Locomotion activity	Skoda, Opportunity, WISDM	Sliding Window technique	Accuracy	88.19 % 76.83 % 96.88%
(Zha et al., 2018) **Res-Bidir-LSTM**	Stacked Residual LSTM	Residual block + LSTM	Daily life, Kitchen activity	UCI HAR, Opportunity	Missing+ Normalization + Reshaping	,Accuracy F1 score	93.6% Accuracy
(Zou et al., 2018) **DeepSense AE-LRCN**	AE + 1D CNN + LSTM	CNN	Locomotion activity	Private CSI data	No	Accuracy, Confusion matrix	97.4 % Accuracy

Discussion

From the works studied, we can realize that although researchers and scientists have made great strides in HAR system, room for improvement and progress remains. In Fact, from the comparative study, we note that the improvements for HAR are taken from different angles.

First, the choice of deep learning methods, CNN, LSTM, and hybridization basically, have achieved better performing results. For example, the different types of hybrid works based on CNN and LSTM approaches (Abd et al., 2020), (Naf et al., 2021) and (Pus and Shr, 2022) note a high accuracy of 97.70%, 97.05% and 96.79%, respectively, on the UCI-HAR dataset. It is worth mentioning that CNN is powerful in extracting the main features. While LSTM is effective in catching long-term time dependencies. Which put hybrid models at the top of researchers' interests.

Second, the choice of datasets, such as WISDM, UCI, Opportunity, Fog, and Skoda, has a great impact on the nature of activities done by specific users, in a specific environment. For instance, the user's activities in a smart home are not the same as in a smart factory environment. Taking the same work, we notice that the accuracy changes from dataset to another. For example, the work of (Mur and Jae, 2017) has an accuracy of 96.7% for UCI-HAR, 97.8% for USC-HAD, 92% for Opportunity, 93% for Fog, and finally 92.6% for Skoda dataset.

Therefore, the questions of HAR what, who, and where, will determine the how, and then the appropriate method of deep learning that will predict the true activity, and suit better the existing computation resources.

3.3.5 Open Issues of HAR Methods in Smart homes Environments

A growing number of new technologies and escalating needs, such as an aging population, have made HAR much more significant. Recent years have seen impressive recognition results from DL-based HAR methods. HAR has experienced rapid development, but there are still some difficulties (Kum and Cha, 2021). According to the literature, the recognition of human activity in smart environments, such as smart homes, have raised open issues, that are presented in some challenges. In this vein, we underline some challenges facing the resolution of HAR problems using deep learning. The barriers to activity recognition-based DL include the following:

1. **Data Acquisition Challenges**
    - **Data requirement:** In HAR, gathering data needs a noteworthy amount of work. There is always a need for more user data (Zha et al., 2022).
    - **Missing Data and Data Quality:** Numerous real-world data collection situations introduce various noise sources that reduce the quality of the collected data, such as devices' interferences (Zha et al., 2022).
    - **Privacy Protection:** The privacy concern has grown among users. In general, people are less willing to consent to a sensor's data collection the more inferential possibility it seems to have. (Zha et al., 2022).
2. **Label Acquisition Challenges**
    - **Time series label acquisition:** Deep supervised learning requires labeled data to function properly. In general, labeling image and audio data using visual or audio confirmation is simple (Zha et al., 2022). It is challenging to identify human activities from time series of HAR sensors.
    - **Labeled Data Shortage:** Annotating many data involves a long time and high cost. Thus, the lack of annotated data makes it difficult to comprehend sensor activity (Zha et al., 2022).
3. **Modeling Challenges**
    - **Data Segmentation:** Many techniques use conventional static sliding window techniques to segment time series. A static time window might be too short and not capture enough series to detect long movements, or it might be too large and capture more than is important to detect certain activities (Zha et al., 2022).
    - **Model Generalization:** When a model works well on data, which have never seen before, it has a high generalizability. When a model works well on training data but inadequately on fresh data, it is overfit. DL-based HAR typically surpasses and generalizes better than other types of methods but when data or model complexity is constrained, DL-based methods must make better use of the available data by using particular solutions (Zha et al., 2022).
    - **Model Robustness:** By combining the advantages of several kinds of sensors to create multi-sensory systems, robustness can frequently be increased (Zha et al., 2022).
4. **Activity Recognition Challenges**
    - **Complex Activity Recognition:** For easy tasks like running, current HAR methods deliver high performance. However, difficult tasks like washing hands, which can involve numerous movements, still exists (Kum and Cha, 2021).

- **Concurrent Activity Recognition:** When a person is engaged in multiple tasks at once, such as listening to a phone call and watching TV, concurrent simultaneous events happen (Kum and Cha, 2021).
- **Multiple Residents' Activities:** Activities including multiple occupants or residents are repeatedly related to the data interaction's scope. It becomes challenging to identify when several people carry out a series of actions, which frequently happens in multi-resident settings (Kum and Cha, 2021).

The following figure 3.7, present the main challenges and the sub challenges of HAR systems in smart homes.

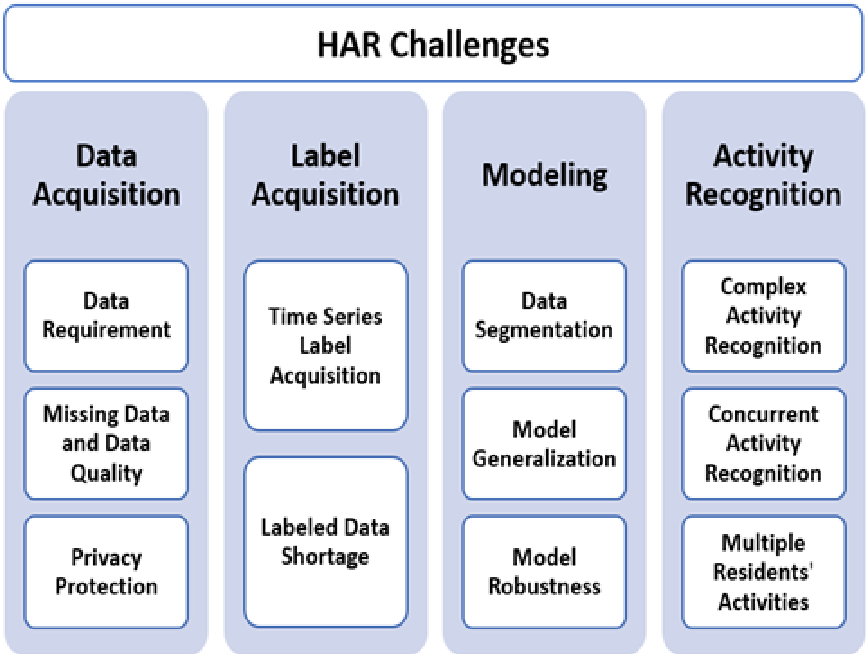

Figure 3.7: Challenges of HAR in Smart Homes

3.4 Conclusion

Recently, HAR has become a crucial technology since it can be applied to many human-centric, and real-life problems. Activity recognition aims to recognize common human activities in real-life settings. Machine learning and specifically deep learning shows the enormous ability for prediction in different fields, including HAR.

In this chapter, we have pointed out the key elements for an efficient algorithm of HAR in smart environments such as smart homes. We have also pointed out the most efficient methods. Subsequently, we presented a comprehensive synthesis of related and recent work, which were compared in a table according to some criteria. The synthesis allowed us to choose our models and techniques for a more efficient activity prediction and recognition.

In addition, we highlight some challenges that hinder the activity recognition process in the DL context. Nevertheless, the smart home deployment depends not only on the HAR software but also on the HAR hardware development, besides their acceptability and usability by the users. The design and experiments of the model will be given in the following chapters.

Chapter4

SYSTEM DESIGN

4.1 Introduction

In this study, we describe a model of HAR in a smart home using sensors integrated into the smartphone. The model is a hybridization between CNN and LSTM. This chapter is reserved for the detailed presentation of the design of the model. We start by illustrating the general architecture of the system. Thereafter, we move on to a detailed explanation of the different parts of the system. The details of the two used models, CNN and LSTM, have already been explained in the previous chapters.

4.2 General System Architecture

The objective of our work is to design an intelligent system capable of predicting and recognizing the daily human activities in smart environments, including smart cities, buildings, and homes.

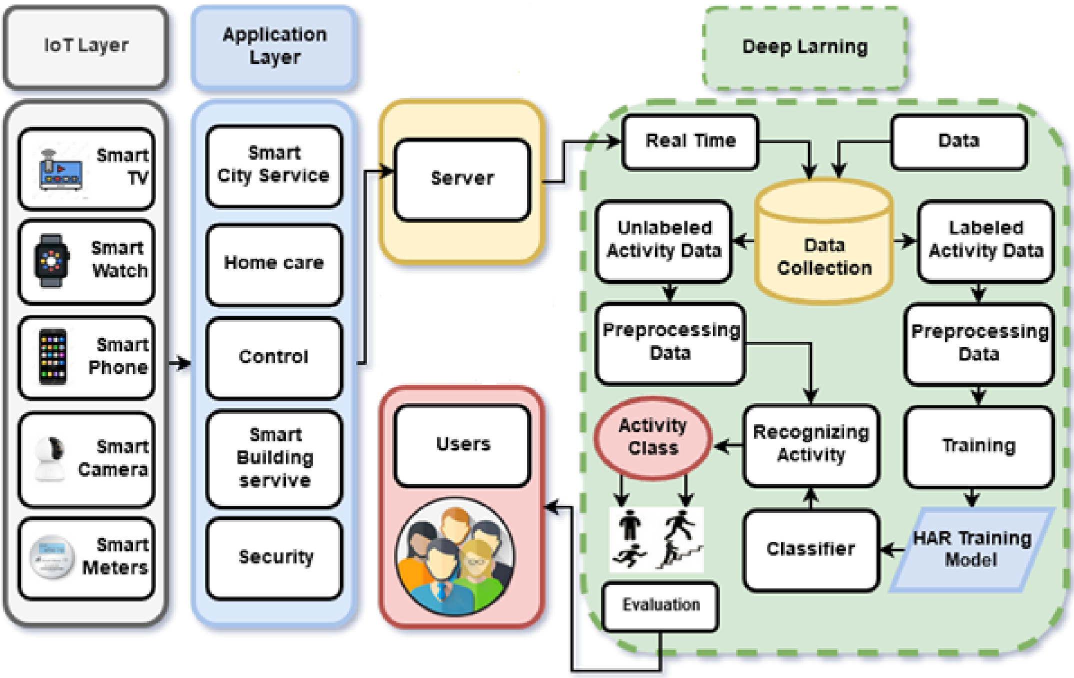

Figure 4.1: Design Architecture of Deep learning for HAR in Smart Environments

Figure 4.1 illustrates the architecture of HAR in a smart home environment based on deep learning. The collection of data comes from different IoT sensors and smart devices. The dataset uses deep learning to process the unlabeled activity data and classify them according to a HAR training model. We provide in this section a general view of the HAR framework parts, followed by a detailed elucidation of the different constituents. As shown in Figure 4.2, the designed framework comprises four stages: data collection, preprocessing, HAR-DL Training Model, and classification.

