

**A FRAMEWORK FOR HUMAN ACTIVITY AND  
BEHAVIOURAL PATTERN RECOGNITION IN  
MULTIMODAL SENSOR SMART HOME  
ENVIRONMENT**

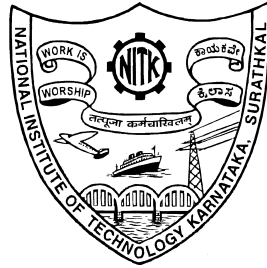
**Thesis**

Submitted in partial fulfillment of the requirements  
for the award of the degree of

**DOCTOR OF PHILOSOPHY**

by

**Ranjit Kolkar**



DEPARTMENT OF INFORMATION TECHNOLOGY  
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA  
SURATHKAL, MANGALORE - 575 025

February, 2024

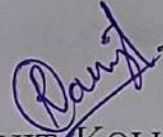


# DECLARATION

I hereby declare that the Research Thesis entitled “A Framework for Human Activity and Behavioural Pattern Recognition in Multimodal Sensor Smart Home Environment”, which is being submitted to National Institute of Technology Karnataka, Surathkal in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy in Information Technology is a bonafide report of the research work carried out by me. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

Place : NITK - Surathkal

Date : February 29, 2024



RANJIT KOLKAR

Reg. No. : 187059IT002,

Department of IT,

NITK Surathkal.




# CERTIFICATE

This is to certify that the Research Thesis entitled, "**A Framework for Human Activity and Behavioural Pattern Recognition in Multimodal Sensor Smart Home Environment**", submitted by **RANJIT KOLKAR** (Reg. no. **187059IT002**), as the record of research work carried out by him, *is accepted as the Research Thesis* submission in partial fulfilment of the requirements for the award of the degree of **Doctor of Philosophy in Information Technology**.

Place : NITK - Surathkal

Date : February 29, 2024



DR. GEETHA V

Research Guide,  
Associate Professor,  
Department of IT,  
NITK Surathkal.



Chairman DRPC,

Department of IT,  
NITK Surathkal.

**CHAIRMAN - DRPC**  
Department of Information Technology  
NITK Surathkal, Srinivasnagar P.O.  
Mangaluru 575 025, INDIA



*I dedicate this thesis to my wife, children,  
parents, siblings and all who were consistent in  
their **Behaviour Patterns** throughout this  
journey.*



# Acknowledgements

I take the privilege to express my heartfelt gratitude to all the people who have constantly supported me throughout my doctoral research. I sincerely thank my research guide, Dr. Geetha V, Department of Information Technology, NITK Surathkal, for her valuable guidance, enthusiasm, inspiration, and dedication throughout my research. This research work results from her being open to the most out-of-the-box ideas, timely suggestions, and firm decisions. I extend my gratitude to Dr. Sidney Rosario for allowing me to relearn the basics and stay updated with the technology.

I sincerely thank the RPAC members, Dr. Pushparaj D. Shetty and Dr. Sowmya S. Kamath, for their apt and timely suggestions and valuable reciprocations. I would also like to thank all the teaching, technical, administrative, and non-teaching staff who have been very kind and helpful throughout my research. Further, I extend my gratitude to Prof. G Ram Mohan Reddy, Department of IT, NITK, who motivated me to research at NITK Surathkal.

I also thank my parents and family for their exhaustive encouragement and support. I extend my special thanks to my wife, Mrs. Deepa, for her cooperation, love, suggestions and constant encouragement at every point in time.

I extend my gratitude to my dear ones Dr. Ashwin T. S., Dr. Shashank Shetty, Dr. Karthik K., Dr. Sanket Salvi, Dr. Natesha, Dr. Gokul K., Dr. Sanjay B., Dr. Manjunath V., Mr. Pragnesh and my other co-researchers who made my journey more enjoyable. Never to forget, the most exciting place for doctoral research with updated technologies usable for research and advanced resources made readily available, facilitating a convenient study, my heartfelt gratitude to NITK Surathkal for making this research memorable and possible.

*Ranjit Kolkar*



# Abstract

Human Activity Recognition ([HAR](#)) has become a subject of significant interest due to its potential applications in various fields, including healthcare, sports, and user profiling. There are four main types of sensor-based HAR: wearable, ambient, camera, and hybrid sensor-based recognition. Smartphones, with their built-in sensors, have emerged as valuable tools for [HAR](#), and other sensors like Passive Infrared ([PIR](#)), load sensors, smart switches, and smartwatches are extensively used in HAR systems along with vision-based sensors. Despite advancements, accurately recognizing human activities remains challenging due to the complexity and diversity of sensors used and the intricate nature of human activities. Each sensor type has advantages and limitations, making selecting appropriate sensors a challenging task requiring a comprehensive understanding of their characteristics. While there are existing applications of [HAR](#), there are still significant opportunities to address various challenges. This work addresses several challenges in improving recognition efficiency, integrating multimodal sensors, achieving synchronization between heterogeneous sensors, collecting long-hour data using these sensors, and developing a cost-effective framework for human activity recognition and behavioral patterns in the daily life of an elderly person.

The thesis work addresses the challenges and develops a framework for [HAR](#) and behavioural pattern recognition using multimodal sensors in a smart home environment. First, we design and develop a deep learning-based solution to recognize the activities based on sensors present in the smartphone. Later, we create and curate a dataset for long-hour human activities in a multimodal sensor-equipped smart home environment and follow to design and develop a human behavioural pattern recognition system in a smart home environment.

The first work focuses on comparing the performance of various deep learning models Convolutional Neural Networks ([CNN](#)), Long Short-Term Memory ([LSTM](#)), and Gated Recurrent Unit ([GRU](#)) for [HAR](#) using smartphone-based sensors. The study explored various datasets and recognition models, providing valuable insights into the overall HAR architecture. The primary objective of

this research is to accurately recognize basic human activities such as walking, sitting, standing, going upstairs, going downstairs, and lying down. The models were trained and evaluated on well-known datasets like Wireless Sensor Data Mining ([WISDM](#)) and University of California, Irvine, Human Activity Recognition ([UCI-HAR](#)). Through rigorous experimentation, the performance of the models on these datasets was significantly improved using the GRU model, laying the foundation for the subsequent research objectives. Additionally, the thesis proposed a novel approach called Spider Monkey Optimization ([SMO](#))-based deep neural network to enhance HAR's accuracy and precision further. The proposed system was evaluated on various datasets involving similar activities, including [UCI-HAR](#), [WISDM](#), Royal Institute of Technology ([KTH](#)) action, and Physical Activity Monitoring using Accelerometers, Gyroscopes, and Magnetometers ([PAMAP2](#)). The optimization improved the performance and the reduced training time, making it practical for real-world applications.

The second work in the thesis involves the collection of long-hour datasets using a multimodal approach. It has been observed from the literature and our previous work that understanding human behaviour patterns solely based on basic activities and smartphone sensors is challenging. Therefore, in this work, we combined smartphone sensor data with ambient sensors to better understand the user's context. The context includes room occupancy detection using [PIR](#) sensors, water bottle level indication using load sensors, and monitoring the status of the TV, bathroom lights, and mirror bulb lights using smart switches. We derived a broader range of activities beyond the basic ones by combining and proposing a hybrid sensor-based data collection approach for two individuals over an extended period.

The third work in the thesis also proposes a novel priority-based labelling technique for data segmentation to retain user context while labelling. This enhanced dataset enables us to gain valuable insights into human behaviour patterns in day-to-day life. Additionally, through a comprehensive analysis of user data, we can derive the user's personality and provide feedback on their behaviour patterns to improve or analyze activities performed over time. The research identifies various applications, such as elderly monitoring systems, personality identification, and behaviour analysis, all aimed at improving health and well-being.

**KEYWORDS:** [HAR](#), [SMO](#), Wearable sensors Smartphone sensors, Deep learning, Ambient sensors, Internet of Things ([IoT](#)), [PIR](#), Elderly monitoring, User profiling, Behaviour patterns.

# Contents

List of Figures	vii
List of Tables	xi
Abbreviations	xiii
Notations	xv
<b>1 Introduction</b>	<b>1</b>
1.1 History . . . . .	2
1.2 Human Activities in Smart Home . . . . .	3
1.3 General HAR Architecture . . . . .	4
1.4 Wearable Sensors-based HAR . . . . .	6
1.5 Ambient Sensors-based HAR . . . . .	7
1.6 Human Activity Pattern Recognition . . . . .	7
1.7 Applications . . . . .	8
1.8 Motivation . . . . .	10
1.9 Problem Statement . . . . .	10
1.10 Research Objectives . . . . .	11
1.11 Framework for HAR and Behavioural Pattern Identification in a Smart Home Environment . . . . .	12
1.12 Research Highlights . . . . .	16
1.13 Organization of the Thesis . . . . .	16
1.14 Summary . . . . .	18
<b>2 Literature Survey</b>	<b>19</b>
2.1 Sensor-based HAR . . . . .	19
2.1.1 Wearable Sensor-based HAR . . . . .	19
2.1.2 Ambient Sensor-based HAR . . . . .	23
2.1.3 Camera Sensor-based HAR . . . . .	26

2.1.4	Hybrid Sensor-based HAR . . . . .	28
2.2	Datasets used in HAR . . . . .	31
2.3	Patterns in the Daily Activities . . . . .	33
2.3.1	Anomaly Detection in the Activities . . . . .	35
2.4	Research Gaps, Issues and Challenges . . . . .	37
2.5	Summary . . . . .	40
<b>3</b>	<b>Human Activity Recognition using Deep Learning Techniques</b>	<b>41</b>
3.1	Proposed Human Activity Recognition using Various Deep Learning Techniques . . . . .	42
3.1.1	Raw Data and Preprocessing . . . . .	42
3.1.2	Feature Extraction . . . . .	45
3.1.3	Deep Learning Models . . . . .	45
3.1.4	Results and Discussion . . . . .	48
3.2	Proposed Human Activity Recognition using Deep Learning Techniques with Spider Monkey Optimization . . . . .	53
3.2.1	Dataset Description . . . . .	56
3.2.2	Flow of Proposed Deep Neural Network with Spider Monkey Optimization (DNN-SMO) Approach . . . . .	56
3.2.3	Results and Discussion . . . . .	62
3.2.4	Discussion . . . . .	68
3.3	Summary . . . . .	69
<b>4</b>	<b>Proposed Framework for Long-hour Data Collection using Multimodal Sensors</b>	<b>71</b>
4.1	Proposed Basic Activities Recognition using a SmartPhone . . . . .	75
4.1.1	Basic Activities Recongnition . . . . .	76
4.1.2	Model Deployment on Smart Phone . . . . .	78
4.1.3	Implementation . . . . .	80
4.1.4	Result and Discussion . . . . .	81
4.2	Proposed User Context Acquisition using Ambient Sensors . . . . .	85
4.2.1	Proposed Single-Person Occupancy Detection using PIR . . . . .	86
4.2.2	Implementations . . . . .	89
4.2.3	Result and Discussion . . . . .	94
4.2.4	Smart Switches to Detect ON/OFF Status of Appliances . . . . .	99
4.2.5	Drinking Water Analysis using Load Sensors . . . . .	104
4.3	Proposed Technique to Derive High-Level Activities . . . . .	109

4.4	Proposed Priority Labelling . . . . .	117
4.4.1	Effect of Priority Labelling . . . . .	121
4.5	Comparison with Existing Datasets . . . . .	124
4.6	Constraints and Limitations . . . . .	125
4.6.1	Sensors and the Activities . . . . .	127
4.6.2	Privacy . . . . .	129
4.6.3	Adaptibility . . . . .	129
4.7	Summary . . . . .	130
<b>5</b>	<b>Human Behavioural Activity Pattern Recognition</b>	<b>131</b>
5.1	Proposed Work on Human Behavioural Pattern Recognition using Adaptive Granularity . . . . .	133
5.1.1	Methodology . . . . .	134
5.2	Implementation . . . . .	138
5.2.1	Granular Entities . . . . .	139
5.2.2	Groups Creation . . . . .	140
5.2.3	Time Complexity of the Algorithms . . . . .	142
5.3	Results and Discussion . . . . .	143
5.3.1	Ground Truth Data . . . . .	143
5.3.2	Accuracy Comparison . . . . .	144
5.3.3	Limitation of the Algorithms . . . . .	146
5.3.4	Activity Occurances by Interval . . . . .	147
5.3.5	Similarity Between Profiles . . . . .	149
5.4	Proposed Comprehensive Analysis of Human Behavioural Patterns	152
5.4.1	Methodology . . . . .	152
5.5	Implementations and Results . . . . .	161
5.5.1	The Duration based Activity Trend Detection . . . . .	161
5.5.2	Analysing Activity Patterns across Time Intervals . . . . .	162
5.5.3	Exploring Activity Insights: Locations, Durations, Categories, and States . . . . .	165
5.5.4	Sleep Analysis . . . . .	168
5.5.5	Activity Sequences Analysis . . . . .	170
5.6	Summary . . . . .	177
<b>6</b>	<b>Conclusions and Future Work</b>	<b>179</b>
	<b>Publications based on Research Work</b>	<b>182</b>



# List of Figures

1.1	Various Sensor-based HAR Systems with Example. . . . .	2
1.2	General HAR Architecture. . . . .	5
1.3	Proposed Framework for HAR and Behavioural Pattern Identification in a Smart Home Environment . . . . .	14
1.4	Thesis Organization . . . . .	17
2.1	Literature Review Organisation . . . . .	20
3.1	Proposed HAR using Deep Learning Techniques . . . . .	43
3.2	General Working of the LSTM node. . . . .	46
3.3	The Proposed GRU Neural Network for HAR. . . . .	48
3.4	Class Distribution of UCI-HAR Dataset (Mekruksavanich and Jitpattanakul 2021) . . . . .	50
3.5	Class Distribution of WISDM Dataset (Kwapisz <i>et al.</i> 2011a) . . . . .	50
3.6	The Accuracy Comparison Across Deep Learning Models on the UCI-HAR Dataset Activity Wise . . . . .	51
3.7	The Accuracy Comparison Across Deep Learning Models on the WISDM Dataset Activity Wise . . . . .	51
3.8	The Accuracy Comparison Across Deep Learning Models on WISDM and UCI-HAR Dataset . . . . .	52
3.9	Proposed DNN Model with Spider Monkey Optimization. . . . .	54
3.10	Sample Video Frame from ACTION KTH Showing Walking Activity (Schuldt <i>et al.</i> 2004). . . . .	57
4.1	The Proposed Framework for Long-hour Dataset Collection using Multimodal Sensors. . . . .	74
4.2	Proposed Basic Activities Recognition using Smart Phone . . . . .	76
4.3	Process to Collect the Basic Activities in Android Application. . . . .	78

4.4	Sample Screenshot of the Android Application After Deploying the Trained Model. The Sample Shows the Walking Activity is Recognised. . . . .	79
4.5	Confusion Matrix of GRU Model Experimented on WISDM Dataset.	82
4.6	Train and Testing Accuracies and Loss for WISDM Dataset. . . . .	82
4.7	Proposed User Context Acquisition using Ambient Sensors . . . . .	86
4.8	Simulated Circuit Representation of the Proposed Single-Person Occupancy Detection using PIR Sensors . . . . .	90
4.9	Floor Design of the Home in which the Long-hour Data is Recorded.	91
4.10	Samples of the Basic Activities Performed by the Subject. Shows Sleeping, Sitting, StairDown, StairUp, and Walking Activities in the Order from Top Left to Bottom Right . . . . .	92
4.11	Duration of Time Spent by a Person in Each Room. . . . .	98
4.12	Percentage Distribution of Duration in Each Room . . . . .	98
4.13	Smart Switch Connected to Relay Switch to Read the ON/OFF Status of Home Appliances. . . . .	101
4.14	Proposed Technique to Derive High-Level Activities . . . . .	109
5.1	Proposed Work on Human Behavioural Pattern Recognition using Adaptive Granularity . . . . .	135
5.2	The Weekly Profile of a Person based on the Events and Groups. The Group (e1, e2) is Repeating on Five Days ( D1, D2, D4, D6, D7 ). The same Group is Common Between Person1 and Person2 Four Times a Week. . . . .	137
5.3	Activity Occurances by interval . . . . .	148
5.4	Activity Shares by Interval for June 1, 2020 . . . . .	149
5.5	Proposed Comprehensive Analysis of Human Behavioural Patterns.	153
5.6	The Sample Shows the Trend of Sleep Activity based on the Duration over Weeks. . . . .	164
5.7	Hourly Time Spent in Each Location Overall . . . . .	166
5.8	Time Spent in Each Location per Day . . . . .	166
5.9	Distribution of Activity Durations across Different Categories . . . . .	168
5.10	Afternoon Sleep(Left) Durations and Night Sleep(Right) Durations in Hours. . . . .	169
5.11	Scatter plot showing the correlation between night sleep and afternoon sleep durations. . . . .	170
5.12	Weekdays and Weekends Sleep Analysis Histogram . . . . .	171

5.13 Drinking Activity Frequency Patterns of Minimum Consecutive Day of Two Days . . . . .	177
5.14 Drinking Activity Frequency Patterns of Minimum Consecutive Day of Five Days . . . . .	178



# List of Tables

2.1	Summary of Literature Review on Wearable Sensors-based in HAR	23
2.2	Summary of Literature Review on Ambient Sensor-based HAR (ASHAR)	27
2.3	Summary of Literature Review on Camera Sensor-based HAR (CSHAR)	29
2.4	Summary of Sensor Modalities in HAR Systems	32
2.5	Existing Datasets for Human Activity Recognition Based on Wearable and Ambient Sensors	33
2.6	Pattern Mining Techniques in Human Activity Recognition	36
2.7	Anomaly Detection in Human Activity Recognition	37
3.1	The Accuracy Comparison Across Deep Learning Models on WISDM and UCI-HAR Dataset	52
3.2	Notations and Description used in the proposed DNN-SMO.	58
3.3	Accuracy Comparison Across Datasets	64
3.4	Precision Comparison Across Datasets	65
3.5	Recall Comparison Across Datasets	66
3.6	F1-score Comparison Across Datasets	66
3.7	Error Rate Comparison Across Datasets	67
3.8	Computation Time Comparison Across Datasets	68
3.9	Overall Performances Comparison Across Datasets	68
4.1	Accuracy of GRU model on WISDM and WISDM Fusion Data Before Deployment	83
4.2	Accuracy of GRU model on WISDM and WISDM Fusion Data After Deployment	83
4.3	The Performance Metrics for Each Activities.	84
4.4	Experimental to Evaluate the Room Occupancy Using the Proposed Algorithm.	94
4.5	Accuracy for Each Room and Average False Detection	95
4.6	Sample Appliance Usage and Derived Activities	102
4.7	Smart Switch Performance Evaluation	103

4.8	Load Sensor Measurements vs. Known Water Levels . . . . .	108
4.9	Notation Used to Derive the High-Level Activities . . . . .	111
4.10	Derived activities using basic activity and ambient sensors. . . . .	115
4.11	Description of Activity Codes . . . . .	116
4.12	Notation Table . . . . .	119
4.13	Experiment Results: Labelling Comparison . . . . .	122
4.14	Simple case of priority setting for the activities. The smallest number indicates the highest priority. . . . .	122
4.15	Sample instance of labelling for a 10-minute window using 2-minute segmented data using the priority. . . . .	123
4.16	Summary and Comparison of various Studies. . . . .	125
4.17	Comparison of Different Datasets and Activity Recognition Models	126
5.1	The Notation and their description used in the Definitions . . . . .	136
5.2	Accuracy Comparison of FBP Algorithm with Other Algorithms. Apriori (Agrawal and Srikant 1994), FP-Growth (Han <i>et al.</i> 2000), MTK (Chuang <i>et al.</i> 2008), estDec+ (Shin <i>et al.</i> 2014) . . . . .	144
5.3	FBP Identification Accuracy with Different Time Granularities and Intervals. Morning: 06:00-11:00, Afternoon: 11:00- 16:00, Evening: 16:00-21:00, Night: 21:00-06:00 . . . . .	145
5.4	Count of Matched Groups between Person 1 and Person 2 in the morning routine of a day (sample) . . . . .	150

# Abbreviations

<b>ADL</b>	Activities of Daily Living
<b>AG</b>	Activity Graph
<b>AL</b>	Auto Labelling
<b>ACI</b>	Ambient or Context Information
<b>AR</b>	Augmented Reality
<b>ASHAR</b>	Ambient Sensor-based HAR
<b>CAR</b>	Combinatorial Activity Rules
<b>CNN</b>	Convolutional Neural Networks
<b>CASAS</b>	Comprehensive Adult Student Assessment Systems
<b>CSHAR</b>	Camera Sensor-based HAR
<b>DL</b>	Deep Learning
<b>DL-Q</b>	Deep Learning based Q-Framework
<b>DNN</b>	Deep Neural Network
<b>DNN-SMO</b>	Deep Neural Network with Spider Monkey Optimization
<b>ECG</b>	Electrocardiogram
<b>EEG</b>	Electroencephalography
<b>EEPROM</b>	Electrically Erasable Programmable Read-Only Memory
<b>EMG</b>	Electromyogram
<b>FBP</b>	Frequent Behaviour Pattern
<b>FPTree</b>	Frequent Pattern Tree
<b>FP-Growth</b>	Frequent Pattern Growth
<b>GPS</b>	Global Positioning System
<b>GRU</b>	Gated Recurrent Unit
<b>HAR</b>	Human Activity Recognition
<b>HAPR</b>	Human Activity Pattern Recognition

<b>HLA</b>	High Level Activities
<b>HMM</b>	Hidden Markov Models
<b>HVAC</b>	Heating, ventilation, and air conditioning
<b>HSHAR</b>	Hybrid Sensor-based HAR
<b>IMU</b>	Inertial Measuring Unit
<b>IoT</b>	Internet of Things
<b>KTH</b>	Royal Institute of Technology
<b>LLA</b>	Low Level Activities
<b>LSTM</b>	Long Short-Term Memory
<b>PAMAP2</b>	Physical Activity Monitoring using Accelerometers, Gyroscopes, and Magnetometers
<b>PIR</b>	Passive Infrared
<b>RF</b>	Random Forests
<b>ReLU</b>	Rectified Linear Unit
<b>RFID</b>	Radio Frequency Identification
<b>RNN</b>	Recurrent Neural Network
<b>SD</b>	Storage Device
<b>SMO</b>	Spider Monkey Optimization
<b>SVM</b>	Support Vector Machines
<b>UCI-HAR</b>	University of California, Irvine, Human Activity Recognition
<b>WISDM</b>	Wireless Sensor Data Mining
<b>WSHAR</b>	Wearable Sensor-based HAR

# Notations

$A$	Activity code or category
$D$	Data associated with the activity
$T$	Represent the timestamp (time interval) of the activity
$e$	Defined as a tuple of three elements, $e = \langle A, D, T \rangle$ .
$g$	Set of entities with similar activity codes or categories, occurring within a specified time interval and sharing similar data characteristics
$\delta$	Maximum time interval between two consecutive entities in the group
$\lambda$	Number of times the $g$ is repeated in an interval
$AS$	Set of available basic activities and sensor activations
$A$	Set of basic activities
$S$	Set of sensor activations
$\oplus$	Exclusive-OR (XOR) operation
$ik$	Itemset assigned to the $k$ -th activity trace
$AS_k$	$k$ -th activity set representing the combination of basic activities and ambient sensor activations
$C_k$	$k$ -th context set representing room locations, sensor activations, and user-provided manual inputs
$DA$	Set of derived activities (high-level activities)
$ha_k$	Derived activity (high-level activity) assigned to the $k$ -th activity trace
$Tr$	Set of activity traces
$An_k$	Label of the derived activity indicating anomaly or non-anomaly
$An_k(tr_k)$	Anomaly classification function for the $k$ -th activity trace
$\text{Supp}(I^* \Rightarrow har)$	Support of association rule between antecedent $I^*$ and high-level activity $har$
$\text{Conf}(I^* \Rightarrow har)$	Confidence of association rule between antecedent $I^*$ and high-level activity $har$
$d_s$	Individual dataset from three different sources
$SM_{PQ}[d_s(t)]$	Pre-processed dataset
$y(t)$	Sensor input dataset

$x(t - 1)$	Previous time step of the sensor input dataset
$UR(0, 1)$	Random number uniformly allocated in the range $[0, 1]$
$SM_{max(q)}$	Upper bound of the $q^{th}$ search space
$SM_{min(q)}$	Lower bound of the $q^{th}$ search space
$SM_Q$	Unified dataset after combining pre-processed datasets
$U(n, m)$	Convolved dataset for feature selection
$s(m, u, v)$	Features extracted from the unified dataset
$d(s - 1)$	Window size used in the previous step in raw data processing
$a[(x_1, y_1), \dots, (x_n, y_n)]$	Collected dataset representing human actions
$W$	Weight vector in the hyperplane equation
$x_i$	Input vector for the hyperplane equation
$B$	Bias term in the hyperplane equation.
$SM_N(p, q)$	Updated dataset on the local leader (LL) stage
$SM^*$	$SM(P, Q) + UR(0, 1)$
$LL(Q)$	$Q^{th}$ dimension based on the local leader location
$SM(R, Q)$	$Q^{th}$ dimension randomly selected from the local group
$GL(Q)$	Location of the global leader in the $Q^{th}$ dimension
$P(SM)$	Probability determination of the dataset
$F(P)$	Fitness value of the $P^{th}$ dataset
$h_{k_j}$	Output of the $j^{th}$ neuron in the $k^{th}$ classification layer
$a_{k_{ij}}$	Offset of the $j^{th}$ neuron in the $k^{th}$ classification layer
$w_{k_j} - j^{th}$	Row of the weight matrix connecting $(k - 1)^{th}$ and $k^{th}$ classification layers
$h_{l_j}$	Output of the $j^{th}$ neuron in the $l^{th}$ layer (softmax)
$sum$	Running sum during dataset updating
$DU(i, j)$	$i^{th}$ and $j^{th}$ dimension based on dataset updating
$LL(k, j)$	$j^{th}$ dimension based on group location of the $k^{th}$ local leader
$y(j, i)$	Updated dataset based on the fitness function

# Chapter 1

## Introduction

The global population of elderly individuals is rising, with life expectancy projected to increase from 71 to 77 years over the next three decades (Farah *et al.* 2019). This trend is particularly significant in many Asian countries, including India, where the population aged 60 and above is growing at a rate exceeding 20% by 2050 (Ministry of Statistics 2016). With this demographic shift, the need for assistive technology and improved care for the elderly has become increasingly important. Developing long-term care solutions, advanced health assistance, and remote monitoring systems can significantly enhance the quality of life for this growing population.

Advancements in sensor hardware technology have paved the way for various research areas to assist individuals in leading better lives (Carmeli *et al.* 2016). One such area is the development of smart home systems, which address these challenges and strive to improve people's quality of life. Within a smart home environment, personalised health assistance can be provided by deploying [HAR](#) systems.

HAR involves accurately recognising and understanding human activities or actions within their context. In recent years, sensors have gained significant attention and found widespread application in research and real-time scenarios. The increased capabilities of sensors have expanded their applicability in numerous domains. HAR finds applications in sports, interactive games, surveillance, human-computer interaction, rehabilitation, ergonomics, and monitoring the elderly and patients (Guo *et al.* 2016; Terada and Tanaka 2010; Bulling *et al.* 2014; Cornacchia *et al.* 2016).

Traditionally, HAR systems relied on video and image analysis to recognise activities. However, with advancements in sensor and communication technologies, researchers have explored alternative modalities, such as wearable and ambient

sensors, for activity detection (Dempsey *et al.* 2005). These sensor modalities can be categorised into four main types: wearable-based (attached to the body), vision-based (camera-based), ambient-based (environment-attached sensors), and hybrid-based (combination of modalities) (Bulling *et al.* 2014; Cornacchia *et al.* 2016). Figure 1.1 visually represents various sensor-based HAR systems.

By leveraging these sensor modalities, HAR systems can capture and interpret human activities more accurately and efficiently. This enables the development of advanced applications that enhance performance, detect abnormal behaviour, improve rehabilitation outcomes, and ensure proper usage of tools and devices in various contexts.

Overall, the combination of sensor technology, HAR systems, and smart home environments holds excellent potential for improving the quality of life for individuals by providing personalised health assistance and efficient monitoring capabilities.

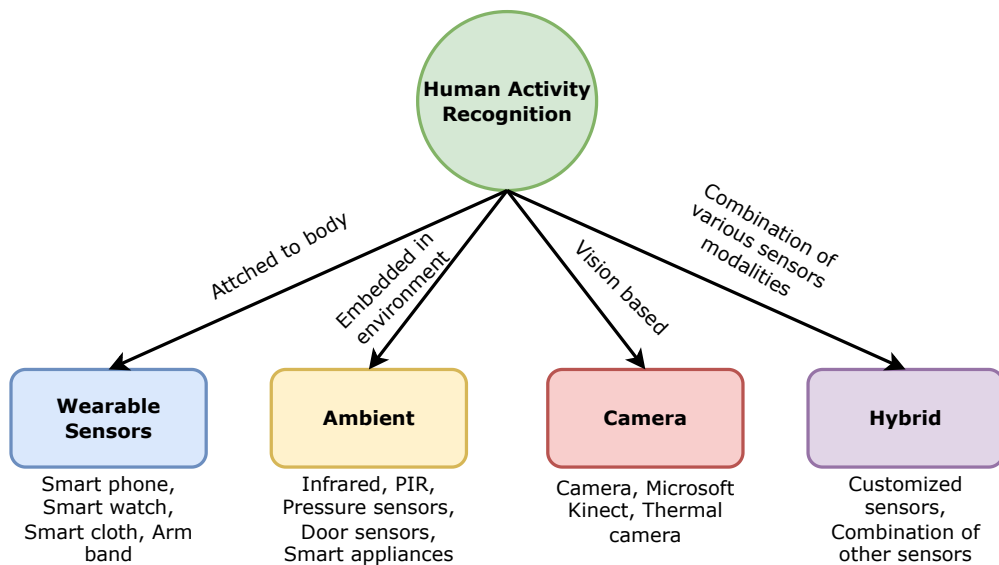


Figure 1.1: Various Sensor-based HAR Systems with Example.

## 1.1 History

Human activity recognition has a rich history dating back to the early 1990s. The field emerged, intending to develop algorithms and systems that automatically recognise and understand human activities based on sensor data. In the early stages, researchers focused on simple activities like walking, running, sitting, and

standing. They utilised wearable sensors such as accelerometers and gyroscopes to capture motion data and applied machine-learning algorithms to classify and recognise these activities.

As sensor technologies advanced and the interest in HAR grew, researchers began exploring new modalities and techniques. Ambient, camera-based, and depth sensors were introduced to capture a more comprehensive view of human activities. The emergence of deep learning techniques, such as CNNs and Recurrent Neural Network (RNN)s, revolutionised HAR by improving accuracy through the automatic learning of hierarchical features and temporal dependencies in sensor data. This led to significant progress in HAR performance and expanded its applications across healthcare, fitness tracking, elderly care, behaviour analysis, and human-computer interaction.

Today, HAR continues to evolve, with ongoing research focusing on novel sensor modalities, fusion techniques, and advanced machine learning algorithms. Integrating emerging technologies like the IoT and Augmented Reality (AR) holds immense potential for creating intelligent systems that can understand and respond to human activities in real-time. The history of HAR demonstrates its significant contributions to various fields and its potential to improve our lives through personalised interventions, anomaly detection, and a deeper understanding of human behaviour patterns.

## 1.2 Human Activities in Smart Home

In the context of smart homes, HAR involves detecting and categorising various activities to enable intelligent and context-aware systems. HAR in smart homes focuses on understanding human behaviour and daily routines to enhance comfort, safety, and energy efficiency. Several activities can be detected and categorised within the smart home context like *Basic, Domestic, Personal Care, Leisure and Entertainment, Sleeping and Resting, Physical Fitness and Social Interaction* activities.

*Basic Activities:* These activities capture fundamental actions that individuals perform daily, such as walking, sitting, standing, and lying down. These activities provide the foundation for higher-level activity recognition.

*Domestic Activities:* Domestic activities encompass tasks related to household chores and routines, such as cleaning, cooking, washing dishes, doing laundry, and vacuuming. Recognising these activities enables automated assistance and optimisation of household tasks.

*Personal Care Activities:* Personal care activities involve grooming, hygiene, and self-care, such as brushing teeth, showering, washing hands, etc. Monitoring these activities can support independent living and assist individuals with specific needs, such as the elderly or individuals with disabilities.

*Leisure and Entertainment Activities:* These activities cover various recreational pursuits, including watching TV, listening to music, playing games, reading, and engaging in hobbies. Recognising leisure activities allows for personalised recommendations, content streaming, and intelligent entertainment systems.

*Sleeping and Resting Activities:* Tracking sleep patterns and monitoring restful periods are essential for understanding individuals' sleep quality and overall well-being. Detecting sleeping activities helps optimise sleep environments, track sleep duration, and provide insights for improving sleep habits.

