People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

University of Ghardaia
Faculty of Science and Technology
Department of Mathematics and Computer Science

T H E S I S

to get the title

IT Master

Specialty : Intelligent Systems for Knowledge Extraction

Presented by

Firas SAADI **et** Nacer Eddine LAKAS

Human Activity Recognition approach for Energy Efficiency based on Deep Learning

---

supported on 2023

Jury members :

M.Abdelkader BOUHANI	*MAA*	*Univ.Ghardaia*	*President*
Dr.Ahmed SAIDI	*MCB*	*Univ.Ghardaia*	*Examiner*
Dr.Houssem Eddine DEGHA	*MCB*	*Univ.Ghardaia*	*Supervisor*

University Year : 2022/2023

Acknowledgements

This thesis has become possible thanks to Almighty God, and I must thank Almighty God who guided us and gave us the strength and perseverance necessary to complete this work. Praise be to Almighty God, the Lord of all worlds. I would also like to express our gratitude to the many people who have assisted us.

Firstly, I would like to thank the thesis supervisor, Houssem Eddine Degha, for the time he dedicated to providing us with the essential methodological tools to conduct this research, for his patience and availability, and above all, for his wise advice that has enriched our thinking.

Secondly, I would like to thank the professors of the Mathematics and Computer Science Department at Ghardaia University, who provided us with the necessary tools for the success of our academic studies.

Lastly, special greetings to our colleagues and friends who shared our educational journey with us.

Dedication

To our loving parents, whose unwavering support and endless encouragement have been the driving force behind our pursuit of knowledge, without their support this work would probably have been

It was never completed. So represent this thesis

A culmination of the support and encouragement they gave us.

To our dear brothers for their unending and unwavering love and to all of our family.

And also to all the people who supported us and were a source for our continuation in our academic journey and the completion of this thesis.

I dedicate this humble work.

Firas and Nacer Eddine

ABSTRACT

In recent years, human activity recognition has garnered significant attention due to its applications in healthcare, smart homes, and surveillance systems. Traditionally, wearable devices and camera-based approaches have been used for activity recognition, but they come with limitations related to user comfort, privacy concerns, and installation requirements. However, the advent of sensor technology and the Internet of Things (IoT) has opened up new possibilities for activity recognition using environmental sensors. This thesis explores the feasibility of utilizing environmental sensors for human activity recognition and compares six different models using exclusively collected environmental sensor data. The research involves establishing a robust data collection framework, applying preprocessing techniques, and evaluating the models using various metrics. The findings will provide valuable insights into the potential of environmental sensors for activity recognition and guide researchers, practitioners, and system developers in making informed decisions regarding model selection and system design.

Keywords : human activity recognition, environmental sensors, dataset, machine learning models, comparative analysis.

RÉSUMÉ

Ces dernières années, la reconnaissance de l'activité humaine a suscité une attention particulière en raison de ses applications dans les soins de santé, les maisons intelligentes et les systèmes de surveillance. Traditionnellement, les appareils portables et les approches basées sur la caméra ont été utilisés pour la reconnaissance des activités, mais ils s'accompagnent de limitations liées au confort de l'utilisateur, aux problèmes de confidentialité et aux exigences d'installation. Cependant, l'avènement de la technologie des capteurs et de l'Internet des objets (IoT) a ouvert de nouvelles possibilités de reconnaissance d'activité à l'aide de capteurs environnementaux. Cette thèse explore la faisabilité d'utiliser des capteurs environnementaux pour la reconnaissance de l'activité humaine et compare six modèles différents utilisant exclusivement des données de capteurs environnementaux collectées. La recherche consiste à établir un cadre de collecte de données robuste, à appliquer des techniques de prétraitement et à évaluer les modèles à l'aide de diverses mesures. Les résultats fourniront des informations précieuses sur le potentiel des capteurs environnementaux pour la reconnaissance des activités et guideront les chercheurs, les praticiens et les développeurs de systèmes dans la prise de décisions éclairées concernant la sélection de modèles et la conception de systèmes.

Mots-clés : reconnaissance de l'activité humaine, capteurs environnementaux, ensemble de données, modèles d'apprentissage automatique, analyse comparative.

ملخص

في السنوات الأخيرة ، حظي التعرف على النشاط البشري باهتمام كبير نظرًا لتطبيقاته في الرعاية الصحية والمنازل الذكية وأنظمة المراقبة. تقليديًا ، تم استخدام الأجهزة القابلة للارتداء والأساليب القائمة على الكاميرا للتعرف على النشاط ، ولكنها تأتي مع قيود تتعلق براحة المستخدم ومخاوف الخصوصية ومتطلبات التثبيت. ومع ذلك ، فإن ظهور تقنية الاستشعار وإنترنت الأشياء IoT قد فتح إمكانيات جديدة للتعرف على النشاط باستخدام المستشعرات البيئية. تستكشف هذه الأطروحة جدوى استخدام المستشعرات البيئية للتعرف على النشاط البشري وتقارن ستة نماذج مختلفة باستخدام بيانات أجهزة الاستشعار البيئية التي تم جمعها حصريًا. يتضمن البحث إنشاء إطار عمل قوي لجمع البيانات ، وتطبيق تقنيات المعالجة المسبقة ، وتقييم النماذج باستخدام مقاييس مختلفة. ستوفر النتائج رؤى قيمة حول إمكانات المستشعرات البيئية للتعرف على الأنشطة وتوجيه الباحثين والممارسين ومطوري الأنظمة في اتخاذ قرارات مستنيرة بشأن اختيار النموذج وتصميم النظام.

الكلمات المفتاحية : التعرف على النشاط البشري ، أجهزة االستشعار البيئية ، مجموعة البيانات ، نماذج التعلم آالي ، المعايير.

ACRONYMS

— **PDA** Personal Digital Assistant
— **ADL** Activity of Daily Living
— **AE** Automatic Encoders
— **API** Application Programming Interface
— **DL** Deep Learning
— **ML** Machine Learning
— **DNN** Deep Neural Networks
— **DRL** Deep Reinforcement Learning
— **FCN** Fully Convolutional Network
— **GAN** Generative Adversary Networks
— **GRU** Gated Recurrent Units
— **ANN** Artificial Neural Networks
— **CNN** Convolutional Neural Networks
— **RNN** Recurrent Neural Networks
— **LSTM** Long Short Term Memory
— **HAR** Human Activity Recognition
— **ICT** Information and Communication Technology
— **ID** Integration Device
— **IoT** Internet of Things
— **NLP** Natural Language Processing
— **RBM** Restricted Boltzmann Machines
— **RFID** Radio Frequency Identification
— **Seq2Seq** Sequence-to-Sequence model
— **TCP/IP** Transmission Control Protocol/Internet Protocol
— **TSC** Time Series Classification
— **UDP/IP** User Datagram Protocol/Internet Protocol

Table des matières

TABLE DES FIGURES

LISTE DES TABLEAUX

CHAPITRE 1

GENERAL INTRODUCTION

In the past few years, there has been a notable surge in interest and focus on the field of human activity recognition. This field holds immense importance across several domains, such as healthcare, smart homes, and surveillance systems. The accurate detection and classification of human activities have profound implications, ranging from enhancing healthcare monitoring to automating and improving the convenience of smart environments.

In the past, human activity recognition systems primarily relied on wearable devices or camera-based approaches. However, these methods had certain drawbacks, including potential discomfort for users, privacy concerns, and installation requirements. Thankfully, the continuous progress in sensor technology and the expansion of the Internet of Things (IoT) have opened up a new and promising path for activity recognition : the use of environmental sensors.

Environmental sensors, including temperature, humidity, movement, and sound sensors, have gained significant popularity in the realm of smart homes, buildings, and cities. These sensors hold the potential to gather valuable information about human activities without the requirement of invasive or intrusive devices. By utilizing the data obtained from environmental sensors, it becomes feasible to create precise and non-intrusive systems for recognizing human activities.

This thesis delves into the possibility of employing environmental sensors for the purpose of human activity recognition. The main goal is to train and compare six distinct models using a dataset that has been exclusively collected from environmental sensors. By emphasizing the utilization of these sensors, our aim is to leverage the potential of environmental data in order to create activity recognition systems that are both reliable and efficient.

The research encompasses the establishment of a sturdy framework for data collection and preprocessing techniques to guarantee the quality and dependability of the dataset. Six well-established models in activity recognition, each employing distinct approaches and underlying algorithms, are carefully chosen for comparison. Feature extraction, selection, and model optimization techniques are implemented to enhance the performance of these models. The evaluation encompasses diverse metrics to assess the efficacy of the models in accurately recognizing human activities, taking into account both accuracy and practical feasibility.

The findings derived from this research will offer valuable insights into the potential application of environmental sensors in human activity recognition. The comparative analysis of the models will illuminate the influence of various algorithms and architectural choices. Ultimately, the results of this thesis will contribute to the progression of human activity recognition systems, providing guidance to researchers, practitioners, and system developers in making informed decisions pertaining to model selection and system design.

In general, the objective of this research is to leverage the capabilities of environmental sensors and showcase their potential in accurately detecting and classifying human activities. By doing so, it aims to contribute to the advancement of the field of human activity recognition.

To accomplish this work, we have divided the dissertation into four chapters. The second chapter serves as an introduction to the field, covering essential knowledge, concepts, deep learning models, and related works. In the third chapter, we present visualizations of the dataset. The fourth chapter focuses on the practical aspects, discussing the results obtained, and concludes with a comprehensive summary of the dissertation.

CHAPITRE 2

LITERATURE REVIEW

2.1 Introduction

In the context of intelligent ecosystems, such as smart cities and residential spaces, the automation of tasks and the seamless integration of smart living technologies have become pivotal in enhancing the quality of life for inhabitants. The proliferation of sensors in smart homes and other environments enables the collection of vast amounts of data, forming the foundation for intelligent decision-making and personalized services. Furthermore, the advancements in artificial intelligence, particularly deep learning algorithms, offer immense potential in analyzing and extracting meaningful insights from the massive datasets generated by these smart devices.

This chapter serves as an introduction to the fundamental pillars of our research : human activity recognition, smart environmentsand lot , environmental sensors, deep learning algorithms, and related works.