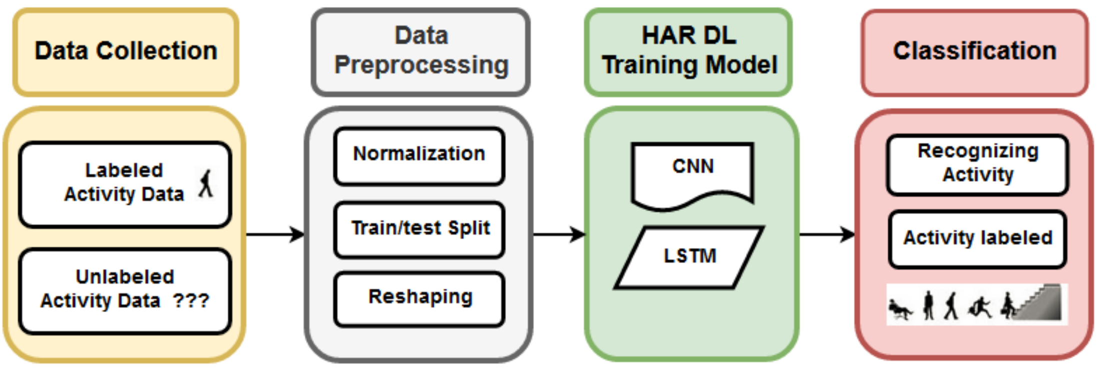

Figure 4.2: System Stages Framework

4.3 Data Collection: WISDM Dataset Description

To assess the performance of our system, we perform experiments with six different classes of daily life activities: sitting, standing, walking, upstairs, downstairs, and jogging. For this study, we chose activity-relevant smart environmental HAR from popular activity datasets, such as the ACTi Tracker dataset, commonly known as the "WISDM dataset" [14]. The WISDM dataset is composed of data collected from the accelerometer sensor of a smartphone placed in different body parts of the volunteer in a controlled laboratory environment as presented in the next figure. The android phone's triaxial accelerometer is used to calculate acceleration. the displacements of (x , y , z) axes are measured by this accelerometer. These axes represent the user's sideways and horizontal orientation (x-axis), downward and upward movement (y-axis), backward and forward movement (z-axis) [14]. The activity was labeled on acceleration data obtained during the periods of start and stop of an activity.

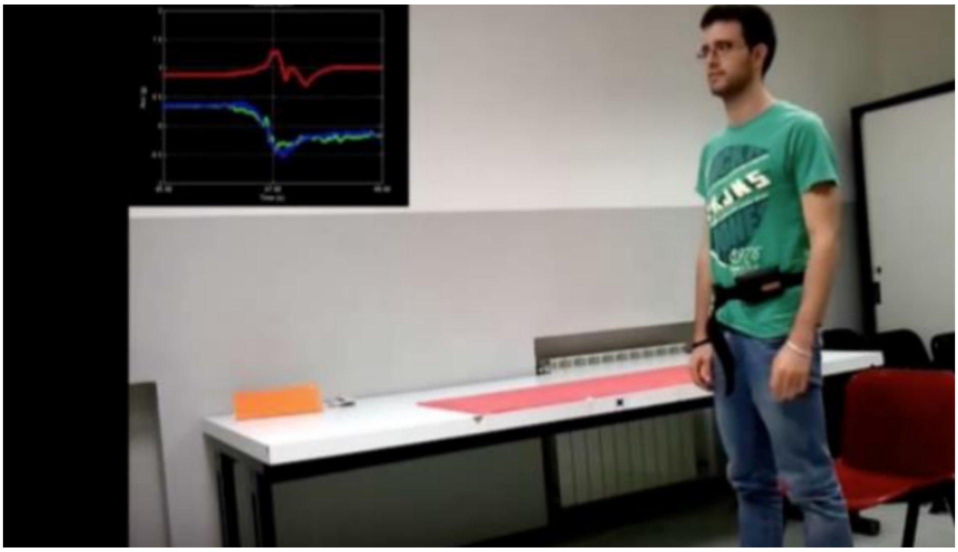

Figure 4.3: HAR using Smartphone Sensors [14]

This dataset [14] was developed in 2013 by the WISDM Lab of Fordham University (Wireless Sensor for Data Mining). The different activities performed a sampling rate of 20Hz (the duration of each data segment is about 5s).

The next table 4.1 presents the class distribution of the dataset that we use for experimentation.

Class Distribution	Percentage
Walking	38.6%
Jogging	31.2%
Upstairs	11.2%
Downstairs	9.1%
Sitting	5.5%
Standing	4.4%

Table 4.1 Class Distribution for WISDM Dataset

The data visualization by activity type, the dataset sample, and the data types are presented as follows:

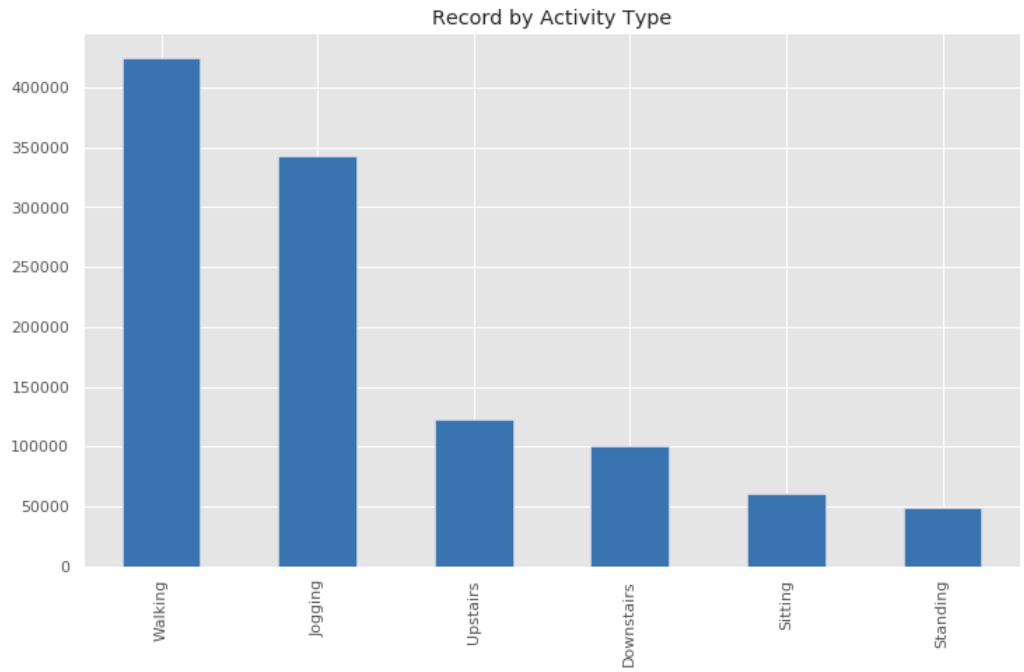

Figure 4.4: Data Visualisation Record by Activity Type

So, as we can see that 70% of activities are walking and jogging.

Data Shape: (1098203, 6)

	user-id	activity	timestamp	x-axis	y-axis	z-axis
0	33	Jogging	49105962326000	-0.7	12.7	0.5
1	33	Jogging	49106062271000	5.0	11.3	1.0
2	33	Jogging	49106112167000	4.9	10.9	-0.1
3	33	Jogging	49106222305000	-0.6	18.5	3.0
4	33	Jogging	49106332290000	-1.2	12.1	7.2
5	33	Jogging	49106442306000	1.4	-2.5	-6.5
6	33	Jogging	49106542312000	-0.6	10.6	5.7
7	33	Jogging	49106652389000	-0.5	13.9	7.1
8	33	Jogging	49106762313000	-8.4	11.4	5.1
9	33	Jogging	49106872299000	1.0	1.4	1.6