*Physical Fitness Activities:* Monitoring physical fitness activities, such as exercising, running, cycling, or yoga, contributes to health tracking and personalised fitness guidance. Recognising these activities enables tailored exercise recommendations and progress monitoring.

*Social Interaction Activities:* Social interaction activities involve communication and engagement with others, including making phone calls, video conferencing, receiving visitors, and interacting with smart home voice assistants. Recognising social activities can enhance communication systems and enable social support within the smart home environment.

By detecting and categorising these activities within the context of a smart home, intelligent systems can adapt to individual needs, automate routine tasks, optimise energy consumption, provide personalised recommendations, and enhance the overall living experience.

### 1.3 General HAR Architecture

The general architecture of HAR systems typically consists of several key components that work together to recognise and classify human activities. The Figure 1.2 shows the high-level overview of the general architecture:

1. *Sensor Data Collection:* HAR systems rely on various sensors to capture data about human activities. These sensors can include wearable devices, ambient sensors, cameras, accelerometers, gyroscopes, pressure sensors, or any other relevant sensors capable of capturing relevant activity information.

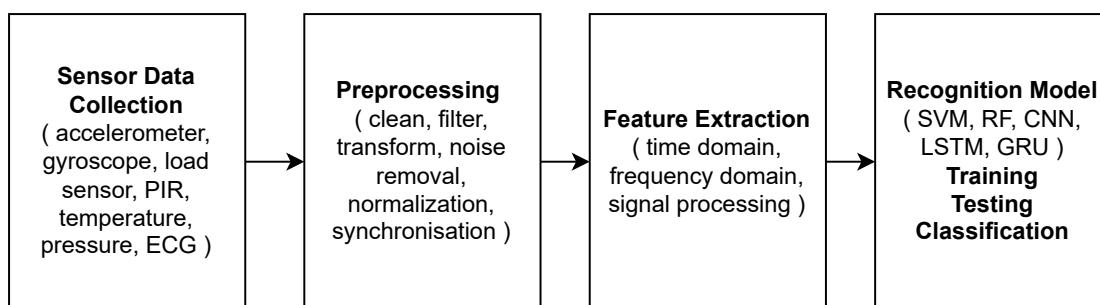


Figure 1.2: General HAR Architecture.

2. *Preprocessing*: The collected sensor data undergoes preprocessing to clean, filter, and transform the raw signals into a suitable format for further analysis. Preprocessing steps may include noise removal, signal normalisation, feature extraction, and data synchronisation.
3. *Feature Extraction*: Meaningful features are extracted from the preprocessed sensor data. Features can be time-domain, frequency-domain, statistical, or more advanced features derived from signal processing techniques or machine learning algorithms. Feature extraction aims to capture relevant information that differentiates various activities.
4. *Activity Recognition Model*: The extracted features are used to train a machine learning or deep learning model for activity recognition. Commonly used models include Support Vector Machines (SVM), Random Forests (RF), CNN, LSTM, GRU or a combination of these models. The model learns the patterns and characteristics of different activities from labelled training data.
5. *Activity Classification*: Once the activity recognition model is trained, it can be used to classify new, unseen sensor data into different activity classes. The model predicts the most likely activity based on the extracted features and learned patterns. The classification output represents the recognised activity.
6. *Post-processing and Decision Making*: Post-processing steps may involve smoothing the activity predictions, applying decision rules or thresholds, and incorporating contextual information to refine the activity recognition results. This step helps improve the accuracy and reliability of the final activity classification.
7. *Output and Feedback*: The recognised activities can be communicated to the user or integrated into smart home systems to trigger appropriate ac-

tions, provide feedback, or generate personalised recommendations based on the detected activities. This feedback loop allows the system to adapt and respond to the user's activities in real-time.

## 1.4 Wearable Sensors-based HAR

Wearable Sensor-based Human Activity Recognition (HAR) is a pivotal component of smart home environments, enabling the monitoring and understanding of occupants' activities through devices worn on their bodies. Wearable sensors offer the advantage of providing real-time and continuous data capture, facilitating more accurate and dynamic activity recognition. When integrated with smart home systems, these devices contribute to a personalised and context-aware living experience.

Wearable devices equipped with accelerometers and gyroscopes detect changes in linear and rotational motion, identifying various basic activities such as walking, running, sitting, and standing, as well as fitness-related activities like cycling and stair climbing. Heart rate monitors and Electrocardiogram (ECG) sensors provide insights into physical exertion and stress, enabling the recognition of fitness activities and varying intensity levels during exercises. Global Positioning System (GPS) sensors in wearables track users' location and movement patterns, recognise outdoor activities like jogging, hiking, or cycling, and provide spatial context within the smart home environment.

Some wearables also incorporate environmental sensors, such as temperature and humidity sensors, adding contextual information to activity recognition and distinguishing between indoor and outdoor activities. Skin conductance sensors offer insights into emotional responses and stress levels, identifying stress-related activities and emotional states. Gesture and motion sensors detect specific hand movements or body gestures, which are valuable for recognising activities like waving or gesturing to control smart home devices.

Smart home systems can continuously monitor occupants' activities by leveraging wearable sensor data, providing valuable insights for personalised services, health monitoring, and behaviour analysis. The real-time nature of wearable sensor-based HAR allows for prompt responses and timely interventions to enhance safety, comfort, and well-being within the smart home environment. As wearables become more integrated into everyday life, their potential for human activity recognition in smart homes will continue to expand, creating a synergistic ecosystem that holds great promise for shaping the future of human-centric

technology applications.

## 1.5 Ambient Sensors-based HAR

Ambient Sensor-based HAR plays a crucial role in smart home environments, providing valuable insights into occupants' daily routines and behaviours without requiring them to wear specific devices. These sensors, alongside wearable devices, collectively contribute to a comprehensive understanding of activities within the smart home setting, enabling a more context-aware and personalised living experience.

Ambient sensors include load sensors, PIR sensors, smart switches, temperature sensors, and light sensors. Load sensors detect weight changes, enabling the recognition of activities like drinking water from a dispenser. PIR sensors detect the presence of individuals in different rooms, allowing for tracking movements and inferring activities based on their locations. Smart switches provide information about the status of home appliances, helping identify leisure and entertainment activities like watching TV.

Temperature sensors monitor changes in room occupancy, and light sensors infer activities based on light patterns. Combining data from these ambient sensors allows the system to make more accurate activity predictions and respond intelligently to the occupants' activities and preferences.

Integrating ambient sensors with wearables and other smart home devices enables a holistic approach to HAR, enhancing activity recognition accuracy and providing seamless, adaptive, and user-centric living experiences. As sensor technology and machine learning algorithms advance, ambient sensor-based HAR will continue to play a pivotal role in creating smarter and more efficient living environments tailored to individual needs. The combination of ambient sensors and wearables represents a promising frontier in human-centric technology applications, contributing to the growth of smarter and more intuitive smart home systems.

## 1.6 Human Activity Pattern Recognition

Human activity pattern recognition is a field of study that focuses on understanding and analysing patterns in human behaviour over time. It involves identifying, classifying, and analysing recurring patterns and trends in individuals' activities.

By recognising and interpreting these patterns, valuable insights about human behaviour, habits, preferences, and routines can be gained.

Human activity pattern recognition aims to uncover meaningful information from the collected data, such as identifying common behavioural patterns, detecting anomalies or deviations from normal behaviour, predicting future activities, and providing personalised recommendations or interventions. This knowledge has diverse applications in various domains, including healthcare, fitness monitoring, elderly care, smart environments, behaviour analysis, and human-computer interaction.

By understanding human activity patterns, researchers and practitioners can develop systems and technologies that adapt to individual needs, enhance well-being, improve efficiency, and provide context-aware services. It enables the design of intelligent systems that can automatically recognise and respond to human activities, leading to advancements in areas such as activity monitoring, assisted living, personalised assistance, and smart automation.

Human activity pattern recognition is crucial in unlocking valuable insights from human behaviour data, enabling us to better understand and support individuals in their daily lives. It offers a promising avenue for developing innovative applications and technologies that enhance human well-being and improve the overall quality of life.

## 1.7 Applications

HAR has found numerous applications in the context of smart homes. By leveraging sensor data and machine learning algorithms, HAR enables the development of intelligent systems that can understand and respond to human activities in a home environment. Some key applications of HAR in smart homes include *Energy Efficiency*, *Security and Safety*, *Health Monitoring*, *Ambient Intelligence*, *Personalised Assistance*, *Aging in place* and others.

- *Energy Efficiency*: HAR can help optimise energy consumption by detecting and predicting patterns in activities related to lighting, heating, and appliance usage. Smart home systems can automatically adjust energy settings to minimise waste and maximise efficiency by understanding when and how occupants interact with different devices.
- *Security and Safety*: HAR can be used for intruder detection, anomaly detection, and activity monitoring to enhance home security. By recognising pat-

terms of regular activities and identifying anomalies or suspicious behaviours, smart home systems can raise alerts and trigger appropriate responses to ensure the safety of occupants and their belongings.

- *Health Monitoring*: HAR plays a crucial role in monitoring the health and well-being of individuals within a smart home. By analysing activity patterns, sleep quality, and mobility, smart home systems can provide insights into an individual's physical health and detect potential health issues or risks. This information can be invaluable for elderly care, assisted living, and remote health monitoring.
- *Ambient Intelligence*: HAR enables the creation of context-aware environments that can adapt to occupants' needs and preferences. Smart home systems can automatically adjust lighting, temperature, and other environmental factors to create a comfortable and personalised living space by recognising specific activities or behavioural patterns.
- *Personalised Assistance*: HAR can provide personalised recommendations and assistance based on individual activities and preferences. For example, a smart home system can suggest recipes based on cooking habits or provide reminders for medication intake based on daily routines. This level of personalisation enhances convenience and supports individuals in their daily lives.
- *Aging in Place*: HAR is particularly valuable for supporting ageing populations to live independently in their own homes. By monitoring activities, detecting falls, and providing assistance when needed, smart home systems equipped with HAR can help ensure the well-being and safety of older adults, enabling them to age in place with confidence.

These are just a few examples of how HAR can be applied in smart homes. The advancements in sensor technologies, machine learning algorithms, and data analytics continue to open up new possibilities for improving smart homes' functionality, comfort, and efficiency through human activity recognition.

## 1.8 Motivation

The elderly population is growing unprecedentedly, increasing the need for assistance and care in the home environment. However, providing personalised assistance to every individual can be challenging regarding workforce, time, and resources. This has prompted the development of smart home technologies that aim to address these challenges and enhance the quality of life for the elderly. HAR is a crucial aspect of smart homes that can enable personalised assistance and monitoring. Most existing research in HAR focuses on specific sensor-based activity recognition, where each type of sensor has strengths and limitations in recognising particular activities. There is a significant opportunity to leverage the benefits of various sensors in a smart home environment by utilising hybrid sensor systems. This can be achieved through sensor fusion at different levels, such as sensor level, data level, feature level, and model level, to improve the overall performance of activity recognition. Moreover, the scope of HAR extends beyond activity recognition alone. Considering the prolonged duration of activities in a home setting, there is a challenge in detecting patterns in the performed activities. Understanding activity patterns can provide valuable insights into individuals' daily routines and behaviours, enabling personalised assistance and anomaly detection. Therefore, there is a need to explore human activity pattern recognition in the context of a smart home environment.

To address these challenges and gaps in the literature, this research aims to conduct a comprehensive literature survey on various sensors used for human activity recognition and human activity pattern identification in the smart home environment. By exploring the potential of sensor fusion, improving the performance of HAR systems, and investigating human activity patterns, this research intends to contribute to the development of intelligent smart homes that can provide personalised assistance and enhance the well-being of individuals.

## 1.9 Problem Statement

Despite the increasing popularity of Smart Home technologies, there remains a critical need for an effective Human Activity Recognition (HAR) system tailored to the needs of the elderly population. This study aims to address the gaps like improving the accuracy of the HAR, long-hour data collection in real time environ-

ment. By developing a comprehensive framework that integrates multiple sensors and leverages their respective characteristics to accurately recognize and analyze human activities and behavioral patterns in the daily routines of elderly individuals living in Smart Homes. The framework will be developed using a long-hour dataset collected specifically for this purpose, with the goal of overcoming existing challenges and limitations identified in the literature. The aim is to contribute to improved quality of life and enhanced support for independent living among the elderly.

*“To design and develop the framework that integrates the human activity and behavioral pattern recognition using the smart phone sensors and ambient sensors to improve the accuracy of the recognition. Also, to design and develop the behavioural pattern analysis using the data mining techniques to assist the elderly and their well being”*

## 1.10 Research Objectives

**Objective 1: To design and develop an effective human activity recognition system using deep learning techniques.**

Publications based on Objective 1:

- (a) Ranjit Kolkar and Geetha V., “Human Activity Recognition using Deep Learning Techniques with Spider Monkey Optimisation”, Multimedia Tools and Applications, Springer journal. <https://doi.org/10.1007/s11042-023-15007-7> (Scopus/SCIE)
- (b) Ranjit Kolkar and Geetha V., “Issues and Challenges in Various Sensor-Based Modalities in Human Activity Recognition System.” Applications of Advanced Computing in Systems. Springer, Singapore, 2021. Book Chapter Page: 171-179, Presented at International Conference on Advances in Systems, Control and Computing (AISCC-2020) held at Malaviya National Institute of Technology Jaipur (MNIT Jaipur), 2020. [https://doi.org/10.1007/978-981-33-4862-2\\_18](https://doi.org/10.1007/978-981-33-4862-2_18)
- (c) Ranjit Kolkar and Geetha V., “Human Activity Recognition in Smart Home using Deep Learning Techniques,” 2021 13th International Conference on Information and Communication Technology and System (ICTS), 2021, pp. 230-234, Presented at a technical co-sponsored IEEE conference and organised by the Informatics Department, Insti-

tut Teknologi Sepuluh Nopember (ITS) Surabaya, Indonesia. (Virtual Conference)

<https://doi.org/10.1109/ICTS52701.2021.9609044>. (Scopus)

- (d) Ranjit Kolkar, Rudra Pratap Singh Tomar and Geetha V. “IoT based Human Activity Recognition Models based on CNN, LSTM and GRU”. IEEE Silchar Subsection Conference (IEEE SILCON 2022) - Track 5- Artificial Intelligence, Presented virtually at National Institute of Technology Silchar in Nov-2022. <https://doi.org/10.1109/SILCON55242.2022.10028803> (Scopus)

**Objective 2: To create and curate a dataset for long-hour human activities in a multimodal sensor-equipped smart home environment.**

Publications based on Objective 2:

- (a) Ranjit Kolkar and Geetha V., “Human Activity Behavioural Pattern Recognition in Smarthome with Long-hour Data Collection”, SNCS Springer.

<https://doi.org/10.1007/s42979-023-02278-y> (Scopus)

**Objective 3: To design and develop a human behavioural pattern recognition system in a smart home environment.**

Publications based on Objective 3:

- (a) Ranjit Kolkar, Geetha V. and Ashwin T. S., “Behaviour Activity Profiler: A multimodal Sensor-based Approach with Adaptive Granularity”, Elsevier Information Science (Scopus/SCI) (Under Review)
- (b) Ranjit Kolkar and Geetha V., “Comprehensive Activity Analysis and Pattern Identification using Multimodal Sensors with Data mining”, ACM Transactions on Knowledge Discovery from Data (Scopus/SCI) (Under Review)

## 1.11 Framework for HAR and Behavioural Pattern Identification in a Smart Home Environment

The proposed framework, as shown in Figure 1.3, aims to design and develop a comprehensive system for human activity recognition and behavioural pattern

identification in a smart home environment.

The proposed framework addresses challenges in recognizing basic human activities efficiently using smartphones. By combining data from multiple sensors, it overcomes limitations of single-sensor setups, allowing for a broader range of activities to be derived. Synchronization of sensor data is essential for comprehensive user behavior understanding. The framework introduces a novel approach to cost-effective, unobtrusive, and private long-hour data collection in smart homes, filling a gap in multimodal sensor applications. The comprehensive analysis of this diverse dataset tackles the challenge of representing data from multimodal sensors effectively. Additionally, the framework introduces context-aware, personality-oriented analysis, advancing the understanding of human activities in smart home environments.

It consists of two main components: data collection and pattern recognition.

Various sensors capture different aspects of human activities in the data collection phase. Smartphone sensors such as accelerometers and gyroscopes are used to gather motion-related data, providing information about the movement and orientation of the device. Ambient sensors like [PIR](#), smart switches, and load sensors are employed to detect specific activities, such as human presence and monitoring water consumption. This multimodal sensor setup ensures comprehensive data collection from different sources.

To train and recognise activities, the initial training is performed using the [WISDM](#) dataset, which provides a wide range of labelled activity data. Deep learning models such as [LSTM](#) and [GRU](#) are employed due to their ability to capture temporal dependencies in time-series data. These models can effectively learn and recognise patterns from the collected data.

To optimise the performance of the trained models, spider monkey optimisation is applied. This metaheuristic algorithm helps fine-tune the model parameters and improve its accuracy and efficiency. The optimised models are then deployed in an Android app, allowing real-time activity recognition on mobile devices. The deployed models are thoroughly validated to ensure their reliability and effectiveness in real-world scenarios.

Additional datasets from multiple users are incorporated further to enhance the performance and diversity of the collected data. These datasets are mapped to the existing [WISDM](#) dataset, expanding the range of activities and increasing the variability in the collected data. This augmentation helps improve the generalisation capabilities of the models and makes them more robust in recognising different activities.

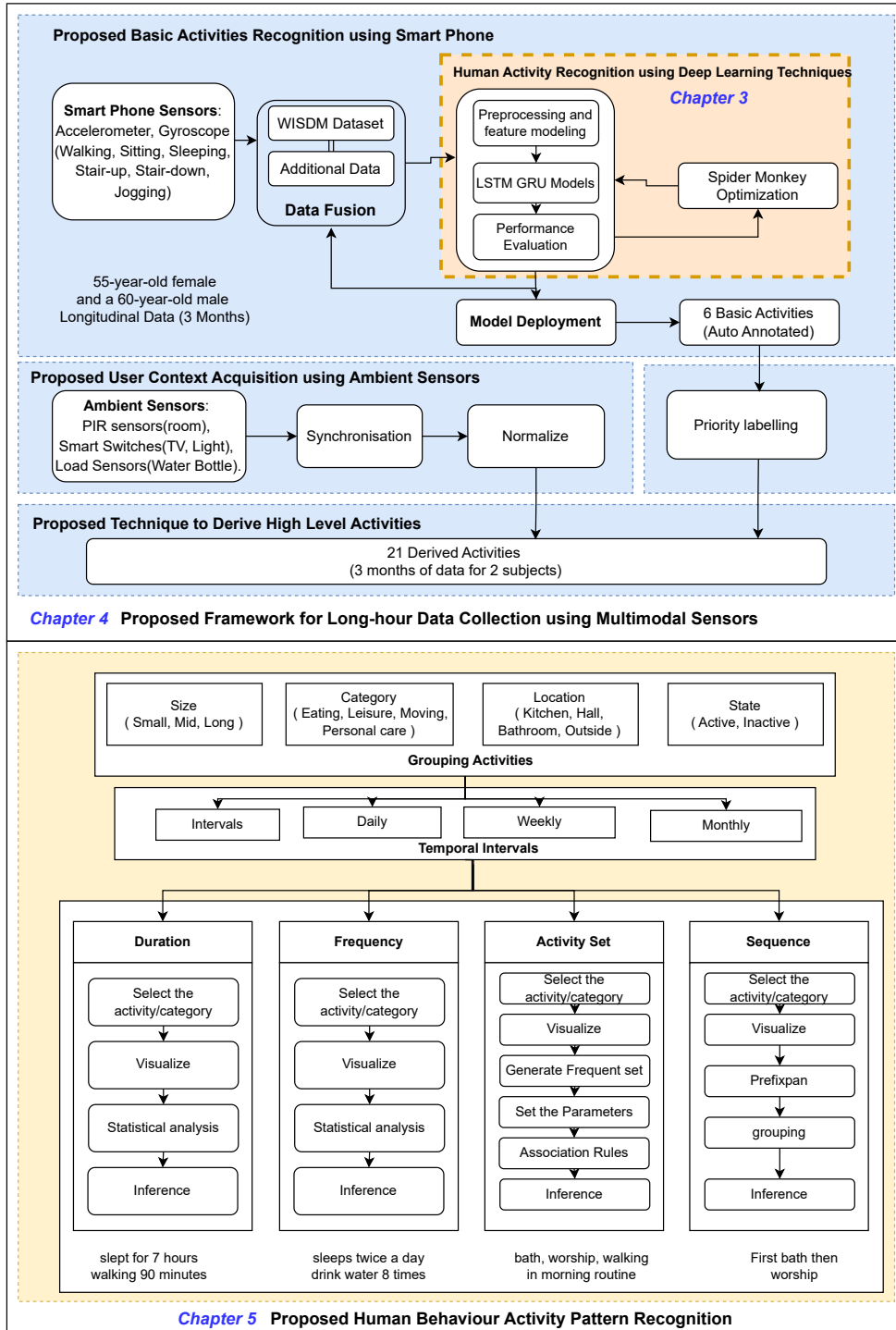


Figure 1.3: Proposed Framework for HAR and Behavioural Pattern Identification in a Smart Home Environment

During the long-hour data collection phase, the mobile app collects activity data over an extended period, typically three months. Simultaneously, ambient sensors record data regularly, typically every two minutes. This synchronised data provides a more comprehensive view of the activities performed in the smart home environment.

The collected data is then preprocessed and labelled using priority labelling. Heuristic and context-based labelling techniques are employed to assign appropriate labels to the data instances based on the information from both mobile and ambient sensors. Manual validation and synthesis of missing data are performed to ensure data completeness and accuracy.

In the pattern recognition phase, the derived data is categorised based on activity size, category, location, and state attributes. Different temporal intervals, including intervals, daily patterns, weekly trends, and monthly insights, are analysed to understand human activity patterns comprehensively.

User profiles are derived using adaptive granularity techniques, which involve creating events, groups, and profiles based on the collected data. This enables a personalised analysis of activity patterns and helps tailor smart home systems to individual users' needs and preferences.

Duration-based analysis involves selecting specific activity categories, visualising the data, conducting statistical analysis, and deriving meaningful insights. Frequency-based analysis identifies frequent patterns and association rules within the activity data. It utilises techniques such as Frequent Pattern Tree (FPTree) and the Apriori algorithm to discover meaningful associations between different activities. Activity set-based analysis further explores these associations by grouping related activities and extracting more specific rules.

The sequence-based analysis uses profile scanning techniques to traverse the data sequences and identify recurring patterns and trends. This analysis helps understand the sequential ordering of activities and identify common activity sequences.

The proposed framework combines multimodal sensor data collection, deep learning-based activity recognition, optimisation techniques, and comprehensive pattern recognition methods to enable effective human activity recognition and behavioural pattern identification in smart home systems. By leveraging various data sources and employing advanced analytical techniques, the framework provides valuable insights into human behaviour and supports the development of intelligent and personalised smart home applications.

## 1.12 Research Highlights

1. Proposed a deep learning framework to recognize basic human activities. This work involves experimentation on existing datasets and data fusion to improve performance over existing works, as well as using optimizations to improve the models.
2. Proposed a framework for long-hour data collection using multimodal sensors. This work involves designing and developing a smartphone application to collect long-hour data. The application automatically annotates recognized activities and utilizes ambient sensors for user context acquisition. This work also involves synchronizing multimodal sensors to capture a broader range of user activities, a task that would be challenging with only a smartphone.
3. Proposed a priority labeling technique for data segmentation for better context-based analysis.
4. Proposed an adaptive granularity approach involving context-based behavioural pattern identification over daily, weekly, and monthly periods. By adapting the granularity of the analysis, this method aims to identify frequent activities and provide a deeper understanding of human behaviour over time.

These contributions collectively aim to advance the field of Human Activity and Behavioural Pattern Recognition in Smart Home environments, offering novel frameworks and approaches to enhance the accuracy, efficiency, and usability of such systems for the benefit of elderly individuals.

## 1.13 Organization of the Thesis

The remainder of this thesis is organised as follows, shown in Figure 1.4. Chapter 2 comprehensively reviews the related work on HAR. It examines the challenges and issues associated with various sensor modalities, including wearable, vision, and ambient-based sensors. The chapter also presents a detailed summary of existing recognition models and publicly available datasets. Additionally, it emphasises the importance and challenges of pattern recognition in the context of smart homes. The problem statement and research objectives are defined based on the findings of the literature survey.

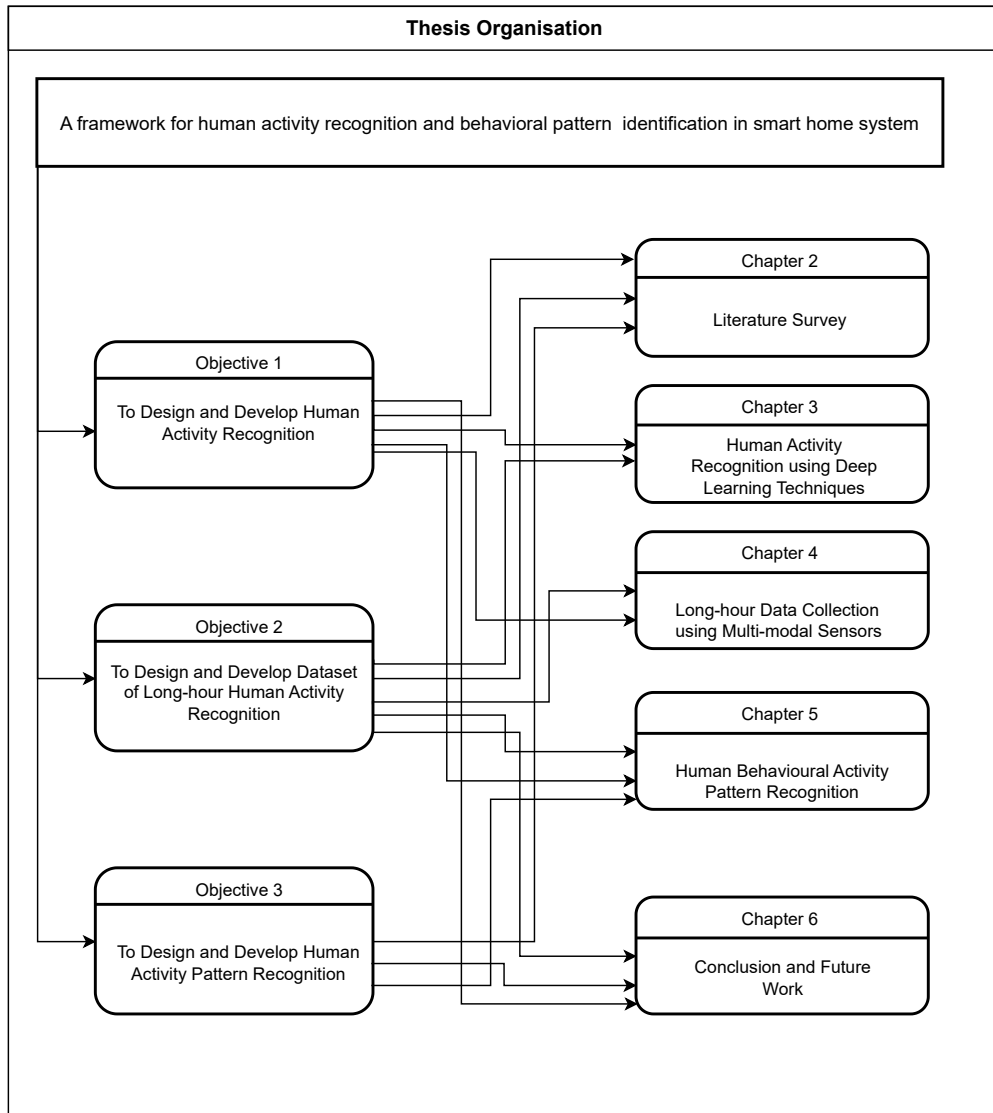


Figure 1.4: Thesis Organization

Chapter 3 focuses on the preliminary work conducted on HAR using Deep Learning techniques. The chapter discusses the various datasets and describes the experiments using different deep-learning models to enhance performance. It presents the implementation of the [GRU](#) based HAR recognition and showcases the results achieved. Furthermore, the chapter proposes utilising Spider Monkey-based optimisation techniques to improve performance.

Chapter 4 is dedicated to designing and developing a dataset collection framework for capturing long-hour human activities. It outlines the strategies employed for collecting multimodal sensor-based datasets in different phases. The chapter also proposes various labelling strategies for human activity pattern recognition.

In Chapter 5, the focus shifts to modelling human activity pattern recognition.

The data collected in Chapter 4 is analysed to derive human activities based on duration, frequency, and sequences. Additionally, adaptive granularity-based human profiling is performed.

Finally, Chapter 6 summarises the contributions of the research work and outlines possible future directions for enhancing the performance of HAR and Human Activity Pattern Recognition (HAPR).

## 1.14 Summary

This section allows an understanding of the comprehensive overview of the research topic of HAR and its significance in the context of smart homes. The HAR's history emphasises its role in improving the quality of life and enabling various applications such as health monitoring, assisted living, and energy management. HAPR is a higher-level analysis of human activities, focusing on identifying patterns, durations, frequencies, and sequences of activities. The importance of HAPR in understanding human behaviour and enabling personalised services in smart homes makes it interesting to carry out the research. It is essential to understand the general framework for HAR in smart homes, which involves using various sensor modalities such as wearable, vision-based, and ambient sensors. The integration of these sensors allows for comprehensive data collection and analysis, enabling accurate activity recognition and pattern discovery.

The introduction to the topic, its challenges and issues associated with HAR, such as sensor placement, data variability, and real-time processing, motivated for conducting research in HAR and HAPR with the potential benefits of improving health outcomes, enhancing energy efficiency, and providing personalised services in smart homes.

The next chapter delves into the Literature Survey, providing a detailed analysis of existing research in the field.

# Chapter 2

## Literature Survey

This chapter focuses on a literature survey in the field of HAR in general and smart Homes. It consists mainly of four types of sensor-based HAR systems: Wearable, Camera, Ambient sensor-based HAR and the combinations among these sensors. The literature survey also includes publicly available datasets, applications developed with the HAR, human activity pattern recognition, and anomaly detection. This section elaborates on the background and related work on the HAR in the smart home environment. It also summarises the research issues and challenges based on the literature survey. Problem statement and research objectives are formulated to address the identified challenges. Figure 2.1 shows the organisation of the literature review.

### 2.1 Sensor-based HAR

An investigation has been conducted to explore diverse sensor-based human activities, including identifying various activities by utilising different sensor modalities. The study observes the strengths and constraints of each sensor modality in recognising activities. Furthermore, it investigates various pattern recognition techniques on diverse datasets to comprehend human behavioural patterns. Additionally, the research addresses prevalent issues and challenges encountered in this domain.

#### 2.1.1 Wearable Sensor-based HAR

Wearable Sensor-based HAR ([WSHAR](#)) systems are mounted on the human body parts. Human performs various activities in daily life that involve the movement of body parts — this wearable sensor tries to acquire the motion of the activities

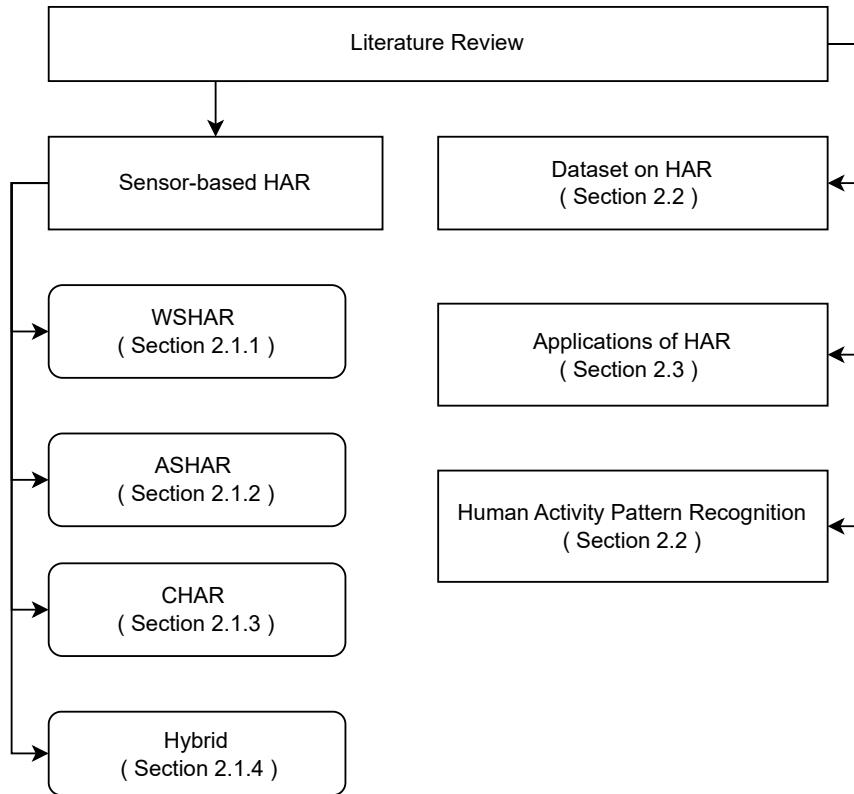


Figure 2.1: Literature Review Organisation

performed by the human. The development of wearable devices, such as smartphones, smartwatches, wristbands, smart clothes or other sensors attachable to the body, makes capturing data from these sensors easy. (Adaskevicius 2014), (Filippoupolitis *et al.* 2017a), (Hassan *et al.* 2018). These devices also can continuously monitor real-time activities and give feedback to the user immediately or periodically. Wearable sensors are used in various applications like health care, sports, fitness, smart home, etc.