2.2 Human Activity Recognition

Human Activity Recognition (HAR) entails the monitoring of an individual's activities using a network of sensors and interconnected devices. This process generates a dataset consisting of time series data reflecting parameter values or state changes. To accomplish this, a range of sensors can be employed, such as RFID, accelerometers, detectors, noise sensors, and motion sensors (Le et al., 2019).

Human Activity Recognition (HAR) has emerged as a prominent and challenging research field due to its increasing significance. HAR systems serve as valuable tools for researchers to gather data on human behavior. As outlined by (Fer et al. 2020), these systems can be categorized into three main types : sensor-based systems, vision-based systems, and multimodal-based systems represented in FIGURE 2.1.

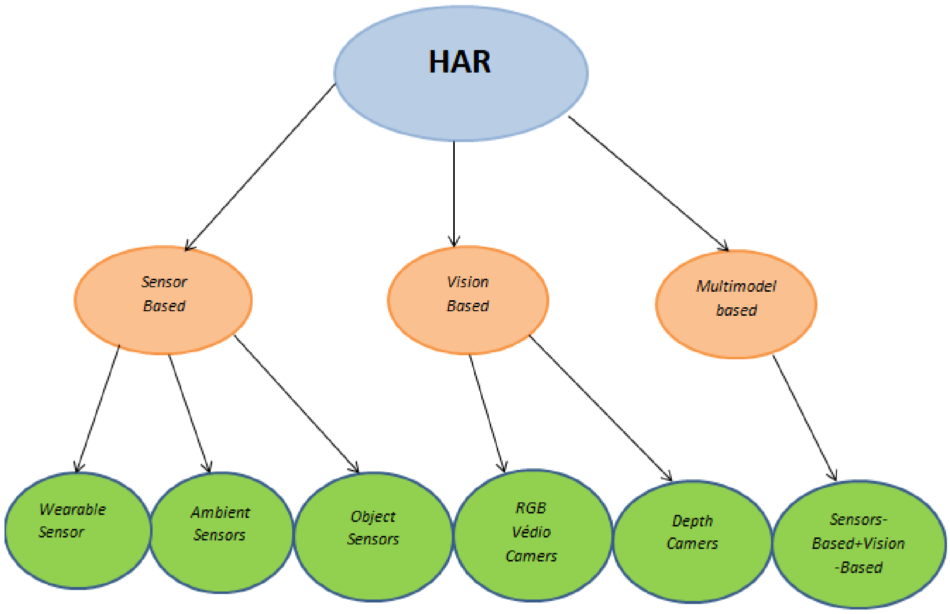

FIGURE 2.1 – Human activity system.

As previously mentioned, there are three main types of HAR systems : sensor-based, vision-based, and multimodal-based. The first category, sensor-based, encompasses three distinct types of sensors. Wearable sensors are attached to a person's limbs or clothing to capture their movements. Examples of wearable sensors used in HAR systems include the Global Positioning GPS System, Inertial Measurement Unit (Gyroscope, Accelerometer, Magnetometer), smartphone sensors, smartwatch sensors, and biosensors. Ambient sensors, on the other hand, utilize temperature, light, pressure, RFID, radar, Wi-Fi, and Bluetooth sensors to gather environmental data. The final type of sensor-based system utilizes sensors mounted on automobiles, trains, and aircraft.

The second type of HAR is vision-based, which relies on depth cameras and RGB video cameras for data collection. However, vision-based systems may encounter challenges as standard cameras are unable to operate effectively in complete darkness (Fer et al., 2020).

The challenge of vision-based HAR in complete darkness can be overcome with the aid of modern technology such as the Kinect camera. This advanced camera is capable of generating a three-dimensional virtual skeleton and providing depth, RGB, and audio information. This integration enhances the robustness, accuracy, and reliability of activity identification. However, it is important to note that the combination of sensor-based and vision-based activity detection algorithms in multimodal HAR systems can increase both the cost and complexity. In comparison, wearable sensors offer several advantages over fixed cameras, despite their drawbacks, which include high cost, complexity, and privacy concerns. Wearable sensors are more cost-effective, easy to handle, and convenient to position.

Machine learning (ML) methods are commonly utilized in HAR systems to detect events from the data collected from various signals. These systems find valuable applications in smart environments, such as smart homes, where they can continuously monitor patients for health diagnoses, medication reminders, or even predict crimes through automated surveillance in public places. To gain a better understanding of the relationship between HAR and deep learning in smart environments like the smart home, it is essential to define the terms "activity" and "HAR process." This clarification will provide insights into how these concepts intersect and complement each other.

2.2.1 Notion of Activity

In our research domain, the term "activity" refers to a collection of physical behaviors that an individual performs within their domestic environment. These activities can be hierarchically structured into actions. For instance, the activity of "sleeping" can be further divided into distinct actions like "entering the bedroom" and "lying down in bed." These actions, in turn, consist of atomic steps that constitute the activity, such as "turning the door handle" or "switching off the light." Thus, activities are composed of actions, and actions are comprised of atomic operations (Ouk, 2019).

2.2.2 General Structure of HAR Systems

The architecture of a HAR system can be divided into four primary stages. In the first stage, wearable sensors are attached to measure the mobility, position, and temperature of a person's body. These sensors require an Integration Device (ID) to establish a connection, which can be a laptop, PDA, smartphone, or a purpose-built embedded system. The main objective of the ID is to preprocess the data received from the sensors and, in certain cases, transmit it to a server application for real-time monitoring, display, and/or analysis. Depending on the desired level of reliability, either UDP/IP or TCP/IP communication protocols are suitable choices (Lar and Lab, 2012). The accompanying FIGURE 2.2 illustrates the general data acquisition architecture for HAR systems.

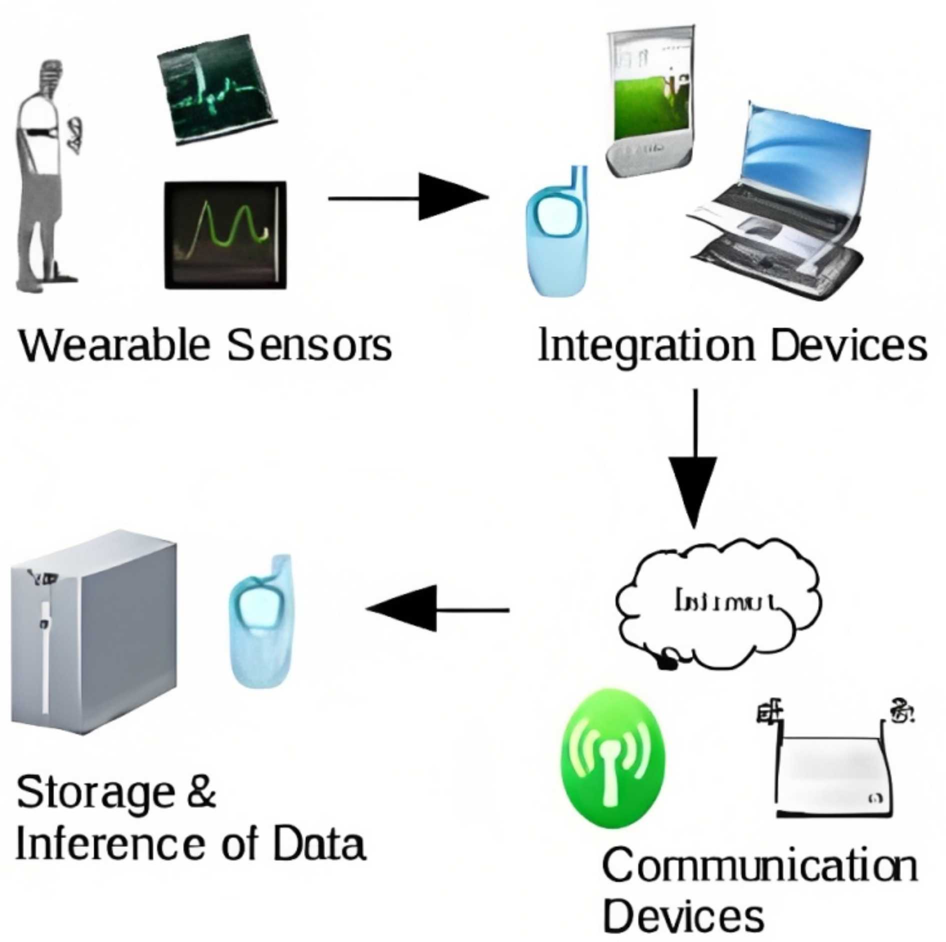

FIGURE 2.2 – Generic data collection architecture for human activity recognition (HAR) [24]

The previous discussion has already established the potential privacy concerns associated with monitoring a person's activities at home. While camera installations are commonly used in various security services, homeowners often feel reluctant to keep cameras and monitoring systems active while they are present in the house. Consequently, sensor-based systems have gained widespread popularity in recognizing routine human activities, particularly in smart homes (Lar and Lab, 2012). The advent of the Internet of Things (IoT) and the availability of powerful yet affordable smart devices have enabled the emergence of practical technical solutions in the form of smart homes equipped with ambient sensors. However, in order to fully harness the capabilities of such hardware, an ideal system also requires well-designed algorithms and solutions.

2.2.3 Human Activity Recognition Process

In recent years, human activity recognition (HAR) has gained significant attention as a challenging research field. HAR involves the utilization of diverse sensors, including pressure detectors, motion sensors, RFID, and electrical power analyzers, to identify the activities of individuals or multiple residents within a home environment. The HAR process consists of several crucial stages, and the following four steps have been identified as particularly important (Bel et al., 2015) :

- • **Preprocessing :** eliminating unprocessed data to handle incompleteness, eliminate noise and redundancy, and normalize the data from sensor streams ;
- • **Features extraction :**the method of removing features from unprocessed data to feed machine learning ;
- **Features selection :** improve the quality of the features while reducing the amount of features to reduce the computing effort needed for classification ;
- • **Classification :** HAR refers to the use of machine learning and logic to identify a specific activity by monitoring and analyzing a person's movements. It involves the network-based daily activity monitoring of residents in a smart home through IoT devices. By monitoring activities, a smart home can provide personalized assistance services to enhance the autonomy, health, and quality of life of its residents, particularly for children, elderly individuals, and those who require assistance.