Figure 4.5: Database Sample

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1098204 entries, 0 to 1098203
Data columns (total 6 columns):
 #   Column     Non-Null Count    Dtype
---  ------     --------------    -----
 0   user       1098204 non-null  int64
 1   activity   1098204 non-null  object
 2   timestamp  1098204 non-null  int64
 3   x-axis     1098204 non-null  float64
 4   y-axis     1098204 non-null  float64
 5   z-axis     1098203 non-null  float64
dtypes: float64(3), int64(2), object(1)
memory usage: 50.3+ MB
```

Figure 4.6: Data Types

Furthermore, the number of activities varies by the user as it is portrayed in the subsequent figure,

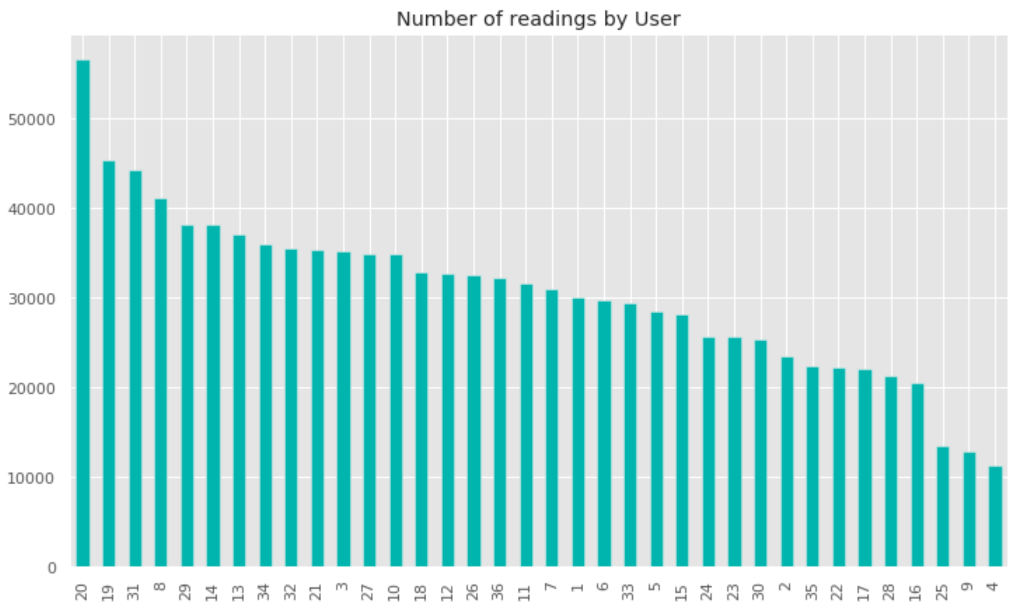

Figure 4.7: Number of Reading by Person

After analyzing the dataset, we schedule the accelerometer readings for a timestamp of 10 seconds. Because each activity has a distinct pattern, we can visually examine how the accelerometer data appears for each activity.

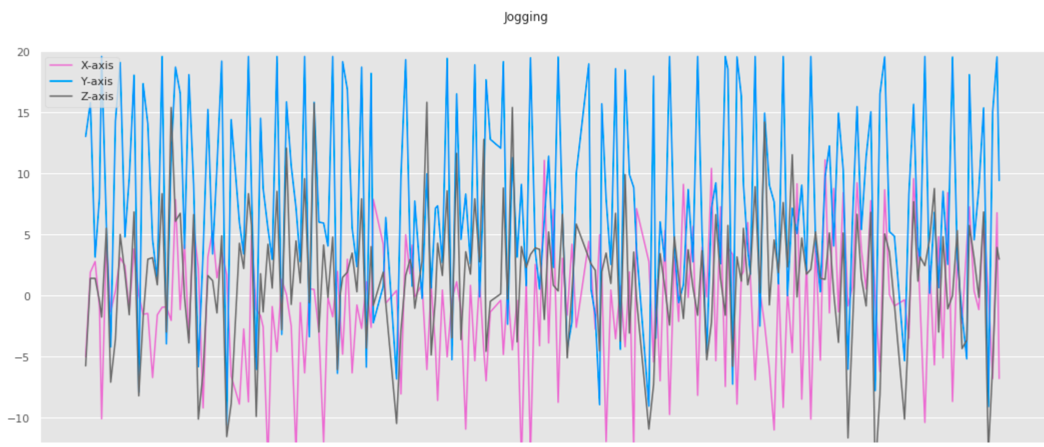

Figure 4.8: Visualization of Jogging Activity

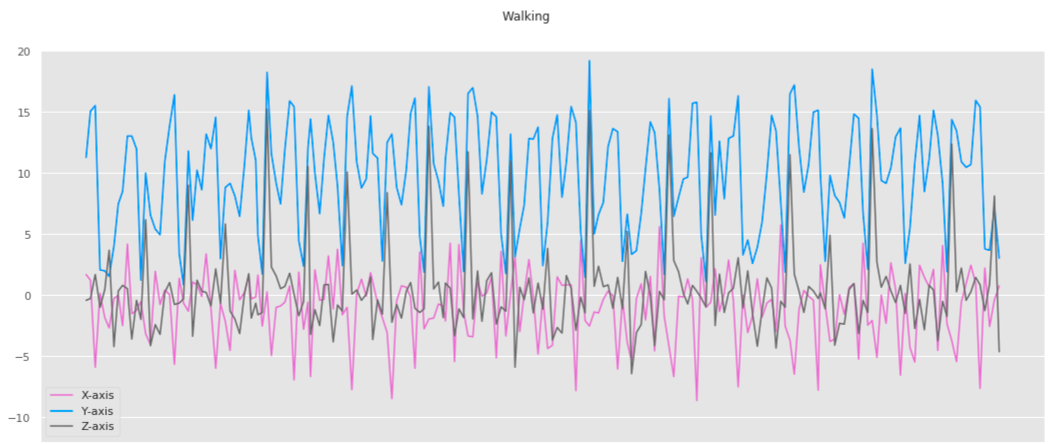

Figure 4.9: Visualization of Walking Activity

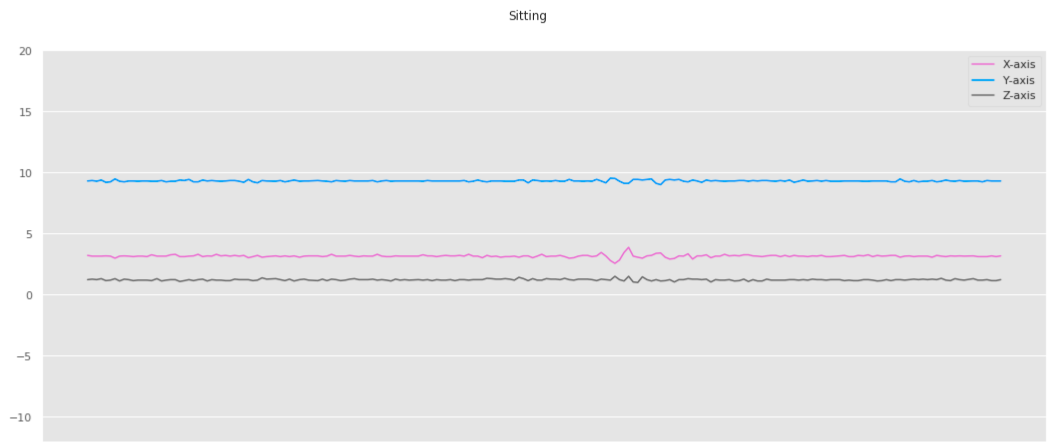

Figure 4.10: Visualization of Sitting Activity

From the above figures, we can see that for jogging and walking activities (Figures 4.8 and 4.9) there is a lot of variation in pattern, while for the sitting activity (Figures 4.10) the pattern is almost flat.

4.4 Data Preprocessing

In smart environments, the collected data are composed of labeled data and unlabeled data. Our objective is hence to train a HAR model with reference to the collected data in order to predict and recognize the activity label that is not provided. To achieve this goal, we start by the normalization of the acquired data and then split the normalized data into two groups, train and test data.

Finally, we will reshape the data to adapt the neural network. The data will not be scaled to not affect the underlying distributions of the different human activities. Hence, we performed data preprocessing using the above-mentioned methods:

- Normalize the data,
- Split the data (training/test),
- Reshaping the data,

Then, this data is transferred to our HAR DL model for training

4.5 HAR-DL Training Model

To overcome the challenge of characterizing the underlying patterns of big amount of data, we use deep learning method that has been shown to be effective in deducing discriminative representations from such data. To predict the label of unlabeled human activities, we adopt a deep learning model that combines a CNN "convolutional neural network" and an LSTM "long short-term memory neural network".

After preprocessing the database, the data is transferred to the HAR DL model for training. The model is a hybridization between a CNN and an LSTM (See figure 4.2). The convolution layer performs the function of a feature extractor, providing an abstract representation of the original data in the shape of a feature vector.

The LSTM layer creates the feature vector's time dynamics. As shown in Figure 4.11, this model contains two convolutional layers, an LSTM layer, and two dense layers. CNN is used to extract features from input data. CNN is composed of two hidden layers that use convolutional filters to derive abstract representations of input data. Followed by pooling processes, CNN considers the characteristics and the features of the data from various perspectives.

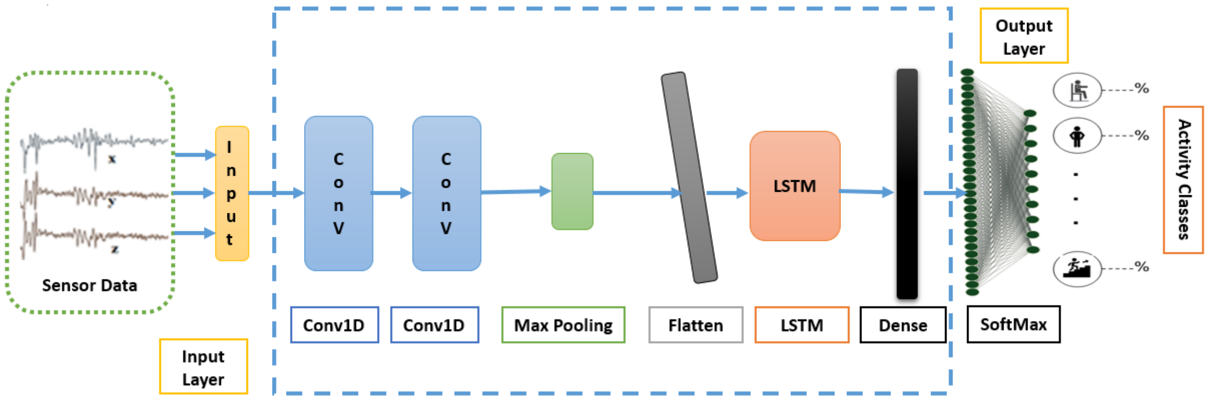

Figure 4.11: Number of Reading by Person

The network's input is sequential data. The convolution layer conducts convolution operations on this data to flatten it and represent it as a feature vector, which is then sent to feed the LSTM layer (sequence-learning layer). LSTM has gates, serving as internal systems that control the flow of information. The LSTM layer builds the feature vector's time correlation and extracts the features containing time information. Following that, the LSTM layer delivers its output to the dense layer, which in turn sends its output to the second dense layer with the softmax activation function, which is used as the output layer.

4.6 Classification

The classification phase includes activity recognition, which leads to the labeling of activities into activity classes. The activity recognition classification is done using the test data after the execution of the HAR-DL training phase and saving the model (See Figure 4.2). The test data undergoes the same preprocessing as the training data. Therefore, we start by normalizing and then shaping the data. After preprocessing test data, the model is loaded and activity recognition can take place, and the results are saved.

4.7 Conclusion

The system designed in this study consists of four phases. First, the phase dedicated to data collection is presented in the dataset understanding and visualization. Then comes the preprocessing phase, which structures the preparation of the initial data, followed by the HAR deep learning training phase, which allows the learning of the activity recognition model in order to enhance the results, and finally, the classification phase that tests the model with real data to detect and label human activities.

The described recognition model for HAR in the environment of a smart home is a composition of CNN and LSTM deep learning techniques. We have gone in detail in this chapter through the different constituents of the designed system. The implementation and evaluation of the efficiency of the hybrid model are the objectives of the next chapter.

Chapter5

IMPLEMENTATION AND EVALUATION

5.1 Introduction

This chapter includes two parts: the first is devoted to the implementation of the designed model, where its different pillars are presented, such as the development environment and the libraries used. The second displays the results obtained through discussions and comparisons.

5.2 Implementation

This part is devoted to the details of the development environment and the programming language used for the realization of our system. We also presented the training and test, database used, and the details of the designed architecture.

5.2.1 Development Environment

Google Colab: Colaboratory or "Colab." Allows writing and running Python code from the browser, delivered by Google (free), founded on Jupyter Notebook, and planned for machine learning training and research. This platform enables machine-learning models to be trained directly in the cloud. Colab allows:

- Improving coding skills in the Python language.
- Developing applications of DL using widespread Python libraries, for instance, PyTorch, Keras, OpenCV, and TensorFlow.
- Using a development environment (Jupyter Notebook), which does not require any configuration.

- Accessing a GPU graphics processor that is free. This feature distinguishes Colab from other services.

Anaconda: Anaconda is a scientific distribution of Python, which allows writing and running Python code through the browser. It is offered by Anaconda Enterprise, uses Jupyter Notebook, and was intended for machine learning research [15].

5.2.2 Programming Language and Libraries

This section will introduce python language and the libraries used for the implementation of the human activity recognition model.

Python:

Python has recently become the most commonly used programming language by computer scientists. This language has risen to prominence in infrastructure management, data analysis, and software development. In particular, Python, in particular, enables developers to concentrate on what they do rather than how they do it. It liberated developers from the form constraints that hampered them in previous languages. As a result, developing code in Python is faster than in other languages (Jai, 2015).

Libraries used

TensorFlow: We used this library to define the basic components of the CNN-LSTM architecture. This library is intended for the implementation of automatic and DL algorithms. It also offers great flexibility in the setting of use for the development of NN [16].

Keras: it is used with TensorFlow. We used this library to implement the different layers, the activation functions, and the preparation of the training base [16].

NumPy: This library was used to adjust the input types based on the configuration of the models employed, which were designed to handle multidimensional arrays or matrices in addition to mathematical functions that operate on these arrays. We used this package specifically for window extraction and image scanning [17].

Sklearn: is one of Python's most beneficial ML packages. Many robust techniques for machine learning and statistical modeling, such as dimensionality reduction, regression, classification, and clustering, are available in the sklearn library [19].

Pandas: is a data analysis open-source and processing tool developed in the language of Python [20]. It is flexible, powerful, fast, and simple to use.

Before proceeding to the building of the designed model, it is indispensable to pass by importing the dataset and making the necessary preprocessing to our data. We present in this section the model building with the details of realization.

As we have already explained in chapter 4, the preprocessing of the database goes through several stages:

- Preparation of the Anaconda execution environment.
- Importing and preparing the WISDM database.
- Importing the necessary libraries Numpy, Matplotlib, Pandas, Sklearn.