The sensors' advancement and ability to acquire the data makes it possible to apply them to many real-time applications. In the context of human activity recognition, these sensors can be Inertial sensors, which include accelerometer, gyroscope, and magnetometer; Physical health sensors, which include ECG, Electroencephalography (EEG), Electromyogram (EMG) Skin temperature; Environmental sensors like camera, microphone, GPS, etc. All these sensors can be integrated into various small-size devices like smartphones, smartwatches, smart clothes gloves, etc.

In the realm of smart homes and smart health environments, people engage in various activities, which can be classified into three levels: *Transitive*, *Basic* and *Complex* activities. *Transitive* activities involve distinct temporal patterns among

actions and include activities like stand-to-sit, sit-to-lie, push-ups, and bicep curls, often relevant in fitness or rehabilitation contexts (Farah *et al.* 2019). *Basic* activities serve as mid-level features for classification tasks, with longer durations compared to transitional actions. These encompass activities such as walking, running, cooking, and using stairs (Lorussi *et al.* 2016). *Complex* activities involve sequential, interleaved, or concurrent patterns of transitional or basic activities, including actions like coffee breaks, relaxation periods, smoking, and conversations (Liu *et al.* 2016). This categorization aids researchers and practitioners in understanding and developing tailored recognition systems for activities in smart environments.

In [WSHAR](#), various sensor modalities play a crucial role in accurately detecting and classifying human activities. These sensors are typically integrated into different platforms users carry while performing activities. The aim is to minimise the obtrusiveness during use and provide a convenient and unobtrusive way of collecting activity data.

One of the most widely used platforms for HAR is smartphones, which come equipped with various sensors such as accelerometers and gyroscopes. Smartphones are ubiquitous and carried by people everywhere, making them an ideal choice for data acquisition in daily activities (Hassan *et al.* 2018; Uddin *et al.* 2018; Mohamed *et al.* 2012; Kang *et al.* 2017; Guo *et al.* 2016). However, they have constraints of limited sensor types and fixed locations, often found in pockets, belts, or bags, which may only be suitable for some types of activities.

Another popular platform is smartwatches, designed with integrated sensors and typically wrist-mounted. Wearing a smartwatch is more convenient and less obtrusive than carrying a smartphone all the time. However, similar to smartphones, smartwatches may not have the exact sensors required for specific tasks (Chernbumroong *et al.* 2013; Filippoupolitis *et al.* 2017a)

Smart clothing presents an intriguing option for integrating multiple sensors, including physical sensors, to collect a wide range of data. These garments are particularly suitable for long-term monitoring applications due to their ease of wear (Adaskevicius 2014). For example, smart shirts are designed with sensors to monitor various physiological activities, including cardiac, respiratory, and sleep-related parameters, often incorporating sensors for heart rate and ECG measurements (Hexoshin 2018). Lorussi *et al.* (2016) developed a comprehensive smart textile platform, comprising sensing shirts, trousers, gloves, and shoes, primarily aimed at assessing stroke patients. This platform incorporates various sensors, such as inertial sensors, textile goniometers, piezoresistive sensors, [EMG](#), and goniometers.

In a different context, Zhou et al. (2016) employed two types of textile-based sensors: a fabric pH sensor for sweat analysis and piezoresistive textiles to capture body movements. Smart clothing designs are also tailored for monitoring aspects like infants' sleep patterns, breathing, and body positions (Mimobaby 2018).

In addition to smart clothing, Inertial Measuring Unit (IMU), which stands for Inertial Measurement Unit, represents a specialized device that measures and reports a craft's velocity and orientation. It achieves this by combining data from various sensors, including accelerometers, gyroscopes, magnetometers, and sometimes barometers. In various applications, IMU sensors or combinations of them have been employed to detect human gestures or activities, demonstrating satisfactory performance (Bulling *et al.* 2014; Georgi *et al.* 2015; Montalto *et al.* 2015). These sensors are versatile and find utility in different contexts for activity recognition.

Specifically designed platforms are built for specific or common research purposes in HAR, in which the sensors required for a specific task are integrated. These platforms can meet the specific sensor requirements, but they may involve extra costs in hardware and research period (Uddin *et al.* 2018; Burns *et al.* 2010; Cook *et al.* 2015). The table 2.1 summarises the wearable sensor-based human activity recognition system. The research challenges in WSHAR are as follows.

1. *Accuracy in Activity Recognition:* Correctly identifying the activities is challenging due to human activities. For example, sensors may not give correct data, or that sensor may not be enough to provide all the necessary information needed for the activity.
2. *High-Level and Long-Term Activity Monitoring:* Most results focus on short-term activity detection. Acquiring the data properly over some time makes it difficult to process. The proper dataset is limited to specific activities and cannot be generalised.
3. *Multi-User and Multi-Sensor Activity Monitoring:* As each individual performs the activities differently, combining multiple users and activities is challenging.
4. *Real World Data Collection:* Most of the result carries their experiment in a lab and controlled environment because of the difficulty level.
5. *Heterogeneous Sensor Data Representation:* The input and output of the sensor node differ drastically; hence, it is very difficult to process and analyse the heterogeneous sensor data.

6. *Imbalanced and Overlapping Data Classes*: Some of the classes occur less frequently, and some of the classes take longer time.

Table 2.1: Summary of Literature Review on Wearable Sensors-based in HAR

Author	Description	Activities	Advantages	Limitations
Liu <i>et al.</i> (2021)	Health care applications using general machine learning approach with supervised learning	Gaits for ICU	Early mobility identification using gaits	specific activity recognition.
Wang <i>et al.</i> (2019)	Swimming posture monitoring using inertial sensors.	Swimming strokes	Specialized sensor, battery and inbuilt memory	Not significant accuracy
Farah <i>et al.</i> (2019)	Microprocessor-controlled wearable walk assist sensors.	Gait	Real-time analysis	Complex sensor system
Uddin <i>et al.</i> (2018)	Ambient sensors for limited activities such as walking, jogging, lying, standing, falling, and upstairs.	Walking, jogging, lying, standing, falling, upstairs	High accuracy	Not real-time, obtrusiveness, limited activity
Hassan <i>et al.</i> (2018)	Robust human activity recognition based on smartphone accelerometer and gyroscope data using deep learning techniques.	Standing, lying, walking, upstairs, sitting, downstairs	Widely used sensor (smartphone)	Poor results due to smartphone placement.
Zhang <i>et al.</i> (2017)	Sleep stage detection using EEG readings during sleep. Simple machine learning classifiers are used and compared.	Sleeping, EEG patterns	Uses physiological signals	Limited activities
Lorussi <i>et al.</i> (2016)	Specialised clothes with sensors for stroke patient rehabilitation and training. Pressure sensors for stroke patients' activities.	Stroke patient rehabilitation and training	Pressure sensors.	Clothes and gloves are easy to use and comfortable and costlier sensors
Chernbumroong <i>et al.</i> (2013)	Uses wrist-watch and inertial sensors like accelerometer and temperature for detecting daily living activities for the elderly. Uses SVM for classification.	Feeding, brushing teeth, walking, sleeping	Inexpensive, small and non-intrusive sensors	Limited activities can be identified.
Proposed Work of HAR Recognition (Section 3.1)	Smartphone based inertial sensors like accelerometer and gyroscope for detecting daily living activities for the elderly. Uses GRU model for classification.	stair up, walking, sleeping, and other	Inexpensive, small and non-intrusive sensors	Limited activities can be identified.

### 2.1.2 Ambient Sensor-based HAR

The literature on [ASHAR](#) encompasses a range of research studies that explore the potential of utilising ambient sensors to detect and classify human activities automatically. [ASHAR](#) presents its own challenges due to the reliance on environmental factors and body movement. Including ambient sensors allows for a more comprehensive understanding of human activities. These sensors can include

magnetic switches to detect door open/close events, temperature sensors for ambient or water temperature monitoring, photo sensors to measure light illuminance levels, pressure pads to assess applied pressure, infrared motion sensors to detect movement, and water flow sensors for taps, among others. Additionally, ambient sensors can encompass home electric appliances, force sensors, smoke/heat sensors, and more. The signal types from these sensors can be binary or continuous.

Context plays a crucial role in accurate human activity recognition, as highlighted by Ni *et al.* (2015). Consideration of contextual factors such as object interaction, space, and time enables precise and correct activity recognition. For instance, recognising the preparing activity can be inferred by considering the coffee machine as the object. Location information helps eliminate ambiguity in identifying activities like lying down or sleeping. Contextual information significantly reduces the time required for activity identification.

In the realm of smart homes, Alam *et al.* (2012) provides an overview of smart home systems and their components. They categorise smart home services and utilities into comfort, remote access, automation, repository, and energy optimisation while also classifying sensors and their purposes into conventional sensors (e.g., temperature, motion, pressure, and location), physiological sensors (e.g., heart rate, blood pressure), and multimedia sensors (e.g., camera, microphone, thermal imaging).

Furthermore, Rashidi and Mihailidis (2012) focuses on building algorithms for analysing human behaviour in smart homes, and Salih and Abraham (2013) presents a review of ambient-intelligence-assisted healthcare monitoring, particularly in the context of wireless sensor network technologies and applications. Data mining techniques are often employed in ambient sensor-based patient monitoring systems to extract meaningful insights from the collected data. These approaches leverage the data generated by ambient sensors to monitor and assess the well-being of individuals in a non-intrusive manner.

Alwan *et al.* (2006, 2007) conducted pilot studies in assisted living units, monitoring Activities of Daily Living (ADL) for several months. The results showed a high acceptance rate of the system among residents and caregivers, suggesting the potential for improved healthcare planning and detection of health status changes. Similarly, (Lotfi *et al.* 2012) conducted a case study on dementia patients, using ambient sensors to identify abnormal behaviour, thereby demonstrating the efficacy of such systems in capturing meaningful data.

Fall detection is another critical aspect of ASHAR, and researchers have made significant progress in this domain. Ariani *et al.* (2012) simulated a fall detection

system and achieved an impressive 89.33% accuracy, highlighting the potential for reliable fall monitoring using ambient sensors. Rantz *et al.* (2008) conducted a pilot study on fall and wandering detection in assisted living environments, reporting improved safety and security through ambient sensors.

The literature describes the technical aspects of ambient sensor systems. Bamis *et al.* (2008, 2010) provided insights into detecting activities and deviations in activity patterns, and Celler *et al.* (1996) proposed a sleep stage classification algorithm based on a polysomnography database, achieving 100% accuracy.

Researchers have explored the fusion of different sensor modalities to improve HAR accuracy and localisation precision. Kinney *et al.* (2004) presented a localisation system that combined infrared and sound sensors, resulting in a 54% improvement in localisation precision compared to standalone systems.

The summary of the table is shown in the Table 2.2. The literature on ambient sensor-based HAR demonstrates the potential of these systems in enhancing healthcare, providing safety, and understanding human behaviour patterns. The studies discussed above showcase the efficacy of ambient sensors in recognising ADL, detecting falls, and improving localisation precision, thereby contributing to the advancement of smart and assistive living environments. The challenges in ASHAR are as follows:

1. Ensuring accurate and reliable data collection from various ambient sensors.
2. Dealing with data variability due to environmental factors and sensor limitations.
3. Developing robust activity recognition algorithms that consider body movement and ambient sensor data.
4. Handling the complexity of contextual information and integrating it into the recognition process.
5. Addressing privacy concerns and ensuring data security in smart home environments. Balancing the trade-off between sensor coverage and user privacy.
6. Overcoming challenges related to data preprocessing, feature extraction, and model training with diverse ambient sensor data.
7. Addressing interoperability issues among ambient sensors and smart home systems.

8. Incorporating adaptive learning techniques to handle changes in the environment and user behaviour over time.
9. Ensuring scalability and efficiency in processing large volumes of ambient sensor data for real-time recognition and monitoring.

### 2.1.3 Camera Sensor-based HAR

In a [CSHAR](#) system, data is acquired through various cameras, such as RGB- and depth-based cameras. These cameras are typically mounted in fixed positions within the environment (Jalal *et al.* 2017). Some researchers consider camera-based activity recognition an ambient-based sensor technology due to its environmental installation. For example, Bian *et al.* (2014) utilised a depth camera for fall detection in a smart home environment, while Khan and Sohn (2011) employed a single camera to detect activities like chest pain, fainting, headache, falls, and vomiting, using silhouettes instead of the whole image frame to address privacy concerns. Microsoft Kinect-based sensors, equipped with infrared cameras, infrared projectors, and microphones, offer accurate depth information and skeleton extraction, facilitating gesture and action recognition of humans. However, Kinect sensors have limited range capabilities.

In the study by Mohamed *et al.* (2012), Kinect sensors were utilised to monitor and assist older and disabled individuals, employing machine learning and deep learning techniques for activity and action classification. Table 2.3 summarises the camera-based human activity detection methods. The challenges in [CSHAR](#) are as follows.

1. *Human behaviour*: Human behaviour is complex and does not have a common pattern over time, and in most cases, individual activities differ, which makes it challenging to recognise the activity. The characteristics of various activities are similar, but they are fundamentally different. Brushing and eating look identical, but they are entirely different activities.
2. *Inter-class similarity*: The characteristics of various activity classes are similar but fundamentally different. Brushing and eating look identical, but they are entirely different activities.
3. *Influence of the environment*: The light or illumination factor variation affects the recognition system and makes it challenging for the system. The

Table 2.2: Summary of Literature Review on [ASHAR](#)

Author	Purpose	Characteristics	Outcomes	Sensors
Ramanujam <i>et al.</i> (2021)	Survey on various sensors	Survey on Deep Learning Models in ambient based sensors	benchmark datasets	wearable and home appliances sensors
Alwan <i>et al.</i> (2006)	Recognition of activities of daily living	Monitored activities in assisted living units for 3 months. High system acceptance. Improved healthcare planning.	High acceptance rate.	PIR, stove sensor, bed pressure sensor
Alwan <i>et al.</i> (2007)	recognition of activities of daily living	Assessed impact of passive health status in assisted living for three months. Reduced billable interventions and cost of care.	Reduced billable interventions and cost of care.	PIR motion sensors, stove sensor, pressure sensors
Ariani <i>et al.</i> (2012)	Fall detection	Simulated fall detection system tested on existing dataset. 89.33% accuracy in fall detection.	89.33% accuracy in fall detection.	PIR motion sensors, pressure mats
Bamis <i>et al.</i> (2008)	Recognition of activities of daily living	Case study on two residences with 7 and 4 months of monitoring. Described system functionality in detecting activities.	Described system functionality in detecting activities.	Video monitoring, PIR motion sensors
Bamis <i>et al.</i> (2010)	Recognition of activities of daily living	Reported progress in sensors and behavior interpretation mechanisms. Demonstrated system functionality.	Demonstrated system functionality.	Video monitoring, PIR motion sensors
Celler <i>et al.</i> (1996)	Sleep stage classification	Proposed classification algorithm based on polysomnography database. Outperformed commercial products in a specific database.	100% accuracy in sleep stage classification.	Doppler radar, microphone
(Kinney <i>et al.</i> 2004)	Localization	Proposed localisation system combining infrared and sound sensors. Improved precision compared to a standalone system.	54% improvement in localisation precision.	PIR motion sensors, sound sensors
Lotfi <i>et al.</i> (2012)	recognition of activities of daily living	Case study on two dementia patients monitored for 20 days and 18 months. Identified abnormal behaviour using ambient sensors.	Identified abnormal behaviour using ambient sensors.	PIR motion sensors, door opening sensors, flood sensors
Rantz <i>et al.</i> (2008)	Fall detection	Pilot study on fall detection and wandering detection. Installed for 8-23 months. Improved sense of safety and security.	Improved sense of safety and security.	PIR motion sensors, video camera
Zouba <i>et al.</i> (2009)	Recognition of activities of daily living	Monitoring of simulated activities in a laboratory setting. 50-80% precision, 66-100% sensitivity.	50-80% precision, 66-100% sensitivity in activity recognition.	Video camera, PIR motion sensors
Proposed user context acquisition using Ambient sensors (Section 4.2)	occupancy detection, alliances status	Identifying the context of a user based on the location and the appliances status	Synchronized and normalized data of a elderly	Relay sensors, load sensors, PIR motion sensors

video's brightness, contrast and shadow make it difficult and need to be taken care of at the time of production. Even the weather conditions influence the

recognition system.

4. *Occlusions*: In the Case of multiple people in the video, it is difficult to process the subjects if they are fully or partially occluded with each other. Even some of the body parts of the individual might also be occluded. For example, the hand is hidden from a side viewpoint in the video.
5. *Scaling*: The people performing the activity may be near or far from the device used to sense the activity. The quality of the video, like resolution, brightness, and lighting, matters a lot. Even the camera adjustments should be consistent and make it difficult to process if the parameters from one camera to another differ.
6. *Camera*: The quality of the video, like resolution, brightness, and lighting, matters a lot. Even the camera adjustments should be consistent and make it difficult to process if the parameters from one camera to another differ.
7. *Moving background objects or humans*: Moving objects vary the features quickly and make recognising the correct activity difficult.

### 2.1.4 Hybrid Sensor-based HAR

There has been growing interest in developing Hybrid Sensor-based HAR ([HSHAR](#)) by integrating data from different sensor modalities to improve classification accuracy. Pansiot *et al.* (2007) proposed a sensor-fusion-based framework that combines an ear-worn accelerometer and a vision sensor installed in the environment, achieving improved classification accuracy. Similarly, Hayashi *et al.* Hayashi *et al.* (2015) investigated the combination of environmental sound and acceleration data using Deep Neural Network ([DNN](#)) for HAR, demonstrating an accuracy rate of 91.7% for nine different daily activities. Liu *et al.* (2016) presented a hybrid sensor modality framework based on probabilistic Hidden Markov Models ([HMM](#)) classification for hand gesture recognition. Their framework fuses data from an inertial sensor and a Kinect depth sensor, achieving an accuracy of 93% after data fusion, surpassing the individual sensor performances.

In the context of combining wearable and ambient sensors, Stikic *et al.* (2008) investigated the integration of Radio Frequency Identification ([RFID](#)) into wearable accelerometers on the wrist for detecting daily activities. Their experimental results demonstrated significantly improved recognition accuracy after sensor fusion. Roy *et al.* (2016) proposed a hybrid approach using wearable and ambient

Table 2.3: Summary of Literature Review on CSHAR

Author	Description	Modalities	Activities	Observations
Siddiqi <i>et al.</i> (2021)	selection of confined features from the series of images and discriminate their category based on reversion	Depth	Pose	Privacy concerns
Chen <i>et al.</i> (2016)	Action graph skeleton-based depth features Maximum likelihood for classification MSR Action3D, SBU Kinect Interaction and UTKinectAction datasets are used	Depth	Pose	limited range, controlled environment
Berlin and John (2016)	Interest points are extracted through Harris corner. Histogram-based features are used, and Deep NN is used. UT interaction dataset is used to detect the interactions.	RGB	Pose	limited interactions are considered.
Zhu <i>et al.</i> (2016)	Co occurrences of the feature of skeleton joints based on depth. RNN and LSTM are used for classification Interaction or activity HDM05, Berkely MHAD action dataset, 98% accuracy	Depth and RGB	Pose and motion	fails to recognise sudden movements and limited to gestures, high computation resources
Li <i>et al.</i> (2016)	Vector of linearly aggregated descriptors for deep dynamics, Deep Convolution NN, uses global features based on RGB data. 99%	RGB	motion	high computation resources are unsuitable for real-time applications.
Mo <i>et al.</i> (2016)	Features are extracted automatically through skeleton data CNN, and multi-layer perception is used for classification CAD60 Dataset is used.	Depth	Pose	Smaller Dataset and the performance issues
Shi <i>et al.</i> (2017)	stream sequential deep trajectory descriptors are used for representation RNN and CNN are used for classification, KTH, HMDB51, UCF101 datasets are used	RGB	motion	not unified modeling. Privacy Concerns
In the proposed framework we would like to use only wearable and ambient sensors for sensor cost effectiveness				

sensors to detect complex daily activities in a multiple-inhabitant smart context, achieving a 30% improvement in accuracy compared to pure smartphone-based solutions. Wang and Miao (2018) introduced a hybrid sensory-based HAR system combining wrist-worn sensors and ambient-mounted PIR sensors, providing a more comprehensive and accurate activity monitoring for older individuals.

Data fusion between different sensor modalities is addressed in various ways. Liu *et al.* (2016) employed data-level fusion by synchronising and fusing data from inertial sensors and vision depth for gesture recognition. Pansiot *et al.* (2007) utilised feature-level fusion, preprocessing data from an ear-worn accelerometer and a wall-mounted camera before feeding them to a Bayesian classifier. Stikic *et al.* (2008) used decision-level fusion by combining the number of activations from infrared sensors with features extracted from acceleration data. Liu *et al.* (2016) applied multi-HMM classification framework for hand gesture recognition, with

decision-level fusion considering the maximum likelihood probability from each classifier. Diethel *et al.* (2017) addressed the challenges of the fusion of heterogeneous sensor modalities using Bayesian models. Nakamura *et al.* (2010) presented a collective framework that synchronises wearable and ambient sensors to monitor a user's location and vitals.

The research on data fusion in hybrid sensory systems for HAR has shown promising results, leading to enhanced recognition accuracy and more comprehensive activity monitoring capabilities. Researchers have explored various fusion approaches, including data-level, feature-level, and decision-level fusion, tailoring their strategies to the specific tasks and goals of the hybrid systems. These developments represent significant contributions towards creating more effective and robust HAR solutions. The following list is the challenges in using the multimodal or hybrid sensor-based HAR. Table 2.4 summarises the sensor-based human activity recognition. The challenges in HSHAR are as follows.

1. *integration of heterogeneous sensor data*: Combining data from different sensor modalities requires handling data with different formats, resolutions, and sampling rates. Integrating these data streams into a cohesive framework for activity recognition poses technical challenges.
2. *Data fusion and feature extraction*: Extracting meaningful features from heterogeneous sensor data and fusing them effectively is a complex task. It involves identifying relevant information from each modality and combining them to comprehensively represent human activities.
3. *Handling data variability*: Different sensor modalities may exhibit varying levels of noise, missing data, or outliers. Dealing with these variations and ensuring the robustness and reliability of the HAR system becomes a challenge.
4. *Privacy concerns*: Some sensor modalities, such as cameras, raise privacy issues due to the potential intrusion of personal space. Balancing the need for accurate activity recognition with privacy protection becomes crucial in designing hybrid sensory HAR systems.
5. *Contextual information integration*: Each sensor modality provides a unique perspective on human activities. Integrating contextual information from ambient sensors, such as temperature or light sensors, with wearable sensors' motion-related data requires developing sophisticated algorithms to utilise these diverse sources of information effectively.

6. *Scalability and resource constraints:* The integration of multiple sensor modalities increases the computational and resource requirements of the HAR system. Developing efficient algorithms and architectures that can handle large volumes of data in real time becomes challenging.
7. *Adaptability and generalizability:* The performance of hybrid sensory HAR systems may vary across different environments, user populations, or activity contexts. Ensuring the adaptability and generalizability of the system to different scenarios is a significant challenge.
8. *System calibration and synchronisation:* Different sensor modalities may require calibration and synchronisation to ensure accurate and synchronised data collection. Handling the technical complexities associated with calibration and synchronisation is crucial for effectively operating the hybrid sensory HAR system.

Addressing these challenges requires interdisciplinary research and the development of advanced algorithms, data fusion techniques, and system architectures. The ultimate goal is to create hybrid sensory HAR systems that can leverage the strengths of multiple sensor modalities to achieve more accurate and robust activity recognition in a wide range of real-world applications in smart home environments.

## 2.2 Datasets used in HAR

Table 2.5 provides a comprehensive overview of existing datasets commonly employed in HAR research. The table presents various datasets used for HAR along with their key attributes. These datasets have been instrumental in advancing research on HAR and have contributed to developing effective recognition systems using various sensing modalities. The datasets cater to diverse contexts, sensor types, and durations, enabling researchers to explore various applications and challenges.

[WISDM Dataset](#) (Kwapisz *et al.* 2011a) is a smartphone-based dataset that captures accelerometer and gyroscope data for single-person indoor activities. The [UCI-HAR Dataset](#) (Anguita *et al.* 2013a) focuses on indoor activities and utilises wearable sensors, such as accelerometers and gyroscopes. The Opportunity Activity Recognition Dataset (Roggen and Saghya 2012) offers a more comprehensive setting with multiple sensor modalities and an indoor smart home environment.

Table 2.4: Summary of Sensor Modalities in HAR Systems

Author	Modality	Sensors	Advantages	Disadvantages
Laudanski <i>et al.</i> (2015), Sztyley <i>et al.</i> (2017)	WSHAR	Accelerometer, heart rate, gyroscope, temperature sensors, etc., embedded in a smartphone, smart band, smartwatch, smart clothes or other devices	Small in size, less cost, flexible, easy to use, capture motion-related data	Cannot provide contextual information, suffer from the problem of arbitrary data caused by activities
Phillips <i>et al.</i> (2017), Jalal <i>et al.</i> (2017)	ASHAR	PIR, RFID, contact sensors, humidity sensors, etc.	Provide important contextual information and are less obtrusive	Complex sensor deployment, less information, and cannot be installed in any space
Tunca <i>et al.</i> (2014), Mehr <i>et al.</i> (2016), Luo <i>et al.</i> (2017)	CSHAR	Camera	Camera provides better information about objects, read global features accurately	Privacy issues, expensive, working in a constrained space, influenced by lighting conditions
Nakamura <i>et al.</i> (2010), Hayashi <i>et al.</i> (2015), Diethe <i>et al.</i> (2017)	HSHAR	Combination of camera and inertial sensors, fusion of PIR sensors and accelerometers, etc.	Capture rich information and utilise the strengths of different sensor modalities	Complex system structure and high cost, data fusion and synchronisation

The mHealth Extrasensory dataset (Chaqfeh *et al.* 2017) provides real-life data from smartphones, incorporating various sensors like an accelerometer, gyroscope, microphone, and GPS. The Daily Life Long Dataset (Bulling *et al.* 2014) is designed for long-term monitoring in a smart home, featuring ambient sensors like Passive Infrared (PIR), pressure, and contact sensors.

Comprehensive Adult Student Assessment Systems (CASAS) Kyoto (Cook and Das 2009) and CASAS Multi Resident (Cook *et al.* 2012) datasets focus on indoor smart home environments, employing ambient sensors such as PIR, door contacts, pressure, and temperature. The Extrasensory dataset (Vaizman *et al.* 2017) extends the real-life smartphone data to include diverse sensor inputs.

Various studies have utilised these datasets to address different HAR challenges, such as activity recognition, fall detection, gait analysis, and health monitoring. Researchers have employed machine learning and deep learning techniques to extract valuable insights and patterns from the data, paving the way for more

sophisticated and context-aware HAR systems.

Overall, these datasets and their research have significantly contributed to advancing the field of Human Activity Recognition and have laid the foundation for developing efficient, accurate, and adaptive activity recognition systems in diverse real-world settings.

Table 2.5: Existing Datasets for Human Activity Recognition Based on Wearable and Ambient Sensors

Dataset Name	Occupancy	Context	Sensing	Capture Duration	Participants
WISDM (Kwapisz <i>et al.</i> 2011a)	Single person	Indoor	Smartphone (Accelerometer, Gyroscope)	Variable	51
UCI-HAR (Anguita <i>et al.</i> 2013a)	Single person	Indoor	Wearable (Accelerometer, Gyroscope)	Variable	30
Opportunity (Roggen and Sgha 2012)	Single person	Indoor (Smart Home)	Wearable (Accelerometer, Gyroscope, Magnetometer)	Several days	4
mHealth Extrasensory (Chaqfeh <i>et al.</i> 2017)	Single person	Real-life	Smartphone (Accelerometer, Gyroscope, Microphone, GPS, etc.)	Variable	60
Daily Life Long (Bulling <i>et al.</i> 2014)	Single person	Indoor (Smart Home)	Ambient (PIR, Pressure, Contact Sensors)	Long-term (Months to Years)	5
CASAS Kyoto (Cook and Das 2009)	Single person	Indoor (Smart Home)	Ambient (PIR, Door Contact, Pressure, Temperature)	Several weeks	1
CASAS Multi Resident (Cook <i>et al.</i> 2015)	Multiple residents	Indoor (Smart Home)	Ambient (PIR, Door Contact, Pressure, Temperature)	Several weeks	7
Extrasensory (Vaizman <i>et al.</i> 2017)	Single person	Real-life	Smartphone (Accelerometer, Gyroscope, Microphone, GPS, etc.)	Variable	60
Proposed Log-hour Data Collection (Chapter 4)	Single person	Real-life	Smartphone (Accelerometer, Gyroscope, PIR, load, relay sensors)	Several months	12

## 2.3 Patterns in the Daily Activities

Once the Activities are recognised in a smart home, there can be a pattern in the human activities. Over the period, there can be a frequency in which activities are performed. A person in a smart home may perform a specific task on a specific day or over the week with a noticeable frequency or count. Further, the activities can also be classified as normal and abnormal activities.

One popular approach is to extend current data mining techniques in the context of activity recognition tasks. These methods look for regularly occurring interactions and discover significant patterns concerning measures such as frequency or periodicity. For example, Heierman and Cook (2003) discovered sub-sequences, or episodes, that are closely related in time. These episodes may be partially ordered and are evaluated based on information theory principles.

Gu *et al.* (2009) use a frequency measure to discover emerging patterns and use feature relevance to segment the boundary of any adjacent activities. There are also methods for the simultaneous discovery of frequent and periodic patterns using the minimum description length principle.

In Abnormal activity detection, commonly occurring abnormal actions in the day-to-day lives of elderly people, such as 'chest pain', 'headache', 'fainting', 'falling backwards' and 'falling forward', can be detected. This will help the health care system to report to the concerned person immediately once abnormal activities are detected. Besides behaviour modelling, detecting behaviour changes (anomaly detection) is another crucial and challenging task. Anomaly detection is the process of detecting behavioural "changes" of occupancy's usual lifestyle pattern, and the concern has been that unusual data is very rarely seen to be modelled as abnormal behaviour for classification (Dhiman and Vishwakarma 2019).

The user's activity characteristics are influenced by their habits. The surveys done on people's daily activities found that 47% of the participants' daily activities might coincide in space, and the consistency in people's daily lives forms their habits. The author proposed habit extraction methods based on the corresponding activity probability and calculation formulas of the habit strength since different habits have different variation characteristics on habit strength and time zone. Further, the self-learning algorithms on time zone and habit strength threshold to obtain the suitable parameters (Wood *et al.* 2005) (Bakar *et al.* 2016).

Researchers have also explored various approaches to uncover meaningful patterns in HAR datasets collected from different sensors in diverse contexts. Some commonly used techniques include frequent item-sets mining, association rules, and granularity-based analysis. For instance, Kwapisz *et al.* Kwapisz *et al.* (2011a) utilised the Apriori algorithm to mine frequent itemsets in the WISDM Dataset collected from smartphones with accelerometers and gyroscopes, identifying activities like walking, jogging, upstairs, and downstairs. Anguita *et al.* Anguita *et al.* (2013a) applied the Frequent Pattern Growth ([FP-Growth](#)) algorithm to the [UCI-HAR](#) Dataset, which includes data from wearable sensors like

accelerometers and gyroscopes, to discover association rules related to activities such as walking, sitting, standing, and laying.

On the other hand, Stisen *et al.* (2015) used a granularity-based approach on the Opportunity Dataset, collected from wearable sensors with accelerometers, gyroscopes, and magnetometers, to define temporal logic for recognising various activities. Fernandez *et al.* (2015) employed the **FP-Growth** algorithm on the mHealth Extrasensory Dataset, gathered from smartphones equipped with multiple sensors, to extract frequent itemsets associated with various activities. Cook *et al.* (2012) and Cook and Das (2009) utilised the Apriori algorithm and temporal logic, respectively, on the Daily Life Long Dataset and **CASAS** Kyoto Dataset, both collected from ambient sensors in smart home environments, to identify patterns related to daily activities. Association rules were explored by Chaqfeh *et al.* (2017) on the Extrasensory Dataset, which comprises smartphone sensor data, to discover correlations among various activities. Mohamed *et al.* (2012) performed granularity-based analysis on Kinect sensor data in Smart Home Activities to analyse human behaviour patterns.