In general, the process of HAR involves three crucial steps that start from collecting data about environmental conditions and human behavior and end with identifying the current activity. These steps include receiving sensor data from different sensor technologies as the first step. Next, the data is processed to remove redundancy and noise, and normalization is performed. In the second step, feature extraction is carried out to extract the most important activity features like spatial and temporal data. Finally, classification is done to determine the activity using ML and DL models trained on the data. Figure 2.3 demonstrates these steps.

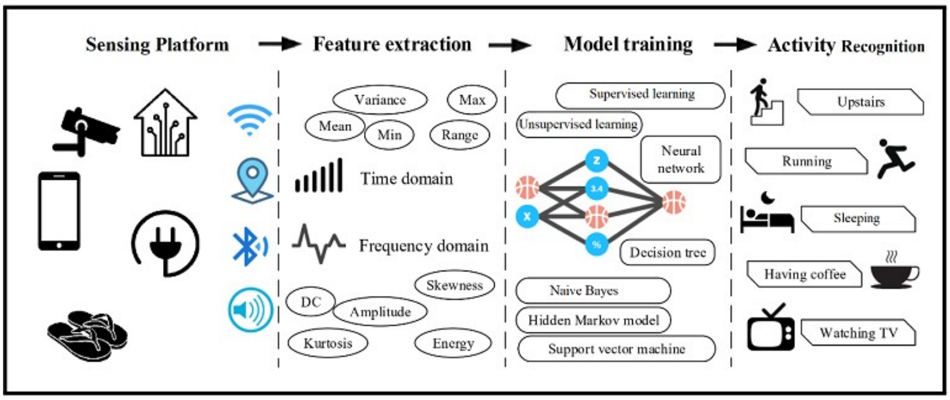

FIGURE 2.3 – An Illustration of Sensor-based HAR Activity Recognition (Mag, 2020)

The primary objective of HAR systems is to supplement or replace human activities within homes. This can be achieved by either predicting and performing activities when necessary or meeting the predefined criteria and preferences set by humans. For example, a HAR system can monitor the health of a resident and utilize sensory devices to promptly alert medical professionals in case of an urgent requirement (Ser et al., 2022).

Pattern categorization challenges in HAR can be attributed to both the limitations of ambient and wearable sensors, as well as the algorithms employed in smart environments. To overcome these challenges, there has been an increasing adoption of deep learning (DL) techniques for activity classification in HAR.

In the subsequent section, we will delve into a comprehensive review of the existing research and developments concerning DL-based HAR approaches.

2.3 Smart Environments and IoT

This section primarily focuses on the technological components that constitute a smart environment, which encompass IoT, sensor technologies, and data processing.

2.3.1 Smart Environments

Smart environments refer to physical spaces, including cities, homes, workplaces, and public places, that are equipped with technology and interconnected devices to enhance functionality, efficiency, and overall quality of life. These environments leverage sensors, IoT devices, artificial intelligence, and automation systems to gather and analyze data, monitor and control various aspects of the environment, and automate tasks to enhance convenience and efficiency.

Smart environments offer numerous benefits, including improved energy efficiency, waste reduction, enhanced safety, and increased comfort. However, they also give rise to concerns regarding privacy, security, and overreliance on technology. Therefore, it is crucial to ensure responsible and sustainable design and implementation of smart environments.

2.3.1.1 Smart Home

Definition

A smart home is a residential space equipped with diverse technologies and devices that enable automation, remote monitoring, and control of various functions. These functions include but are not limited to lighting, temperature regulation, security, and entertainment. The devices within a smart home are interconnected and connected to the internet, allowing them to be controlled conveniently using a smartphone or other internet-connected devices.

Smart homes integrate a range of sensors, IoT devices, and automation systems to gather and analyze data. These systems enable automatic adjustments or manual control of various settings. For instance, a smart thermostat can learn the temperature preferences of the homeowner and make adjustments accordingly. Similarly, smart lights can be programmed to activate and deactivate at specific times or in response to detected motion.

FIGURE 2.4 – Smart Home [20]

Smart homes offer numerous advantages, such as heightened comfort, convenience, improved energy efficiency, and enhanced home security. By reducing energy consumption and mitigating potential home damage, they can also lead to cost savings. As the technology behind smart homes continues to advance, their capabilities and potential benefits are expected to expand even further.

Smart Home Applications

- **- Home automation** : Smart homes can automate routine tasks, such as turning lights on and off or adjusting the thermostat, based on predefined schedules or user preferences.

- **- Energy management** : Smart homes can monitor and manage energy usage, identifying inefficiencies and providing suggestions for reducing consumption.

- **- Home security** : Smart homes can be equipped with cameras, motion sensors, and other security devices to monitor and protect the home and its inhabitants.

- **- Entertainment** : Smart homes can integrate audio and video systems to provide immersive entertainment experiences, from streaming music to watching movies.

— **- Health monitoring** : Smart homes can incorporate health monitoring devices, such as blood pressure monitors or fitness trackers, to track and analyze personal health data.

— **- Voice control** : Many smart homes feature voice-controlled assistants, such as Amazon's Alexa or Google Assistant, to enable hands-free control of various devices and functions.

2.3.1.2 Smart Building

Definition

A smart building refers to a structure that integrates diverse technologies and systems to optimize its functionality, efficiency, and overall performance. These technologies and systems encompass sensors, automation systems, energy management systems, and IoT devices. They are interconnected, enabling the collection and analysis of data to enhance building operations.

Smart buildings leverage technology to automate tasks, optimize energy consumption, enhance occupant comfort and safety, and improve building management and maintenance. For instance, sensors can detect occupancy levels and adjust lighting and temperature accordingly. Predictive maintenance algorithms can identify and address maintenance issues proactively, preventing them from escalating into significant problems.

Smart buildings are prevalent in diverse settings, such as commercial and residential buildings, hospitals, universities, and government facilities. They provide numerous advantages, including decreased energy consumption, reduced operating costs, enhanced occupant productivity and comfort, and improved safety and security. As technology in smart buildings progresses, their potential benefits are expected to expand even further.

FIGURE 2.5 – smart building [21]

2.3.1.3 Smart Cities

Definition

A smart city refers to an urban environment that leverages advanced technology and data analysis to optimize its operations, improve the quality of life for its residents, and foster sustainable development. These cities integrate various technologies and systems, such as IoT devices, sensors, data analytics, and automation systems. They utilize these tools to gather and analyze data, monitor and manage city infrastructure, and enhance service delivery.

Smart cities strive to enhance the efficiency and effectiveness of urban services, including transportation, energy, water, waste management, public safety, and healthcare. By leveraging technology, these cities aim to optimize resource utilization, minimize environmental impact, and elevate the quality of life for their citizens.

FIGURE 2.6 – smart city [22]

Importance of Smart Cities

Smart cities hold immense significance as they embody a fresh approach to urban development centered around leveraging technology and data to enhance citizens' quality of life. Through the utilization of advanced technologies like the Internet of Things (IoT), sensors, and artificial intelligence, smart cities can optimize resource allocation, minimize energy usage, enhance transportation systems, bolster public safety measures, and provide more streamlined public services. Moreover, they foster economic growth and innovation while promoting sustainability and resilience. In essence, smart cities play a pivotal role in cultivating livable, sustainable, and resilient urban environments where individuals can thrive.

2.4 Environmental sensors

Environmental sensors installed in smart homes are specialized devices that monitor and measure various environmental factors within a living space. These sensors, typically embedded in smart thermostats, smoke detectors, and security systems, are designed to gather real-time data on parameters like temperature, humidity, air quality, and noise levels. By continuously monitoring these environmental variables, the sensors contribute to creating a comfortable and healthy living environment for occupants. Additionally, these sensors play a crucial role in human activity recognition within smart homes. By analyzing the data collected from the environmental sensors, advanced algorithms and machine learning techniques can infer specific activities or behaviors of individuals. This information can be used to automate and personalize various aspects of the smart home, such as adjusting temperature and lighting, optimizing energy usage, and enhancing security. By leveraging the insights provided by environmental sensors, smart homes can improve occupants' quality of life, promote energy efficiency, and enhance overall home automation experiences.

2.5 Deep Leaning Algorithms

In this section, our focus will be on deep learning, specifically delving into its models and discussing the two main algorithms employed for human activity recognition

2.5.1 Introduction to Machine Learning

Machine learning, a subset of artificial intelligence (AI), encompasses the development of algorithms that enable computers to learn from data and make predictions or decisions without explicit programming.

The fundamental concept behind machine learning involves training models using data rather than relying on predefined software rules. These models leverage statistical or mathematical techniques and can be trained to identify patterns, make predictions, or take actions based on input data.

Various types of machine learning algorithms exist, including supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning entails training a model using labeled data, where the desired output is known. The model learns to predict outputs based on inputs and is evaluated based on its accuracy in predicting the correct output.

Unsupervised learning involves training a model on unlabeled data, aiming to uncover patterns or structures within the data. This type of learning is often utilized for tasks like clustering or anomaly detection.

Reinforcement learning centers on training a model to make decisions based on rewards and punishments. The model learns to take actions that maximize accumulated rewards over time.

Machine learning finds application in diverse fields, such as image recognition, natural language processing, speech recognition, recommendation systems, fraud detection, and more.

2.5.2 Introduction to Deep Learning

Deep learning, a subset of machine learning, focuses on training deep neural networks with multiple layers to identify patterns within vast and intricate datasets. These networks are designed to resemble the structure of the human brain, comprising interconnected nodes across layers that process and learn from data.

Deep learning algorithms excel at processing unstructured data, such as images, audio, and text. This is because deep neural networks can autonomously learn and extract features from raw data, eliminating the need for manually defined features.

The training process in deep learning involves inputting extensive data into the network, adjusting the weights and biases of the nodes in each layer, and evaluating the network's performance on a validation set. This iterative process continues until the network achieves the desired level of accuracy.

A key advantage of deep learning lies in its scalability to handle large and intricate datasets. Deep learning models can be trained on massive volumes of data, resulting in improved accuracy and performance.