```
from matplotlib import pyplot as plt
import numpy as np
import pandas as pd

from sklearn import metrics
from sklearn.metrics import classification_report
from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from sklearn import linear_model, datasets,metrics
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import train_test_split
```

Figure 5.1: Used libraries

The data preprocessing will be include:

- Normalization of the measurements to be between 0 and 1.
- Transformation of the measurements to be in a 3-D array of [samples, timesteps, features].

- Breaking the 3-D array into a training, validation, and test dataset.
- Reshaping the data to adapt the neural network architecture.

5.2.3 Building the Hybrid Model

The architecture of CNN-LSTM encompasses using two combinations: first, CNN layers to extract relevant features from row data, and second, LSTM to support sequence learning for the purpose of detecting and recognizing human activities. As a result, CNN will examine the input data as blocks to extract features. LSTM will then interpret the time-dependent features retrieved from each block.

CNN-LSTM is a model that interprets the output of CNN models on a time-relationship basis. We took the next steps toward building the CNN-LSTM hybrid model.

```
def make_cnn_lstm_model(lstm_neurons,dense_neurons,drop_out):
  model = Sequential()
  model.add(TimeDistributed(Conv1D(filters=64, kernel_size=3, activation='relu'),
                            input_shape=(None,n_length,n_features)))
  model.add(TimeDistributed(Conv1D(filters=64, kernel_size=3, activation='relu')))
  model.add(TimeDistributed(Dropout(drop_out)))
  model.add(TimeDistributed(MaxPooling1D(pool_size=2)))
  model.add(TimeDistributed(Flatten()))
  model.add(LSTM(lstm_neurons))
  model.add(Dropout(drop_out))
  model.add(Dense(dense_neurons, activation='relu'))
  model.add(Dense(num_classes, activation='softmax'))
  model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
  return model
```

```
n_steps, n_length = 20, 10
X_train_cnn = X_train_res.reshape((X_train_res.shape[0], n_steps, n_length, n_features))
X_test_cnn = X_test_res.reshape((X_test_res.shape[0], n_steps, n_length, n_features))
```

```
model_cnn_lstm = make_cnn_lstm_model(lstm_neurons= 100, dense_neurons = 100, drop_out = 0.5)
print(model_cnn_lstm.summary())
```

Figure 5.2: CNN-LSTM hybrid model

The summary of the CNN-LSTM hybrid model:

```
Model: "sequential_7"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
time_distributed_6 (TimeDist (None, None, 8, 64)       640
_________________________________________________________________
time_distributed_7 (TimeDist (None, None, 6, 64)       12352
_________________________________________________________________
time_distributed_8 (TimeDist (None, None, 6, 64)       0
_________________________________________________________________
time_distributed_9 (TimeDist (None, None, 3, 64)       0
_________________________________________________________________
time_distributed_10 (TimeDis (None, None, 192)         0
_________________________________________________________________
lstm_7 (LSTM)                (None, 100)               117200
_________________________________________________________________
dropout_6 (Dropout)          (None, 100)               0
_________________________________________________________________
dense_16 (Dense)             (None, 100)               10100
_________________________________________________________________
dense_17 (Dense)             (None, 6)                 606
=================================================================
Total params: 140,898
Trainable params: 140,898
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 5.3: Summary of the CNN-LSTM hybrid model

Training the model:

```
batch_size = 1024
epochs = 50
cnn_lstm = model_cnn_lstm.fit(X_train_cnn, y_train_val, batch_size=batch_size,
                      epochs=epochs, callbacks=callbacks_list, validation_split=0.2,verbose=1)
```

Figure 5.4: Training the model

Testing the model:

```
score = model_cnn_lstm.evaluate(X_test_cnn, y_test_res, verbose=1, batch_size=batch_size)

print('\nAccuracy on test data: %0.2f' % score[1])
print('\nLoss on test data: %0.2f' % score[0])
```

Figure 5.5: Testing the model

5.3 Evaluation and comparisons

We will present in this part: the experimental configuration, the performance results and comparisons.

5.3.1 Experimental setup

We evaluated and validated the efficiency of the described CNN-LSTM hybrid model using the WISDM database. We trained our model on Google Colab with a Google Python 3 computer, with 16GB of RAM and 110GB of storage on the Google Chrome browser. This model was implemented in Python (V3.8) in the TensorFlow (V2.3) backend and used Adam as the optimizer.

As a metric of recognition quality, we use both the training accuracy and the testing accuracy. Accuracy is calculated as the percentage of correct classifications across all classifications as follows:

Accuracy = the correct classification number/the entire classification number.

5.3.2 Results

After executing the HAR deep learning training model, we used the test data to recognize and classify activity data. Then, we compare the results with the actual data using the performance metrics, which are the accuracy and the loss. Figure 5.8 depicts our model's accuracy and loss throughout training and validation.

```
Epoch 1/50
32940/32940 [==============================] - 4s 124us/step - loss: 0.9966 - acc: 0.6423 - val_loss: 0.5844 - val_acc: 0.7960
Epoch 2/50
32940/32940 [==============================] - 2s 59us/step - loss: 0.5150 - acc: 0.8134 - val_loss: 0.4225 - val_acc: 0.8380
Epoch 3/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.4182 - acc: 0.8426 - val_loss: 0.3360 - val_acc: 0.8697
Epoch 4/50
32940/32940 [==============================] - 2s 57us/step - loss: 0.3553 - acc: 0.8639 - val_loss: 0.3155 - val_acc: 0.8801
Epoch 5/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.3058 - acc: 0.8849 - val_loss: 0.2419 - val_acc: 0.9107
Epoch 6/50
32940/32940 [==============================] - 2s 57us/step - loss: 0.2576 - acc: 0.9079 - val_loss: 0.2965 - val_acc: 0.8959
Epoch 7/50
32940/32940 [==============================] - 2s 56us/step - loss: 0.2330 - acc: 0.9179 - val_loss: 0.1766 - val_acc: 0.9360
Epoch 8/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.1938 - acc: 0.9322 - val_loss: 0.1669 - val_acc: 0.9422
Epoch 9/50
32940/32940 [==============================] - 2s 57us/step - loss: 0.1883 - acc: 0.9363 - val_loss: 0.1582 - val_acc: 0.9457
Epoch 10/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.1593 - acc: 0.9460 - val_loss: 0.1335 - val_acc: 0.9515
Epoch 11/50
32940/32940 [==============================] - 2s 57us/step - loss: 0.1489 - acc: 0.9496 - val_loss: 0.1208 - val_acc: 0.9585
Epoch 12/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.1440 - acc: 0.9500 - val_loss: 0.1281 - val_acc: 0.9506
Epoch 13/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.1219 - acc: 0.9592 - val_loss: 0.1035 - val_acc: 0.9619
Epoch 14/50
32940/32940 [==============================] - 2s 58us/step - loss: 0.1157 - acc: 0.9597 - val_loss: 0.1177 - val_acc: 0.9530
Epoch 15/50
32940/32940 [==============================] - 2s 56us/step - loss: 0.1053 - acc: 0.9635 - val_loss: 0.1313 - val_acc: 0.9513
Epoch 16/50
32940/32940 [==============================] - 2s 55us/step - loss: 0.0984 - acc: 0.9674 - val_loss: 0.0950 - val_acc: 0.9671
Epoch 17/50
32940/32940 [==============================] - 2s 59us/step - loss: 0.1022 - acc: 0.9648 - val_loss: 0.0848 - val_acc: 0.9681
```