The outcomes of these studies were valuable in revealing insights into human behaviour and activity patterns, leading to an improved understanding of activity recognition in various contexts. Frequent itemsets, association rules, and granularity-based techniques provided meaningful patterns that can be utilised in developing more accurate and context-aware human activity recognition systems. These findings have applications in areas like healthcare, smart homes, and activity monitoring, contributing to enhanced user experience, well-being, and automation of tasks in daily life.

### 2.3.1 Anomaly Detection in the Activities

This section summarises the abnormal activities and anomaly detection in the smart home environment. Anomalies can be detected by wearable sensor-based, camera or vision sensor-based. These activities are classified into Spatial anomaly activities, Temporal anomaly activities, Duration-based anomaly and Sequence-based anomalies. The following table 2.7 summarises the research in anomaly detection.

Table 2.7 provides insights into the diverse methods employed for anomaly detection in human activity recognition. For instance, Yin *et al.* Yin *et al.* (2008) employ an iterative adaptation procedure and KNLR adaptation techniques using specialised sensors to discern rare abnormal activities from common normal ac-

Table 2.6: Pattern Mining Techniques in Human Activity Recognition

Authors	Pattern Mining Technique	Dataset	Sensors Used	Activities Identified	Outcome
Kwapisz <i>et al.</i> (2011a)	Frequent Itemsets (Apriori)	WISDM Dataset	Smartphone (Accelerometer, Gyroscope)	Walking, Jogging, Upstairs, Downstairs	Discover frequent activity patterns.
Anguita <i>et al.</i> (2013a)	Association Rules (FP-Growth)	UCI HAR Dataset	Wearable (Accelerometer, Gyroscope)	Walking, Sitting, Standing, Laying	Uncover correlations among activities.
Stisen <i>et al.</i> (2015)	Granularity-based Analysis	Opportunity Dataset	Wearable (Accelerometer, Gyroscope, Magnetometer)	Various Daily Activities	Define temporal logic for activity recognition.
Fernandez <i>et al.</i> (2015)	Frequent Itemsets (FP-Growth)	mHealth Extrasensory Dataset	Smartphone (Accelerometer, Gyroscope, Microphone, GPS, etc.)	Various Daily Activities	Extract frequent activity patterns.
Cook <i>et al.</i> (2012); Cook and Das (2009)	Frequent Itemsets (Apriori), Temporal Logic	Daily Life Long Dataset, CASAS Kyoto	Ambient (PIR, Door Contact, Pressure, Temperature)	Daily Activities	Identify patterns in smart home environments.
Chaqfeh <i>et al.</i> (2017)	Association Rules	Extrasensory Dataset	Smartphone (Accelerometer, Gyroscope, Microphone, GPS, etc.)	Various Activities	Discover correlations among activities.
Mohamed <i>et al.</i> (2012)	Granularity-based Analysis	Kinect Sensor Data	Smart Home Activities	Human Behavior Patterns	Analyze patterns in smart home environments.
Proposed behavioural pattern analysis(Chapter 5)	Adaptive granularity and comprehensive	Smartphone, PIR, load, relay sensors	Smart Home Activities	Human Behavior Patterns	Analyze behavioural patterns of elderly

tivities, such as slipping or falling. On the other hand, Gutchess *et al.* Gutchess *et al.* (2007) leverage automated video surveillance and Hidden Markov Models (HMM) to analyse path tracking and identify anomalies like pedestrian running or unusual timing activities. Teng *et al.* Teng *et al.* (2005) focus on wearable sensors and accelerometer data to detect anomalies related to unconsciousness by analysing active and inactive signals. Fahad *et al.* Fahad and Rajarajan (2015) delve into video-based anomaly detection in smart homes using score-based methods and timing sequence anomalies from Kasteren-HA and Kasteren-HB smart home datasets.

The work of Zaidi *et al.* Zaidi *et al.* (2017) employs SVM classifiers for video-based activity detection, with manual annotations for abnormal activities like

Table 2.7: Anomaly Detection in Human Activity Recognition

Author	Methodology	Anomaly	Remarks
Yin et al., 2008 Yin et al. (2008)	Uses special sensors (light, temperature, microphone, accelerometer) to detect rare abnormal activities versus frequent normal activities. Iterative Adaptation procedure and KNLR adaptation techniques are used	Slipping on the floor, Falling down on the floor	Over time, abnormal activities themselves can become normal. Obtrusiveness.
Gutchess et al., 2007 Gutchess et al. (2007)	Automated video surveillance using HMM for classification. Path tracking is analyzed to detect anomalies	Pedestrian running, Unusual timing activities	Activities are assumed and worked only on the datasets.
Teng et al., 2015 Teng et al. (2005)	Wearable sensor-based anomaly detection based on accelerometer data. Uses active and inactive signals generated by sensors	Unconsciousness	Active and inactive sensors do not infer any daily activities.
Fahad et al., 2015 Fahad and Rajarajan (2015)	Video-based anomaly detection in smart home using Kasteren-HA and Kasteren-HB smart home dataset. Uses score-based methods (regularity or duration score)	Timing sequence-based anomaly	Based on the path only, motion-based actions can be inferred.
Zaidi et al., 2017 Zaidi et al. (2017)	Video-based activity detection using SVM classifiers. Manual annotations for abnormal activities	Hiding, Running, Shouting, Unorthodox movement	Manual annotation may not be feasible for high accuracy. Not suited for real-time analysis.
Poh et al., 2019 Poh et al. (2019)	Anomaly detections compared with LSTM and HMM-based classifiers. Video-based activity detection using annotated abnormal and normal activities	Daily-based abnormal activities	Results can be improved using Bidirectional LSTM.
Proposed anomaly detection (Section 5.6)	classification of unnatural, abnormal and normal activities	Daily-based abnormal activities	User Context based anomalies.

hiding, running, shouting, and unorthodox movement. Poh et al. Poh et al. (2019) compare anomaly detections using LSTM and HMM-based classifiers for video-based activity detection, focusing on daily-based abnormal activities.

## 2.4 Research Gaps, Issues and Challenges

Through an extensive review of the existing literature, several critical research issues and challenges have emerged in the domain of HAR within smart home

environments. These challenges pose intriguing opportunities for further investigation and innovation. The identified research issues and challenges in HAR within smart home environments align closely with the literature review's objectives and this research's overarching goals. The following key issues have been identified:

1. *Sensor Selection*: The challenge of selecting appropriate sensors directly relates to the objective of designing and developing an effective HAR system using deep learning techniques. The choice of sensors impacts the data quality and directly influences the accuracy and robustness of the system.
2. *Feature Extraction*: The challenge of feature extraction is critical to creating and curating a dataset for long-hour human activities. Selecting informative and discriminative features is crucial in building a comprehensive and representative dataset.
3. *Data Fusion*: Addressing the data fusion challenge aligns with developing a human behavioural pattern recognition system. Effectively integrating data from multiple modalities is essential in capturing comprehensive activity patterns.
4. *Data Annotation and Labeling*: Streamlining the data annotation and labelling process connects with the objective of dataset creation. Efficient data annotation tools facilitate the development of large-scale and accurately labelled datasets.
5. *Deployment Challenges*: Overcoming deployment challenges is essential for implementing HAR systems in real-world smart home environments. The accurate positioning of sensors is crucial for achieving reliable and context-aware activity recognition.
6. *Complex Human Behaviour*: Understanding and recognising complex human behaviour aligns to develop an effective HAR system using deep learning techniques. Advanced algorithms are required to capture intricate activity patterns and variations.
7. *Inter-Class Similarity*: Accurately differentiating between similar activities is vital for the effectiveness of HAR systems. This challenge directly links to the objective of designing robust and accurate HAR systems.

8. *Environmental Influence*: Addressing the impact of environmental factors on recognition systems connects to developing a human behavioural pattern recognition system in a smart home environment. This involves building models that can adapt to varying environmental conditions.
9. *Privacy and Security*: Ensuring data privacy and security is critical when dealing with sensor data in smart home environments. This challenge is relevant to all objectives as data security is a fundamental concern in HAR research.
10. *Calibration and Configuration of Sensors*: Proper calibration and configuration of sensors are vital for accurate data collection, which aligns with all objectives to develop effective HAR systems and datasets.
11. *Adaptability of Sensors*: Ensuring the adaptability of sensors connects with the objective of creating wearable and ambient sensor-based datasets and designing effective HAR systems for smart homes.
12. *Real-Time Processing*: Achieving low-latency processing while maintaining high accuracy is crucial for real-time HAR applications, aligning to develop efficient HAR systems using deep learning techniques.
13. *Multimodal Data Fusion*: Addressing the challenge of data fusion aligns to create multimodal sensor-based datasets and design hybrid sensor-based HAR systems.
14. *Label Noise and Annotation Errors*: Handling label noise and annotation errors is essential for ensuring the accuracy and reliability of HAR systems, which connect with all objectives.
15. *Transfer Learning*: Transfer learning can enhance HAR systems and align to develop efficient HAR techniques using deep learning.

The following research gaps are considered to address in the proposed framework; Feature extraction, Data Fusion, Data Annotation, Synchronization of heterogeneous data, Real time long hour data collection. These endeavours contribute to the advancement of ambient intelligence in smart home environments, improving residents' quality of life and innovation in human activity recognition.

## 2.5 Summary

The literature review presented a comprehensive survey of various Human Activity Recognition (HAR) aspects in smart home environments. We explored different techniques and approaches used in the field and the datasets and applications utilized. Various challenges and issues are highlighted for various sensor modalities. By leveraging the insights from the literature review, we will be addressing most of these challenges in the upcoming chapters to design and develop an efficient human activity recognition system using deep learning techniques, curated datasets for long-hour human activities, and develop a human behavioural pattern recognition system in a smart home environment.

## Chapter 3

# Human Activity Recognition using Deep Learning Techniques

Deep learning models, such as [CNNs](#) and [RNNs](#), have demonstrated remarkable performance in various domains, including machine vision, robotics, and speech recognition (Zhihao *et al.* 2020; Chattopadhyay and Maitra 2022). These models have the ability to automatically extract abstract features from raw data, eliminating the need for manual feature engineering. However, the performance of deep learning models heavily depends on the selection of hyperparameters, which control the learning process and affect the model's efficiency (Shankar *et al.* 2020). The challenge lies in finding the optimal set of hyperparameters, as the impact of each hyperparameter is interrelated and can influence the model's performance (Zela *et al.* 2018). The traditional approach of manually searching for the best hyperparameters is time-consuming and impractical in complex deep learning architectures (Feurer and Hutter 2019).

To address these challenges, this research proposes a novel approach to enhance HAR using deep learning techniques and [SMO](#) algorithms. The deep learning models, including RNNs, CNNs, and LSTMs, are utilized to build a framework that can effectively predict, classify, and recognize human activities. The [SMO](#) algorithm is employed to optimize the hyperparameters of the deep learning models, improving their performance and efficiency. By combining the strengths of deep learning and SMO, this research aims to achieve accurate and reliable human activity recognition in real-world scenarios.

Overall, the proposed approach offers a promising solution to enhance the performance of [HAR](#) by leveraging the power of deep learning models and optimizing their hyperparameters using [SMO](#) algorithms. The improved accuracy

and efficiency of the proposed framework can benefit various applications, including healthcare monitoring, assisted living systems, and activity tracking in smart homes.

## 3.1 Proposed Human Activity Recognition using Various Deep Learning Techniques

This work analyses data collected from wearable sensors using publicly available datasets [UCI-HAR](#) and [WISDM](#). These datasets contain inertial movement data captured by sensors like accelerometers and gyroscopes. The data collected from these sensors are time-series recorded at regular intervals based on the specific activity (e.g., standing, walking, etc.).

Deep Learning (DL) methods extract meaningful information from the collected data. DL-based techniques, including Convolutional Neural Networks, Long Short-Term Memory, and Gated Recurrent Units, are applied to process and analyse the data. These DL models are specifically chosen due to their ability to handle sequential data and capture complex patterns in the sensor readings.

The effectiveness of the methods is evaluated using the [UCI-HAR](#) and [WISDM](#) datasets, as described in the dataset section of this research. These datasets provide labelled samples of various human activities, allowing for the training and testing of the models. Applying these models to the datasets aims to accurately recognise and classify different activities based on the features extracted from the sensor data.

The work focuses on utilising wearable sensor data, leveraging DL techniques such as CNN, LSTM, and GRU, and evaluating their performance. The objective is to develop robust and accurate models for activity recognition based on the collected data, enabling advancements in human activity pattern analysis and applications in areas such as smart homes. Figure 3.1 shows the proposed HAR using deep learning techniques.

### 3.1.1 Raw Data and Preprocessing

The data collected from wearable sensors capture changes in inertial movements, specifically through accelerometer and gyroscope sensors. Accelerometer sensors measure acceleration forces, while gyroscope sensors measure angular velocity. By combining the readings from these two types of sensors, the movement of the sensors attached to the body can be recorded in different directions (x, y, z-axis).

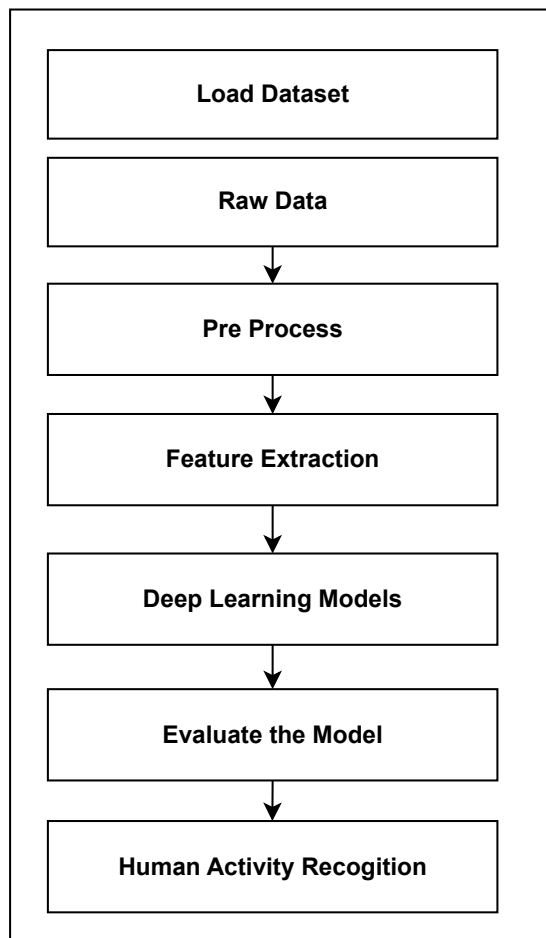


Figure 3.1: Proposed HAR using Deep Learning Techniques

The [WISDM](#) dataset consists of time series data, which needs to be segmented into smaller samples. In the context of capturing sensor data from the embedded accelerometer and gyroscope, the term "3-axial" refers to the three-dimensional nature of the measurements along the X, Y, and Z axes. The accelerometer captures linear acceleration, representing the change in velocity along the three axes, while the gyroscope measures angular velocity, representing the rotation rate around the three axes.

The data was recorded at a constant rate of 50 Hz, meaning measurements were taken 50 times per second. This high sampling rate allows for capturing detailed and accurate motion information. Before analysing the sensor data, preprocessing was performed to enhance data quality and reduce noise. Noise filters were applied to remove any unwanted disturbances or random fluctuations in the sensor signals.

To facilitate further analysis and feature extraction, the preprocessed sensor signals were divided into fixed-width sliding windows. Each window had a duration of 2.56 seconds, and consecutive windows had a 50% overlap, meaning that each subsequent window started halfway (1.28 seconds) into the previous window. The data captured across different time intervals can be compared and analysed using overlapping windows, providing a continuous and coherent view of the motion patterns. Each window contained 128 readings, which means that within a 2.56-second window, there were 128 measurements for both the accelerometer and gyroscope data. These readings are essential for feeding the data into machine learning models or other analysis techniques to recognise and understand human activities or movements. The sliding window approach allows the dataset to be structured to capture temporal dependencies and extract meaningful features for activity recognition and other motion-related applications.

To ensure the reliability of the results, data cleaning procedures are implemented. Duplicates and null values are removed from the datasets, ensuring the integrity of the collected data. Additionally, efforts are made to balance the classes in the datasets, which is essential to avoid biases and ensure a fair evaluation of the models.

By conducting experiments on these datasets and applying data preprocessing techniques, the study aims to demonstrate the effectiveness of the proposed approach in recognising and classifying human activities accurately. The detailed preprocessing steps and dataset characteristics are crucial in providing reliable and meaningful results, contributing to the advancement of human activity recognition in wearable sensor applications.

### 3.1.2 Feature Extraction

A combination of Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) architectures is employed to extract relevant features from the sensor data. The CNN component performs one-dimensional convolutions, enabling the model to capture spatial features and patterns from the raw sensor data. By applying activation functions like Rectified Linear Unit (ReLU) and employing kernel size filters, the CNN introduces non-linearity and enhances the model's ability to learn complex patterns. Max pooling is then applied to retain the most salient features from the generated feature maps, maximising the meaningful elements.

The choice of utilising CNN and RNN architectures for feature extraction is justified by their ability to handle time series data and capture spatial and temporal patterns. The CNN is well-suited for extracting spatial features from raw sensor data, eliminating the need for manual feature engineering. The subsequent integration of RNN allows the model to capture temporal dependencies and effectively recognise activity patterns that evolve. This combined approach enables the model to learn meaningful representations of human activities, facilitating accurate and robust activity recognition and classification.

### 3.1.3 Deep Learning Models

In the time series dataset, the data in a window depends on the input of the previous data. LSTM layers help in extracting the temporal features in sequential data. The main work of the LSTM is by having three different gates, namely Input Gate- Equation (3.2), Output Gate- Equation (3.1), and Forget Gate- Equation (3.3) also shown in the Figure 3.2. The information  $x_t$  goes from state  $s_{t-1}$  to  $s_t$  state- Equation (3.5). Nevertheless, from the  $x_t$  information, the model can selectively read, write and forget the information. In the selective write operation, only some data is written to the next state. The written portion should be significant to the model to reduce the loss.  $W$  and  $U$  matrix contains the weights of the input and recurrent connections. The sigmoid function is used to make the fraction between 0 and 1.

$$o_t = \sigma(W_o h_{t-1} + U_o x_t + b_o) \quad (3.1)$$

$$i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i) \quad (3.2)$$

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f) \quad (3.3)$$

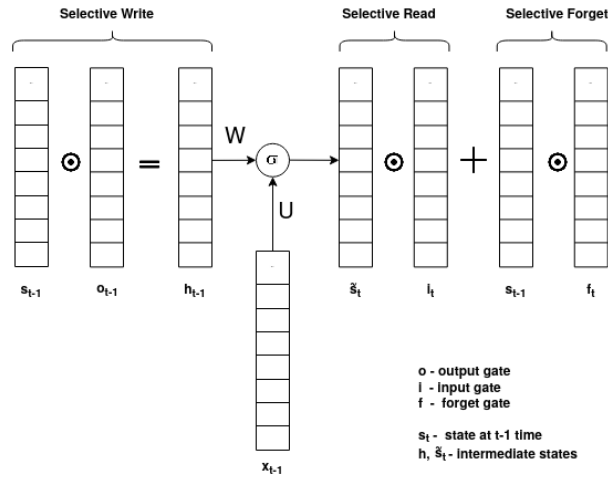


Figure 3.2: General Working of the LSTM node.

$$\tilde{s}_t = \sigma(W h_{t-1} + U x_t + b) \quad (3.4)$$

$$s_t = f_t \odot s_{t-1} + i_t \odot \tilde{s}_t \quad (3.5)$$

$$h_t = o_t \odot \sigma(s_t) \quad (3.6)$$

Typical CNN architecture has the Convolution layer, pooling layer, flattening and fully connected layers. This work uses CNN and RNN to use the time series data. One-dimensional convolution network performs well in feature extraction from raw data. That means there is no need for domain expertise to extract the features manually here. A feature map is obtained by obtaining the filters of the input data and collected as an output. In CNN, ReLU is used as the activation function and kernel size filter. Max pooling is used to consider the essential features from the previous feature maps. This maximises the elements in the feature map.

In the proposed methodology, the LSTM is used as a specific type of RNN that excels at capturing long-term dependencies in time-series data, such as human activity recognition. RNNs are well-suited for this task as they can consider previous events using loops that persist information. LSTM address the challenge of long-term dependency by carefully regulating the flow of information using different gates. This work utilises a single LSTM network with 100 units. The LSTM layers receive input from the 1D convolutional layer. Dense layers and activation functions are incorporated into the network to introduce non-linearity to the neuron outputs. Finally, a softmax layer is employed to normalise the output probabilities as the task involves multiclass (six) classification.

The combination of two CNN layers and one LSTM layer is stacked together.

This approach, known as CNN-LSTM piling, allows leveraging the advantages of both CNN and LSTM layers. However, it is observed through experiments that stacking multiple layers of CNN and LSTM can significantly increase computation load without proportional performance improvement. Therefore, this work chooses the combination of two CNN layers and one LSTM layer. Another variation is the hybrid layers, where the CNN and LSTM layers are combined.

A dense layer is placed between the CNN and LSTM layers to create a fully connected network. This dense layer helps in establishing connections between the layers. Additionally, this work explores the utilisation of a Bi-directional LSTM network, which involves training the LSTM network twice on the time-series input data: once on the original data and another time on the reverse of the input. This approach increases the number of learning parameters and enables better training with the available data.

Another variation of the LSTM network is the GRU. GRUs address the vanishing gradient problem in RNNs and are characterised by their ability to retain and filter relevant information using update and reset gates. The update gate determines which data from the previous step is passed to the next step, akin to the input and forget gates in LSTM. The reset gate decides which data to forget and not propagate to the next step. These mechanisms effectively solve the vanishing gradient problem encountered in traditional RNNs.

The proposed GRU neural network follows the following steps, as illustrated in Figure 3.3:

1. Input: Raw sensor data, with a vector size of 128.
2. CNN layers: Apply one-dimensional convolution operations on the input, with the ReLU activation function. Another convolution operation follows this.
3. Dropout: To mitigate overfitting, a dropout layer with a rate of 0.5 is included in the network.
4. Max pooling: As multiple layers are present, the pooling layer operates on each feature map independently to generate a new set of pooled feature maps, preserving the order of the CNN layers.
5. Flatten: This step is necessary to create a proper input vector for the subsequent GRU layer.

6. GRU layers: With 32 units, the output from the previous step is fed into another GRU layer.
7. Dense: The output from all the preceding layers is concatenated using a dense layer.
8. Softmax Activation: The final layer, preceded by batch normalisation of all the layers, is responsible for outputting the probabilities for the multiple classes of activities.

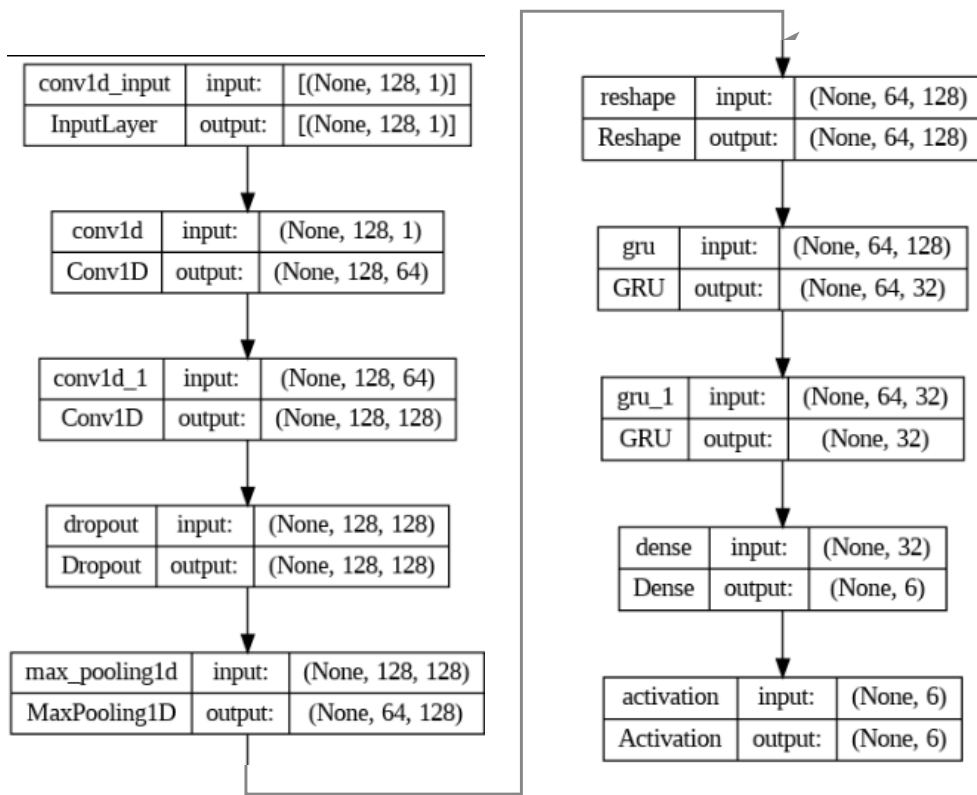


Figure 3.3: The Proposed GRU Neural Network for HAR.

### 3.1.4 Results and Discussion

The following section discusses the experiment setup, the datasets used, the performance measures, and the results and discussion.

#### 3.1.4.1 Datasets

All the experiments were implemented using Python 3.6 programming language and executed using NVIDIA DGX Station server with 4X Tesla V100 and 500

TFLOPS. The external memory of a minimum of 16GB is used to store the datasets and related data.

The **UCI-HAR** Dataset (Anguita *et al.* 2013a) is utilised in this work for human activity recognition. This dataset consists of recordings from 30 subjects performing various activities such as walking, walking upstairs, downstairs, sitting, standing, and lying. The activities are captured using a waist-mounted smartphone with inertial sensors like an accelerometer and gyroscope. The dataset is a multivariate time-series dataset with 10,299 instances and 561 attributes. The signals are segmented into windows of 2.56 seconds, and the dataset has a sampling rate of 50 Hz. The dataset is divided into a 70:30 ratio for training and testing purposes.

The **WISDM** Dataset (Kwapisz *et al.* 2011a) is another dataset used in this work for human activity recognition. This dataset is collected in a laboratory-controlled environment and involves 36 subjects performing activities like walking, sitting, jogging, going downstairs, going upstairs, and standing. Similar to the UCI-HAR dataset, smartphones with inertial sensors capture the activities. The dataset contains 1,098,207 examples with a sampling rate of 20 Hz. It has a total of 6 attributes, and the class distribution of the dataset is as follows: walking (38.6%), jogging (31.2%), upstairs (11.2%), downstairs (9.1%), sitting (5.5%), and standing (4.4%).

The class distribution of both datasets is illustrated in Figure 3.4 and Figure 3.5. These datasets provide a diverse range of human activities and serve as valuable resources for training and evaluating this work’s human activity recognition models.

### 3.1.4.2 Performance Measures

We have accuracy as performance measures to evaluate experiments of the deep learning models. Accuracy is the ratio of correctly classified samples to the total number of pieces. Activities can be classified as True Positives (TP) and True Negatives (TN) when they are classified correctly and as False Negatives (FN) and False Positives (FP) when they are wrongly classified. The performance measures are TP, TN, FP, and FN. Precision is the ratio of positives predicted correctly to the total number of samples classified as positives.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (3.7)$$

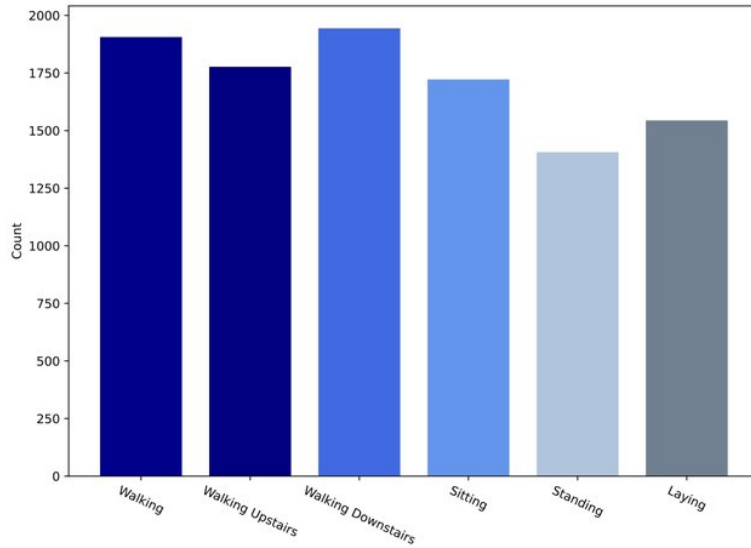


Figure 3.4: Class Distribution of UCI-HAR Dataset (Mekruksavanich and Jitpatanakul 2021)

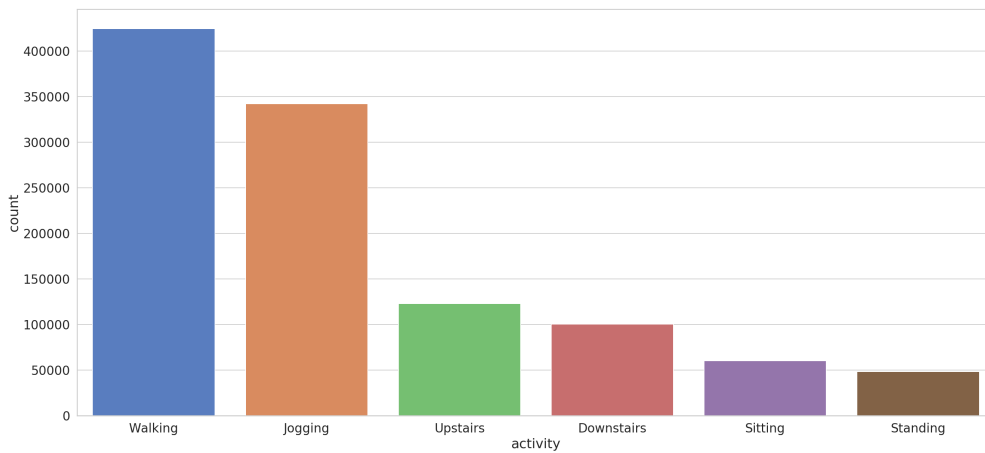


Figure 3.5: Class Distribution of WISDM Dataset (Kwapisz *et al.* 2011a)

### 3.1.4.3 Results and Discussion

We ran the deep learning models mentioned in the methodology on the two datasets above. These experiments are carried out to recognise the six activities. The data is divided into 70% subjects for training and 30% for testing as given in the dataset. The data is divided into 2.56 seconds of the window with a 50% overlapping window.