Deep learning finds applications in a broad range of fields, including image and speech recognition, natural language processing, autonomous vehicles, and numerous others. It has become a crucial area of research and development within the field of artificial intelligence.

Among the deep learning algorithms that rely on a neural network :

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

2.5.2.1 Artificial Neural Network (ANN) :

An Artificial Neural Network (ANN) is a computational model inspired by the structure and functionality of the human brain. It is a mathematical model composed of interconnected processing units called artificial neurons or nodes. The basic building block of an ANN is a neuron, which receives inputs, applies weights to them, performs a mathematical operation, and generates an output. The outputs of neurons in one layer serve as inputs to the neurons in the next layer, forming a network of interconnected layers.

— **Input Layer :** It receives the initial data or features to be processed.
— **Hidden Layer(s) :** These layers perform complex computations by applying weights to inputs and applying activation functions. They extract features and learn representations from the input data.
— **Output Layer :** It produces the final output, which depends on the problem being solved. For example, in a classification task, the output layer may represent different classes, while in a regression task, it may produce a continuous value.

2.5.2.2 Convolutional Neural Network (CNN) : Convolutional Neural Networks (CNNs) are deep neural networks used for image classification, object detection, and computer vision tasks. They use a series of convolutional layers to capture spatial and temporal dependencies in the data by applying learnable filters to extract features such as edges, corners, and shapes. The filters used in a CNN can identify patterns in the image regardless of their location, due to a sliding window approach. CNNs also typically include pooling layers to reduce feature map size and fully connected layers for classification or regression tasks the general structure of a CNN network is depicted see FIGURE 2.7.

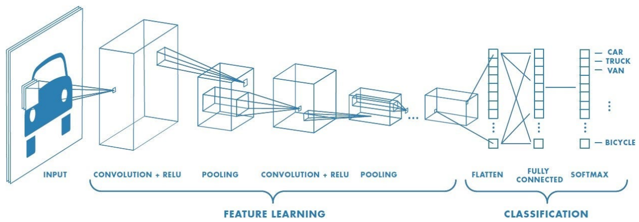

FIGURE 2.7 – General Structure Of a CNN (Kha et al., 2020)

Convolutional Layers : Convolutional layers are the backbone of a CNN. They consist of a set of filters that are convolved with the input image to produce a set of feature maps. Each filter is a small matrix of learnable parameters that slides over the input image, computing the dot product between its values and the corresponding pixels in the input. The result of the convolution operation is a set of feature maps, where each map represents the presence of a specific feature in the input see the example in FIGURE 2.8. Convolutional layers are the key to the spatial invariance property of CNNs, meaning that they are able to recognize the same pattern regardless of its position in the input image.

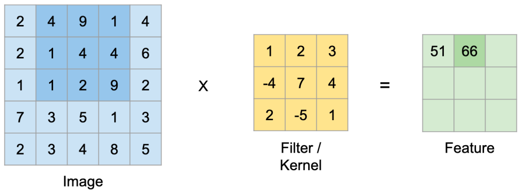

FIGURE 2.8 – Convolution Operation [10]

Pooling Layers : Pooling layers are used to reduce the spatial dimensionality of the feature maps produced by the convolutional layers. This is important because it reduces the number of parameters in the network, which in turn helps to reduce overfitting. There are several types of pooling operations, including max pooling, which takes the maximum value within a patch of the feature map, and average pooling, which takes the average value see FIGURE 2.9. **Fully Connected Layers :** Fully connected layers are used

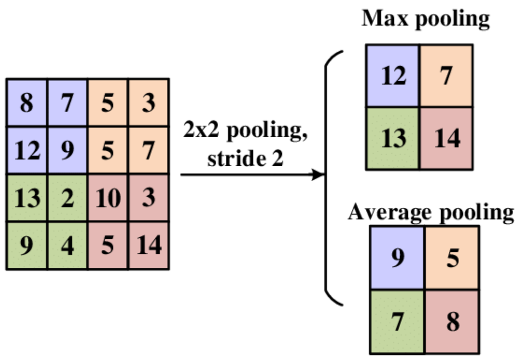

FIGURE 2.9 – Maximal pooling and Average pooling [23]

to perform the final classification or regression task. They take the output of the convolutional and pooling layers, which are typically flattened into a one-dimensional vector, and apply a set of weights to produce the final output. In the case of image classification, the output is typically a set of probabilities over a set of classes

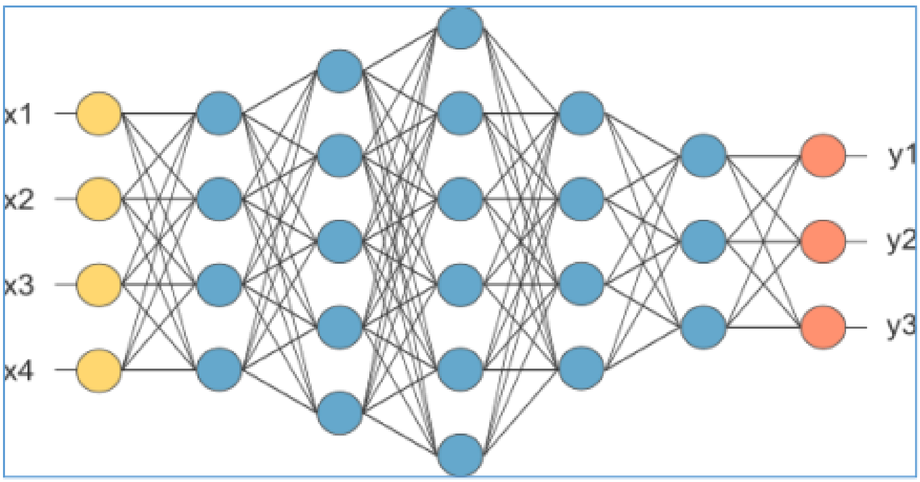

FIGURE 2.10 – Fully Connected Layer (Ind et al., 2018)

2.5.2.3 Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNNs) are a specific type of neural network that can handle sequential data. Unlike feed forward neural networks, which handle each input separately, RNNs are equipped with an internal memory that enables them to process input sequences and maintain context. As a result, they are particularly useful for tasks such as language modeling, speech recognition, and machine translation.

The fundamental concept behind RNNs involves using the output from the previous time step as input for the current time step. This allows the network to remember past inputs and use that information to make predictions or classifications. Essentially, RNNs are designed to detect sequential patterns or dependencies in data.

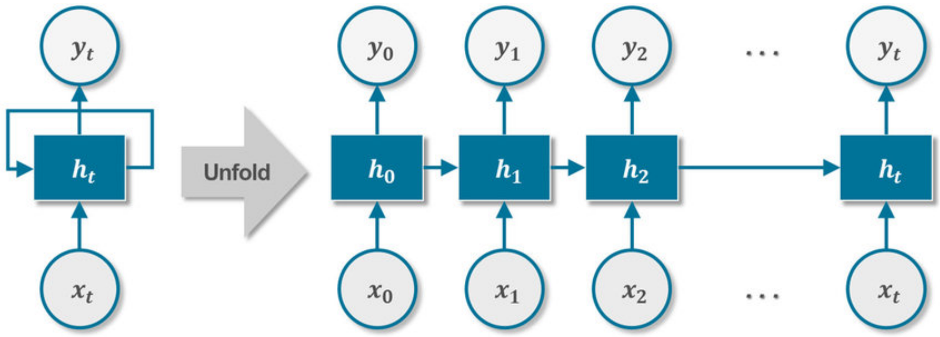

FIGURE 2.11 – RNN architecture [25]

2.5.2.4 Long Short Term Memory (LSTM) Networks

The main problem with traditional RNNs that contributed to the emergence of LSTM is the vanishing gradient problem. This occurs when the gradients that are propagated across the network during back propagation become very small, making it difficult to update the network weights. As a result, the network is unable to capture the long-term dependencies in the data, which limits its ability to model sequential data as shown in the figure.

LSTMs address this problem by introducing a memory cell, which allows the network to selectively remember or forget information over long periods of time.

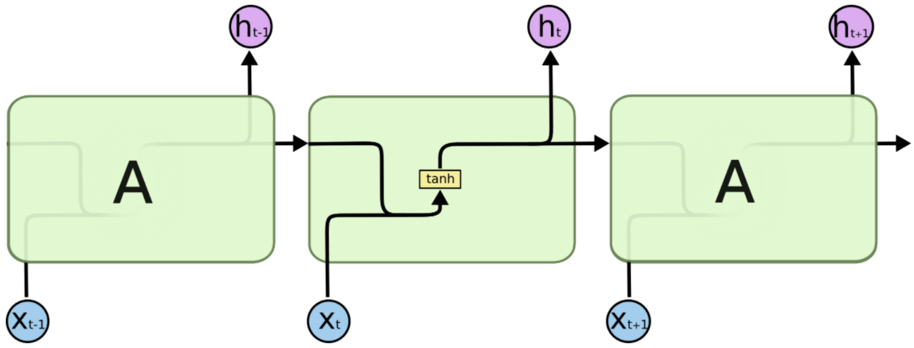

FIGURE 2.12 – RNN architecture model (She, 2020)

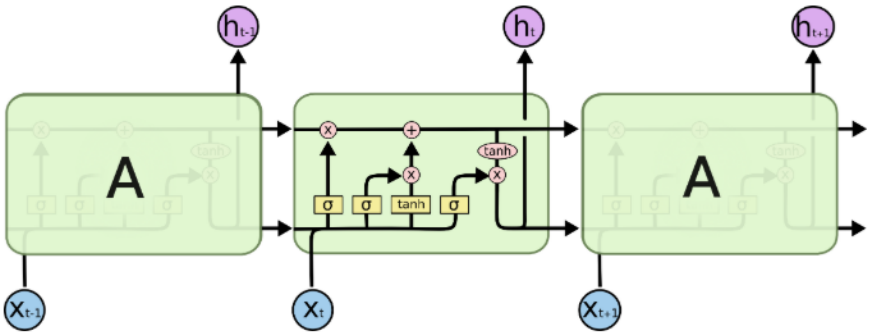

FIGURE 2.13 – LSTM architecture model (She, 2020)

In order to know more details on the internal structure, we present the following figure :