Figure 5.6: Execution screenshot

Several experiments on training were conducted to find the most suitable values for the training parameters. Then, we finally decided to train the model on 50 epochs with 1024 batch sizes. Therefore, the accuracy of test data of CNN-LSTM achieves 97%, whereas the loss is 0.09.

```
13726/13726 [==============================] - 0s 12us/step

Accuracy on test data: 0.97

Loss on test data: 0.09
```

Figure 5.7: Results of accuracy and loss

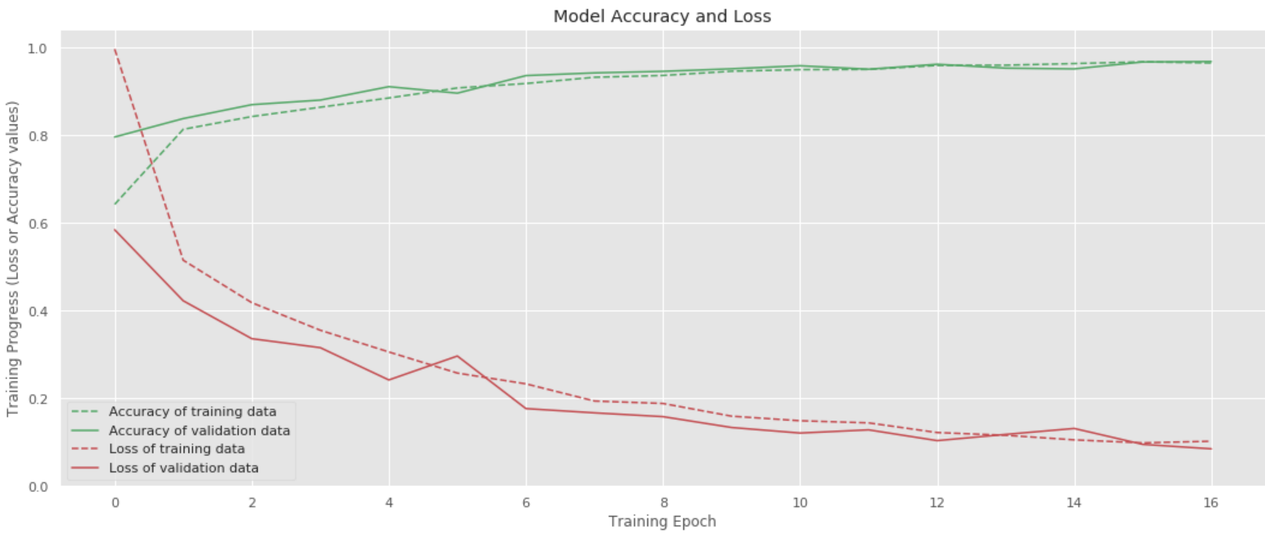

Figure 5.8: Visual Accuracy and Loss Results

Figure 5.8 shows the training loss rate as a dashed line in red, the validation loss rate as a solid line in red, the accuracy of training data as a dashed line in green, and the validation accuracy rate as a solid line in green. The training loss rate is upper than the validation loss rate. Validation accuracy is nearly identical to training accuracy. The activity recognition accuracy rate of training and validation progressively converges to one as the iteration number increases. Furthermore, the loss rate gradually approaches zero. Hence, activity recognition accuracy is gradually improving.

The confusion matrix for predicting different activities of the CNN-LSTM hybrid model on the WISDM dataset is denoted in the table below. The diagonal numbers in the matrix indicate the correctly classified activities, whereas the other numbers indicate the misclassified activities.

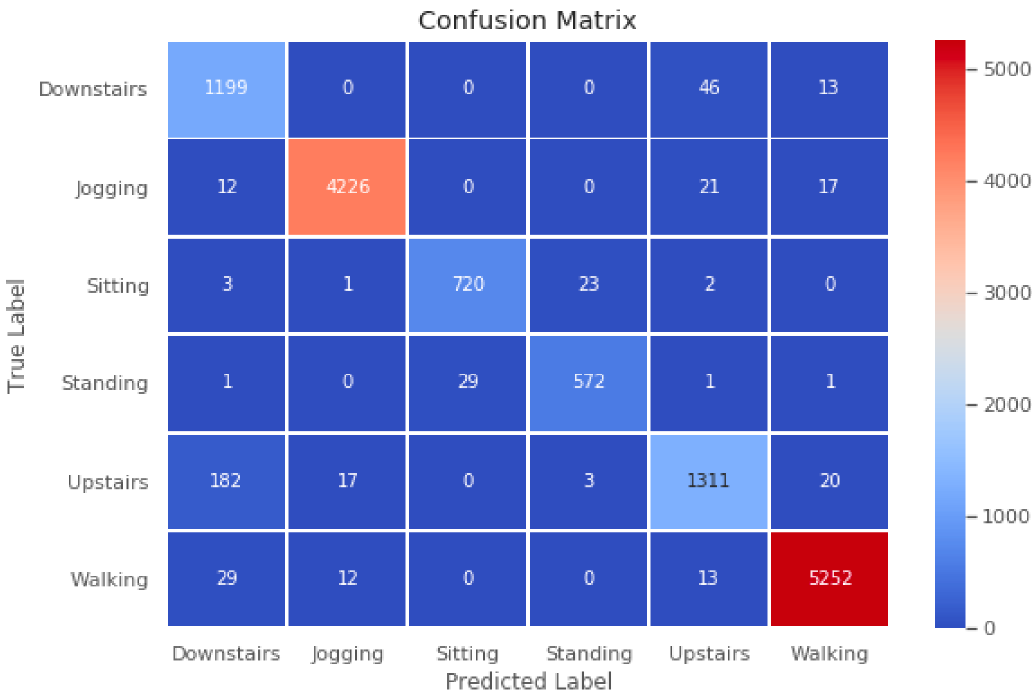

Table 5.1: Confusion Matrix of Hybrid CNN-LSTM Model

As seen in the confusion matrix, several pair activity classes, such as (jogging, standing) and (walking, sitting), have no misidentified instances since they contain all zeros. However, (sitting, standing), (upstairs, jogging), and (downstairs, walking) have high confusion rates, especially for (upstairs, downstairs), which records the highest rates of misclassified instances. Because these activities have comparable properties, the accuracy rate is low. It is vital to highlight that among these activities, walking and jogging are the two most common activities in our dataset. They have the highest recognition rate of accuracy ,which is closely linked to their specific actions. Even though sitting and standing are minority classes, the hybrid model is accurately able to distinguish them. The accuracy of activities performed upstairs and downstairs is not as high as for walking and jogging activities. This is expected as these two activities are very similar and the initial data may not be sufficient to precisely discriminate between them.

With regard to this data, the hybrid model structure is proved.

The model architecture for CNN-LSTM is as follows:

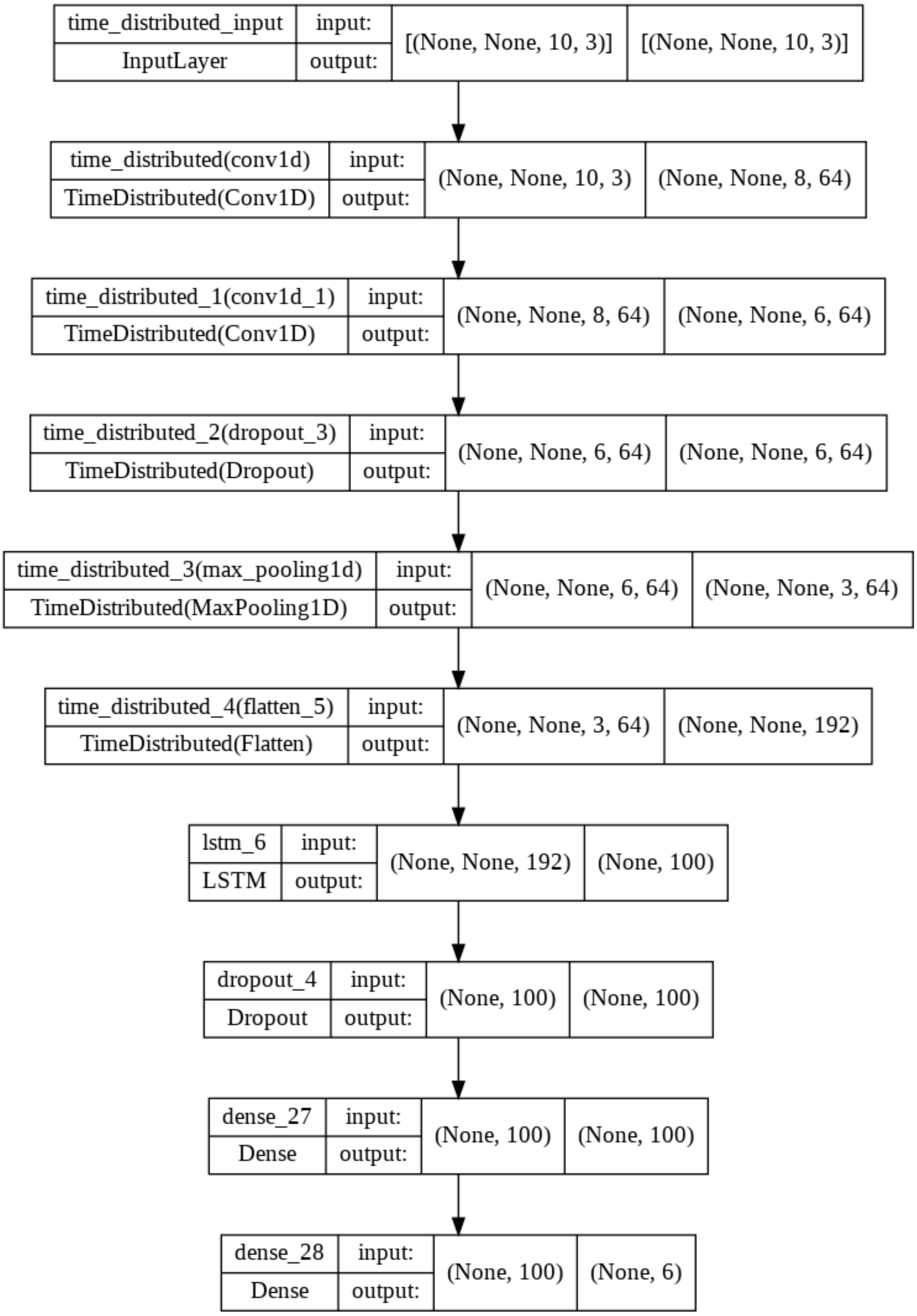

Figure 5.9: Model Architecture for CNN-LSTM

5.3.3 Comparison between the Hybrid Model and Other Models

Comparison with Implemented Models

We implemented, besides the hybrid model, the models LSTM and CNN, and then we compared the three models. The results acquired with the WISDM dataset are displayed in Table 5.2 and figure 5.10.

Models	Accuracy	Loss
LSTM	0.89	0.32
CNN	0.94	0.16
CNN-LSTM	**0.97**	**0.09**

Table 5.2: Model Comparison with Implemented Models

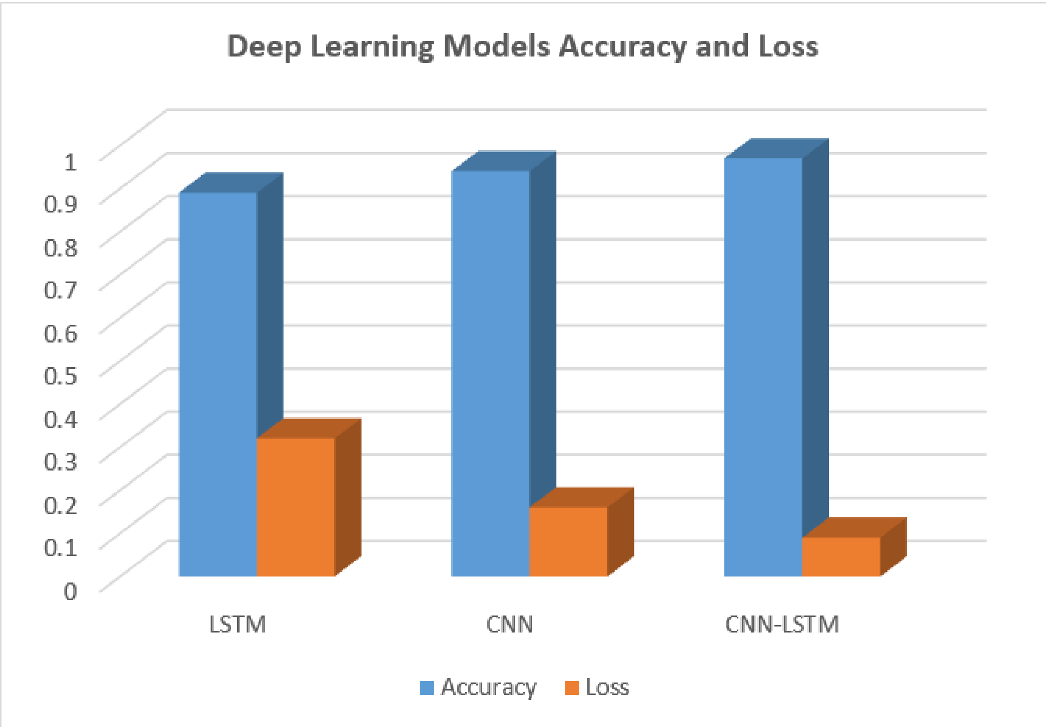

Figure 5.10: Accuracy and Loss for DL Models

A first comparison between the LSTM, CNN, and CNN-LSTM models shows that the CNN-LSTM gives the best accuracy and loss results compared to the above models.

Comparison with Other Reference Models

Similarly, we compared the hybrid model with other reference models from the literature (Li and Wan, 2022), (Che et al., 2016), (Ign, 2018), (Zen et al., 2014), and (Naf et al., 2021), with the same WISDM dataset.

The accuracy results obtained by the hybrid models CNN-LSTM and CNN-BiLSTM are the highest among these models, as indicated in Table 5.3 and figure 5.11. Currently, the hybrid model CNN-LSTM recorded an accuracy of 97%, whereas the CNN as well as LSTM models achieved a low accuracy of 93.32% and 92.1%, respectively.

Models	**Accuracy**
(Li and Wan, 2022) **BiLSTM**	**97.32%**
(Che et al., 2016) **LSTM**	92.1%
(Ign, 2018) **CNN**	93.32%
(Zen et al., 2014) **CNN- Sharing**	96.88%
(Naf et al., 2021) **CNN-BiLSTM**	**98.53%**
Hybrid **CNN-LSTM**	**97%**

Table 5.3: Accuracy and Loss for DL Models

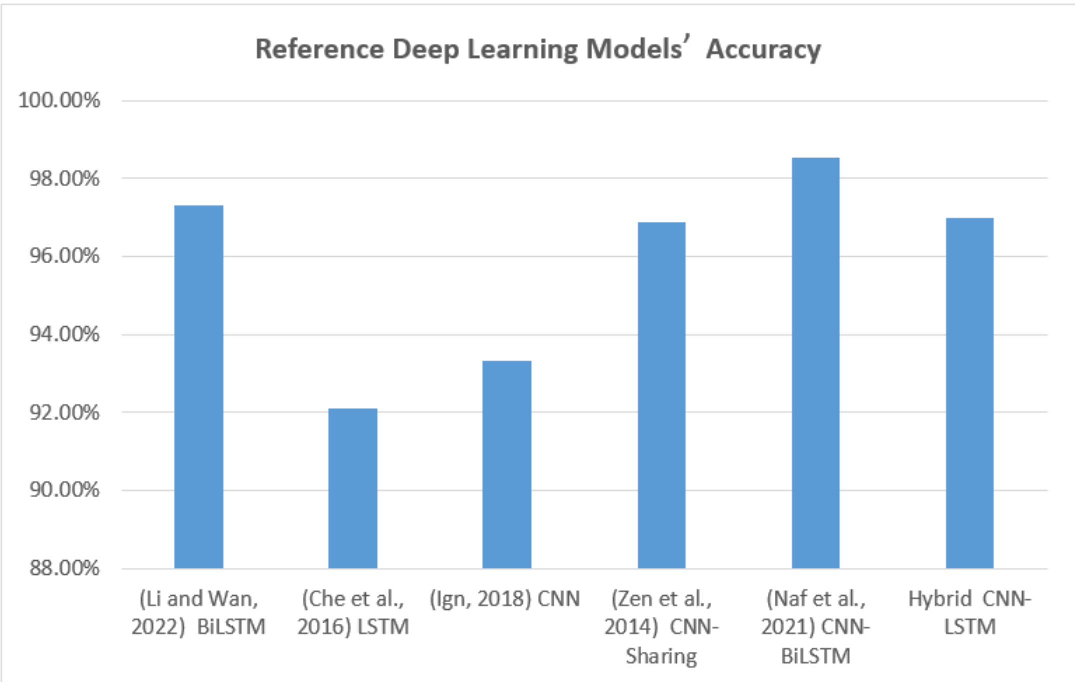

Figure 5.11: Accuracy and Loss for DL Models

We notice that when these models are coupled with internal additional processing, such as the CNN model with the sharing of partial weights, the accuracy improved from 93.32% in CNN to 96.88% in CNN-Sharing, and the same for the LSTM model with bidirectional internal processing, the accuracy increased from 92.1% in LSTM to 97.32% in BiLSTM. Therefore, when those improved models are combined as the improved hybrid model CNN-BiLSTM, they achieve a noteworthy accuracy result of 98.53%, higher than our simple hybrid model of 97%. Hence, the hybrid models improved the activity recognition and prediction results in a significant way.

5.4 Conclusion

This chapter has been divided into two parts: implementation, and evaluation. In the first part, we presented the working environment, the programming language, the training and testing bases, as well as the details of how the training phase for the hybrid model was built and implemented. The experimental configuration, the results, and a comparison with other reference works from the literature were all introduced in the second part. Based on the obtained results, the hybrid CNN-LSTM model gives the best results in detecting and recognizing human activities within the realm of smart environments.

GENERAL CONCLUSION

DL for human activity identification and recognition, or simply HAR, plays a crucial role in the daily lives of people due to its efficiency in learning in-depth knowledge about the activities performed by humans from the IoT sensor devices embedded in smart environments.