Figure 3.6 and 3.7 show that recognising the sitting and standing activity is more accurate than most models experimented. The equation 3.7 calculates the accuracy. In contrast, activities like downstairs, upstairs and walking and jogging have lesser accuracy than others. Also, it is observed from Figure 3.8 that the

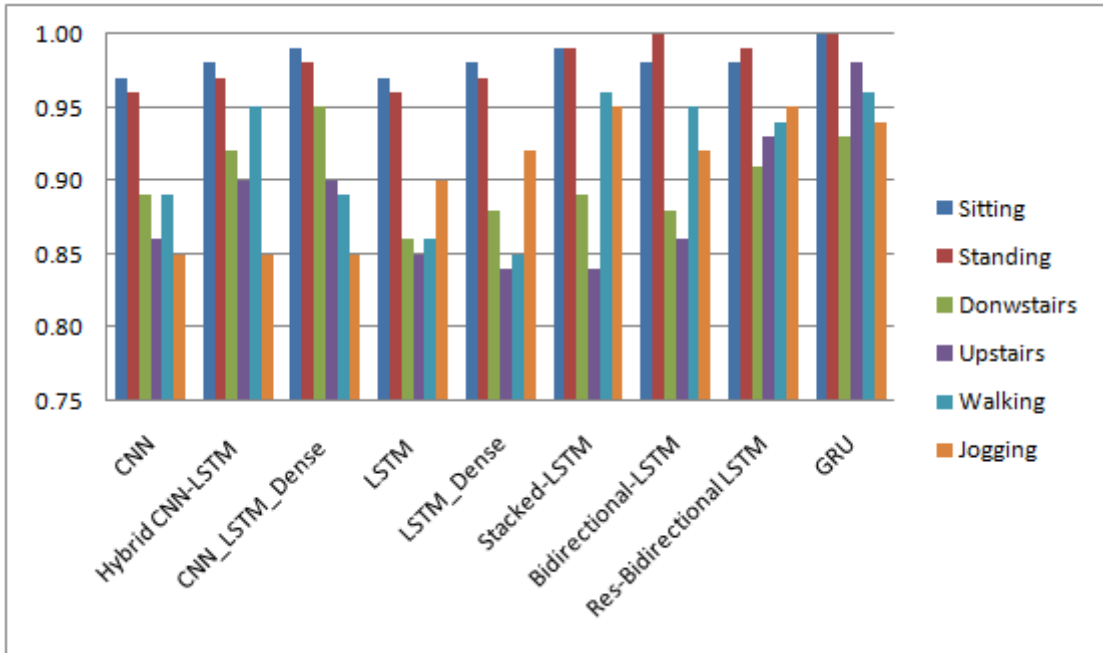


Figure 3.6: The Accuracy Comparison Across Deep Learning Models on the UCI-HAR Dataset Activity Wise

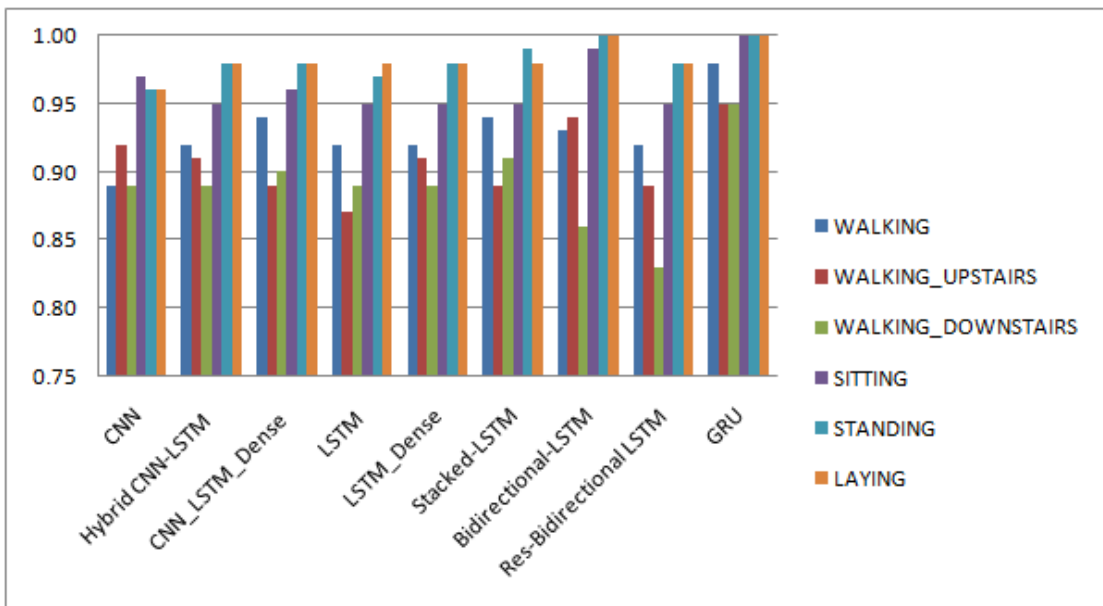


Figure 3.7: The Accuracy Comparison Across Deep Learning Models on the WISDM Dataset Activity Wise

accuracy for recognising UCI-HAR dataset activities is lesser than the WISDM dataset. This is because of the nature of the dataset explained in the dataset section. The accuracy of the various models is summarised in Table 3.1. The Gated Recurrent Network provides the best result of 98% accuracy in recognising

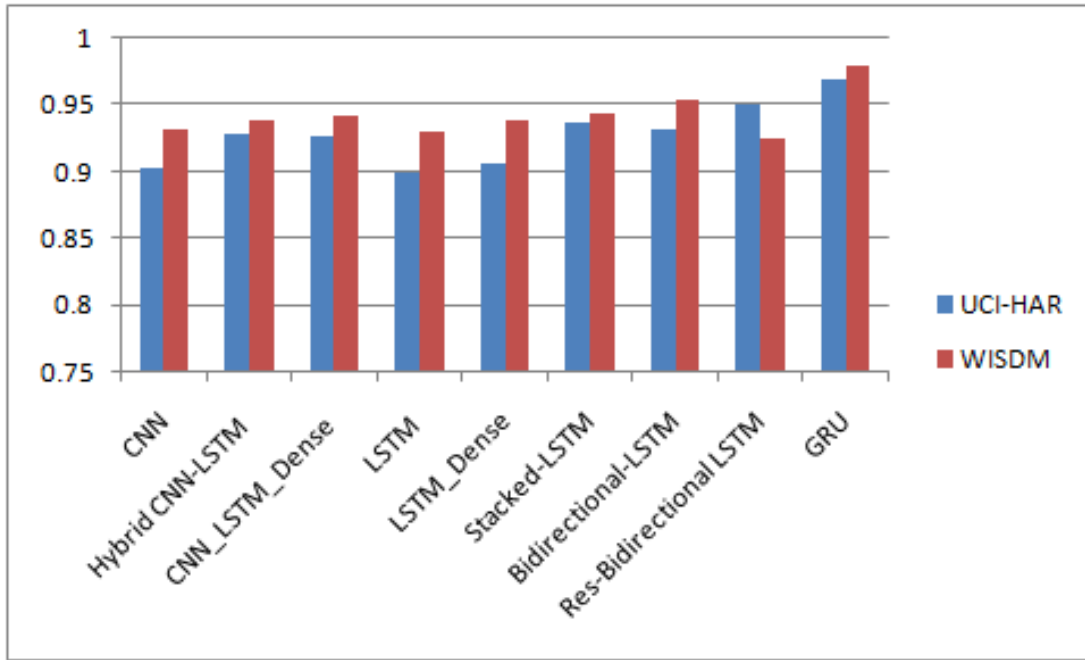


Figure 3.8: The Accuracy Comparison Across Deep Learning Models on WISDM and UCI-HAR Dataset

Table 3.1: The Accuracy Comparison Across Deep Learning Models on WISDM and UCI-HAR Dataset

DL Models	Dataset	
	UCI-HAR	WISDM
CNN	90.33	93.17
Hybrid CNN-LSTM	92.83	93.83
CNN-LSTM-Dense	92.67	94.17
LSTM	90.00	93.00
Stacked-LSTM	93.67	94.33
Bidirectional-LSTM	93.17	95.33
Res-Bidirectional LSTM	95.00	92.5
<b>Proposed GRU</b>	<b>96.83</b>	<b>98.00</b>

the overall activities for the WISDM dataset and 96.83% for the UCI-HAR dataset. The highest accuracy is because of the shorter time sequence in the dataset. Also, the time to train the GRU model is less than the LSTM because of the two gates used compared to the three gates in LSTM. The terms are abbreviated as

Hybrid CNN-LSTM as HC-LSTM, CNN-LSTM-Dense as CNN-LD, LSTM-Dense as LSTM-D, Stacked-LSTM as S-LSTM, Bidirectional-LSTM as B-LSTM, Res-Bidirectional-LSTM as RB-LSTM.

In this work on human activity recognition using accelerometer and gyroscope sensor data, the GRU architecture has demonstrated superior performance. The simplified structure of GRU, which includes fewer parameters compared to LSTM, proves to be advantageous in efficiently capturing long-term dependencies and adapting to varying patterns within the sequential data. This efficiency, coupled with the computational considerations of this dataset, makes GRU particularly well-suited for achieving high accuracy in the task.

There are various deep learning models available for the recognition of human activities. The deep learning models overcome the problem of selecting the features manually. The GRU model outperformed the accuracy in recognition of the activities from both the datasets of UCI-HAR and WISDM. The ability to use the time series data with the varied sequence has provided the advantage to the GRU model to recognise. The result also showed that some class of activities has the highest accuracy rates for all the models.

Moreover, the effectiveness of the models can be further enhanced by utilising optimisation techniques. These methods are pivotal in selecting optimal features and hyperparameters, augmenting training efficiency and overall performance metrics. The [DNN-SMO](#) is proposed to improve the performance in the following section.

## 3.2 Proposed Human Activity Recognition using Deep Learning Techniques with Spider Monkey Optimization

The proposed work aims to develop a novel [DNN-SMO](#) for HAR using wearable and camera sensor data. HAR is essential in various domains, such as injury detection, entertainment, ambient assisted living, and smart environments. However, the complexity of the data and the high computation time pose challenges in accurate activity recognition.

The DNN-SMO approach addresses these challenges by leveraging the power of deep learning and optimisation techniques. The experiment utilises four datasets, three based on wearable sensors, while the fourth is based on camera data from the [KTH](#) action dataset. The datasets are loaded and cleaned, ensuring they suit

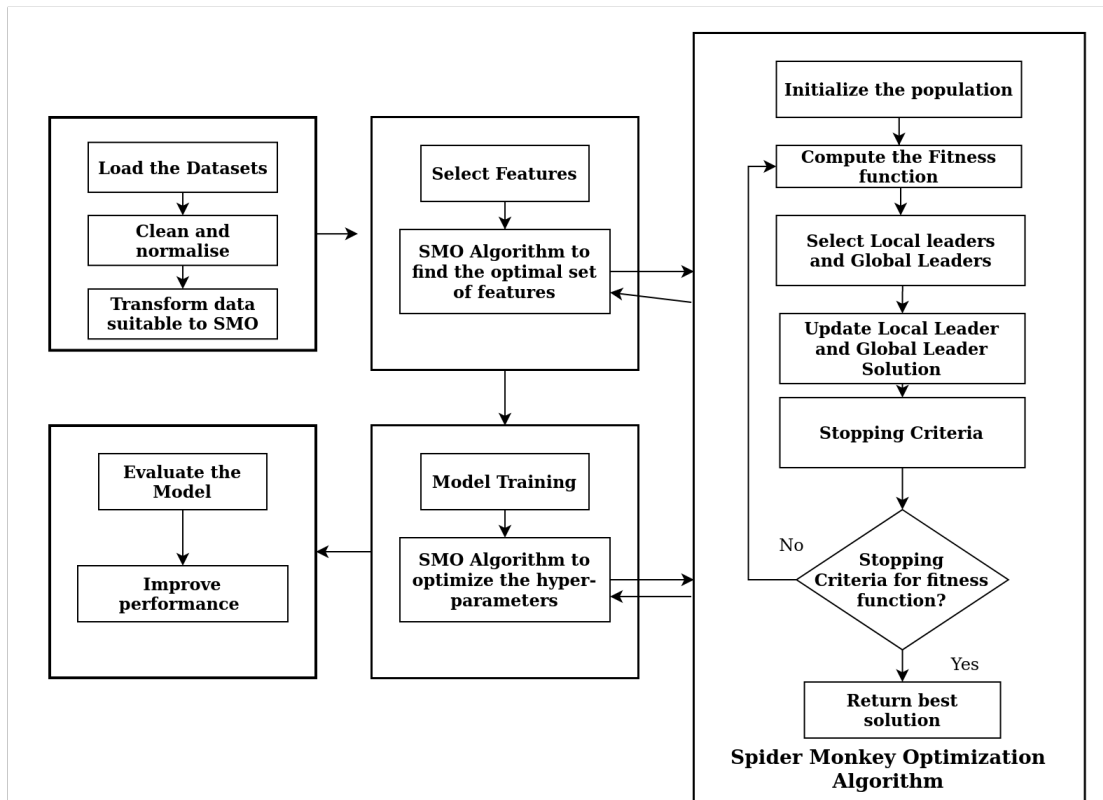


Figure 3.9: Proposed DNN Model with Spider Monkey Optimization.

the model. Preprocessing steps are applied to transform and prepare the data for the [SMO](#) algorithm.

Feature selection is a crucial step in HAR, and the proposed approach optimises the features using the Spider Monkey Optimization algorithm. This optimisation process helps identify the most relevant and informative features for accurate activity recognition. The Deep Learning Model is then trained using the optimised features, and hyperparameters are tuned based on the [SMO](#) algorithm.

The model is evaluated using the best parameters obtained from the optimisation process. The performance of the [DNN-SMO](#) model is assessed on the selected features, and the classification of human activities such as walking, sitting, up-stairs, standing, and running is performed.

The proposed methodology, depicted in [Figure 3.9](#), demonstrates a comprehensive framework for HAR using deep learning and optimisation techniques. By combining the strengths of deep neural networks in learning complex patterns and the efficiency of the Spider Monkey Optimization algorithm in feature selection, the proposed approach aims to enhance the accuracy and efficiency of activity recognition.

The proposed methodology consists of several key steps to achieve accurate

human activity recognition. These steps are outlined as follows:

**Dataset Training:** The collected human activity dataset, which includes daily living activities, is used to train the system. This dataset serves as the foundation for the subsequent steps.

**Development of DNN-SMO:** A novel DNN-SMO mechanism is developed. This mechanism incorporates the fitness function of the spider monkey into the hidden layer of the Deep Neural Network. The activity recognition performance is enhanced by integrating the spider monkey's optimisation capabilities.

**Feature Extraction and Fusion:** The local leader and global leader fitness functions are employed to improve feature extraction. These fitness functions aid in identifying the most relevant features from the sensor data. The extracted features are then fused at the feature level to capture comprehensive information before the classification process.

The fitness function is often used in optimization problems to quantify the difference between estimated and target values. It is represented as:

$$F = \sum_{i=1}^N (x_i - y_i)^2 \quad (3.8)$$

Where:

- $F$  is the fitness value.
- $x_i$  is the estimated value of the  $i$ th feature.
- $y_i$  is the  $i$ th target value.
- $N$  is the number of target values.

This function is used to guide optimization algorithms in finding the optimal set of features and hyperparameters.

**Human Activity Classification:** The proposed DNN-SMO approach effectively classifies various human activities such as standing, running, and sitting. Accurate activity recognition is achieved by leveraging the optimised features and the trained DNN-SMO model.

These key steps collectively form the basis of the proposed methodology for human activity recognition. The approach aims to improve the accuracy and performance of activity classification by integrating deep learning techniques, optimisation algorithms, and feature fusion.

### 3.2.1 Dataset Description

The datasets used in this work play a crucial role in evaluating the proposed HAR systems and optimising the fitness function. Multiple wearable and camera-based sensor datasets are considered to ensure the effectiveness and diversity of the algorithms.

**UCI-HAR** Dataset (Anguita *et al.* 2013b) and the **WISDM** Dataset (Kwapisz *et al.* 2011b) are explained in the Section 3.1.4.1. The class distribution of both datasets is also given in Figure 3.4 and 3.5. Along with these, two more datasets are experimented in this work.

**Action Database Dataset** (Schuldt *et al.* 2004): The database has six distinct types of human actions: walking, jogging, running, boxing, hand waving, and hand clapping. These actions were performed multiple times by a group of 25 subjects in four different scenarios. These scenarios include outdoor settings with and without scale variation (s1 and s2), an outdoor scenario with different clothing (s3), and an indoor environment (s4). In total, the database currently contains 2391 sequences of these actions. The sequences in this database were captured against uniform backgrounds using a static camera with a frame rate of 25 frames per second (fps). Subsequently, the sequences were downsampled to a spatial resolution of 160x120 pixels, resulting in an average sequence duration of approximately four seconds.

**PAMAP2** (Reiss and Stricker 2012): The benchmark PAMAP2 dataset uses the triaxial accelerometers, gyroscopes, and magnetometers on the subject's ankle, chest, and wrist regions. Eighteen household activities are labelled in the dataset. The activities like walking, running, cycling, ascending and descending stairs, house cleaning, lying down, sitting, and standing are considered for the experiment. There are a total of 9 subjects used for the data collection.

### 3.2.2 Flow of Proposed DNN-SMO Approach

All the experiments were implemented using Python 3.6 programming language and executed using NVIDIA DGX Station server with 4X Tesla V100 and 500 TFLOPS. The external memory of a minimum of 16GB is used to store the datasets and related data.

Initially, the data is read from the dataset mentioned. These data need to be preprocessed to suit the model chosen. Here, the datasets are filtered to remove the noise. Furthermore, the proper sliding windows are chosen to segment the activities. The next stage is to select the features for the data to feed to train the model. The collected activity datasets are initially trained into the proposed



Figure 3.10: Sample Video Frame from ACTION KTH Showing Walking Activity (Schuldt *et al.* 2004).

**DNN-SMO** approach. Here, the critical parameter is the human activity dataset obtained from the multimodal sensor to analyse human activity like walking and sitting. **SMO** is a population-based algorithm, and it is divided into six stages (Sharma *et al.* 2020). Table 3.2 describes all the notations used in the following section.

### 3.2.2.1 Pre-processing

The study involves three different datasets, each collected from separate sources. The preprocessing stage begins by individually processing each dataset to ensure data quality and consistency. Bandpass filters, such as Butterworth filters, are applied to remove noise from each dataset, followed by normalisation to eliminate negative values and bring the data within a standardised range.

For each dataset  $D_s$ , where  $s = 1, 2, 3$ , the following equation 3.9 represents the pre-processing procedure:

$$SM_{PQ}[d_s(t)] = \sum (y(t)x(t-1) + UR(0, 1) * [SM_{max(q)} - SM_{min(q)}]) \quad (3.9)$$

Table 3.2: Notations and Description used in the proposed DNN-SMO.

Notation	Description
$D_s$	Individual dataset from three different sources.
$SM_{PQ}[d_s(t)]$	Pre-processed dataset.
$y(t)$	Sensor input dataset.
$x(t - 1)$	Previous time step of the sensor input dataset.
$UR(0, 1)$	Random number uniformly allocated in the range $[0, 1]$ .
$SM_{max(q)}$	Upper bound of the $q^{th}$ search space.
$SM_{min(q)}$	Lower bound of the $q^{th}$ search space.
$SM_Q$	Unified dataset after combining preprocessed datasets.
$U(n, m)$	Convolved dataset for feature selection.
$s(m, u, v)$	Features extracted from the unified dataset.
$d(s - 1)$	Window size used in the previous step during raw data processing.
$a[(x_1, y_1), \dots, (x_n, y_n)]$	Collected dataset representing human actions.
$W$	Weight vector in the hyperplane equation.
$x_i$	Input vector for the hyperplane equation.
$B$	Bias term in the hyperplane equation.
$SM_N(p, q)$	Updated dataset on the local leader (LL) stage.
$SM^*$	$SM(P, Q) + UR(0, 1)$
$LL(Q)$	$Q^{th}$ dimension based on the local leader location.
$SM(R, Q)$	$Q^{th}$ dimension randomly selected from the local group.
$GL(Q)$	Location of the global leader in the $Q^{th}$ dimension.
$P(SM)$	Probability determination of the dataset.
$F(P)$	Fitness value of the $P^{th}$ dataset.
$h_{k_j}$	Output of the $j^{th}$ neuron in the $k^{th}$ classification layer.
$a_{k_{ij}}$	Offset of the $j^{th}$ neuron in the $k^{th}$ classification layer.
$w_{k_j}$	$j^{th}$ row of the weight matrix connecting $(k - 1)^{th}$ and $k^{th}$ classification layers.
$h_{l_j}$	Output of the $j^{th}$ neuron in the $l^{th}$ layer (softmax).
$sum$	Running sum during dataset updating.
$DU(i, j)$	$i^{th}$ and $j^{th}$ dimension based on dataset updating.
$LL(k, j)$	$j^{th}$ dimension based on group location of the $k^{th}$ local leader.
$y(j, i)$	Updated dataset based on fitness function.

Here,  $d_s(t)$  denotes the dataset with window size  $x(t)$ ,  $y(t)$  is the sensor input dataset, and  $UR(0, 1)$  represents a random number uniformly allocated in the range of  $[0, 1]$ . The terms  $SM_{max(q)}$  and  $SM_{min(q)}$  represent the upper and lower bounds of the  $q^{th}$  search space, respectively.

After the preprocessing of individual datasets, they are combined into a unified dataset for further analysis. This combination is performed by amalgamating

the preprocessed datasets into a single dataset,  $SM_Q$ , while retaining individual identities. Each dataset is labelled to indicate its source; additional metadata can be included for reference.

The unified dataset  $SM_Q$  is input for subsequent phases, such as feature extraction, selection, and classification. This approach allows the study to leverage the collective information from all three datasets, resulting in a more comprehensive and robust analysis of human activity recognition.

By carefully preprocessing each dataset and amalgamating them into a single dataset while preserving their characteristics, the study can benefit from the diversity and richness of information offered by multiple datasets, leading to improved performance and accuracy in identifying human activities.

### 3.2.2.2 Feature Extraction and Selection

This phase extracts a comprehensive set of features from the unified dataset  $SM_Q$  to capture relevant information for identifying human activities. The following features are calculated for each axis of the data:

- Average acceleration
- Standard Deviation
- Average absolute difference between the value of each of the 200 readings within the ED (Euclidean Distance) and the mean value over those 200 values (for each axis)
- Average Resultant Acceleration
- Time Between Peaks
- Binned Distribution

Combining these features results in approximately 156 features for both [UCI-HAR](#) and [WISDM](#) datasets. These features encompass temporal and frequency-based attributes, contributing to a more comprehensive analysis of human activities.

To improve the efficiency and effectiveness of the activity recognition model, a feature selection step is performed using the [SMO](#) algorithm. The primary goal of feature selection is to identify a subset of features most relevant to recognising specific human activities while minimising the number of features used.

The feature selection process is formulated as an optimisation problem, where the classifier's performance is maximised. The [SMO](#) algorithm efficiently explores

different feature subsets, aiming to find the optimal combination that yields the best performance for activity recognition. The **SMO** algorithm converges on the optimal subset of features, ensuring that only the most informative features are considered for further analysis.

The following equation represents this feature selection step 3.10:

$$U(n, m) = \sum s(m, u, v) * [d(s - 1) + 1] \quad (3.10)$$

Here,  $U(n, m)$  represents a convolved dataset measured according to the collected dataset, and  $(u, v)$  denotes different layers of features. The parameter  $d(s - 1)$  corresponds to the window size used in the previous step during the raw data processing. The adaptive window method enhances identification accuracy, providing better end-to-end information for human activity recognition.

The collected dataset  $a[(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)]$  is also defined in the context of the hyperplanes, represented by the following equations 3.11, 3.12, and 3.13:

$$W.x_i + B \geq +1; Y_{i'} = \pm 1 \quad (3.11)$$

$$W.x_i + B \geq -1; Y_{i'} = -1 \quad (3.12)$$

$$\text{minimize } ||W|| \text{ subject to } Y_{i'}(W.x + B) \geq 1 \quad (3.13)$$

Here,  $x_i \in \mathbb{R}^n$  and  $y_{i'} \in \{\pm 1\}$  represent the labels related to all human actions in the collected dataset. The optimisation process aims to find the hyperplanes that maximise the distance (denoted by  $W$ ) between the maximum and minimum hyperplanes while ensuring the correct classification of human activities.

The feature extraction and selection process efficiently identifies the most relevant features for activity recognition, reducing computational complexity and improving the accuracy and efficiency of the classification model.

### 3.2.2.3 Classification

The classification stage involves training the **DNN** model using the proposed **SMO** algorithm. The goal is to obtain an optimal solution by adjusting the weights of the classifier. Once the model is trained, the extracted output is fed into the classification layer to classify and detect human activities. The proposed optimisation fitness function enhances the activity identification process within the **DNN** network.

To achieve the best performance, the **SMO** algorithm is utilised to experiment with optimal feature selection and proper hyperparameters for the **DNN** model.

This includes determining the number of layers, selecting the appropriate loss function, fine-tuning the learning rate, and optimising the kernel size.

The DNN network is based on artificial neural networks and comprises multiple layers. The input layer is represented as  $i(h')$ , and there are multiple classification layers denoted as  $h'_k$ . Each classification layer contains numerous neurons that take the previous layer's output as input. The linear activation function in the classification layer is described by the equation 3.14:

$$h_{k_j} = \text{sigmoid}(a_{k_{i_j}} + w_{k_j}[h_{k-1}]) \quad (3.14)$$

Here,  $h_{k_j}$  represents the output of the  $j^{\text{th}}$  neuron in the  $k^{\text{th}}$  classification layer,  $a_{k_{i_j}}$  denotes the offset of the  $j^{\text{th}}$  neuron in the  $k^{\text{th}}$  classification layer, and  $w_{k_j}$  refers to the  $j^{\text{th}}$  row of the weight matrix that connects the  $(k-1)^{\text{th}}$  classification layer to the  $k^{\text{th}}$  classification layer. The sigmoid linear activation function is defined in equation 3.15:

$$h_{l_j} = \frac{e^{a_{k_j} + w_{l_j} h_{l-1}}}{\sum e^{a_{i_i} + w_{l_i} h_{l-1}}} \quad (3.15)$$

The  $l^{\text{th}}$  layer is assumed to be the functional layer of the softmax. The dataset is then updated in the local leader (LL) stage using the following equation 3.16:

$$\begin{aligned} SM_N(p, q) = [SM^*] * [ & [(LL(Q) - SM(P, Q))] \\ & + U_R(-1, 1)] * [SM(R, Q) - SM(P, Q)] \end{aligned} \quad (3.16)$$

Here,  $SM^*$  represents  $SM(P, Q) + U_R(0, 1)$ , and  $LL(Q)$  denotes the  $Q^{\text{th}}$  dimension based on the local leader location. The  $SM(R, Q)$  is a randomly selected  $Q^{\text{th}}$  dimension from the local group. If the fitness of the previous action fails to classify the activity accurately, the process moves to the global leader (GL) stage using the equation 3.17:

$$\begin{aligned} SM_N(p, q) = [SM^*] * [ & [(GL(Q) - SM(P, Q))] \\ & + U_R(-1, 1)] * [SM(R, Q) - SM(P, Q)] \end{aligned} \quad (3.17)$$

In this equation,  $GL(Q)$  represents the location of the global leader in the  $Q^{\text{th}}$  dimension. Different types of dataset experiences ( $Q = 1, 2, 3, \dots, m$ ) are used to update the location of every dataset. The best dataset is accessed to improve its quality, and its probability determination is given by the equation 3.18:

$$P(SM) = \frac{F(P)}{\sum_{P=1}^n F(P)} \quad (3.18)$$

Here,  $F(P)$  denotes the fitness value of the  $P^{th}$  dataset, and the value is compared with the LL location. A better fitness value indicates the best quality dataset obtained from the LL and GL fitness functions. Finally, the correct activities of the given dataset are classified based on the fitness function in equation 3.19:

$$y(j, i) = x(j, i) + \left\{ U_R(0, 1) * [LL(k, j) - DU(i, j)] \right\} + \left\{ U_R(0, 1) * \left\{ \frac{sum}{DU} \right\} \right\} \quad (3.19)$$

Here,  $sum = sum + [x(j, i) - y(j, i)]$ ,  $DU(i, j)$  represents the  $i^{th}$  and  $j^{th}$  dimension based on the dataset updating, and  $LL(k, j)$  represents the  $j^{th}$  dimension based on the group location of the  $k^{th}$  local leader. This proposed DNN-SMO approach encompasses feature extraction, feature selection, and identification of human activity.

The multimodal sensor datasets are used as input to the proposed DNN-SMO framework. The dataset is trained and processed through the preprocessing layer, where the dataset quality is improved based on the proposed fitness function. The flow chart for the proposed approach is shown in Figure 3.9.

### 3.2.3 Results and Discussion

The deep learning model evaluates the proposed system using spider monkey optimisation. The experiment is conducted on the three datasets. The **UCI-HAR** dataset is the labelled dataset. The parameters for the models are optimised by various experiments conducted on the datasets. **DNN** is the vital model in the HAR system. In this research, the multimodal sensor dataset is used to predict the effectiveness of the proposed replica. The proposed efficient approach improves the activity classification, such as walking, sitting, etc. Finally, the comparison is performed among the projected techniques with some existing techniques.

Here, multimodal sensor datasets are taken as the input, which is trained to the proposed **DNN-SMO** framework. Then, this trained dataset is entered into the preprocessing layer. Based on the proposed approach fitness function, the dataset's quality will improve in this stage.

Moreover, the training loss of the [UCI-HAR](#) dataset is decreased over each iteration of the training process. Also, the training accuracy of the [UCI-HAR](#) dataset increased based on the training iteration of the proposed model. Simultaneously, testing loss decreases, and testing accuracy increases based on the training iteration. Here, the modelling accuracy of the training process is increased over the testing accuracy with different datasets. Simultaneously, the overall dataset's training and validation performance loss is decreased.

After that, take the [WISDM](#) dataset to train and classify the human activity efficiently with the help of equation 3.19. The confusion matrix is plotted between the true label and each activity. Normalised confusion matrices are mainly based on the trials of testing WISDM dataset samples.

After the classification, [WISDM](#) datasets are classified based on human activity like sitting, lying, walking, walking upstairs and downstairs, etc.

### 3.2.3.1 Computational Complexity

[DNN](#) models are computationally expensive due to their large number of parameters. Training a [DNN](#) requires significant computation, including matrix multiplications and non-linear activations, which can be computationally intensive. The complexity of training a [DNN](#) can be expressed as  $O(nmp)$ , where  $n$  is the number of samples in the dataset,  $m$  is the number of features, and  $p$  is the number of parameters in the network. The [SMO](#) algorithm also has a computational complexity, which is affected by the number of features and iterations. The complexity of the [SMO](#) algorithm can be expressed as  $O(n*k)$ , where  $n$  is the number of samples in the dataset and  $k$  is the number of iterations. The computational complexity of the [DNN-SMO](#) model is higher than that of the [DNN](#) alone, as the optimisation process requires additional computation.

### 3.2.3.2 Implementaion Challenges

Data collection and preprocessing: Collecting and preprocessing sensor data, such as accelerometer and gyroscope data, from smartphones, was challenging. It requires careful consideration of the sampling rate, sensor placement, and data cleaning to ensure that the collected data is appropriate for training the [DNN](#) and is discussed in preprocessing. Determining the appropriate [DNN](#) architecture: Determining the appropriate [DNN](#) architecture is challenging. It requires a good understanding of the problem and the dataset and the ability to experiment with different architectures to find the one that works best.

**SMO** algorithm implementation: Implementing the **SMO** algorithm was challenging, as it required a good understanding of the algorithm and its parameters. It also requires careful tuning of the algorithm’s parameters to find the optimal configuration for the problem. Also, the hardware setup to work on these methods and experiments must be considerably high.

### 3.2.3.3 Performance Evaluation

The performance of the proposed technique is calculated by evaluating the key performance with some other recent works. The performance metrics such as accuracy, Precision, Recall, F1-score, Error rate, and Computation time are considered. Furthermore, the performance of the proposed method is compared with existing techniques like Deep Learning based Q-Framework (**DL-Q**) (Zhou *et al.* 2020), end-to-end **DNN** (Hassan *et al.* 2021), and **SVM** (Subasi *et al.* 2020).

### 3.2.3.4 Accuracy(A)

The rate of accuracy count is validated as the accurate classification used in the collected dataset. Thus, the evaluations of accuracy measures with existing approaches are shown in Table 3.3.

Table 3.3: Accuracy Comparison Across Datasets

Dataset	Accuracy			
	DL-Q	End to end DNN	SVM	Proposed method
ACTION KTH	78.30	83.00	90.20	95.78
UCIHAR	79.90	84.00	90.30	98.85
WISDM	81.00	84.50	92.00	98.92
PAMAP2	84.20	85.80	90.00	95.22

The Accuracy validation is shown in equation 3.20. Where TN is a True Negative, TP is a true positive, FN is a false negative, and FP is a false positive.

$$A_c = \frac{(TN + TP)}{(TN + TP + FN + FP)} \quad (3.20)$$

There is a significant increase in the accuracy of using the proposed method for all three datasets. Table 3.3 shows an increase of 5.58%, 8.55%, 6.92% and 5.22% in accuracy compared to the maximum of other methods in the same domain.

### 3.2.3.5 Precision (Pr)

The precision calculation is evaluated based on the number of activity classifications by the total number of collected datasets. Thus, the Precision of the processed dataset is evaluated as the number of correct positive predictions alienated by the total number of positive predictions. Precision for the HAR system is intended as an appropriate human activity. The calculation of Precision is in equation 3.21, and the result is shown in Table 3.4.

Table 3.4: Precision Comparison Across Datasets

Dataset	Precision			
	DL-Q	End to end DNN	SVM	Proposed method
ACTION KTH	87.92	90.60	93.00	94.54
UCIHAR	89.00	91.32	93.50	96.00
WISDM	90.00	92.50	94.00	98.12
PAMAP2	84.00	92.50	92.00	94.42

$$P_r = \frac{(TP)}{(TP + FP)} \quad (3.21)$$

There is a significant increase in the accuracy of using the proposed method for all three datasets. The Precision of 94.54%, 96%, 98.12% and 94.42% is achieved in our model for the three datasets. Table 3.4 shows an increase of 1.54%, 2.50%, 4.12% and 2.42% in the Precision compared to a maximum of other methods in the same domain.

### 3.2.3.6 Recall (Rc)

The Recall is the number of correct positive forecasts divided by the total number of true positives and false negatives. Recall for HAR is calculated as a classified activity divided by the total collected dataset. Also, the Recall is calculated using the formula equation 3.22.

$$P_r = \frac{(TP)}{(TP + FN)} \quad (3.22)$$

There is a significant increase in the Recall using the proposed method for all three datasets. Our model achieved the Recall of 92.50%, 96.70%, 98.90% and 96.34% in our model for the four datasets. Table 3.5 shows an increase of 1.54%,

Table 3.5: Recall Comparison Across Datasets

Dataset	Recall			
	DL-Q	End to end DNN	SVM	Proposed method
ACTION KTH	82.30	80.10	80.00	92.50
UCIHAR	82.50	82.00	80.30	96.70
WISDM	82.90	85.00	82.00	98.90
PAMAP2	78.90	82.22	85.42	96.34

2.10%, and 4.12% in the Recall compared to a maximum of other methods in the same domain.