2.6 related work

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

TABLE 2.1 – The table shows the related works

(Ign, 2018) **CNN**	UCL-HAR WISDM	Data centering+ Normalization	CNN	CNN+ Statistical features	Locomotion activity	Accuracy	90.42% 93.32%
(Ino et al., 2013) **DRNN**	UCI-HAR	No	LSTM	No	Locomotion activity	Accuracy	95.42%
(Jia and Yin, 2015) **DCNN**	UCIHAR USC-HAD SHO	No	2D CNN	No	Daily life activity	Accuracy Comput-cost	97.59% 97.83% 99.93% Accuracy
(Li and Wan, 2022) **BIiLSTM**	Homemade WISDM PAMAP2	Mean + Standard Deviation	LSTM	Residual block + BIiLSTM	Sport activity health activity	Accuracy	96.95% 97.32% 97.15%
Ma et al, 2019) **AttnSense**	STISEN Skoda PAMAP2	Data Augment + Fast Fourier Transform + Segmentation	ID CNN+ GRU	CNN	Locomotion Factory maintenance	Fl score	96.5% 93.1% 89.3%
(Mek and Jit, 2021) **CNN-BiGRU**	UTwente	Noise reduction + Missing+ Filling + Normalization	ID CNN + BiGRU	CNN	Simple + complex activity	Accuracy Precision Recall F1 score	98.78% Accuracy
(Moh et al., 2020) **DCNN AdaBoost**	ADL	Segmentation + Converting binary string data into greyscale images	1D CNN	CNN	Daily life activity	Accuracy	99.5%
(Mur and Jae, 2017) **DCNN**	UCL-HAR USC-HAD Opportunity FOG Skoda	No	LSTM+ RNN	No	Daily activity Locomotion health-related activity factory maintenance	Accuracy Precision Recall Fl score	96.1% 97.8% 92% 93% 92.6% Accuracy
(Naf et al, 2021) **CNN-BiLSTM**	WISDM UCI-HAR	Time Series Data + Segmentation	1D CNN+ BiLSTM	CNN	Locomotion activity	Accuracy	98.53% 97.05%

TABLE 2.2 – The table shows the related works

(Nan et al, 2020) Multichannel CNN-LSTM	NRA	Data centering+ Normalization	multichannel 1D CNN+LSTM	CNN	Segmentation + Sliding window	Accuracy Comput-cost	81.1% Accuracy
(Ord and Rog, 2016) DeepConvLSTM	Opportunity Skoda	No	Conv+LSTM	conv	Locomotion activity Factory maintenance	F1 score	95.8% F1 score
(Par et al, 2018) Residual- RNN	MIT	No	RNN	No	Daily life activity	Accuracy RMSE	90.85%
(Pus and Shr, 2022) CNN-GRU	UCI-HAR	Fourier Transform	1D CNN+ GRU	CNN	Daily life activity	Accuracy	96.79%
(Rad et al, 2018) MA-DNN MA-CNN	STISEN GAIT Sleep-Stage Indoor/Outdoor detection	Fast Fourier+ Transformation (FFT)+ Empirical Cumulative Distribution Function (ECDF)	1D CNN	CNN	Locomotion Factory Dailt activity Indoor/Outdoor detection	Fl score	81.6% 89.5% 66.4% 82.3% for respective dataset
(Rav et al, 2016) CNN	ActiveMiles WISDM Skoda FoG	No	1D CNN	handcrafted	Locomotion activity Factory maintenance	Accuracy	95.1% 98.2% 95.9% 91.5%
(Sey et al, 2018) CAE	Private data collected by Radar	Synthetic Minority Oversampling Technique (SMOTE)	1D CNN+ AE	CNN	Locomotion activity	Accuracy	94.2%
(Ron and Sun, 2016) Convnet	Private data by smartphone	No	1D CNN	CNN	Locomotion activity	Accuracy	94.8%
(Sin et al, 2017) ID CNN	Kasteren	No	1D CNN	CNN	Three Houses activity	Accuracy	95.3% 86.8% 86.23% for 3 houses
(Tan et al, 2020) Lego CNN	UCI-HAR WISDM, PAMAP2	Sliding Window technique	1D CNN	Lego filters	Locomotion Activity	Accuracy F1 score	96.9% 98.82% 92.79%

TABLE 2.3 – The table shows the related works

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

TABLE 2.4 – The table shows the related works

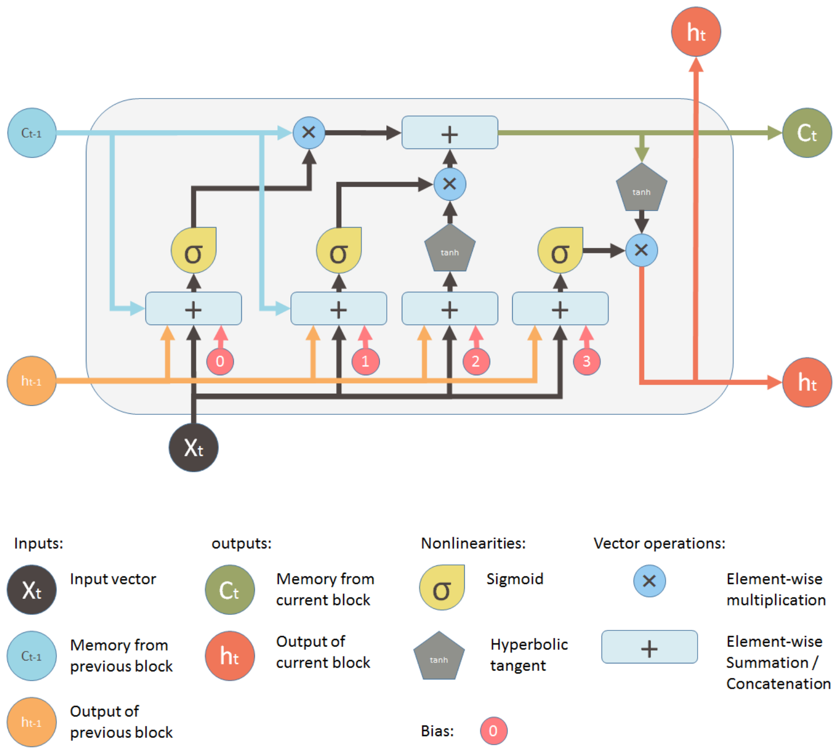

FIGURE 2.14 – structure details of LSTM [12]

Discussion

Upon reviewing the literature, it is evident that researchers and scientists have made considerable advancements in the field of HAR (Human Activity Recognition) systems. However, there are still opportunities for further improvement and progress. A comparative analysis reveals that enhancements in HAR are being pursued from various angles and approaches.

Significant advancements have been made by utilizing deep learning techniques such as CNN, LSTM, and their hybridization, leading to improved results. Hybrid models combining CNN and LSTM approaches have demonstrated high accuracy on the UCI-HAR dataset, with reported percentages of 97.70%, 97.05%, and 96.79% in works by (Abd et al. 2020), (Naf et al. 2021), and (Pus and Shr 2022), respectively. It is noteworthy that CNN excels at extracting key features, while LSTM effectively captures long-term time dependencies. As a result, hybrid models have garnered substantial interest among researchers.

Furthermore, the choice of datasets plays a significant role in shaping the characteristics of activities performed by specific users in particular environments. Various datasets, such as WISDM, UCI, Opportunity, Fog, and Skoda, exhibit distinct activity patterns. For instance, activities in a smart home environment differ from those in a smart factory setting. Notably, the accuracy of activity recognition models varies across different datasets. In the study conducted by (Mur and Jae 2017), the accuracy levels were observed to be 96.7% for UCI-HAR, 97.8% for USC-HAD, 92% for Opportunity, 93% for Fog, and 92.6% for the Skoda dataset.

Hence, the factors of what, who, and where in HAR are crucial in determining the appropriate deep learning method that can effectively predict the true activity and align with the available computational resources.

2.7 Conclusion

This chapter is structured into four main sections. Firstly, we delve into the concept of human activity and provide a concise overview of intelligent environments and IoT devices. We define and explain relevant terms and explore environmental sensors. Additionally, we present various architectures of deep learning models. In the following section, we will examine research studies and visualizations pertaining to the dataset

CHAPITRE 3

METHODOLOGY

3.1 Introduction

This chapter presents the methodology employed in this research to achieve the objectives of training and comparing six different models for human activity recognition using a dataset collected from environmental sensors. It outlines the research design, data collection process, preprocessing techniques, model selection, feature extraction, and model evaluation.

3.2 Research Design

The research design for this study follows a systematic and empirical approach. It involves the collection of a dataset from environmental sensors, such as temperature, humidity, movement, and sound sensors...etc. The dataset was collected over a period of 4 weeks, where sensor readings were recorded only when there was a change in sensor values. The data collection process aimed to capture significant changes in the environment related to human activities Figure 3.1.

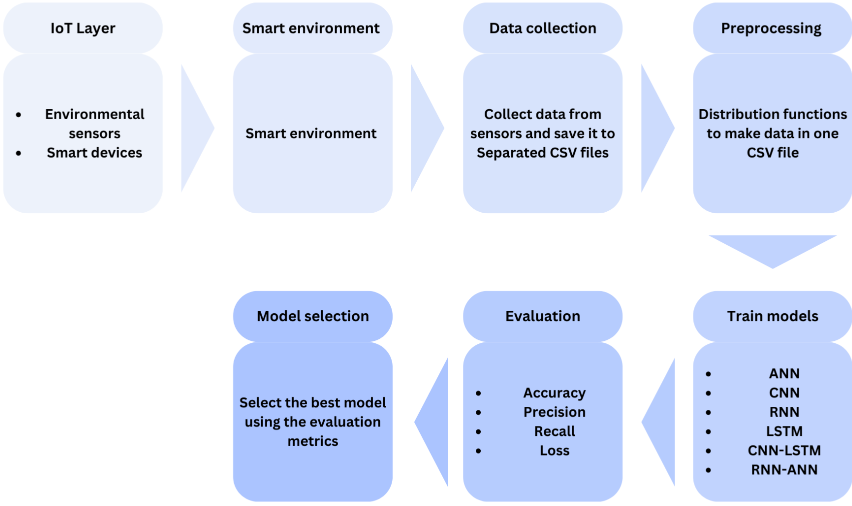

FIGURE 3.1 – General architecture.