In this thesis, we discussed the importance of the human activity recognition issue in smart environments and we described a hybrid model to solve it. We studied the most recent studies on the HAR in smart environments and the methods they used to obtain or improve the results in this field. Since the synthesis and comparison of related work made in chapter 2, we have found several points that have facilitated the work on an efficient and effective recognition model for HAR. We have designed a hybrid activity recognition model in smart environments that combines convolutional neural networks with one of the utmost extensively used recurrent neural network methods, longshort-term memory. CNN is preferable for extracting features from input data, while LSTM is advocated for detecting and recognizing activities that have a natural order. The reason behind this is that CNN is more able to learn deep features contained in recursive patterns, while LSTM is used to model the features extracted by CNN and output the feature vector containing time-order relationships between sensor readings. As for perspectives on this work:

- It would be interesting to generalize our system into a human activity recognition system in many datasets of smart cities. The actual implementation of this work with the necessary infrastructure is for a smart home environment.
- Using the datasets generated from the new simulators for smart environments for human activities and behaviors, such as the dataset of the SBS Simulator (Deg et al., 2019), and comparing the existing datasets in order to study the challenging issue of concurrent and multi-resident activities

- The real-life data is noisier than our WISDM dataset, which was collected in the lab. Therefore, we will collect more data by extending the number of human activities and using the data augmentation technique to increase the sample and add noise to it.
- Apply the Autoencoder and the Transformer methods in smart environments to explore their potential in human activity recognition.
- Combining the context-awareness-based ontologies with the hybrid methods may lead to an efficient recognition of human activities.

BIBLIOGRAPHY

(Abd et al., 2020) Abdel-Basset, M., Hawash, H., Chakrabortty, R. K., Ryan, M., Elhoseny, M., Song, H. (2020). ST-DeepHAR: Deep learning model for human activity recognition in IoHT applications. IEEE Internet of Things Journal, 8(6), 4969-4979.

(Ala et al., 2020) Alawneh, L., Mohsen, B., Al-Zinati, M., Shatnawi, A., Al-Ayyoub, M. (2020). A comparison of unidirectional and bidirectional lstm networks for human activity recognition. In 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops) (pp. 1-6). IEEE.

(Ali, 2022) Ali Hamad, R. (2022). Towards Reliable, Stable and Fast Learning for Smart Home Activity Recognition (Doctoral dissertation, Halmstad University Press).

(Alo et al., 2019) Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. Electronics, 8(3), 292.

(Aru et al., 2017) Arulkumaran, K., Deisenroth, M. P., Brundage, M., Bharath, A. A. (2017). A brief survey of deep reinforcement learning. arXiv preprint arXiv:1708.05866.

(Bel and Que, 2017) Bellas, M., and Maxime Quemin 2017). Smart building : construction de bâtiment intelligent, apport de lélectronique imprimée. Rapport.

(Bel et al. 2015) Belley, C., Gaboury, S., Bouchard, B., Bouzouane, A. (2015). Nonintrusive system for assistance and guidance in smart homes based on electrical devices identification. Expert Systems with Applications, 42(19), 6552-6577.

(Bou et al., 2021)a Bouchabou, D., Nguyen, S. M., Lohr, C., LeDuc, B., Kanellos, I. (2021). A survey of human activity recognition in smart homes based on IoT sensors algorithms: Taxonomies, challenges, and opportunities with deep learning. Sensors, 21(18), 6037.

(Bou et al., 2021)b Bouchabou, D., Nguyen, S. M., Lohr, C., Leduc, B., Kanellos, I. (2021, January). Fully convolutional network bootstrapped by word encoding and embedding for activity recognition in smart homes. In International Workshop on Deep Learning for Human Activity Recognition (pp. 111-125). Springer, Singapore.

(Bou, 2014) Boudellal, M. (2014). Smart home: habitat connecté, installations domotiques et multimédia. Dunod.

(Che et al., 2016) Chen, Y., Zhong, K., Zhang, J., Sun, Q., Zhao, X. (2016, January). LSTM networks for mobile human activity recognition. In Proceedings of the 2016 International Conference on Artificial Intelligence: Technologies and Applications, Bangkok, Thailand (pp. 24-25).

(Che et al., 2019) Chen, K., Yao, L., Zhang, D., Guo, B., Yu, Z. (2019). Multi-agent attentional activity recognition. arXiv preprint arXiv:1905.08948.

(Das and Coo, 2005) Das, Sajal K., and Diane Cook (2005). Designing smart environments: A paradigm based on learning and prediction. Mobile, wireless, and sensor networks: Technology, applications, and future directions (pp. 337-357).

(Deg et al., 2019) Degha, H. E., Laallam, F. Z., Said, B. (2019). Intelligent context-awareness system for energy efficiency in smart building based on ontology. Sustainable computing: informatics and systems, 21, 212-233.

(Dev et al., 2019) Devanne, M., Papadakis, P. (2019, October). Recognition of activities of daily living via hierarchical long-short term memory networks. In 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC) (pp. 3318-3324). IEEE.

(Fan et al., 2019) Fan, X., Wang, F., Wang, F., Gong, W., Liu, J. (2019). When rfid meets deep learning: Exploring cognitive intelligence for activity identification. IEEE Wireless Communications, 26(3), 19-25.

(Fer et al., 2020) Fereidoonian, F., Firouzi, F., Farahani, B. (2020). Human activity recognition: From sensors to applications. In 2020 International Conference on Omni-layer Intelligent Systems (COINS) (pp. 1-8). IEEE.

(Fu, 2021) Fu, B. (2021). Sensor Applications for Human Activity Recognition in Smart Environments.

(Gho et al., 2019) Ghods, A., Cook, D. J. (2019). Activity2vec: Learning adl embeddings from sensor data with a sequence-to-sequence model. arXiv preprint arXiv:1907.05597.

(Goc et al., 2018) Gochoo, M., Tan, T. H., Liu, S. H., Jean, F. R., Alnajjar, F. S., Huang, S. C. (2018). Unobtrusive activity recognition of elderly people living alone using anonymous binary sensors and DCNN. IEEE journal of biomedical and health informatics, 23(2), 693-702.

(Goo et al., 2016) Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep learning. MIT press.

(Hal et al., 2000) Hall, R. E., Bowerman, B., Braverman, J., Taylor, J., Todosow, H., Von Wimmersperg, U. (2000). The vision of a smart city (No. BNL-67902; 04042). Brookhaven National Lab.(BNL), Upton, NY (United States).

(Hee and Yoo, 2018) Cho, H., Yoon, S. M. (2018). Divide and conquer-based 1D CNN human activity recognition using test data sharpening. Sensors, 18(4), 1055.

(Hes and Tes, 2016) Hessen, H. O., Tessem, A. J. (2016). Human activity recognition with two body-worn accelerometer sensors (Master's thesis, NTNU).

(Hoc and Sch, 1997) Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

(Hon and Kha, 2017) Hon, M., Khan, N. M. (2017, November). Towards Alzheimer's disease classification through transfer learning. In 2017 IEEE International conference on bioinformatics and biomedicine (BIBM) (pp. 1166-1169). IEEE.

(Hoy, 2016) Hoy, M. B. (2016). Smart buildings: an introduction to the library of the future. Medical reference services quarterly, 35(3), 326-331.

(Ign, 2018) Ignatov, A. (2018). Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. Applied Soft Computing, 62, 915-922.

(Ind et al., 2018) Indolia, S., Goswami, A. K., Mishra, S. P., Asopa, P. (2018).

Conceptual understanding of convolutional neural network-a deep learning approach.

(Ino et al., 2018) Inoue, M., Inoue, S., Nishida, T. (2018). Deep recurrent neural network for mobile human activity recognition with high throughput. Artificial Life and Robotics, 23(2), 173-185.

(Jai, 2015) Jain, K. (2015). Scikit-learn (sklearn) in Pythonthe most important Machine Learning tool I learnt last year!. Analytics Vidhya, January 5.

(Jar et al., 2009) Jarrett, K., Kavukcuoglu, K., Ranzato, M. A., LeCun, Y. (2009, September). What is the best multi-stage architecture for object recognition?. In 2009 IEEE 12th international conference on computer vision (pp. 2146-2153). IEEE.

(Jia and Yin, 2015) W. Jiang and Z. Yin, (2015).Human activity recognition using wearable sensors by deep convolutional neural networks. In Proceedings of the 23rd ACM international conference on Multimedia. Acm, pp. 13071310.

(Jia et al., 2015) Jiang, W., Yin, Z. (2015, October). Human activity recognition using wearable sensors by deep convolutional neural networks. In Proceedings of the 23rd ACM international conference on Multimedia (pp. 1307-1310).

(Kha et al., 2020) Khan, A., Sohail, A., Zahoora, U., Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. Artificial intelligence review, 53(8), 5455-5516.

(Kum and Cha, 2021) Kumar, P., Chauhan, S. (2021). Human activity recognition with deep learning: Overview, challenges possibilities. CCF Transactions on Pervasive Computing and Interaction, 339(3), 1-29.

(Lar and Lab, 2012) Lara, O. D., Labrador, M. A. (2012). A survey on human activity recognition using wearable sensors. IEEE communications surveys tutorials, 15(3), 1192-1209.

(Le et al., 2019) Le, X. H., Ho, H. V., Lee, G., Jung, S. (2019). Application of long short-term memory (LSTM) neural network for flood forecasting. Water, 11(7), 1387.

(Lec et al., 2015) LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

(Li and Wan, 2022) Li, Y., Wang, L. (2022). Human Activity Recognition Based on Residual Network and BiLSTM. Sensors, 22(2), 635.

***(*Ma et al., 2019)** Ma, H., Li, W., Zhang, X., Gao, S., Lu, S. (2019, August). AttnSense: Multi-level Attention Mechanism For Multimodal Human Activity Recognition. In IJCAI (pp. 3109-3115).

***(*Mag, 2020)** Magnani, A. (2020). Human Action Recognition and Monitoring in Ambient Assisted Living Environments.

***(*Meg and Dje, 2019)** Meghehout, A., Khawla Djeddai, (2019). Apport du bâtiment intelligent dans la gestion de lénergie: Cas dun équipement administratif à Jijel (Master dissertation, Jijel University).

***(*Mek and Jit, 2021)** Mekruksavanich, S., Jitpattanakul, A. (2021). Deep convolutional neural network with rnns for complex activity recognition using wrist-worn wearable sensor data. Electronics, 10(14), 1685.