### 3.2.3.7 F-1 Score (Fs)

The Recall is the number of correct optimistic forecasts divided by the total number of true positives and false negatives. Recall for HAR is calculated as a classified activity divided by the total collected dataset. Also, the Recall is calculated using the equation 3.23.

Table 3.6: F1-score Comparison Across Datasets

Dataset	F1-Score			
	DL-Q	End to end DNN	SVM	Proposed method
ACTION KTH	82.30	86.10	94.00	94.50
UCIHAR	82.50	84.00	89.30	93.70
WISDM	82.90	87.00	93.00	95.90
PAMAP2	79.22	83.40	92.00	93.98

$$F_s = 2 * \left\{ \frac{(P_s * R_c)}{(P_s + R_c)} \right\} \quad (3.23)$$

The F1-Score of 94.50%, 93.70%, 95.90% and 93.98% is achieved in our model for the four datasets. Table 3.6 shows an increase of 0.50%, 4.40%, 2.90% and 2.98% in the F1-score compared to a maximum of other methods in the same domain.

### 3.2.3.8 Error Rate (Er)

The error rate is estimated based on the fault during classification in the execution process. Also, fault classification is termed an error occurrence. It is estimated as the total frequently classified activities, elaborated in the equation 3.24. The error rate comparison is shown in Table 3.7.

Table 3.7: Error Rate Comparison Across Datasets

Dataset	Error Rate (%)			
	DL-Q	End to end DNN	SVM	Proposed method
ACTION KTH	11.00	5.67	6.55	2.80
UCIHAR	10.23	5.11	5.34	1.55
WISDM	5.40	4.50	3.20	2.35
PAMAP2	6.12	5.20	1.40	3.05

$$E_r = \frac{(C_d * T_d)}{(T_d)} \quad (3.24)$$

Where  $C_d$  is represented as the activities classified correctly in a dataset and  $T_d$  is denoted as the total collected activities in the dataset. The Error rate of 2.80%, 1.50%, 2.35% and 3.05% is noted in our model for the four datasets.

### 3.2.3.9 Computation Time (Ct)

Computation time is based on the time taken to classify the human activity identification; also, the computation time process is termed run time. Moreover, the algorithm's efficiency can also minimise execution time; it is the critical metric in the HAR framework. Therefore, the comparison of computation time is shown in Table 3.8.

Computation time is 5.5, 3.64, 3.4 and 2.3 seconds for the four datasets. Table 3.8 shows an increase of 0.6, 1.84, 2.25 and 4.24 seconds in the computation time compared to a maximum of other methods in the same domain.

The all metrics comparison validation shows that the proposed novel DNN-SMO applies to all applications to classify the HAR system. The overall performance of the proposed model is elaborated in Table 3.9. The proposed model attained 98.9% maximum accuracy, 98.12% Precision, 98.9% recall measure, 98.9% F-1 score, and a minimum error rate of 3.05%.

Table 3.8: Computation Time Comparison Across Datasets

Dataset	Computation Time(seconds)			
	DL-Q	End to end DNN	SVM	Proposed Method
ACTION KTH	8.50	6.95	6.10	5.50
UCIHAR	4.60	3.45	5.48	3.64
WISDM	5.48	4.50	5.65	3.40
PAMAP2	4.48	3.50	6.54	2.30

Table 3.9: Overall Performances Comparison Across Datasets

Dataset	Accuracy	Precision	Recall	F1-score	Execution Time (s)	Error Rate
ACTION KTH	95.78	94.54	92.50	96.30	5.50	2.80
UCI-HAR	98.90	98.12	98.90	98.90	3.64	2.35
WISDM	98.85	96.00	96.70	98.84	3.40	1.55
PAMAP2	95.82	94.42	96.34	93.08	3.05	2.30

The accuracy of the model is improved due to the use of the Deep Neural Network Model. Also, the scope for improvement in optimising the multiple parameters helped to achieve the highest accuracy for the four datasets. The experiments and analysis carried out with the [WISDM](#) dataset gave the highest accuracy of 98.92% as compared to other datasets. The accuracy improved because the dataset and the data distribution helped the model to recognise the activities accurately. However, for the [PAMAP2](#) dataset, the highest accuracy achieved is 95.22% because it has more activities than other datasets. Also, the fusion of multimodal sensor data will not give the best results compared to single sensor datasets. Using the proposed SMO-based optimiser helped reduce the execution time because of the reduction in the feature set.

### 3.2.4 Discussion

This work surpasses existing efforts because previous approaches have focused on general Human Activity Recognition (HAR). Our innovation lies in the integration of data fusion techniques, carefully selecting the features and fine tuning the models which significantly enhance the accuracy of our data collection. Through experimentation with existing datasets and models, we have observed superior performance with our approach.

### 3.3 Summary

There are various deep learning models available for the recognition of human activities. Many deep-learning approaches are performed with CNN-LSTM, LSTM Dense, Stacked LSTM, and Bidirectional LSTM. The deep learning models overcome the problem of selecting the features manually. The GRU model outperformed the accuracy in recognition of the activities from both the datasets of UCI-HAR and WISDM. The ability to use the time series data with the varied sequence has given the advantage to recognise the GRU model. The result also showed that some class of activities has the highest accuracy rates for all the models. However, while doing the experiments, we observed that the performance could be improved by optimising certain parameters that will also help reduce training time. So this research introduced an innovative DNN-SMO method. We experimented with this model on the datasets collected from multimodal sensors like cameras, wearable sensor devices, etc. Here, the collected datasets are processed, extracted, and selected, and finally, activity is recognised based on the fitness of the proposed approach. Finally, the efficiency of the proposed model was validated with recent old works and pertained to good results by attaining less processing time and a high accuracy rate. Moreover, it attained high accuracy, f-measure, error rate, and execution time performance. Hence, the attained accuracy is 98.9% and a lower error rate of 3.4% than other methods.

However, the major limitation identified based on the results and analysis is that it recognises only basic or low-level activities. Recognising the human behavioural patterns requires identifying the activities based on the long-hour time series data. Hence, the long-hour data collection is considered for further research.



## Chapter 4

# Proposed Framework for Long-hour Data Collection using Multimodal Sensors

There are already existing datasets for human activity recognition using smartphone-based data with accelerometers and gyroscope sensors. Many datasets (Bokhari *et al.* 2021; Janarthanan *et al.* 2020; Tanberk *et al.* 2020; Dua *et al.* 2021; Kim 2020; Nandy *et al.* 2020) are available on which the experiment is being done for HAR systems. An accelerometer is a sensor device that monitors the acceleration of any person or object in its immediate rest frame. Several accelerometer sensor-based devices include wearable devices, smartphones, and other electrical equipment to recognize human activities and their activity patterns. However, these sensors provide the data to recognize only basic human activities like walking, sleeping, running, and sitting. Also, these data sets are lab-controlled environments and only recognize the basic activities. So, the long-hour dataset is created to detect human activity patterns over a period.

It is essential to have a long-hour data set to identify human behavioural patterns. With the existing datasets, which use only smartphones, it is challenging to have continuous daily living activities. Hence, the work focuses on creating a dataset that records human activity over a long period in a home environment, such as daily, weekly, and monthly analyses. We have used the existing mobile phone sensors to avoid the burden of wearing specific wearable devices and other sensors. The data recording is done in three phases. First, the six basic activities are recognized using the deep-learning models on the WISDM dataset. Second, the trained model is deployed in the mobile application to recognize the

basic activities, and the ambient sensor data is recorded simultaneously. In the third phase, the basic activities and the ambient sensor data are combined to derive more activities like watching TV in the Hall.

With smartphone-based sensors, the activities that can be recognized are limited to walking, stairs up, stairs down, sleeping, and falling. The limitation is because of the placement of the mobile phone inside the pocket. With mobile phone-based sensors, the activities done using hand movements cannot be recognized. However, using the ambient sensors attached to the bed, chair, TV, and location-detecting sensors, it is possible to derive more activities like using the bathroom, using the kitchen, watching TV, sleeping, and sitting on a chair. Table 4.10 briefs the activities from the mobile sensors and activities derived from the ambient sensors. So, an Android-based application is developed to collect the data for long hours. The user needs to configure it by specifying the basic parameters of sensors and duration at the beginning. To develop the Android application, we used the Android software development toolkit and Android Studio as an integrated development environment to design the user interface and develop the application.

The proposed dataset can be used to analyze a person's activity patterns, which can be applied to various applications such as personality profiling, anomaly detection, automated assistance and habit recording.

The present work enhances HAR systems and addresses key challenges in this field. The following contributions highlight the importance and impact of this research.

- **Data Fusion for Enhanced Performance:** A novel data fusion approach is designed, integrating raw data from the existing [WISDM](#) dataset. This fusion technique effectively improves the overall performance of the HAR system by enhancing the accuracy of activity recognition. By leveraging data fusion, the system achieves better recognition rates for various activities, reducing the occurrence of false annotations and ensuring more reliable results. This advancement is crucial in real-world applications, where accurate activity recognition is essential for providing meaningful insights and context-aware services.
- **Automated Labelling with Android Application:** An Android-based application is developed and deployed, equipped with a trained model for automatic labelling of new datasets related to basic human activities. This contribution significantly streamlines the data labelling process, especially for long-hour

dataset collections spanning several weeks. By automating the labelling procedure, the research mitigates the challenges associated with manual annotation, saving time and effort while maintaining high accuracy in labelling various activities.

- **Enhanced Activity Recognition through Ambient Sensors:** The work successfully synchronizes output from ambient sensors with long-hour data, deriving additional activities beyond the basic set. The effective configuration and placement of PIR sensors play a pivotal role in detecting the room in which a person resides, enriching the context and improving the quality of activity recognition. Integrating simple relay switches for on/off detection further enhances the system's ability to derive more activities, capturing more subtle user behaviour. This achievement extends the scope of HAR to encompass a broader range of human activities, making it more relevant for real-world scenarios and context-aware applications.
- **Efficient Single-Person Occupancy Detection Algorithm:** A specialized single-person occupancy detection algorithm is developed, proficiently identifying the presence of individuals within a room and resolving occupancy overlaps between rooms. This algorithm is critical in providing essential contextual information for activity recognition. By accurately detecting occupancy, the system gains insights into individual-centric activities, thus refining the recognition process and ensuring precise and contextually meaningful results. Moreover, the proposed algorithm offers an economical and effective approach to occupancy detection, making it practical for large-scale deployment in smart home environments and other related applications.
- **Priority-Based Two-Minute Data Segmentation:** A unique data segmentation approach generates labelled data at regular intervals to map derived activities. The application of priority-based averaging further optimizes the data size, enabling efficient and effective utilization of person-centric activities. This approach reduces the computational burden and streamlines the processing of large datasets, enhancing the system's performance and responsiveness.

These contributions significantly advance the field of Human Activity Recognition, facilitating more accurate, context-aware, and efficient systems. By addressing various challenges related to data fusion, automated labelling, ambient sensor integration, occupancy detection, and data segmentation, the research lays

a strong foundation for developing advanced HAR technologies with broader applications and real-world impact.

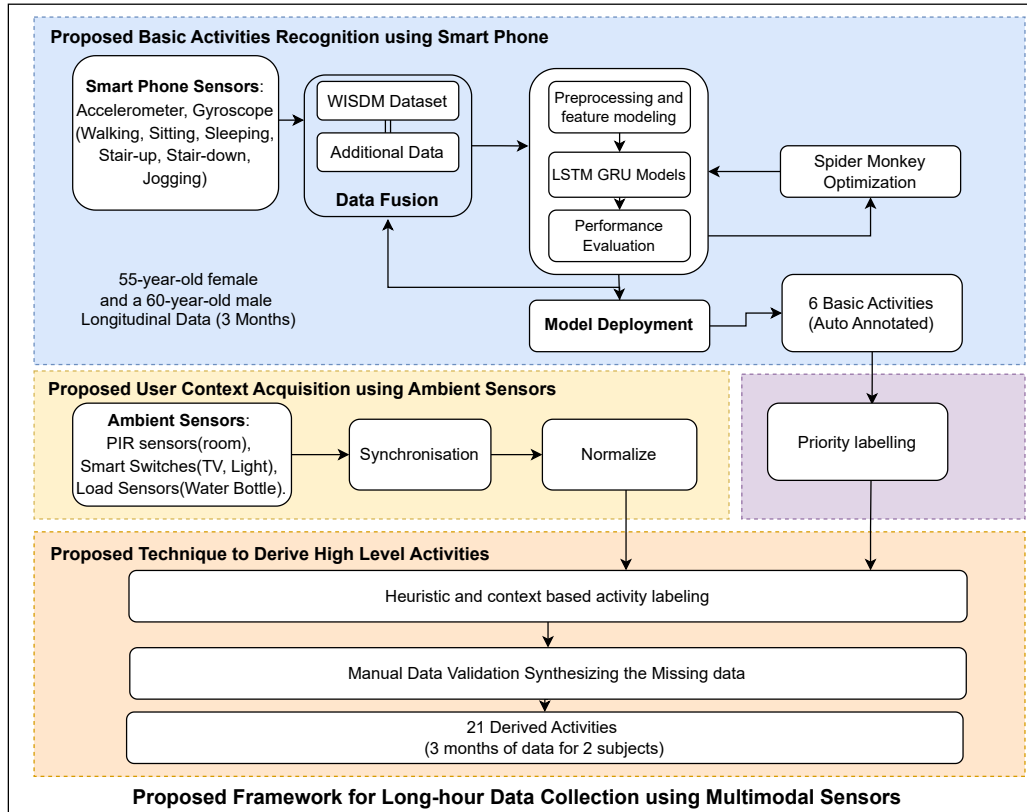


Figure 4.1: The Proposed Framework for Long-hour Dataset Collection using Multimodal Sensors.

The framework for long-hour dataset collection using multimodal sensors is illustrated in Figure 4.1.

In the first step, the system employs a deep learning model to recognize basic daily human activities, including walking, sleeping, upstairs, downstairs, and sitting. This initial phase establishes the foundation for activity recognition, ensuring accurate identification of basic activities. Next, the trained deep learning model is integrated into an Android application, enabling the automatic recognition and annotation of activities over long durations. "Long-hour" refers to continuously recording data over extended periods, such as several hours or days. This automated labelling process is crucial, considering the challenge of manually annotating vast amounts of long-duration data. The Android application efficiently applies the trained model to label the various activities, streamlining the dataset curation process and maintaining high accuracy in annotation.

The work also incorporates the output data from ambient sensors, including

room location information, the on/off status of appliances, and the water bottle indicator. These additional inputs from ambient sensors enhance the contextual information available for activity recognition. Leveraging data from both smartphone and ambient sensors, the system proceeds to derive high-level activities based on their combined outputs. The derived high-level activities offer a more comprehensive understanding of users' behaviour and context, encompassing a broader range of human activities beyond the basic set recognized in the first step. This enrichment of activity recognition is achieved by effectively synchronizing and combining data from different sensor modalities, leading to more subtle and context-aware results. The system provides a comprehensive view of users' behaviour and activity patterns, making it well-suited for context-aware applications and enhancing the overall performance of Human Activity Recognition systems.

## 4.1 Proposed Basic Activities Recognition using a SmartPhone

For recognising fundamental activities in this study, we opted for the [WISDM](#) dataset (Janarthanan *et al.* 2020). Our rationale for this selection lies in the dataset's provision of publicly available raw data, which is pivotal for conducting data-level fusion. Moreover, we followed the same preprocessing approach as the baseline work outlined in Chapter 3. Furthermore, the chosen dataset encompasses the activities we aimed to incorporate into our long-hour data collection initiative. It is worth noting that the data was gathered using smartphone sensors placed within the trouser pocket. This approach was taken to ensure user comfort and convenience, as it is less intrusive than sophisticated sensor placements and arrangements. However, our work has inherent limitations, summarised in this section.

The [WISDM](#) dataset is a laboratory-controlled human activity recognition dataset available to the public. It consists of data from 36 subjects engaging in six activities: walking, sitting, jogging, downstairs movement, upstairs movement, and standing. The data is captured using smartphones with inertial sensors like accelerometers and gyroscopes. The dataset maintains a sampling rate of 20 Hz and comprises a total of 1,098,207 examples. The dataset features six attributes, and the distribution among activity classes is as follows: Walking (38.6%), Jogging (31.2%), Upstairs movement (11.2%), Downstairs movement (9.1%), Sitting (5.5%), and Standing (4.4%).

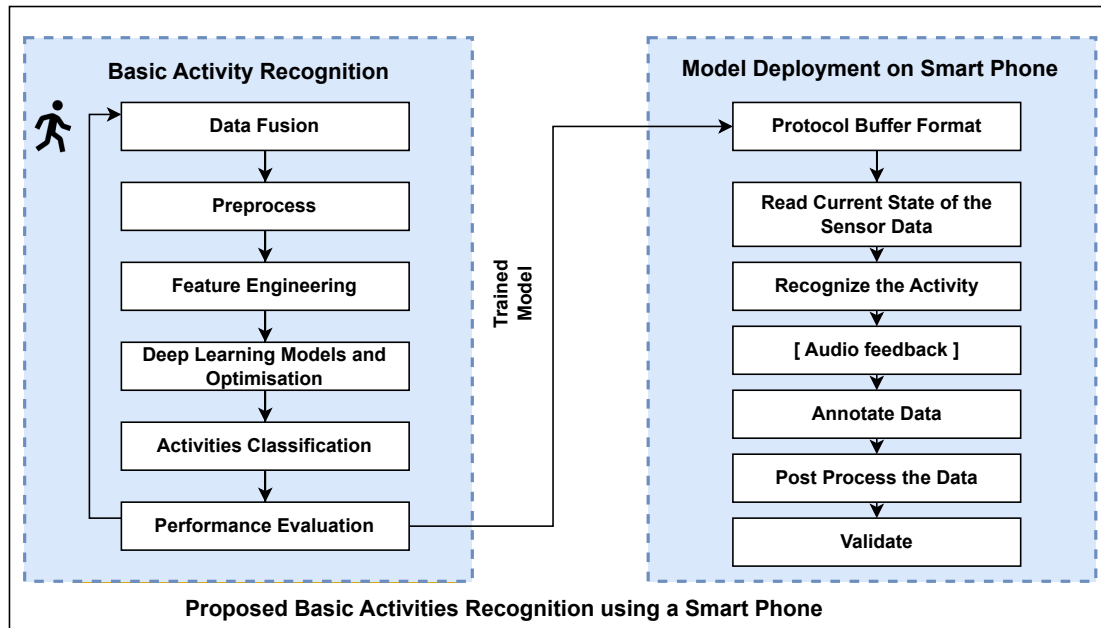


Figure 4.2: Proposed Basic Activities Recognition using Smart Phone

## 4.1.1 Basic Activities Recognition

### 4.1.1.1 Data Fusion

Upon deploying the model with the [WISDM](#) dataset, it was noted that the accuracy of ground truth evaluation could be enhanced compared to the dataset itself. This discrepancy might arise from variations in smartphone sensor configurations and individual walking styles across individuals trained versus that testing.

To address this, data fusion was conducted at the data level, involving an additional 12 individuals, out of which two are used for long-hour data collection. An additional 64,261 data examples from these individuals were incorporated into the existing dataset. To ensure data consistency, the same data recording strategy was employed as that used in the original dataset. The activities recorded mirrored those in the original dataset and were sampled at a rate of 20 samples per second (50 ms intervals). The data was segmented into 10-second intervals, from which features were extracted. Utilising simple statistical methods such as averages, standard deviations, average absolute differences, average resultant accelerations, and time intervals between peaks, these features were transformed into single examples or tuples comprising 46 values.

Before commencing data collection, formal approval was obtained from the Institutional Review Committee. The data collection process involved student and family member volunteers who were provided with Android mobile phones

equipped with the requisite application. Participants were given preliminary instructions and were required to provide basic information such as name, age, and gender, along with their consent to contribute data. Participants selected the activity they were about to perform, initiated recording by pressing the start button on the app, and carried out the designated activity. The data collection team supervised the process, ensuring its proper execution. Upon pressing the start button, a 20-second countdown beep signalled the commencement of recording, allowing participants to prepare and start the activity. The team used a visual representation and basic statistical analysis to monitor and validate the data collection process. Each file was labelled with the subject's ID and the performed activity. This process is illustrated in Figure 4.3.

#### 4.1.1.2 Preprocessing

The smartphone sensors utilised in this study capture variations in inertial movements. Specifically, accelerometer sensors gauge acceleration forces, while gyroscope sensors measure angular velocity. The system captures movement across the x, y, and z axes by amalgamating data from these sensors. To prepare the input signals for analysis, noise filters are employed, and a sampling rate of 2.56 seconds is applied. Moreover, Butterworth low-pass filters are utilised to eliminate noise from the data.

For the experimentation phase, the [WISDM](#) dataset is the primary dataset. This dataset comprises time series data, necessitating segmentation into 128 timestamps per sample. Each data point in the dataset encompasses x, y, and z features indicative of the sensor's orientation. The accelerometer captures both body and gravitational acceleration, whereas the gyroscope registers angular velocity and movement. Before analysis, data cleaning procedures are executed, encompassing the removal of duplicates and null values. Class distribution is balanced within the dataset to ensure equal representation across different activities.

#### 4.1.1.3 Feature Engineering

For feature extraction, a combination of [CNN](#) and [RNN](#) architectures is utilised to leverage the characteristics of time series data. The one-dimensional [CNN](#) is particularly effective in extracting features directly from raw data, eliminating the need for manual feature extraction by domain experts. A feature map is generated as the output by applying filters to the input data. The activation function used in the [CNN](#) is the [ReLU](#), and kernel size filters are employed. Max pooling is applied

to select the most essential features from the previous feature maps, maximising the elements in the feature map. This approach enables effective and automatic feature extraction without relying on prior domain knowledge.

#### 4.1.1.4 Deep Learning Models and Optimisation

The GRU Model uses the gated mechanism in the Recurrent Neural Networks to recognise the activities. The GRU model is the variation of the LSTM model explained earlier. A detailed explanation of the data processing, the model implementation and its results and discussion are described in Section 4.1.3. It is observed that the accuracy for a model applied to WISDM data is 82%, and with the fusion data, the accuracy is 91%.

### 4.1.2 Model Deployment on Smart Phone

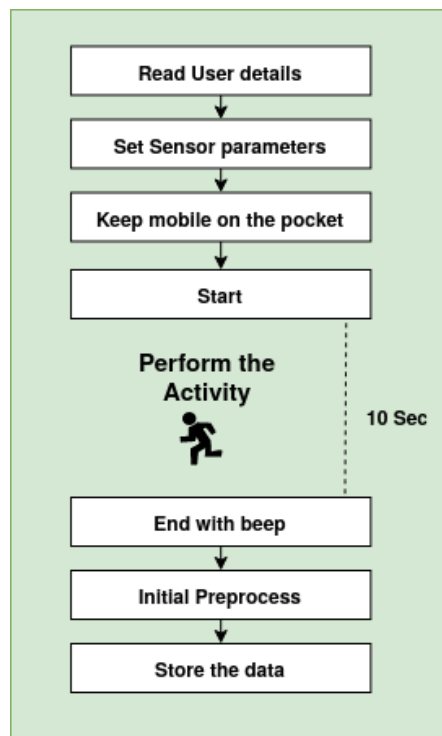


Figure 4.3: Process to Collect the Basic Activities in Android Application.

Once the model is trained and built, the trained model is integrated into our human activity recognition app. The application has three features: Sensor settings, Profile creation, Activity Recognition and Record data. Sensor settings: One can choose the sensors displayed on the application to record the data. The default sensors and the time interval are kept according to this work's objective

and set by the data collector. The data collector is the expert (developer) from the domain. Profile creation module will read the information about the subjects like name, age, profession, and gender, which will provide the context to the experiment. The activity recognition feature runs in the background, and it is to recognise the activity. Once the activity is recorded, the data is stored in the mobile phone memory in comma-separated text format. Optional audio feedback about the recognised activity is given to verify the subject of the activity recognition.

The sensor placement is inside the trousers pocket, either the left or right pocket. The phone's orientation in the pocket is considered and adjusted while recording the data. Also, the mobile inside the packet is recognised and processed while creating and recognising the dataset. The data is recorded with five users with varied ages, an average age of 40.7, and gender (three males and two females).

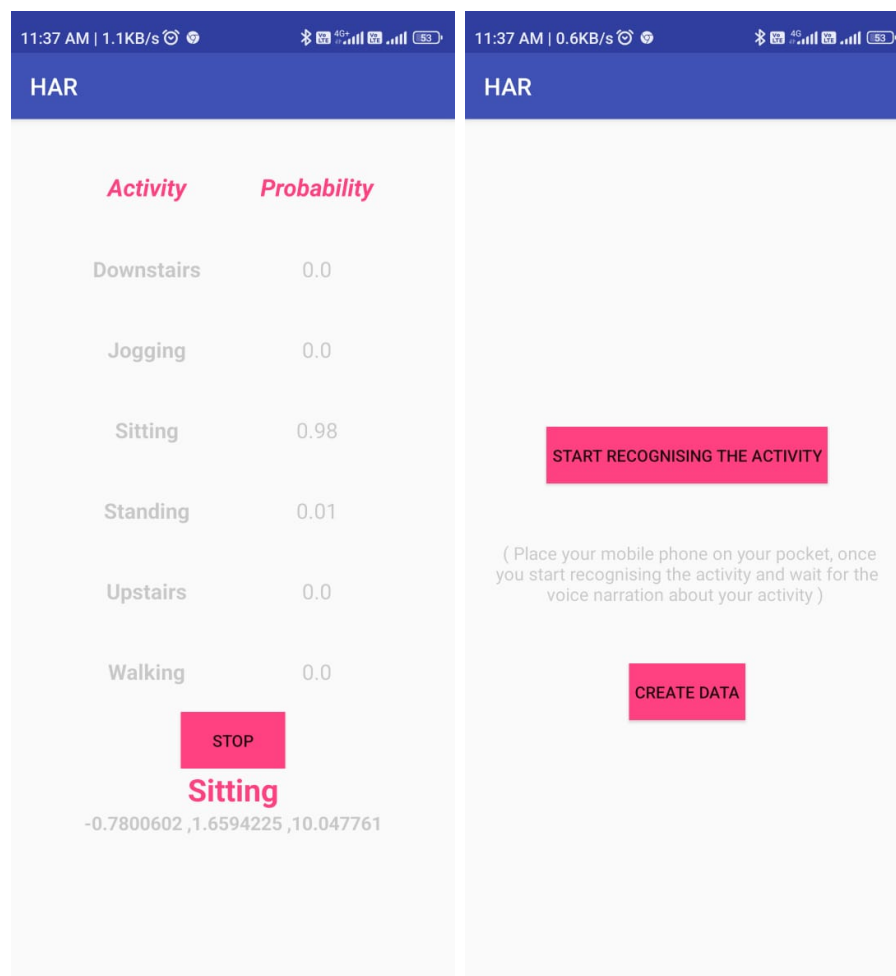


Figure 4.4: Sample Screenshot of the Android Application After Deploying the Trained Model. The Sample Shows the Walking Activity is Recognised.

We asked the users to keep their mobile phones inside their pockets with the application developed to record their activities. The android-based application records the recognised activities every 5 seconds and is stored in the file. The file is in CSV format with the timestamp and the label of the activities.

After deploying the trained model, a recognised sitting activity sample screenshot of the application is shown in Figure 4.4. The recognised activities are stored in the file by taking the maximum of the activities recognised in 1 minute. We have stored only labels or activity per minute. Also, only the start and end times are noted when repeating activities continuously for a longer time. This reduced the file size while storing. This file size reduction was significant as we stored hours of data in the mobile phone memory.

### 4.1.3 Implementation

This model has experimented on the WISDM dataset and the WISDM-fusion. The noise removal and the initial segmentation are carried out. The data is checked for missing data and filled using simple linear interpolation. The data is time-series and segmented to a sliding window of 128 with a 50% overlap to preserve the temporal relationship.

Feature extraction plays an essential role in classification. As it is time-series data, it has both temporal and spatial features. The preprocessed data is fed into the two layers of GRU with 32 neurons each to extract the temporal features, and CNN layers are used to extract the spatial features. The CNN layers have experimented with multiple layers, and in this work, 32 and 128 filters with a kernel size of 64 each are used in the CNN layers. The dropouts of 0.07 and 0.2 are used after CNN and GRU layers. The following are the steps in building the GRU model for classification activities.

1. Input: read the raw data, read from sensors. The vector size is 128.
2. CNN layers: Convolution operations of 1 dimension are applied on input. The Relu Activation function is used with this operation. Another convolution operation is followed by the first.
3. Dropout: This layer is added to avoid over-fitting problems in a multi-layer neural network. The dropout of 0.07 is used.
4. Max pooling: As there are multiple layers, the pooling layer separately operates upon each feature map to create a new set of the same pooled feature maps to order the CNN layers.

5. Flatten: This step is necessary to make a proper input vector to the next GRU layer.
6. GRU layers: With 32 units, it is again fed to another GRU layer
7. Dense: The output of all the above layers is concatenated using a dense layer.
8. Softmax Activation: This is the final layer for outputting the multiple class activities. This layer follows the batch normalisation of all the layers.

#### 4.1.4 Result and Discussion

In the first phase of work, we evaluated the WISDM and WISDM Fusion datasets and noted the model's accuracy before deployment, shown in Table 4.1. There are a total of 36 user data present in the original dataset. Initially, we took out eight users from the dataset for test data and trained the model with the remaining data. Figure 4.5 shows the confusion matrix with the highest accuracy for standing and walking activity. This is because there is fewer true negative for this data. Figure 4.6 shows the Train and Testing accuracy and the Training and Testing Loss.

It is observed from Table 4.2 that the accuracy is lesser after deploying the trained model on the mobile phone. This is because the environment in which the WISDM data is created differs from the deployment environment. Even the users involved in testing after the deployment is varied because of the varied human style of performing certain activities. So, we added 11 users' data to the existing dataset to make WISDM Fusion data. The data level fusion is done. The experiment setup follows a similar preprocessing strategy used in the original dataset. The dataset is trained again and tested with the same model. It is observed from Table 4.2 that the accuracy is increased after adding new users' data(WISDM Fusion). We added the new users not only to improve the accuracy of the data in the new environment but also to use the same users while creating the long-hour dataset to collect the data for human pattern recognition.

The performance of the deployed GRU model was evaluated on both the WISDM and WISDM Fusion datasets. Table 4.2 presents the accuracy of the GRU model in recognising various activities after deployment.

The WISDM dataset consists of diverse activities, and the GRU model demonstrated remarkable accuracy in recognising these activities. Among the different activities, *Walking* achieved an accuracy of 98%, with 461 out of 471 instances correctly recognised. *Sitting* was recognised with an accuracy of 94%, accurately

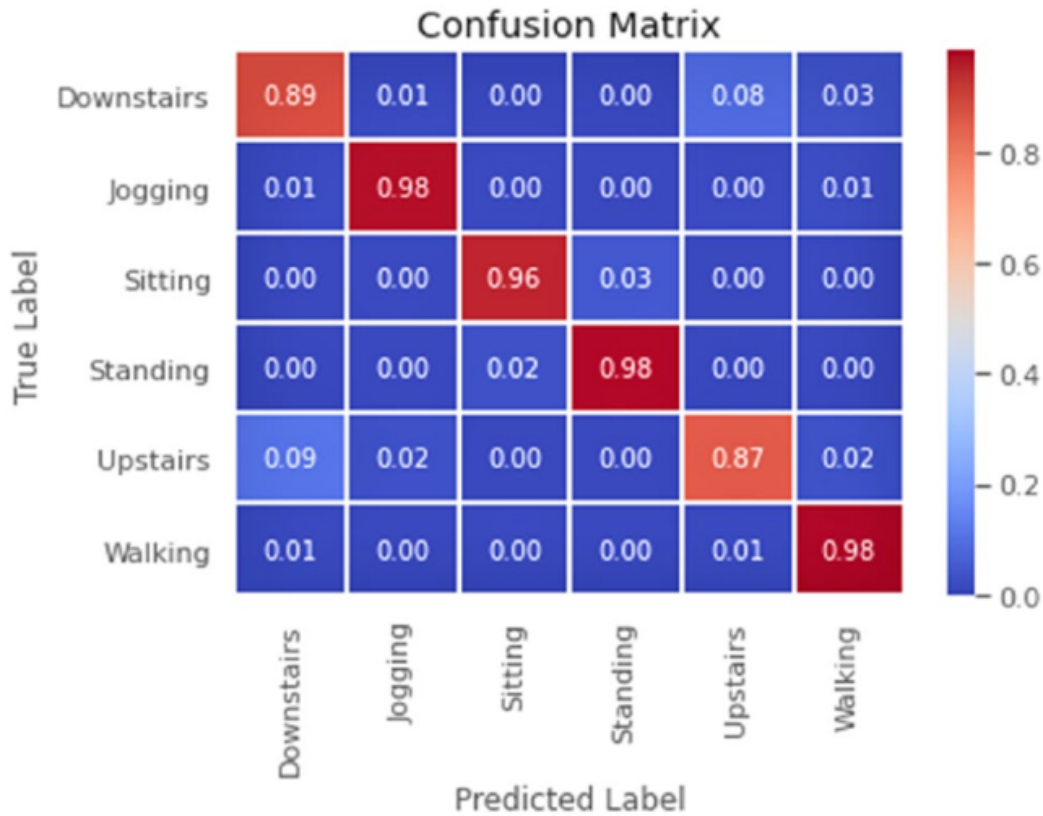


Figure 4.5: Confusion Matrix of GRU Model Experimented on WISDM Dataset.