3.3 Data Collection

The data collection process involved installing the environmental sensors in a apartment as you can see in Figure 3.2 and recording sensor values whenever changes occurred. The date corresponding sensor values were saved for each sensor separately in CSV files. This approach ensured that only relevant data points were captured, reducing redundant and unnecessary data.

Location	Activity
Entrance	Entering, Leaving
Kitchen	Preparing, Cooking, and Washing the dishes
Living Room	Eating, Watching TV, Computing
Toilet	Using the toilet
Staircase	Going up, Going down
Walkway	(no activity specific to this place
Bathroom	Using the sink, Using the toilet, Showering
Office	Computing, Watching TV
Bedroom	Dressing, Reading, Napping
Common to all places Cleaning	Cleaning

TABLE 3.1 – Table of activities

The dataset encompassed 20 classes of varied activities represented in Table 3.1, this dataset is known as the “Orange4Home dataset”, the data collection process spanned 4 weeks, with an occupant in the apartment, and 236 heterogeneous sensors distributed throughout the apartment. In our system, we use

(a) Ground floor.

(b) First floor.

FIGURE 3.2 – The apartment used in data collection

only the 30 most meaningful sensors.

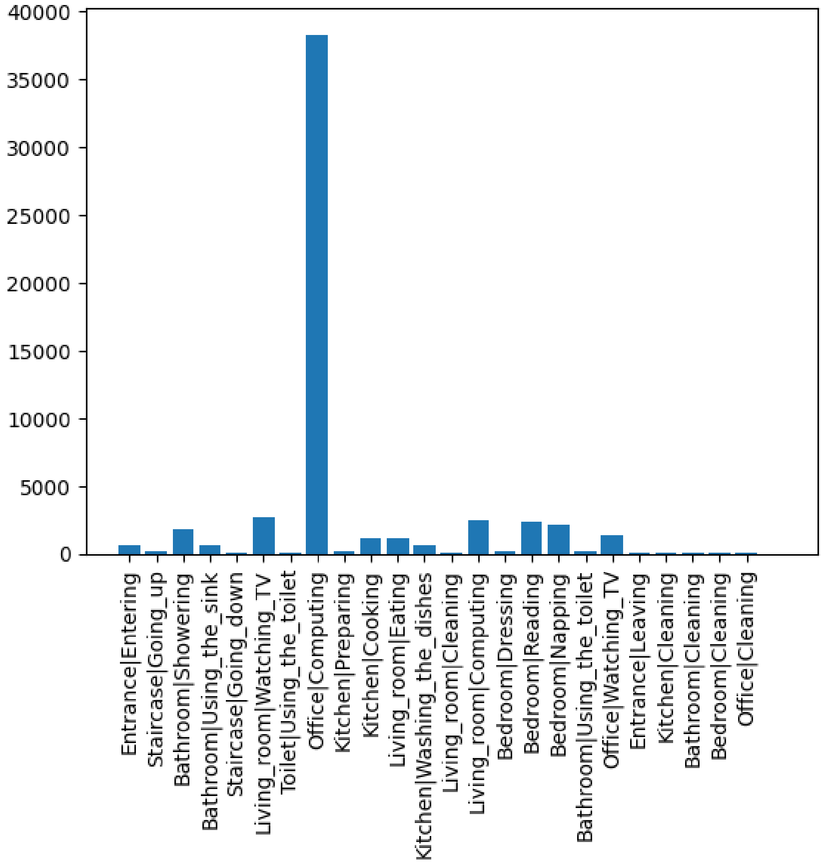

FIGURE 3.3 – Data visualization.

Upon visualizing the dataset based on activity types, it becomes apparent that computing activity accounts for approximately 69% of the total as we can see in Figure 3.3.

3.4 Preprocessing Techniques

In the preprocessing stage, the collected data from the different sensor files were consolidated into a single CSV file see Figure 3.4. This file contained a "Date" column that captured values for each second during the entire four-week period. The other columns in the file represented the sensor readings for temperature, humidity, movement, and sound. This consolidation allowed for a unified dataset with synchronized timestamps for further analysis using our functions see Figure 3.4.

	A	B
1	data	bedroom_luminosity
2	2017-01-30 12:55:38	98.96
3	2017-01-30 12:55:39	202.88
4	2017-01-30 12:55:41	268
5	2017-01-30 12:55:42	293.92
6	2017-01-30 12:55:55	308
7	2017-01-30 12:55:56	296
8	2017-01-30 12:56:00	309.92
9	2017-01-30 12:56:01	320.96
10	2017-01-30 12:56:02	334.72

Figure 3.4 – CSV file for bedroom luminosity sensor records.

```
from datetime import datetime
from numpy.lib.function_base import append
df120=pd.read_csv('/content/drive/MyDrive/articalDATA/120.csv')
df120['bedroom_luminosity'][1]
bedroom_luminosity = []
j = 0
val = 0.0
for i in range(len(df['data'])):
  bedroom_luminosity.append(val)
  if(j>len(df120)):
    break
  if((datetime.strptime(df['data'][i], '%Y-%m-%d %H:%M:%S') <= datetime.strptime(df120['data'][j], '%Y-%m-%d %H:%M:%S'))):
    continue
  elif((datetime.strptime(df['data'][i], '%Y-%m-%d %H:%M:%S') > datetime.strptime(df120['data'][j], '%Y-%m-%d %H:%M:%S'))):
    if(j < len(df120) - 1):
      j += 1
    val = (df120['bedroom_luminosity'][j])


df['bedroom_luminosity'] = bedroom_luminosity
```

Figure 3.5 – A function to distribute the values of bedroom luminosity to the data frame.

3.5 Model Selection

Six different models were selected for training and comparison in this study. The model selection process was based on their popularity and demonstrated performance in human activity recognition tasks. The chosen models include ANN,CNN,RNN,LSTM,CNN-LSTM and RNN-ANN, each with its own unique algorithms and architectures. The selection of diverse models enabled a comprehensive analysis of their performance and comparison.

3.6 Feature Extraction

To extract relevant features from the consolidated dataset, feature extraction techniques were applied. These techniques aimed to capture the distinctive patterns and characteristics of different human activities. Features such as mean, standard deviation, and frequency domain features were extracted from the sensor readings to provide meaningful representations of the activities performed in the environment.

3.7 Model Training and Evaluation

The selected models were trained using the preprocessed dataset. During the training phase, hyperparameter tuning was conducted to optimize the models' performance and prevent overfitting. To evaluate the models' effectiveness in recognizing human activities, the dataset was split into training and testing sets. The models were then evaluated using various performance metrics, including accuracy, precision, and recall. Computational efficiency and resource requirements were also considered to assess the models' practical feasibility.

3.8 Ethical Considerations

Ethical considerations were taken into account throughout this research. Privacy and consent of individuals in the environment where the sensors were deployed were respected. Data anonymization techniques were applied to protect the privacy of individuals. The research adhered to ethical guidelines, ensuring compliance with relevant regulations and institutional policies.

3.9 conclusion

This chapter presented the methodology employed in this research, which involved the collection of a dataset from environmental sensors and the subsequent preprocessing, model selection, feature extraction, and model training and evaluation. The research design aimed to capture significant changes in sensor values related to human activities. The methodology outlined in this chapter provides the foundation for the subsequent chapter, where the results and analysis of the study are discussed.

CHAPITRE 4

IMPLEMENTATION AND EVALUATION

4.1 Introduction

This chapter consists of two parts. The first part focuses on the implementation process, where we discuss the design of the framework and highlight the various components involved, including the development environment and libraries utilized. In the second part, we present and analyze the results obtained from each model, aiming to identify the most effective model for the task at hand.

4.2 Implementation

In this section, we will explore the details of the development environment and the programming language utilized for implementing our system. Furthermore, we will delve into the training and testing procedures, the dataset used[14], and provide comprehensive insights into the architecture we have designed.

4.2.1 Development Environment

Google Colab :

Colaboratory, commonly referred to as "Colab," is a web-based platform developed by Google, enabling users to write and run Python code. It is a free service provided by Google, built on the Jupyter Notebook framework. Colab is specifically designed for machine learning training and research purposes. Notably, one of its significant advantages is the capability to train machine learning models directly in the cloud. Colab offers a range of features, including :

- • 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.

4.2.2 Programming Language and Libraries

In this section, we will provide an introduction to the Python programming language and the libraries that were used to implement the human activity recognition (HAR) models.

Python

Python has become immensely popular among computer scientists and is now considered the most widely used programming language. Its popularity spans across various domains, including infrastructure management, data analysis, and software development. Python's appeal lies in its capability to empower developers to concentrate on their tasks without being burdened by implementation complexities. Unlike previous languages, Python frees developers from strict syntax limitations, allowing them to write code more efficiently. As a result, Python provides a faster development experience compared to other languages.

Libraries used

- **TensorFlow :**The TensorFlow library was used to define the core components for all of our architectures. Its primary purpose is to implement machine and deep learning algorithms. In addition, it offers a high degree of adaptability in its application to create neural networks[15].

- **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[15].

- **NumPy :** We utilized this library to adapt the input types according to the configurations of the employed models, which were specifically designed to handle multidimensional arrays or matrices, along with mathematical functions that operate on these arrays. This package was particularly employed for window extraction and image scanning purposes[16].

- **Sklearn :** The sklearn library, one of the most advantageous ML packages in Python, provides numerous powerful techniques for machine learning and statistical modeling. These techniques include dimensionality reduction, regression, classification, and clustering [18].

- **Pandas** : is a data analysis open-source and processing tool developed in the language of Python [19]. It is flexible, powerful, fast, and simple to use.

The data preprocessing will be include :

- Consolidate the data collected : put the CSV sensor files into one CSV.
- Remove records when there is no activity.
- Encode the string features from the data.
- Split the data.
- Reshape the data.

4.2.3 Build models

We have categorized the models into two groups : simple models and hybrid models. In this section, we will present a comprehensive overview of these models.