***(*Mha, 2019)** Mhamdi, A. (2019). Stratégie de gouvernance dans le model de la smart city Cas de la ville dAlger ń Projet Algiers smart city ż (Master dissertation, Blida University).

***(*Mur and Jae, 2017)** Murad, A., Pyun, J. Y. (2017). Deep recurrent neural networks for human activity recognition. Sensors, 17(11), 2556.

***(*Mut and Han, 2020)** Mutegeki, R., Han, D. S. (2020). A CNN-LSTM approach to human activity recognition. In 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC) (pp. 362-366). IEEE.

***(*Naf et al., 2021)** Nafea, O., Abdul, W., Muhammad, G., Alsulaiman, M. (2021). Sensor-based human activity recognition with spatio-temporal deep learning. Sensors, 21(6), 2141.

***(*Nam and Par, 2011)** Nam, T., Pardo, T. A. (2011, June). Conceptualizing smart city with dimensions of technology, people, and institutions. In Proceedings of the 12th annual international digital government research conference: digital government innovation in challenging times (pp. 282-291).

***(*Nan et al., 2020)** Nan, Y., Lovell, N. H., Redmond, S. J., Wang, K., Delbaere, K., van Schooten, K. S. (2020). Deep learning for activity recognition in older people using a pocket-worn smartphone. Sensors, 20(24), 7195.

(Ord and Rog, 2016) Ordóñez, F. J., Roggen, D. (2016). Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors, 16(1), 115.

(Ouk, 2019) Oukrich, N. (2019). Daily human activity recognition in smart home based on feature selection, neural network and load signature of appliances (Doctoral dissertation, Université Mohamed V; Ecole Mohammadia d'Ingénieurs-Université Mohammed V de Rabat-Maroc).

(Par et al., 2018) Park, J., Jang, K., Yang, S. B. (2018, February). Deep neural networks for activity recognition with multi-sensor data in a smart home. In 2018 IEEE 4th World Forum on Internet of Things (WF-IoT) (pp. 155-160). IEEE.

(Pus and Shr, 2022) Pushpalatha, S., Math, S. (2022). Hybrid deep learning framework for human activity recognition. International Journal of Nonlinear Analysis and Applications, 13(1), 1225-1237.

(Rad et al., 2018) Radu, V., Tong, C., Bhattacharya, S., Lane, N. D., Mascolo, C., Marina, M. K., Kawsar, F. (2018). Multimodal deep learning for activity and context recognition. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 1(4), 1-27.

(Rav et al., 2016) Ravi, D., Wong, C., Lo, B., Yang, G. Z. (2016). Deep learning for human activity recognition: A resource efficient implementation on low-power devices. IEEE 13th international conference on wearable and implantable body sensor networks (BSN) (pp. 71-76). IEEE.

(Ron and Sun, 2016) Ronao, C. A., Cho, S. B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. Expert systems with applications, 59, 235-244.

(Rua et al., 2018) Ruano, A., Silva, S., Duarte, H., Ferreira, P. M. (2018). Wireless sensors and IoT platform for intelligent HVAC control. Applied Sciences, 8(3), 370.

(Sar, 2021) Sarker, I. H. (2021). Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. SN Computer Science, 2(6), 1-20.

(Sep et al., 2019) Sepasgozar, S. M., Li, H., Shirowzhan, S., Tam, V. W. (2019). Methods for monitoring construction off-road vehicle emissions: A critical review for identifying deficiencies and directions. Environmental Science and Pollution Research, 26(16), 15779-15794.

(Ser et al. 2022) Serpush, F., Menhaj, M. B., Masoumi, B., Karasfi, B. (2022). Wearable Sensor-Based Human Activity Recognition in the Smart Healthcare System. Computational Intelligence and Neuroscience, 2022.

(Sey et al., 2018) Seyfiolu, M. S., Özbayolu, A. M., Gürbüz, S. Z. (2018). Deep convolutional autoencoder for radar-based classification of similar aided and unaided human activities. IEEE Transactions on Aerospace and Electronic Systems, 54(4), 1709-1723.

(She, 2020) Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306.

(Sin et al., 2017) Singh, D., Merdivan, E., Psychoula, I., Kropf, J., Hanke, S., Geist, M., Holzinger, A. (2017, August). Human activity recognition using recurrent neural networks. In International cross-domain conference for machine learning and knowledge extraction (pp. 267-274). Springer, Cham.

(Sun et al., 2018) Sun, J., Fu, Y., Li, S., He, J., Xu, C., Tan, L. (2018). Sequential human activity recognition based on deep convolutional network and extreme learning machine using wearable sensors. Journal of Sensors.

(Szt, 2019) Sztyler, T. (2019). Sensor-based human activity recognition: Overcoming issues in a real world setting. Universitaet Mannheim (Germany).

(Tan et al., 2020) Tang, Y., Teng, Q., Zhang, L., Min, F., He, J. (2020). Layer-wise training convolutional neural networks with smaller filters for human activity recognition using wearable sensors. IEEE Sensors Journal, 21(1), 581-592.

(Ull et al., 2019) Ullah, M., Ullah, H., Khan, S. D., Cheikh, F. A. (2019). Stacked lstm network for human activity recognition using smartphone data. In 2019 8th European workshop on visual information processing (EUVIP) (pp. 175-180). IEEE.

(Wan and Liu, 2020) Wang, L., Liu, R. (2020). Human activity recognition based on wearable sensor using hierarchical deep LSTM networks. Circuits, Systems, and Signal Processing, 39(2), 837-856.

(Wan et al., 2016) Wang, A., Chen, G., Shang, C., Zhang, M., Liu, L. (2016, June). Human activity recognition in a smart home environment with stacked denoising autoencoders. In International conference on web-age information management (pp. 29-40). Springer, Cham.

(Wan et al., 2019) Wang, F., Feng, J., Zhao, Y., Zhang, X., Zhang, S., Han, J. (2019). Joint activity recognition and indoor localization with WiFi fingerprints. IEEE Access, 7, 80058-80068.

(Wang et al. 2019) Wang, J., Chen, Y., Hao, S., Peng, X., Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. Pattern Recognition Letters, 119, 3-11.

(Wen and Yin, 2015) Jiang, W., Yin, Z. (2015, October). Human activity recognition using wearable sensors by deep convolutional neural networks. In Proceedings of the 23rd ACM international conference on Multimedia (pp. 1307-1310).

(Xu et al., 2021) Xu, Y., Liu, X., Cao, X., Huang, C., Liu, E., Qian, S. Zhang, J. (2021). Artificial intelligence: A powerful paradigm for scientific research. The Innovation, 2(4), 100179.

(Zah et al., 2021) Zahraa, S., Saguna, S., Christer, Å. (2021). Distributed Machine Learning for Anomalous Human Activity Recognition using IoT Systems.

(Zen et al., 2014) Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P., Zhang, J. (2014, November). Convolutional neural networks for human activity recognition using mobile sensors. In 6th international conference on mobile computing, applications and services (pp. 197-205). IEEE.

(Zha et al., 2018) Zhao, Y., Yang, R., Chevalier, G., Xu, X., Zhang, Z. (2018). Deep residual bidir-LSTM for human activity recognition using wearable sensors. Mathematical Problems in Engineering, 2018.

(Zha et al., 2022) Zhang, S., Li, Y., Zhang, S., Shahabi, F., Xia, S., Deng, Y., Alshurafa, N. (2022). Deep learning in human activity recognition with wearable sensors: A review on advances. Sensors, 22(4), 1476.

(Zou et al., 2018) Zou, H., Zhou, Y., Yang, J., Jiang, H., Xie, L., Spanos, C. J. (2018, May). Deepsense: Device-free human activity recognition via autoencoder long-term recurrent convolutional network. In 2018 IEEE International Conference on Communications (ICC) (pp. 1-6). IEEE.

Webography

[**1**] https://www.intechopen.com/chapters/74934 (Retrieved on 05/05/2022).

[**2**] 5. july std 2019.pdf (journalstd.com) (Retrieved on 05/05/2022).

[**3**] Home automation for a smart house - Eco Passive Houses (Retrieved on 06/05/2022).

[**4**] https://www.cea.fr/comprendre/Pages/nouvelles-technologies/essentiel-sur-domotique-maison-connectee.aspx (Retrieved on 05/05/2022).

[**5**] https://www.ucly.fr/wp-content/uploads/2020/02/smart-cities.pdf (Retrieved on 10/05/2022).

[**6**] https://www.thalesgroup.com/en/markets/digital-identity-and-security/iot/inspired/smart-cities (Retrieved on 02/08/2022).

[**7**] https://www.twi-global.com/technical-knowledge/faqs/what-is-a-smart-city (Retrieved on 10/08/2022).

[**8**] https://nexusintegra.io/smart-city-what-is-how-it-works/ (Retrieved on 10/08/2022).

[**9**] https://www.smartcityexpo.com/ (Retrieved on 10/08/2022).

[**10**] https://machinelearningmastery.com/deep-learning-models-for-human-activity-recognition/ (Retrieved on 17/05/2022).

[**11**] https://medium.com/@achoulwar901/the-art-of-convolutional-neural-network-abda56dba55c (Retrieved on 20/05/2022).

[**12**] https://stanford.edu/ shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks (Retrieved on 25/05/2022)

[**13**] https://medium.com/tech-break/recurrent-neural-network-and-long-term-dependencies-e21773defd92 (Retrieved on 25/05/2022)

[**14**]https://archive.ics.uci.edu/ml/datasets/WISDM+Smartphone+and+Smartwatch+Activity +and+Biometrics+Datast (Retrieved on 28/05/2022)

[**16**] https://www.tensorflow.org/?hl=fr (Retrieved on 02/06/2022)

[**17**] https://numpy.org/ (Retrieved on 02/06/2022)

[**18**] https://matplotlib.org/ (Retrieved on 02/06/2022)

[**19**] https://scikit-learn.org/stable/index.html (Retrieved on 02/06/2022)

[**20**] https://pandas.pydata.org (Retrieved on 02/06/2022)