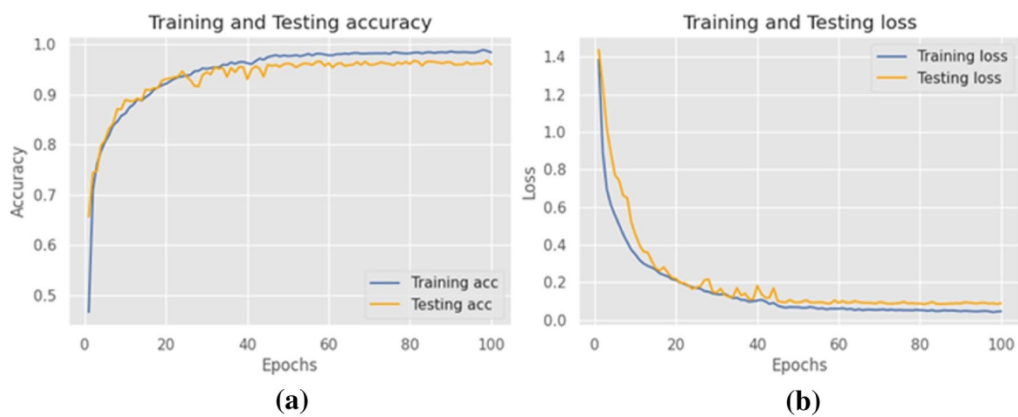


Figure 4.6: Train and Testing Accuracies and Loss for WISDM Dataset.

classifying 352 out of 373 instances. *Upstairs* and *Downstairs* activities achieved accuracies of 85% and 88%, respectively, correctly recognising 232 out of 272 instances for *Upstairs* and 270 out of 307 instances for *Downstairs*. *Jogging* and *Standing* activities were recognised with accuracies of 92% and 96%, classifying 489 out of 530 instances and 185 out of 192 instances, respectively.

Table 4.1: Accuracy of GRU model on WISDM and WISDM Fusion Data Before Deployment

Activities	Accuracy	
	WISDM	WISDM Fusion
Walking	0.98	0.99
Sitting	0.98	0.97
Upstairs	0.87	0.89
Downstairs	0.89	0.91
Jogging	0.96	0.97
Standing	0.98	0.98
<b>Average</b>	<b>0.94</b>	<b>0.95</b>

Table 4.2: Accuracy of GRU model on WISDM and WISDM Fusion Data After Deployment

Activities	Activity data (s)	Total Activities	Correctly Recognized	Accuracy (WISDM)	Correctly Recognized	Accuracy (WISDM Fusion)
Walking	2356	471	461	0.98	453	0.96
Sitting	1866	373	352	0.94	362	0.97
Upstairs	1362	272	232	0.85	235	0.86
Downstairs	1536	307	270	0.88	275	0.90
Jogging	2653	530	489	0.92	506	0.95
Standing	962	192	185	0.96	183	0.95
<b>Total/Avg</b>	10735	2145	1989	<b>0.92</b>	2014	<b>0.93</b>

The WISDM Fusion dataset, which combines data from multiple sources, also demonstrated promising recognition accuracy. *Walking* maintained an accuracy of 96%, with 453 out of 471 instances correctly recognised. *Sitting* exhibited a high accuracy of 97%, accurately classifying 362 out of 373 instances. *Upstairs* and *Downstairs* activities achieved accuracies of 86% and 90%, respectively, correctly recognising 235 out of 272 instances for *Upstairs* and 275 out of 307 instances for *Downstairs*. *Jogging* and *Standing* activities were recognised with accuracies of 95% and 95%, classifying 506 out of 530 instances and 183 out of 192 instances, respectively.

Aggregating the results for all activities, the GRU model achieved an average accuracy of 92% on the WISDM dataset and 93% on the WISDM Fusion dataset. This performance underscores the robustness and effectiveness of the GRU model in accurately recognising various physical activities from sensor data. The results not only showcase the overall strong performance of the deployed GRU model but also underscore its adaptability in handling diverse activities and its ability to leverage fused data for improved recognition accuracy.

Table 4.3: The Performance Metrics for Each Activities.

Activity	Precision	Recall	F1 Score	Specificity
Walking	0.95	0.97	0.96	1.00
Sitting	0.98	1.00	0.99	0.97
Upstairs	0.82	1.00	0.90	0.97
Downstairs	0.88	1.00	0.93	0.98
Jogging	0.91	0.93	0.92	1.00
Standing	1.00	0.78	0.87	0.96
Average	0.92	0.94	0.93	0.98

Table 4.3 presents the performance metrics for each class and the average metrics across all classes. Precision is the proportion of true positive predictions among all positive predictions. It measures the model’s ability to avoid false positives. In this table, the precision values range from 0.82 (*Upstairs*) to 1.00 (*Standing*). A higher precision indicates that the model has a lower false positive rate for that class.

Recall is the proportion of true positive predictions among all actual positive instances. It measures the model’s ability to avoid false negatives. The recall values range from 0.78 (*Standing*) to 1.00 (*Sitting* and *Upstairs*). A higher recall indicates that the model has a lower false negative rate for that class.

The F1 score is the harmonic mean of precision and recall, providing a balanced measure of the model’s performance. It takes into account both false positives and false negatives. The F1 scores range from 0.87 (*Standing*) to 0.99 (*Sitting*). Higher F1 scores indicate a good balance between precision and recall.

Specificity is the proportion of true negative predictions among all actual negative instances. It measures the model’s ability to avoid false positives for negative instances. The specificity values range from 0.96 (*Standing*) to 1.00 (*Walking* and *Jogging*). A higher specificity indicates a lower false positive rate for negative instances. This shows that the model performs well in distinguishing *Walking*, *Sitting*, and *Jogging* activities, with high precision, recall, and F1 scores. These classes have high average metrics, indicating good overall performance.

However, the model needs help in distinguishing *Upstairs*, *Downstairs*, and *Standing* activities. While the model has high precision for *Standing*, it has a relatively low recall, suggesting that it may miss some *Standing* instances. Similarly, the model’s precision is relatively lower for *Upstairs* and *Downstairs*, indicating that it may have some false positives in these activities.

The basic activity recognition is carried out on the WISDM dataset. The GRU model shows an accuracy of 94%. However, once deployed on the mobile phone,

the accuracy is downsized to 92%. This is because of the testing environment and varied users while recognising the activity. New users' data is added to the existing data, and the WISDM Fusion data is created with additional users. Adding the new user's data improved the accuracy to 93% after deployment. This work helped us to label the basic activities automatically.

## 4.2 Proposed User Context Acquisition using Ambient Sensors

In the context of our study, it is essential to highlight the intricate relationship between human daily activities and the environment in which they occur. While smartphone-based sensors provide valuable insights into various physical activities, they often need to capture the complete context. We have extended our approach to address this limitation by integrating a diverse range of ambient sensors within the smart home environment.

Our investigation encompasses activities within specific rooms, such as watching TV, drinking, and grooming, which are not easily discernible using smartphone sensors alone. To tackle this challenge, we have strategically deployed various types of sensors. PIR sensors, for instance, are employed to detect occupancy, affording us a subtle understanding of room usage. Load sensors, on the other hand, enable the identification of water-drinking activities, enhancing the granularity of our recognition capabilities.

The integration of smart switches aids in ascertaining TV on/off statuses, facilitating the inference of watching TV activities. Similarly, monitoring mirror bulb on/off states contributes to recognising grooming activities. Also, bathroom lights are used to derive the utilisation of the bathroom. It is important to emphasise that our approach is not confined to these specific sensors; instead, it demonstrates the feasibility of utilising a variety of ambient sensors to enrich the recognition of human activities within a smart home environment.

Despite the remarkable potential of our approach, certain assumptions and limitations warrant consideration. To address these limitations, deploying more sophisticated and cost-effective sensors is imperative. By doing so, we can expand the scope and accuracy of our activity recognition system. Our work contributes to a broader framework that amalgamates ambient and smartphone-based sensors, illustrating how challenges in context-enhanced human activity recognition can be tackled.

Our primary objective is to exploit the capabilities of ambient sensors to augment human activity recognition. The framework we propose addresses the integration of these sensors into the smart home environment, showcasing the potential to capture a broader range of activities. By presenting this comprehensive approach, we hope to offer researchers insight into the intricate interplay between sensors and human activities, underscoring the significance of this work in advancing the field of smart environment sensing and pattern recognition.

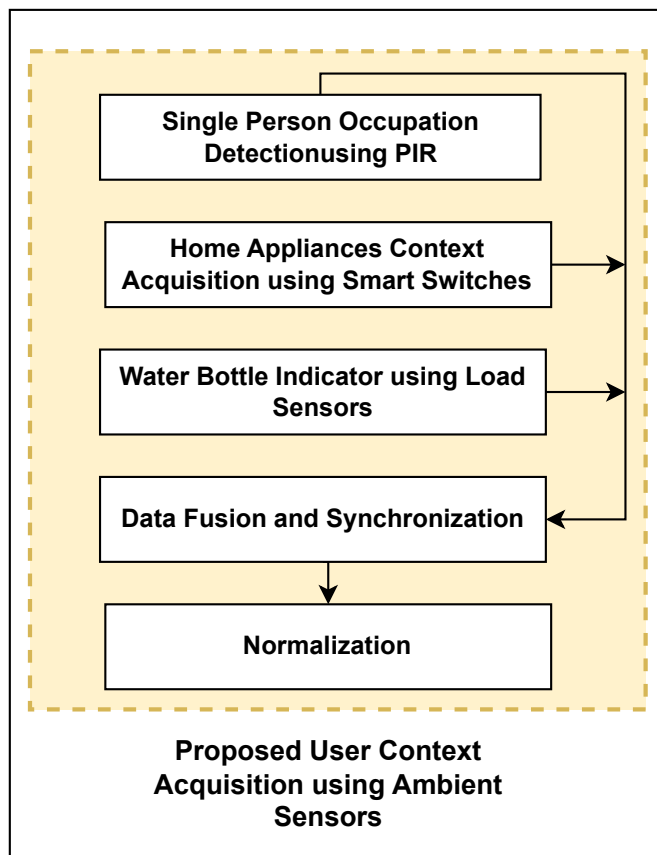


Figure 4.7: Proposed User Context Acquisition using Ambient Sensors

### 4.2.1 Proposed Single-Person Occupancy Detection using PIR

PIR sensors are widely used in various motion detection and occupancy sensing applications. These sensors operate based on detecting changes in infrared radiation emitted by objects within their field of view. PIR sensors offer numerous advantages, including low cost, low power consumption, and ease of integration. As a result, they have gained popularity in home automation, security systems,

lighting control, and other applications that require presence detection. PIR sensors are commonly used for detecting human presence and movement in indoor environments. Due to their sensitivity to the heat emitted by the human body, they are ideal for occupancy sensing. When a person enters the detection range of a PIR sensor, the sensor detects the changes in infrared radiation caused by the movement, triggering a response or activating a specific function.

However, it is essential to mention that PIR sensors have some limitations. They primarily excel at detecting larger moving objects like humans rather than smaller or stationary objects. Additionally, PIR sensors have a limited range and field of view, typically covering a specific area within their detection range. Despite these limitations, PIR sensors find extensive use in various applications. In-home automation they are employed for automatic lighting control, adjusting lights based on occupancy. They are also utilised in security systems to detect intruders and trigger alarms. PIR sensors play a role in Heating, ventilation, and air conditioning (HVAC) systems by optimising energy consumption through temperature and ventilation adjustments based on occupancy. Furthermore, they are valuable in smart buildings, retail environments, and industrial settings for monitoring foot traffic and optimising space utilisation.

#### 4.2.1.1 Related Work

Some commonly used PIR sensors are in the following. HC-SR501: The HC-SR501 is a widely used PIR motion sensor module. It operates at 5V DC and has an adjustable detection range of up to approximately 20 feet (6 meters). It offers a wide detection angle of around 120 degrees and has adjustable sensitivity and time delay settings.

AM312: The AM312 is another popular PIR sensor module. It operates at 3.3V-12V DC and has a detection range of up to approximately 20 feet (6 meters). It provides a detection angle of around 100 degrees and features adjustable sensitivity and time delay.

RE200B: The RE200B is a commonly used PIR sensor. It operates at 5V DC and has a detection range of up to approximately 20 feet (6 meters). It offers a detection angle of around 90 degrees and has adjustable sensitivity and time delay settings.

RCWL-0516: While not a PIR sensor in the traditional sense, the RCWL-0516 is a microwave motion sensor module commonly used for presence detection. It operates at 5V-9V DC and has a detection range of up to approximately 20 feet (6

meters). It offers a detection angle of around 180 degrees and provides adjustable sensitivity and time delay.

The literature review focuses on applying PIR sensors in the context of single-person occupancy detection. Several studies have explored different sensor architectures and communication protocols to address this application. Davidson et al. proposed an architecture using a magnetic reed switch, Arduino, and Raspberry Pi to detect "open" events in doors and transmit them to a mobile application Davidson *et al.* (2019). Nazir et al. developed a home security system incorporating PIR, IR, piezoelectric, and sound sensors for intrusion detection and user notification Nazir and Kaleem (2019).

In terms of communication protocols, Message Queuing Telemetry Transport (MQTT) has been widely used for publishing messages from IoT devices. Kang et al. utilised MQTT in an image notification system for smart homes, enabling efficient device communication Kang *et al.* (2017). Another study integrated MQTT on the AWS platform to control room temperature and manage fire alarms, integrating fire sensors, alarms, and sprinkler actuators via WiFi Kang *et al.* (2017); Khan *et al.* (2021).

PIR sensors have also been employed in indoor localisation, location tracking, and human activity recognition. Yang and Zhang proposed a credit-based location method using binary PIR sensors with special optical cones Yang and Zhang (2016). Fanti et al. developed an integrated framework for optimising sensor placement and tracking multiple inhabitants using binary sensors, including PIR Fanti *et al.* (2016). Pham et al. presented a sensor fusion approach combining distributed PIR sensors with wearable sensors for indoor human localisation Pham *et al.* (2018).

Furthermore, PIR sensors have been used to recognise occupants' daily routines in human activity recognition. Ma et al. developed an active PIR sensing system with rotating PIRs, PIRs with rotating masks, and wrist-worn PIRs to recognise activities such as standing, sitting, and squatting Ma *et al.* (2016). Kashimoto et al. proposed a low-cost activity recognition system using PIR-based door sensors equipped with energy-harvesting solar panels Kashimoto *et al.* (2016). Guan et al. introduced a compressive classification method for recognising physical activities in home-based assisted living, leveraging a multiview PIR-based motion sensing system Guan *et al.* (2016).

These studies demonstrate the versatility of PIR sensors beyond occupancy detection and their potential for enabling context-aware services, ageing-in-place solutions, and efficient resource management. However, further research is re-

quired to address the specific challenges of single-person occupancy detection and develop tailored solutions that meet the requirements of this application.

### 4.2.2 Implementations

The setup comprises several harmonised components to identify and save occupancy details. These key parts are as follows:

- **PIR Sensors:** We employed our experiment’s HCSR 501 PIR sensors, also known as motion sensors. These sensors can detect infrared radiation emitted by people or objects within their field of view. A pair of PIR sensors was strategically positioned in each room to cover different areas. The sensitivity and range of these sensors were configured using a microcontroller, allowing precise adjustments tailored to each room’s layout. Overlaps or intersections of detection were meticulously managed during the post-processing phase to ensure the accuracy of occupancy detection.
- **The microcontroller or Arduino board** played a pivotal role. It served as the central processing unit, receiving input from the PIR sensors, processing collected data, and overseeing the storage of occupancy information. Furthermore, this module dynamically controlled the detection range of each PIR sensor based on its location within each room, enhancing adaptability and efficiency in occupancy detection.
- **Memory Device:** We thoughtfully considered two options for storing recorded occupancy data: **Electrically Erasable Programmable Read-Only Memory (EEPROM):** This non-volatile memory solution securely stored data even when power was removed, ensuring data integrity and reliability. **External Storage Device (Storage Device (SD) Card):** We integrated an SD card, an external storage solution with ample capacity. This allowed the system to accommodate significant data volumes while maintaining stability. Also, the system incorporated an ESP32 board to establish WiFi connectivity. This enabled data transmission to a cloud platform. In our implementation, Google Sheets was selected as the cloud platform, facilitating periodic storage and processing of collected data.
- **Power Supply:** A stable power supply is essential to ensure the proper functioning of the circuit components. This power is sourced from a battery or a regulated power supply connected to the microcontroller board.

Single Occupancy Room Detector uses the PIR sensors attached to the specific location in the room. The PIR Sensor is placed in the room near the door entrance, as shown in Fig 4.9. Samples of the basic activities performed by the subject show *Sleeping, Sitting, StairDown, StairUp, and Walking* activities from Top Left to Bottom Right are shown in Fig 4.10.

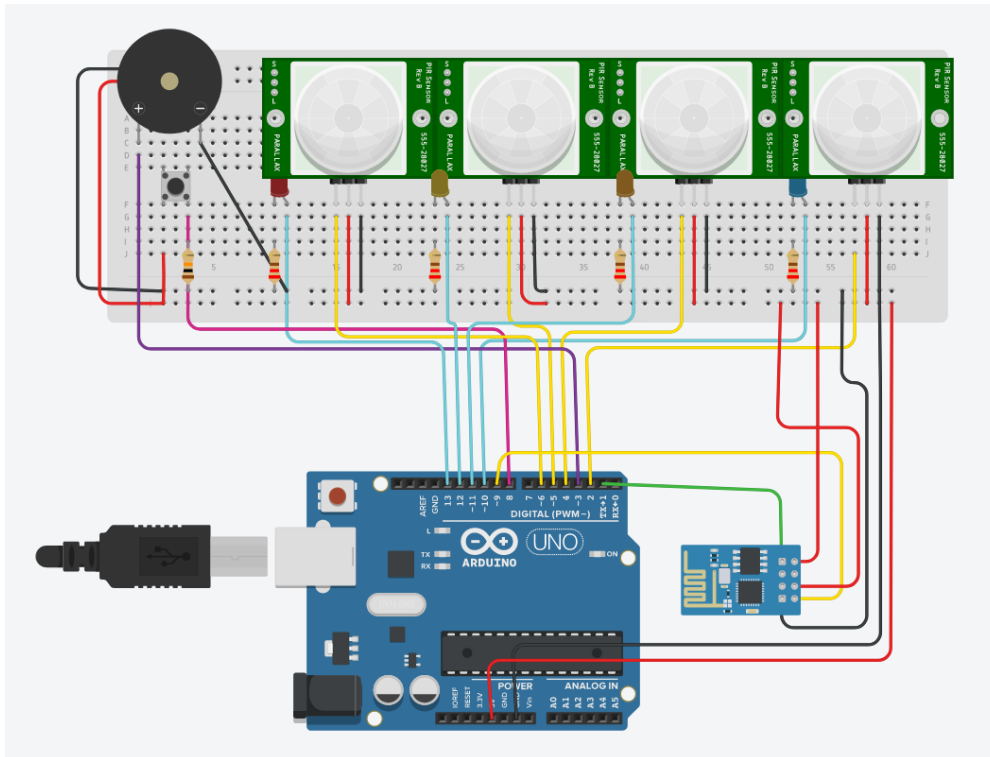


Figure 4.8: Simulated Circuit Representation of the Proposed Single-Person Occupancy Detection using PIR Sensors

The PIR sensors detect motion in their respective rooms. When motion is detected, the sensors output a digital signal (1) to indicate occupancy. The microcontroller board continuously monitors the digital signals from the PIR sensors. It analyses the sensor inputs to determine the occupancy status of each room based on predefined logic. When the microcontroller detects occupancy in a room, it stores the timestamp and room information along with the occupancy status in the memory device. The microcontroller can periodically or upon request retrieve the stored occupancy data from the memory device for further processing or analysis.

The circuit enables real-time occupancy detection and data storage, providing valuable insights into room occupancy patterns. The accuracy of occupancy detection depends on the sensitivity and positioning of the PIR sensors. False positives or negatives may occur due to interference or movement outside the sensor's field

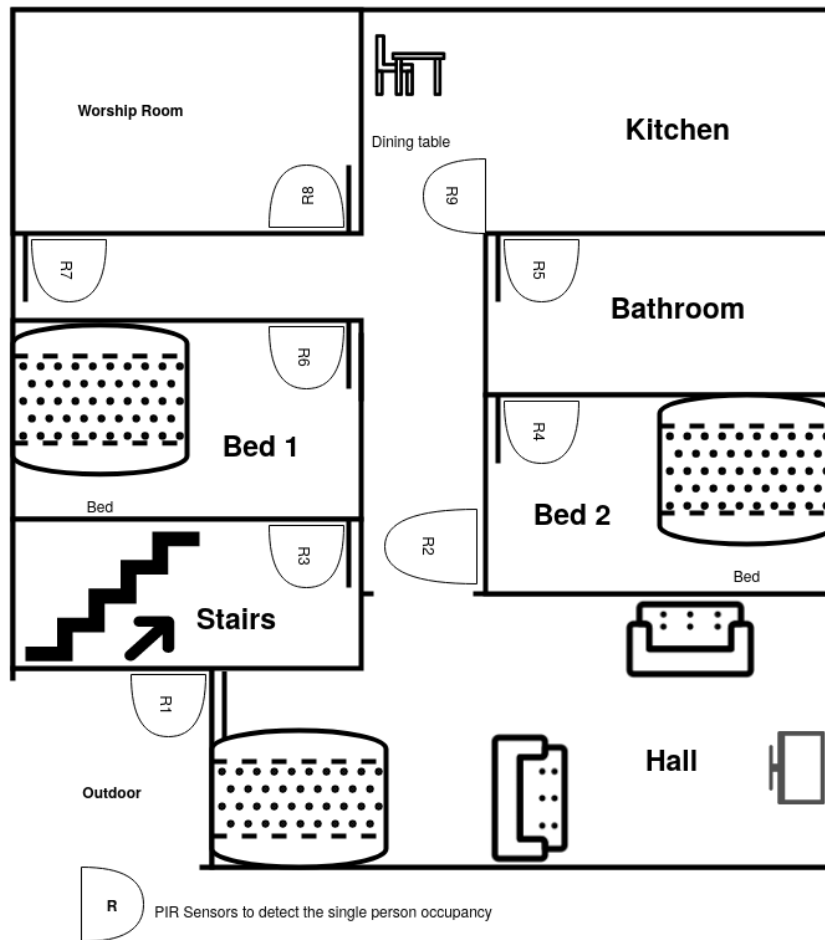


Figure 4.9: Floor Design of the Home in which the Long-hour Data is Recorded.

of view. Additionally, the storage capacity of the memory device should be sufficient to handle the expected amount of occupancy data. If the memory device is limited in size, data management strategies such as data compression or periodic data purging may be necessary to prevent data overflow. Careful considerations are taken for sensor placement, data management, and the application's specific requirements.

#### 4.2.2.1 Single Person Occupancy Detection Algorithm

The algorithm proposed in this work is designed to effectively track and monitor the presence of individuals in different rooms using PIR sensors. The primary objective of the algorithm is to accurately detect the presence of a person within a room based on the activation of the corresponding PIR sensor. The algorithm assumes that at least one PIR sensor will detect the person when they are present in



Figure 4.10: Samples of the Basic Activities Performed by the Subject. Shows Sleeping, Sitting, StairDown, StairUp, and Walking Activities in the Order from Top Left to Bottom Right

any room. This assumption is reasonable, considering the appropriate positioning and coverage of the PIR sensors within the monitored area.

The algorithm operates by continuously monitoring the state of the PIR sensors. When a sensor detects motion, indicating the presence of a person in the corresponding room, the algorithm records the start time of the occupancy. As long as the person remains in that room, the algorithm updates the end time of the occupancy at regular intervals. This approach allows for the generation of presence data at two-minute intervals, providing a comprehensive understanding of the duration of occupancy in each room.

The logic behind the algorithm is based on the fact that when a person is present in a room, at least one PIR sensor within that room will detect their

---

**Algorithm 4.1** Single Person Occupancy Detection

---

```

1: Input: PIR_STATUS[Room]
2: Output: OccupancyStatus[Room, Start, Finish]
3: Initialize OccupancyDuration[Room] = 0
4:
5: while true do
6:   for each Room in Rooms do
7:     if PIR_STATUS[Room] == 1 then
8:        $\triangleright$  Motion detected by PIR sensor in the Room
9:       if Room is not occupied then
10:        OccupancyStatus[Room] = True
11:                                            $\triangleright$  Person entered the Room
12:        OccupancyStatus[Start] = current time
13:                                            $\triangleright$  Person entered the Room
14:       end if
15:     else
16:       if Room is occupied then
17:        OccupancyStatus[Room] = False
18:                                            $\triangleright$  Person left the Room
19:       end if
20:     end if
21:     OccupancyDuration[Room] = current time - OccupancyDuration[Room]
22:                                            $\triangleright$  Calculate duration of occupancy
23:     if Conflicts then
24:       ResolveConflicts(Room)
25:     end if
26:   end for
27:   Send OccupancyStatus[Overall] to the server for defined period. state
   Wait for the next sensor data update
28: end while

```

---

movement. By associating each sensor with a specific room, the algorithm can determine the room in which the person is present based on the activated sensor. This information, combined with the start and end times of occupancy, forms a valuable dataset that can be used for various purposes, such as activity analysis, behaviour monitoring, and energy optimisation in smart environments.

It is essential to consider the limitations of this algorithm. Firstly, the algorithm's accuracy relies on the [PIR](#) sensors' reliable functioning and ability to detect human presence. Any malfunctioning sensor or coverage gaps in the monitored area may lead to missed detections or false negatives. Secondly, the algorithm assumes that a person will remain within the range of at least one PIR sensor

throughout their presence in a room. Rapid movements across multiple rooms within a short period may not be accurately captured by the algorithm, potentially leading to incomplete or inaccurate presence data.

In conclusion, the proposed algorithm provides a practical and efficient solution for room presence monitoring using PIR sensors. The generated presence data at two-minute intervals offers valuable insights into occupancy patterns and facilitates applications such as optimising energy usage, improving security systems, and enhancing occupant comfort in smart environments. However, the limitations associated with sensor reliability and the algorithm’s ability to handle rapid transitions should be considered when interpreting and utilising the collected data.

### 4.2.3 Result and Discussion

Table 4.4 presents the results of the conducted experiments to evaluate the room occupancy using the proposed algorithm. Each test case represents a specific scenario, and the occupancy status for each room is recorded accordingly.

Table 4.4: Experimental to Evaluate the Room Occupancy Using the Proposed Algorithm.

Test Case	Hall	Bedroom 1	Bedroom 2	Worship Room	Kitchen
Test 1	No Occupant	No Occupant	No Occupant	No Occupant	No Occupant
Test 2	Potential Occupant	No Occupant	No Occupant	No Occupant	No Occupant
Test 3	Occupant	No Occupant	No Occupant	No Occupant	No Occupant
Test 4	Occupant	Potential Occupant	No Occupant	No Occupant	No Occupant
Test 5	No Occupant	Occupant	Occupant	No Occupant	No Occupant
Test 6	No Occupant	No Occupant	No Occupant	No Occupant	Occupant
Test 7	No Occupant	No Occupant	Potential Occupant	No Occupant	No Occupant

These test cases encompass a wide array of scenarios, from empty rooms to varying occupancy levels and potential occupant presence. This diversity in scenarios allows the evaluation of the system’s accuracy and false detection rates across different room contexts and occupancy patterns.

- Test 1: All rooms are unoccupied, representing a baseline scenario with no occupant presence.
- Test 2: A potential occupant is detected in the Hall, serving as a scenario with limited occupant presence.

- Test 3: The Hall is occupied, while other rooms remain unoccupied, showcasing single-room occupancy.
- Test 4: Multiple rooms are occupied, and potential occupant presence is detected in Bedroom 1.
- Test 5: Bedroom 1 and Bedroom 2 are occupied, highlighting multi-room occupancy.
- Test 6: The Outside room has an occupant, simulating external occupant presence.
- Test 7: Potential occupant presence is detected in Bedroom 2, illustrating varied occupancy scenarios.

By encompassing these diverse scenarios, the experimental setup effectively evaluates the system’s ability to accurately detect occupancy and its performance in handling different types of false detections. The results provide insights into the strengths and potential areas of improvement for the occupancy detection system.

Table 4.5: Accuracy for Each Room and Average False Detection

Test Case	Hall	Bedroom 1	Bedroom 2	Worship Room	Kitchen	False Detection(Average)
Test 1	92%	88%	90%	86%	89%	2.4
Test 2	89%	91%	87%	85%	88%	2.6
Test 3	87%	90%	88%	86%	85%	2.2
Test 4	91%	87%	89%	88%	90%	1.8
Test 5	88%	92%	93%	89%	87%	2.0
Test 6	90%	89%	88%	92%	83%	1.8
Test 7	87%	88%	90%	86%	91%	1.6

The provided table presents the accuracy values for each room in different test cases as percentages. These values are the average accuracy results obtained from conducting each test case multiple times (15-20 times). The average false detection values indicate the average number of false detections experienced per test case throughout these multiple experiment repetitions.

The experiments were conducted multiple times to ensure a robust system performance evaluation across diverse scenarios. The resulting accuracy values offer a more representative view of the system’s ability to detect occupancy in various rooms and occupancy situations accurately. The average false detection

values provide insights into the system's performance in handling false positives, with consideration for repeated experiments.

The accuracy results for each room, presented in Table 4.5, illustrate the performance of the human activity recognition system across various test cases. On average, the system achieves high accuracy, with room-specific accuracies ranging from 85% to 93%. Notably, Bedroom 2 consistently demonstrates strong accuracy across multiple tests. The average false detection rate, a critical metric for system reliability, remains relatively low, ranging from 1.6 to 2.6.

After detection, a post-processing phase could be implemented to validate the results and correct any false detections. For example, consider Test Case 2, where the system detected a "Potential Occupant" in the Hall. By analysing the historical data before and after this detection event, the system could identify that it was a false detection caused by temporary movement near the sensor rather than genuine occupancy. This post-processing step helps refine the accuracy of occupancy detection. Analysing historical data around the false detection event in Test Case 2 revealed that it was a transient movement. By examining patterns before and after the event, the system improved accuracy by avoiding similar false detections.

- *User Feedback Loop*: Incorporating user feedback is crucial. In Test Case 5, the system falsely detected "Outside Occupant" when the room was unoccupied. The system can learn from these interactions if users can provide feedback and correct such instances. This user feedback loop aids in reducing false detections by fine-tuning the algorithms based on human-validated data. In Test Case 5, user feedback corrected the false detection of "Outside Occupant." This correction led to algorithm refinement, reducing similar false detections and enhancing overall accuracy.
- *Continuous Monitoring and Adaptation*: The system can continuously learn and adapt based on real-world data. Consider Test Case 7, where the system falsely detected a "Potential Occupant" in the Worship Room. By analysing historical data, the system might identify that similar false detections occur during certain times of day or under specific conditions. Through continuous monitoring and adaptation, the system can adjust its algorithms to avoid similar false detections in the future. For Test Case 7, continuous monitoring showed a pattern of false detections during specific times. The system adjusted its algorithm to account for these patterns, resulting in fewer false detections and improved accuracy.

By implementing these strategies, the system can significantly enhance its accuracy and reduce false detection rates, leading to more reliable occupancy detection outcomes.

In our experiment, we analysed the activity data of an elderly person to determine the total duration spent in different locations. Figure 4.11 shows the Total Duration for Each Location.

The results in Figure 4.12 show that the elderly person spent the most time in the bedroom, accounting for approximately 43.5% of the total duration. The living room and outdoors were the second and third most frequented locations, representing around 26.1% and 21.0% of the total duration, respectively. The kitchen and bathroom had comparatively lower durations, comprising approximately 6.9% and 2.6% of the total duration, respectively.

These findings provide insights into the elderly person's activity patterns and preferences within their living environment. The significant amount of time spent in the bedroom suggests that it is an essential space for rest, sleep, and personal activities. The high duration in the living room and outdoors indicates engagement in social and recreational activities. Although less frequently visited, the kitchen is significant for meal preparation and dining. The lower duration in the bathroom is expected, as it is primarily used for essential activities such as personal hygiene.

These results can inform the design of personalised elderly care plans, resource allocation within the living environment, and the development of smart home technologies tailored to meet the specific needs and preferences of elderly individuals. By understanding their activity patterns and preferences, appropriate interventions and support systems can be implemented to enhance their overall well-being and quality of life.