4.2.3.1 Simple models :

ANN :

```
model = Sequential()
model.add(Dense(64, activation='sigmoid', input_dim=X_train.shape[1]))
model.add(Dropout(0.2))
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(256, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(24, activation='softmax'))
```

(a) Model ANN

Model: "sequential_6"

Layer (type)	Output Shape	Param #
dense_8 (Dense)	(None, 64)	1920
dropout_3 (Dropout)	(None, 64)	0
dense_9 (Dense)	(None, 512)	33280
dropout_4 (Dropout)	(None, 512)	0
dense_10 (Dense)	(None, 256)	131328
dropout_5 (Dropout)	(None, 256)	0
dense_11 (Dense)	(None, 24)	6168

Total params: 172,696
Trainable params: 172,696
Non-trainable params: 0

(b) Structure of ANN model

FIGURE 4.1 – Simple Building Artificial Neural Networks

CNN :

```
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=3, activation='relu', input_shape=(29,1)))
model.add(MaxPooling1D(pool_size=2))
model.add(Conv1D(100, kernel_size=3, activation='relu'))
model.add(Dropout(0.2))
model.add(MaxPooling1D(pool_size=2))
model.add(Conv1D(100, kernel_size=3, activation='relu'))
model.add(Dropout(0.2))
model.add(Flatten())
model.add(Dropout(0.2))
model.add(Dense(100, activation='relu'))
model.add(Dense(24, activation='softmax'))
```

(a) Model CNN

Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 27, 64)	256
max_pooling1d (MaxPooling1D)	(None, 13, 64)	0
conv1d_1 (Conv1D)	(None, 11, 100)	19300
dropout (Dropout)	(None, 11, 100)	0
max_pooling1d_1 (MaxPooling 1D)	(None, 5, 100)	0
conv1d_2 (Conv1D)	(None, 3, 100)	30100
dropout_1 (Dropout)	(None, 3, 100)	0
flatten (Flatten)	(None, 300)	0
dropout_2 (Dropout)	(None, 300)	0
dense (Dense)	(None, 100)	30100
dense_1 (Dense)	(None, 24)	2424

Total params: 82,180
Trainable params: 82,180

(b) Structure of CNN model

FIGURE 4.2 – Simple Building Convolutional Neural Networks

RNN :

```
model = Sequential()
model.add(SimpleRNN(50, activation='sigmoid', input_shape=(29, 1)))
model.add(Dense(100, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(100, activation='relu'))
model.add(Dense(50, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(24,activation='softmax'))
```

(a) Model RNN

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 simple_rnn_1 (SimpleRNN)    (None, 50)                2600

 dense_3 (Dense)             (None, 100)               5100

 dropout_2 (Dropout)         (None, 100)               0

 dense_4 (Dense)             (None, 100)               10100

 dense_5 (Dense)             (None, 50)                5050

 dropout_3 (Dropout)         (None, 50)                0

 dense_6 (Dense)             (None, 24)                1224

=================================================================
Total params: 24,074
Trainable params: 24,074
Non-trainable params: 0
_________________________________________________________________
```

(b) Structure of RNN model

Figure 4.3 – Simple Building Recurrent Neural Networks

LSTM :

```
model = Sequential()
model.add(LSTM(50, activation='sigmoid', input_shape=(29, 1)))
model.add(Dropout(0.2))
model.add(Dense(100, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(100, activation='relu'))
model.add(Dense(24,activation='softmax'))
```

(a) Model LSTM

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 50)                10400

 dropout (Dropout)           (None, 50)                0

 dense (Dense)               (None, 100)               5100

 dropout_1 (Dropout)         (None, 100)               0

 dense_1 (Dense)             (None, 100)               10100

 dense_2 (Dense)             (None, 24)                2424

=================================================================
Total params: 28,024
Trainable params: 28,024
Non-trainable params: 0
_________________________________________________________________
```

(b) Structure of LSTM model

Figure 4.4 – Simple Building Long Short-Term Memory

4.2.3.2 Hybrid Model :

CNN-LSTM :

```
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=5, activation='sigmoid', input_shape=(29, 1)))
model.add(Conv1D(filters=64, kernel_size=5, activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(Dropout(0.2))
model.add(Conv1D(filters=128, kernel_size=5, activation='relu'))
model.add(Conv1D(filters=128, kernel_size=5, activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(Dropout(0.2))

model.add(LSTM(100, activation='relu'))
model.add(Dropout(0.2))
#model.add(LSTM(100, activation='relu'))
model.add(Dropout(0.2))
model.add(Flatten())
model.add(Dense(24,activation='softmax'))
```

(a) Model hybrid CNN-LSTM

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_5 (Conv1D)           (None, 25, 64)            384

 conv1d_6 (Conv1D)           (None, 21, 64)            20544

 max_pooling1d_3 (MaxPooling  (None, 10, 64)           0
 1D)

 dropout_5 (Dropout)         (None, 10, 64)            0

 conv1d_7 (Conv1D)           (None, 6, 128)            41088

 conv1d_8 (Conv1D)           (None, 2, 128)            82048

 max_pooling1d_4 (MaxPooling  (None, 1, 128)           0
 1D)

 dropout_6 (Dropout)         (None, 1, 128)            0

 lstm_2 (LSTM)               (None, 100)               91600

 dropout_7 (Dropout)         (None, 100)               0

 dropout_8 (Dropout)         (None, 100)               0

 flatten_1 (Flatten)         (None, 100)               0

 dense_2 (Dense)             (None, 24)                2424

=================================================================
Total params: 238,088
Trainable params: 238,088
Non-trainable params: 0
_________________________________________________________________
```

(b) Structure of CNN-LSTM hybrid model

FIGURE 4.5 – Hybrid building Convolutional Neural Networks and Long Short-Term Memory

RNN-ANN :

```
# Create the Sequential model
model = Sequential()

# Add a SimpleRNN layer with 50 units, sigmoid activation, and input shape (timesteps, features)
model.add(SimpleRNN(50, activation='sigmoid', input_shape=(29, 1)))
model.add(Dense(100, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(100, activation='relu'))
model.add(Dense(100, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(50, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(24, activation='softmax'))
```

(a) Model hybrid RNN-ANN

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 simple_rnn (SimpleRNN)      (None, 50)                2600

 dense (Dense)               (None, 100)               5100

 dropout (Dropout)           (None, 100)               0

 dense_1 (Dense)             (None, 100)               10100

 dense_2 (Dense)             (None, 100)               10100

 dropout_1 (Dropout)         (None, 100)               0

 dense_3 (Dense)             (None, 50)                5050

 dropout_2 (Dropout)         (None, 50)                0

 dense_4 (Dense)             (None, 24)                1224

=================================================================
Total params: 34,174
Trainable params: 34,174
Non-trainable params: 0
```

(b) Structure of RNN-ANN hybrid model

FIGURE 4.6 – Hybrid building Recurrent Neural Networks and Artificial Neural Networks

4.3 Evaluation and comparisons

We will evaluate the models on a range of evaluations, including :

— **- Accuracy = the correct classification number/the entire classification number.**

— **- Recall=the correct classification number/the correct classification number+The model predicts it true, but in the negative class**

— **- Prescision= the correct classification number/the correct classification number+The model predicts that she is in the positive category, but she is actually in the negative category.**

4.4 Results

Once all the HAR models were trained, we obtained a set of results. These results have been compiled and organized in the following table :

Models	**Accuracy**	**Recall**	**Precision**	**Loss**
ANN	0.92	0.90	0.95	0.24
CNN	0.93	0.93	0.96	0.21
RNN	0.83	0.75	0.97	0.5
LSTM	0.67	0.67	0.67	1.44
CNN-LSTM	0.95	0.94	0.97	0.16
RNN-ANN	0.67	0.67	0.67	1.44

TABLE 4.1 – A table showing the results of the models

Based on the information presented in the table, it can be concluded that the CNN-LSTM model yields the best results.

We will provide some visualization of the CNN-LSTM model :

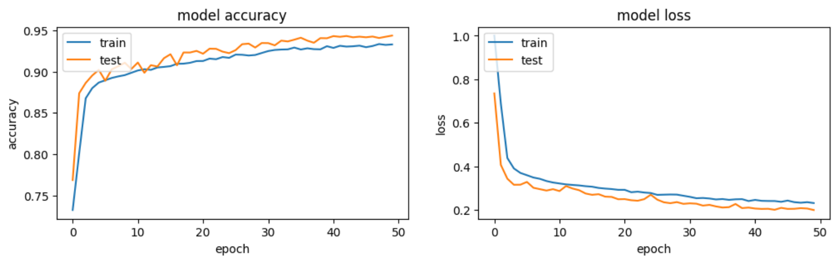

FIGURE 4.7 – Visual Accuracy and Loss Results

The chart on the left illustrates the increasing accuracy, while the chart on the right depicts the decreasing loss. It is worth noting that the training loss rate is higher than the validation loss rate. The verification accuracy is nearly identical to the training accuracy. As the number of repetitions increases, the activity recognition accuracy rate for both training and validation steadily converges to one. Additionally, the loss rate gradually approaches zero, indicating a gradual improvement in the accuracy of activity recognition.

The model architecture for CNN-LSTM is as follows :

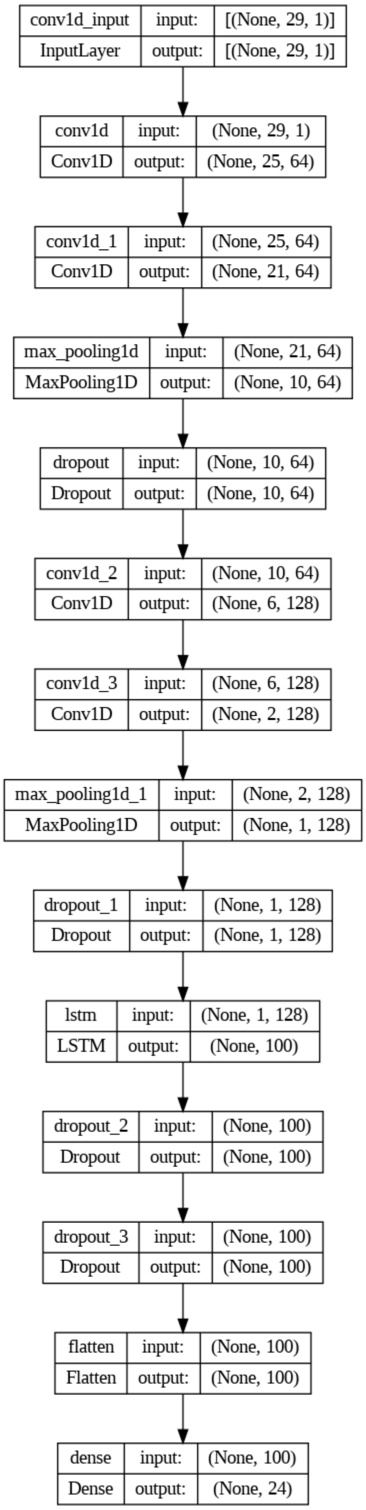

FIGURE 4.8 – Form layers CNN-LSTM

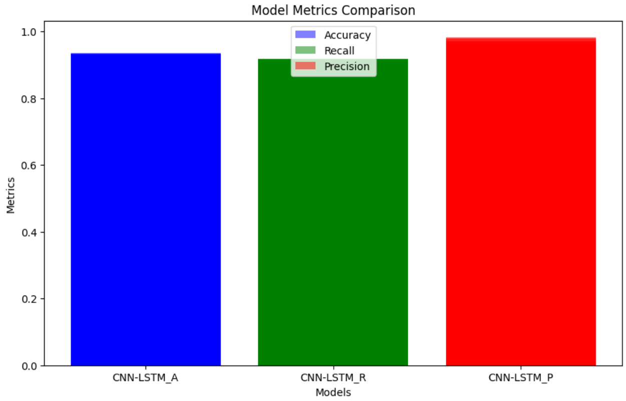

FIGURE 4.9 – Bar charts for model evaluation

4.5 Harnessing Human Activity Recognition for Enhanced Energy Efficiency

Human activity recognition plays a pivotal role in enhancing energy efficiency. By accurately identifying and understanding human behavior, energy consumption patterns can be analyzed and optimized for maximum efficiency. This technology allows for the development of intelligent systems that adapt and respond to human presence, thereby reducing energy waste. Through sensors, cameras, or wearable devices, human activity can be monitored and analyzed in real-time. This information can then be used to automate lighting, heating, and cooling systems, as well as adjust energy usage based on occupancy levels. By incorporating human activity recognition into energy management strategies, organizations, and households can significantly reduce their carbon footprint, lower energy costs, and create a more sustainable future.

4.6 Conclusion

This chapter is divided into two parts. In the first part, we introduced the work environment, programming language, and training and testing procedures. We also provided details on the structures of the different models and presented the results in a table. Based on the obtained results, the hybrid CNN-LSTM model demonstrated the best performance in detecting and identifying human activities in intelligent environments. These results have provided valuable insights and perceptions for further analysis.

CHAPITRE 5

GENERAL CONCLUSION

The Human Activity Recognition (HAR) approach, based on deep learning, holds immense potential for enhancing energy efficiency in various domains and is of significant importance in people's daily lives. It enables a comprehensive understanding of human activities through the integration of IoT sensors into smart environments.

In this thesis, we have emphasized the importance of human activity recognition in intelligent environments and developed several models. Through analysis of the results, we have determined that the CNN-LSTM model surpasses the others, delivering exceptional accuracy and precision. This superiority can be attributed to the CNN's ability to extract features from input data, the LSTM's effectiveness in detecting and recognizing activities in their natural order, and the integration of CNN-extracted features into a feature vector that captures temporal relationships among sensor readings.

This research contributes to the development of human activity recognition systems and provides guidance to researchers, practitioners, and system developers in selecting models and designing systems. By harnessing the capabilities of environmental sensors, accurate detection and classification of human activities can be achieved, leading to improvements in healthcare monitoring, enhancing energy efficiency, and overall comfort in smart environments.

CHAPITRE 6

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., AlAyyoub, 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 al-gorithms : 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). Multiagent 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 contextawareness 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 Research26(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 denois-ing 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 longterm 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 05/03/2023).

[2] july std 2019.pdf (journalstd.com) (Retrieved on 05/05/2023).

[3] Home automation for a smart house - Eco Passive Houses (Retrieved on 02/02/2023).

[4] https://www.cea.fr/comprendre/Pages/nouvelles-technologies/essentiel-sur-domotiquemaison-connectee.aspx (Retrieved on 05/02/2023).

[5] https://www.ucly.fr/wp-content/uploads/2020/02/smart-cities.pdf (Retrieved on 10/02/2023).

[6] https://www.thalesgroup.com/en/markets/digital-identity-and-security/iot/inspired/smartcities (Retrieved on 02/08/2023).

[7] https://www.twi-global.com/technical-knowledge/faqs/what-is-a-smart-city (Retrieved on 10/03/2022).

[8] https://nexusintegra.io/smart-city-what-is-how-it-works/(Retrieved on 10/08/2023).

[9] https://www.smartcityexpo.com/ (Retrieved on 10/04/2023).

[10] https://www.google.com/search?q=+Convolution+Operation&tbm=isch&ved=2ahUKEwi17rj0ue3\-AhXfmycCHViWCEAQ2\-cCegQIABAA&oq=+Convolution+Operation&gs\-lcp=CgNpbWcQAzIFCAAQgAQyBAgAEB4yBAgAEB4yBAgAEB4yBAgAEB4yBAgAEB4yB
sclient=img&ei=WhSgZLXZD9-3nsEP2KyigAQ&bih=625&biw=1349&hl=fr#imgrc=d-0sh50DOHv-AM (Retrieved on 20/04/2023).

[11] https://medium.com/@achoulwar901/the-art-of-convolutional-neural-network-
(Retrieved on 20/05/2023).

[12] https://www.google.com/search?q=structure+details+of+LSTM&tbm=isch&ved=2ahUKEwjY8ovAve3\-AhVMpycCHUkmC2EQ2\-cCegQIABAA&oq=structure+details+of+LSTM&gs\-lcp=CgNpbWcQAzoECCMQJzoGCAAQBxAeOgUIABCABDoECAAQH
-FmDvG2gBcAB4A4ABuQOIAfMIkgEHMC41LjQtMZgBAKABAaoBC2d3cy13aXotaW1nsAEKwAEB&sclient=img&ei=HhigZNjrDMzOnsEPycysiAY&bih=625&biw=1349&hl=fr#imgrc=ohetHbPG39hiFM (Retrieved on 25/05/20223).

[13] https://medium.com/tech-break/recurrent-neural-network-and-long-term-depe (Retrieved on 25/05/2023).

[14] https://gitlab.inria.fr/monitoring/themisrv18smarthome/-/tree/master/case (Retrieved on 10/01/2023)

[15] https://www.tensorflow.org/?hl=fr (Retrieved on 17/01/2023).

[16]https://numpy.org/(Retrieved on 17/01/2023).

[17] https://matplotlib.org/ (Retrieved on 17/01/2023).

[18] https://scikit-learn.org/stable/index.html (Retrieved on 17/01/2023).

[19] https://pandas.pydata.org (Retrieved on 17/01/2023)

[20] https://www.google.com/search?sxsrf=AB5stBjY\-8Zxejb7oXg0qHyK70lUdeJ\-lw:1688210924721&q=Smart+Home&tbm=isch&sa=X&ved=2ahUKEwiKg-Xjs\-3\-AhVBPewKHbIkC34Q0pQJegQIDBAB&biw=1366&bih=625&dpr=1#imgrc=kIFPYUTccTLrDM

[21] https://www.google.com/search?q=smart\%20building&tbm=isch&hl=fr&tbs=rimg:CRbE\-giWrdeQYZs28uXQ9EqCsgIMCgIIABAAOgQIABAAwAIA2AIA4AIA&sa=X&ved=0CBoQuIIBahcKEwiAmurIt\-3\-AhUAAAAAHQAAAAAQDQ&biw=1349&bih=625#imgrc=3OCdQneQodqW6M(25/05/20223)

[22] https://www.google.com/search?q=smart+cities&tbm=isch&ved=2ahUKEwj5o4WquO3\-AhXNpicCHTvvAjgQ2\-cCegQIABAA&oq=smart+cities&gs\-lcp=CgNpbWcQARgAMgQIIxAnMgUIABCABDIFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQyBQg sclient=img&ei=sRKgZLmHMM3NnsEPu96LwAM&bih=625&biw=1349&hl=fr#imgrc=a1CPBv-6XFBGvM(25/05/20223)

[23] https://www.google.com/search?sxsrf=AB5stBiZX0V8nrmcUHt2kpPQCZS9U5B02g:1688212964254&q=Maximal+pooling+and+Average+pooling&tbm=isch&sa=X&ved=2ahUKEwjrlqiwu\-3\-AhV2zgIHHcDUATwQ0pQJegQIDBAB&biw=1366&bih=625&dpr=1#imgrc=pAN7qxILmodWJM(22/05/20223)

[24] https://www.google.com/search?q=Generic\%20data\%20collection\%20architecture\%20for\%20human\%20activity\%20recognition\%20(HAR)&tbm=isch&tbs=rimg:CY\-Ww7YhRPQmYeuEwsh\-eQxYsgIOCgIIABAAOgQIABAAQAHAAgDYAgDgAg hl=fr&sa=X&ved=0CBoQuIIBahcKEwi4rKzGyO3\-AhUAAAAAHQAAAAAQBw&biw=1349&bih=625#imgrc=THnQG62fKa8ZUM&imgdii=KJ9iSdSOtxgoQM(22/05/20223)

[25] https://www.google.com/search?q=RNN+architecture.jpg&tbm=isch&
ved=2ahUKEwjVl_Pn7_D_AhUEpycCHWzFBhoQ2-cCegQIABAA&oq=RNN+architecture.
jpg&gs_lcp=CgNpbWcQAzoECCMQJzoGCAAQBxAeOgQIABAeOgYIABAFEB46BggAEAgQHlDuCljlKGDz
sclient=img&ei=ht-hZNW8KYTOnsEP7Iqb0AE&bih=657&biw=1349&hl=fr#imgrc=
evVAtPGPg2pJpM(22/05/20223)