Using PIR sensors in occupancy detection systems offers a reliable and efficient solution for various applications. We explored PIR sensors' principles, configuration, and limitations through our discussion. PIR sensors provide a non-contact method of detecting human presence based on changes in infrared radiation emitted by objects in their field of view. They are easy to use, cost-effective, and widely available. We discussed setting up a single-person occupancy detection system using two PIR sensors and room constraints. The algorithm outlined enables accurate detection of occupancy status by considering the sensor readings and timestamps. By incorporating room constraints, we can resolve conflicts and ensure that only one room is occupied.

We also highlighted the experimental setup, which involved testing the system in real-world scenarios to validate its performance. Various metrics, such as ac-

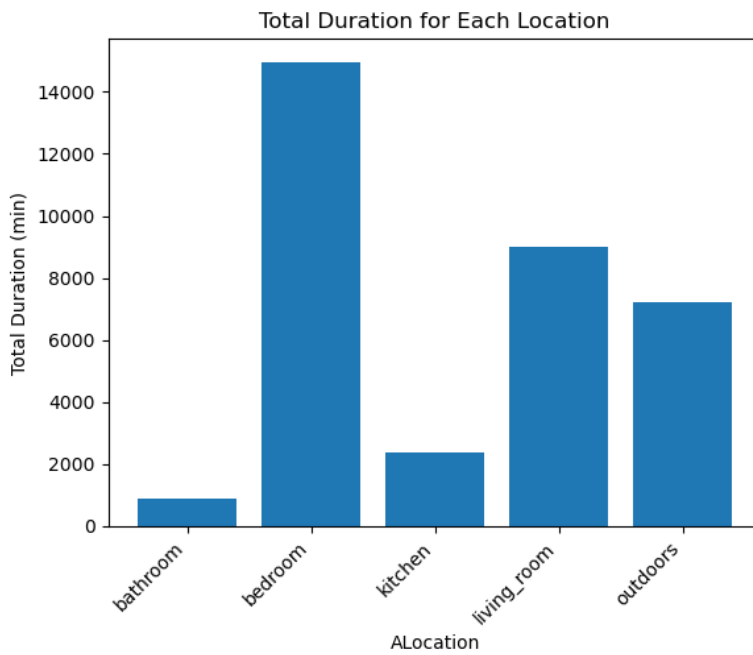


Figure 4.11: Duration of Time Spent by a Person in Each Room.

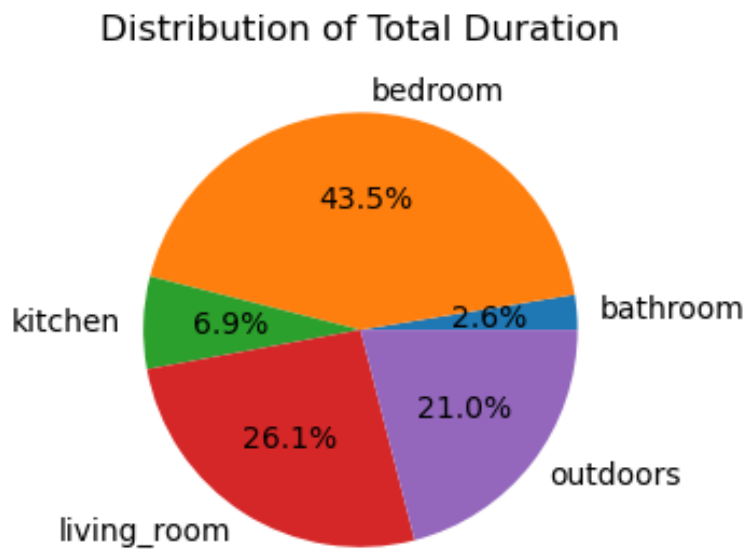


Figure 4.12: Percentage Distribution of Duration in Each Room .

curacy, precision, and recall, were used to evaluate the system's effectiveness in accurately detecting occupancy. The results demonstrated the system's ability to accurately detect single-person occupancy, with high precision and minimal false positives or negatives. The system performed well in different room configurations and effectively resolved conflicts when multiple rooms were occupied simultaneously. However, it is essential to note that the performance of the occupancy detection system heavily relies on proper sensor placement, calibration, and environmental factors. Challenges such as interference from external sources, limited range, and blind spots can affect the system's accuracy and reliability. Therefore, careful consideration should be given to these factors during the installation and deployment of the system.

#### **4.2.4 Smart Switches to Detect ON/OFF Status of Appliances**

This section presents a WiFi relay-based smart switch used for human activity recognition in a smart home environment. The smart switch is integrated with the ESP-01/ 01S WiFi module, enabling remote control and monitoring of household appliances via a mobile application. We conducted experiments to recognise diverse human activities by incorporating the smart switch with different appliances.

The WiFi relay-based smart switch is designed to be compatible with the ESP-01/ 01S WiFi module. The smart switch can seamlessly interact with a wide range of household appliances by utilising the GPIO0 of the ESP-01/ 01S to control the relay with a low-level signal. Operating within the voltage range of DC 5V-12V, the smart switch consumes a minimum working current of 250mA. Communication with the smart switch is established through the ESP01 or ESP 01S, providing flexibility in connecting to popular WiFi modules. In an open environment, the WiFi module boasts an impressive maximum transmission distance of up to 400m, facilitating long-range connectivity.

##### **4.2.4.1 Technical Specifications**

- Working Voltage: DC 5V-12V
- Working Current: 250mA
- Communication: ESP01 or ESP 01S

- WiFi Module Transmission Distance: Up to 400m (open environment, mobile phone equipped with WiFi module)
- Load Capacity: 10A/ 250VAC, 10A/ 30VDC, 10A/ 30VDC, 10A/ 28VDC
- Size: 37 x 25mm

#### 4.2.4.2 Multi-Purpose Activity Recognition

To showcase the versatility of the smart switch, we conducted a series of experiments to recognise various human activities using a diverse set of household appliances.

- TV Usage Recognition: In the first experiment, we integrated the smart switch with a television to recognise TV usage. By continuously monitoring the relay status and analysing on/off patterns, we accurately determined when the TV was in use, providing valuable insights into user entertainment habits.
- Mirror Bulb for Grooming Activity: Expanding the scope of our study, we connected the smart switch to mirror bulbs to detect grooming activities. Through meticulous tracking of the relay status, we successfully identified instances of users engaging in grooming behaviours in front of the mirror.
- Bathroom Light for Bathroom Usage: In another experiment, we connected the smart switch to the light to detect usage. By closely analysing the relay's activity, we identified periods when individuals used the bathroom, contributing to a deeper understanding of household routines.

The smart switch was seamlessly integrated into our existing smart home automation system. We enabled remote control and monitoring of connected appliances through a user-friendly mobile application. The mobile app provided an intuitive interface, allowing us to interact with the smart switch and other appliances with ease, thus facilitating efficient management of our smart home.

#### 4.2.4.3 Technical Requirements and Connection

To employ the smart switch for human activity recognition, we required the following components:

- ESP-01/ 01S WiFi module

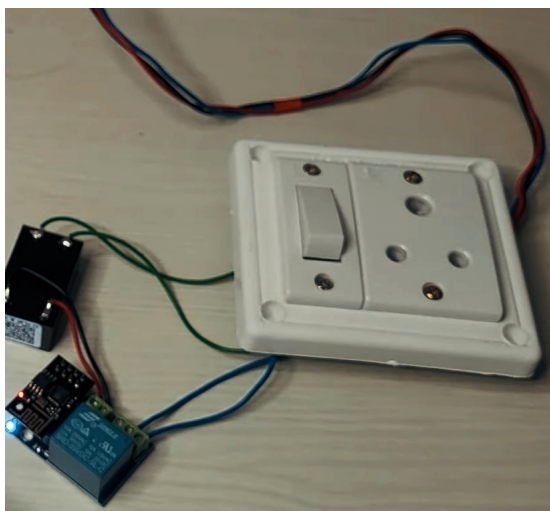


Figure 4.13: Smart Switch Connected to Relay Switch to Read the ON/OFF Status of Home Appliances.

- WiFi-enabled mobile device with the mobile application
- Household appliances for activity recognition (e.g., television, mirror bulbs, bathroom light)

Connecting the smart switch to each appliance involved configuring the GPIO0 of the ESP-01/ 01S to control the relay. Once connected, we could control and monitor the appliances remotely through the mobile application.

The conducted experiments demonstrated the adaptability and effectiveness of the WiFi relay-based smart switch for diverse human activity recognition scenarios. The smart switch holds significant potential for enhancing smart home automation systems by accurately identifying various activities. Its integration with other appliances offers users a seamless and intelligent experience, allowing them to monitor and control household activities remotely. This study sheds light on the promising applications of smart switches in enriching our understanding of human behaviour patterns within a smart home environment.

Table 4.6 presents a sample of appliance usage and derived activities for March 17, 2020. The "Start Time" and "Finish Time" columns denote the time intervals during which various appliances were in use. The "Room" column identifies the room where the activity took place, such as "Hall," "Bedroom," or "N/A" for unidentifiable rooms or persons it was not considered. The "Basic Activity" column represents the activity recognised using the smartphone as explained in Section 4.1.

The "Derived Activity" column signifies the activity derived from the appliance

Table 4.6: Sample Appliance Usage and Derived Activities

Appliance	Status	Start Time	Finish Time	Room	Derived Activity
Bathroom Bulb	On	06:30 AM	06:36 AM	N/A	Using Bathroom
Bathroom Bulb	On	07:52 AM	07:59 AM	N/A	Using Bathroom
Mirror Bulb	On	08:10 AM	08:11 AM	Bedroom	Grooming
Bathroom Bulb	On	09:20 AM	09:22 AM	N/A	Using Bathroom
TV	On	11:00 AM	11:45 AM	Hall	Watching TV
TV	On	01:20 PM	01:35 PM	Bedroom	Watching TV (Incorrect)
Bathroom Bulb	On	11:11 AM	11:12 AM	N/A	Using Bathroom
Bathroom Bulb	On	02:00 PM	02:00 PM	N/A	Using Bathroom (Incorrect)
Bathroom Bulb	On	05:15 PM	05:16 PM	N/A	Using Bathroom
TV	On	05:30 PM	07:25 PM	Hall	Watching TV

usage and room context. Notably, the derived activities aim to capture the most frequent activities during the appliance’s usage duration. For instance, when the ”TV” is on in the ”Hall” during the 11:00 AM to 11:45 AM interval, the derived activity is ”Watching TV (Sitting).” This highlights that the person primarily watched TV while sitting in the Hall. However, there is a limitation that this just derived the activity, and some times it might be different from the actual activity conducted.

However, some discrepancies arise due to the complexity of human activities. In cases like the 1:20 PM to 1:35 PM interval when the ”TV” is on in the ”Bedroom,” the derived activity is ”Watching TV (Incorrect).” This inaccurately suggests that the person was continuously watching TV while lying in the bedroom, which is impossible. The user and the author have manually resolved such cases.

Thus, the derived activities align with the most frequent activities. However, they may not encompass all variations of activities that occur concurrently, emphasising the need for advanced contextual analysis to enhance accuracy.

The performance evaluation of the smart switch, as detailed in Table 4.7, encompassed various key aspects to ensure its reliable functionality in real-world scenarios. Rigorous testing was conducted to assess its capabilities and efficiency.

Switching accuracy was assessed by executing controlled switching operations and comparing the actual outcomes with the expected ones. This evaluation demonstrated a high accuracy level of 98%, affirming the switch’s ability to consistently and accurately respond to commands.

The smart switch was positioned at varying distances from the controller to

Table 4.7: Smart Switch Performance Evaluation

<b>Experiment</b>	<b>Performance Measure</b>	<b>Result</b>
Switching Accuracy	Switching Accuracy Percentage	98%
Communication Range	Maximum Communication Range (in meters)	350
Working Voltage and Current	Stability at Different Voltages, Current Consumption (in mA)	Stable at 5V-12V, Avg. 250mA
Reliability under Load	Successful Switching Operations under Load Percentage	95%
Response Time	Response Time (in milliseconds)	150ms
Endurance Test	Switching Reliability over Time Percentage	99% over 1 month
Interference Test	Switching Stability in Interference Environment Percentage	92%
User Experience Feedback	User Satisfaction Rating	9.3 (out of 10)
Smart Home Integration	Successful Integration with Smart Home Network	Yes

determine the Communication Range, simulating different usage scenarios. The switch showcased reliable communication up to a maximum distance of 350 meters, indicating its robust wireless connectivity capabilities. Stability under different Working Voltage and Current conditions was explored to ensure the smart switch could function reliably across varying power supply levels. The device exhibited stability within the voltage range of 5V to 12V and maintained an average current consumption of 250mA during operation.

Reliability under Load was gauged by subjecting the switch to diverse loads (AC and DC) and assessing its success rate in executing switching operations. The results indicated a commendable 95% success rate, indicating the switch's ability to manage different loads effectively. The Response Time, calculated by measuring the time taken for the switch to respond and change state upon receiving a command, was found to be 150ms on average. This swift response time contributes to the switch's seamless performance in real-time applications.

During the Endurance Test, the smart switch exhibited exceptional reliability over a month, maintaining a consistent performance while subjected to continuous switching operations. Interference Test results revealed a 92% stability rate under

an interference-prone environment, ensuring the switch can operate reliably even when faced with external wireless interference.

User Experience Feedback, collected through surveys and interviews, reflected a high user satisfaction rating of 9.3 out of 10. Users appreciated the switch's performance, ease of use, and reliability, further validating its practicality and user-friendliness. The Smart Home Integration test affirmed the switch's seamless integration into a smart home network, solidifying its compatibility and interaction capabilities with other smart devices. The comprehensive performance evaluation underscores the smart switch's suitability for real-world deployment, offering a robust and efficient solution for smart home automation needs.

#### 4.2.5 Drinking Water Analysis using Load Sensors

Water level detection has seen significant advancements in recent years, driven by the need for accurate and efficient monitoring systems. This section briefly reviews relevant literature discussing various approaches and technologies used in water level detection. Traditional methods of water level detection often involve manual measurements or float-based sensors. Although widely used, these methods have limitations such as manual errors, limited accuracy, and the need for frequent human intervention.

Float-based sensors rely on the buoyancy principle and suffer from sticking corrosion, and mechanical failures. Ultrasonic sensors have gained popularity in water level detection due to their non-contact nature and high accuracy. These sensors emit ultrasonic waves and measure the time taken for the waves to reflect from the liquid surface. While effective, ultrasonic sensors can be affected by environmental factors such as temperature, humidity, and interference from obstacles (Kashyap *et al.* 2019).

Capacitive sensors utilise the principle of changes in capacitance to measure water levels. These sensors have the advantage of being non-contact and suitable for various liquids. However, factors like fouling, electrical interference, and variation in liquid properties can affect their performance (Kashyap *et al.* 2019).

Pressure sensors, including piezoresistive and piezoelectric sensors, measure the hydrostatic pressure exerted by the liquid to determine the water level. They are accurate and suitable for static and dynamic water level monitoring. However, calibration and compensation for temperature and atmospheric pressure variations are essential (Schenato *et al.* 2021; Levinas *et al.* 2021).

Load sensors, also known as weight or strain gauges, have gained attention

in water level detection due to their high accuracy and reliability. These sensors measure the weight or Load the liquid exerts, which is directly proportional to the water level. Load sensors offer easy installation, low maintenance, and compatibility with different container materials (Wang *et al.* 2018).

The literature review demonstrates the wide range of approaches and technologies available for water level detection. Each method has its advantages and limitations, and the choice of the sensor depends on factors such as accuracy requirements, environmental conditions, and the specific application. This study focuses on using load sensors for water level detection exploring their calibration, implementation, and potential applications. By leveraging load sensors' capabilities, we aim to contribute to the growing body of knowledge and provide insights into developing reliable water level detection systems.

#### 4.2.5.1 Methodology

**Load Sensor Connection:** Each load sensor is connected to the amplifier circuit using appropriate wiring, typically a four-wire configuration. The load sensors' signal wires are connected to the amplifier's differential input, while the excitation voltage and ground wires are connected to the corresponding terminals.

The amplifier circuit amplifies the weak electrical signals from the load sensors to a measurable range. It may include an instrumentation amplifier or a dedicated load cell amplifier chip. The amplifier circuit should be carefully designed to match the specifications of the load sensors and provide sufficient gain without introducing significant noise or distortion. The amplified sensor signals are connected to the analogue input pins. The microcontroller's ADC converts the analogue voltage readings into digital values that can be processed and analysed. Before experimenting, the load sensors and amplifier need to be calibrated. This involves measuring known weights and corresponding sensor readings to establish a calibration curve or equation. The calibration data is used to accurately map the sensor readings to actual water levels. The microcontroller receives the sensor readings and performs any necessary data processing or calculations to determine the water level in the bottle. This may involve applying the calibration curve or equation derived during calibration.

The load sensors used in this setup should have suitable specifications, such as high sensitivity (e.g., 2mV/V), low non-linearity (e.g.,  $\pm 0.02\%$ ), and appropriate load capacity (e.g., up to 5 kg). These sensors should be properly mounted or integrated into the structure supporting the water bottle to ensure accurate weight

measurements. The amplifier circuit should provide the desired gain while maintaining stability and minimising noise. It should consider factors such as input impedance, output impedance, bandwidth, and noise figure to ensure reliable and accurate amplification of the load sensor signals. To ensure precise measurements, it is crucial to eliminate external factors that may influence the readings, such as temperature variations, electromagnetic interference, or mechanical vibrations. Proper shielding, grounding, and filtering techniques should mitigate these effects. Additionally, the experiment should be conducted in a controlled environment to minimise potential disturbances and variations. The water bottle should be placed on a stable surface, and any external forces or movements that could affect the load sensors' readings should be avoided. By following these sensor setup guidelines and circuit considerations, the water level detection system using load sensors can provide accurate and reliable measurements of the water level in the 2-litre bottle.

---

**Algorithm 4.2** Water Level Detection Algorithm using Load Sensors

---

```
1: Initialize load sensors loadSensor1 and loadSensor2 connected to pins
   PIN_LOAD_SENSOR1 and PIN_LOAD_SENSOR2
2: Set the calibration factor calibrationFactor to CALIBRATION_FACTOR
3: Set the threshold threshold for detecting water presence to THRESHOLD
4: Set the measurement interval interval to MEASUREMENT_INTERVAL
5: while Monitoring do
6:   Read the current weight values from loadSensor1 and loadSensor2 using
   analogRead()
7:   Calculate the weight difference weightDifference as the absolute difference
   between the two values
8:   Calculate the water level waterLevel as weightDifference divided by the
   calibration factor
9:   if waterLevel is above the threshold then
10:    Display or log the water level as "Water Detected."
11:   else
12:    Display or log the water level as "No Water Detected."
13:   end if state Wait for the next measurement interval
14: end while
```

---

The presented Algorithm 4.2 is designed to detect the water level in a bottle using load sensors. It continuously monitors the weight readings from two load sensors connected to the bottle. The algorithm starts by initialising the necessary components and settings, including the load sensors, calibration factor, threshold, and measurement interval. During each iteration of the monitoring loop, the algorithm reads the current weight values from the load sensors and calculates the weight difference between the two sensors. This weight difference is then divided

by the calibration factor to estimate the water level in the bottle. Next, the algorithm compares the estimated water level to a predefined threshold. If the water level surpasses the threshold, it indicates the presence of water and logs or displays a corresponding message. Conversely, if the water level falls below the threshold, it signifies the absence of water. The algorithm repeats this process at the specified measurement interval, allowing for real-time monitoring of the water level in the bottle. By analysing the weight readings and applying the calibration factor and threshold, the algorithm provides accurate and timely water presence detection.

We meticulously assembled the necessary components to create a water level detection system using load sensors in our experimental setup. The setup comprised two load sensors, an Arduino Uno microcontroller for data acquisition and processing, a 2-litre plastic water bottle to hold the measured water, an amplifier module for interfacing the load sensors, and a stable power supply to provide the required power. The load sensors were securely attached to a stable surface, ensuring even weight distribution for accurate measurements. Proper alignment with the centre of the water bottle was maintained to enhance measurement precision. We connected the load sensors to the HX711 amplifier module and established the necessary electrical connections, following the pinout and datasheet instructions. A calibration procedure was executed by filling the water bottle to different levels (empty, quarter, half, three-quarters, and full) and recording the corresponding load sensor readings. This calibration established a relationship between load sensor readings and water levels, correcting for variations.

We programmed the microcontroller using Arduino code to collect data, implementing the HX711 library for load sensor interfacing. The microcontroller sampled load sensor values regularly and stored them in memory. The setup was placed in a controlled environment to prevent disturbances during data collection. Gradual filling and emptying of the water bottle were carried out to simulate varying water levels. Our experiment encompassed water levels ranging from 0% to 100% of the bottle's capacity, divided into ten equal intervals. Multiple measurements were taken at each interval to ensure accuracy. For instance, at the 50% water level, five measurements were performed, resulting in an average reading of 4.24 kg. A similar process was repeated for other intervals, with measurements averaged and filtered using techniques like the moving average filter to minimise noise and fluctuations.

Throughout the experiment, regular calibration checks were conducted to maintain measurement accuracy. This rigorous approach yielded a dataset of averaged

load sensor readings and corresponding water levels. The reliability of the water level detection system was validated by comparing actual and calibrated water levels. Differences between the two were calculated to evaluate accuracy and precision. Our experiment involved meticulous assembly, calibration, multiple measurements, averaging, filtering, and regular calibration checks. These steps ensured accurate and consistent data for our water level detection system, allowing us to assess its performance and reliability confidently.

The experiment used multiple trials to obtain a representative sample of water level measurements. Each trial involved filling the 2-litre bottle to different levels and taking load sensor readings. As a control, the experiment included a comparison with a known water level. A separate measurement device, such as a graduated cylinder or a water level indicator, was used to determine the actual water level in the bottle. This allowed for validating the load sensor measurements against a known reference. Here is an example dataset that demonstrates the comparison between the load sensor measurements and the known water level:

Table 4.8: Load Sensor Measurements vs. Known Water Levels

Trial	Load Sensor Reading (L)	Known Water Level (L)
1	1.8	1.85
2	1.7	1.75
3	1.6	1.65
4	1.5	1.55
5	1.4	1.45

This dataset compares the load sensor readings with the known water levels for each trial. By analysing the differences between the load sensor readings and the known water levels, we can evaluate the accuracy and precision of the load sensor in detecting the water level. To further validate the results, statistical analysis techniques such as calculating the mean difference standard deviation and performing hypothesis testing can be applied. These analyses provide insights into the consistency and reliability of the load sensor measurements and their agreement with the known water levels.

#### 4.2.5.2 Assumptions and Limitations

The load sensors provide accurate and precise readings of the weight applied to them. The relationship between the load sensor readings and the water levels is linear and consistent. The water bottle is placed on the load sensors stably and consistently for each measurement.

The accuracy of the load sensor readings may be influenced by external factors such as temperature changes or vibrations. The calibration curve or equation derived may have limitations in accurately predicting water levels outside the calibrated range. The experiment is conducted under controlled conditions, and the results may vary in real-world scenarios with different environmental conditions. The experiment focuses on a specific bottle size (2 litres) and may not directly apply to other bottle sizes or shapes. The measurement precision may be limited by the resolution of the load sensors or other equipment used in the experiment.

### 4.3 Proposed Technique to Derive High-Level Activities

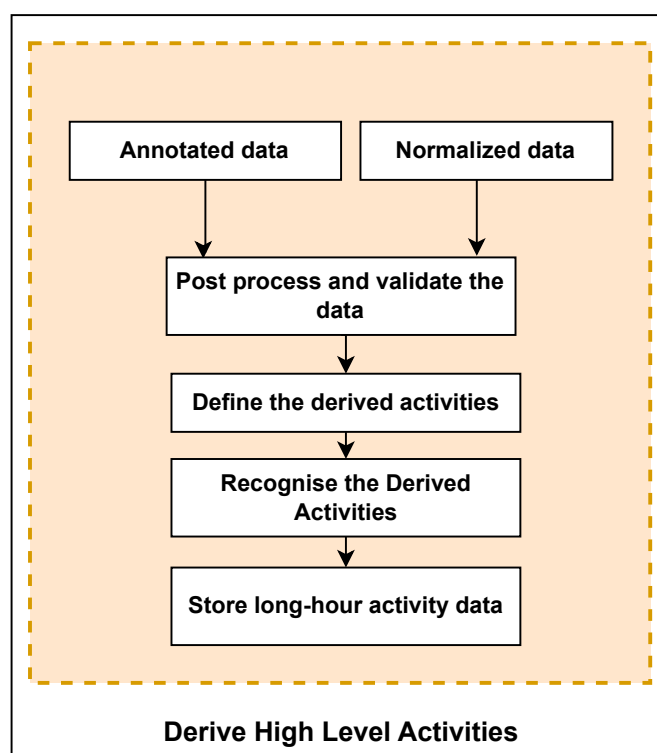


Figure 4.14: Proposed Technique to Derive High-Level Activities  
label fig:Proposed Technique to Derive High-Level Activities

In this work, we will combine data from multiple sources, including smart-phone and ambient sensors (PIR), load sensors, and smart switches), to infer user activities. This proposed framework comprises three key components: (1) Basic Activities, (2) Itemsets, and (3) Activity Traces derived from multimodal sensor data.

The first step in the framework involves defining Basic Activities, which encompass a set of predefined low-level actions and sensor activations extracted from smartphone sensors. This forms a standardised foundation for activity recognition, capturing activities like "Standing," "Sitting," "Walking," "Jogging," "Upstairs," and "Downstairs" using data from the smartphone's accelerometer, gyroscope, and other sensors. Additionally, ambient sensor data, including PIR sensors for motion detection, load sensors for appliance usage, and smart switches for monitoring the status of appliances, contribute to understanding user context.

Next, the concept of Itemsets combines the information gathered from smartphones and ambient sensors with contextual data. The Itemsets comprehensively represent higher-level Derived Activities, incorporating room locations, specific appliance usage, and user inputs. This integration allows for the recognition of activities such as "Lying in Bedroom," "Sitting in Kitchen," "Watching TV Lying," and "Sleeping in Worship room," providing a more meaningful understanding of user actions within their living environment.

To realise the complete recognition process, each activity instance is represented as an Activity Trace, consisting of the Derived Activity, the corresponding Itemset, and an Anomaly/Non-Anomaly label. The Anomaly and Non-Anomaly Classification is performed based on the combination of Basic Activities and Derived Activities, enabling the system to discern normal patterns from unusual or anomalous behaviour. This capability is of utmost importance in context-aware applications, where detecting anomalies can trigger appropriate responses or alerts to ensure the user's safety and well-being. The Notations are summarised in the Table 4.9.

*Definition 1 (Basic Activities):* Let  $AS$  be a set of available basic activities. Then,  $AS$  is defined as the union of  $A$  and  $S$  with the exclusive-OR operation, represented by  $\oplus$ , to account for the availability of data from either basic activities or ambient sensor activations:

$$AS = \{a \mid a \in A\} \oplus \{s \mid s \in S\}$$

Where: -  $A$  is the set of basic activities defined as  $A = \{a_1, a_2, \dots, a_g\}$ , with  $g$  representing the number of basic activities. -  $S$  is the set of sensor activations defined as  $S = \{s_1, s_2, \dots, s_h\}$ , with  $h$  representing the number of sensors available for activations.

The operator  $\oplus$  refers to an exclusive-OR (XOR) operation, indicating that the activation will return either the set of basic activities or sensor activations based

Table 4.9: Notation Used to Derive the High-Level Activities

Notation	Description
$AS$	Set of available basic activities and sensor activations
$A$	Set of basic activities
$S$	Set of sensor activations
$\oplus$	Exclusive-OR (XOR) operation
$ik$	Itemset assigned to the $k$ -th activity trace
$AS_k$	$k$ -th activity set representing the combination of basic activities and ambient sensor activations
$C_k$	$k$ -th context set representing room locations, sensor activations, and user-provided manual inputs
$DA$	Set of derived activities (high-level activities)
$ha_k$	Derived activity (high-level activity) assigned to the $k$ -th activity trace
$Tr$	Set of activity traces
$An_k$	Label of the derived activity indicating whether it is an anomaly or non-anomaly
$An_k(tr_k)$	Anomaly classification function for the $k$ -th activity trace
$\text{Supp}(I^* \Rightarrow har)$	Support of association rule between antecedent $I^*$ and high-level activity $har$
$\text{Conf}(I^* \Rightarrow har)$	Confidence of association rule between antecedent $I^*$ and high-level activity $har$

on data availability. This definition allows us to represent the available activations for the activity recognition process, considering the data from both basic activities and sensor activations.

*Definition 2 (Itemset):* Let  $I$  be a set of itemsets. Then, an itemset  $ik \in I$  is defined as:

$$ik = \{AS_k, C_k\}$$

where  $AS_k \subseteq AS$  represents the  $k$ -th activity set, and  $C_k$  represents the  $k$ -th context set. The activity set  $AS$  is a combination of various basic activities, such as "Standing," "Sitting," "Walking," "Jogging," "Upstairs," and "Downstairs." In contrast, the context set  $C$  includes ambient conditions, such as room locations and sensor activations (e.g., TV Switch ON/OFF, Mirror Bulb ON/OFF), and user-provided manual inputs.

For example, the set  $AS$  may comprise several basic activities, but a specific itemset  $AS_k$  may only include "Sitting" and "Standing" among the available basic activities. Similarly, the context set  $C$  may include different locations

(e.g., Kitchen, Bedroom, Hall) and time-coded values (e.g., Morning, Afternoon, Evening, Night). In our study, we represent time codes as Mo: 05.00 a.m. - 12.00 p.m., Af: 12.01 p.m. - 04.30 p.m., Ev: 04.31 p.m. - 07.30 p.m., and Ni: 07.31 p.m. - 12.00 a.m. In cases of overlapping time codes or different locations for the same activity, we choose the one with the maximum number of instances.

The high-level activities are now called Derived Activities (DA) and are represented as  $DA = \{har|r = 1, \dots, R\}$ , where  $R$  is the number of DAs. Each activity trace is represented as  $tr_k = \{da_k, ik\}$ , where  $da_k$  is the derived activity assigned to the  $k$ -th trace and  $ik$  is the itemset assigned to the same trace. For instance, an activity trace  $tr_1$  may represent the derived activity "Socialising" along with the itemset  $ik = ["Walking," "Standing," "Kitchen," "Mirror ON"]$ , indicating that the user was engaged in socialising while walking and standing in the kitchen with the mirror switched on.

In summary, Definition 2 defines itemsets that combine low-level actions and sensor activations with contextual information, facilitating the construction of a classifier based on Combinatorial Activity Rules (CAR) for each activity.

*Definition 3 (Activity Trace):* Let  $Tr$  be a set of activity traces. An activity trace  $tr_k \in Tr$  is defined as:

$$tr_k = \{da_k, ik\}$$

where  $da_k$  represents the derived activity assigned to the  $k$ -th trace, and  $ik$  is the itemset assigned to the same trace.

For example, consider an activity trace  $tr_1$  representing the derived activity "Personal Grooming" along with the itemset  $ik = ["Sitting", "Standing", "Home", "Morning"]$ , indicating that the user was engaged in personal grooming while sitting and standing at home in the morning.

To compute the support and confidence for a given antecedent  $I^*$ , we define:

$$\text{Support}(I^* \Rightarrow har) = \frac{\sum_{tr_k \in Tr} f(I^*, har, tr_k)}{|Tr|}$$

$$\text{Confidence}(I^* \Rightarrow har) = \frac{\sum_{tr_k \in Tr} f(I^*, har, tr_k)}{\sum_{tr_k \in Tr} f(I^*, tr_k)}$$

where  $f(I^*, har, tr_k)$  is the number of activity traces that contain both  $I^*$  and the derived activity  $har$ .

For example, let's consider the antecedent  $I^* = ["Sitting", "TV", "Hall"]$  and the derived activity  $har = "Watching TV sitting in Hall"$ . From the activity

traces, we find that there is only one instance of the itemset, ["Sitting", "TV", "Hall"]  $\Rightarrow$  "Watching TV sitting in Hall".

Therefore, we can compute the support and confidence for this association rule, which would be  $\text{Support}(I^* \Rightarrow \text{"Watching TV sitting in Hall"}) = \frac{1}{4}$  and  $\text{Confidence}(I^* \Rightarrow \text{"Watching TV sitting in Hall"}) = \frac{1}{2}$ .

The activity traces may include unique itemsets for different derived activities, but real-life scenarios can be more complicated, leading to activity handling challenges. People may perform activities differently, resulting in similar sensor activations for different derived activities. To address this, we can incorporate the frequencies of actions or sensor activations into the confidence calculation for association rules. By considering the frequency of actions or sensor activations triggered during a certain activity, we can find temporary patterns that might improve associative learning performance.

*Definition 4 (Anomaly and Non-Anomaly Classification):* Let  $Tr$  be a set of activity traces with their derived activities and labels. For each activity trace  $tr_k \in Tr$ , the derived activity  $da_k$  is assigned based on the combination of basic activities and ambient sensor data, and the label  $An_k$  indicates whether the derived activity is an anomaly or non-anomaly.

Each activity trace  $tr_k$  is represented as  $tr_k = \{da_k, ik, An_k\}$ , where: -  $da_k$  represents the derived activity assigned to the  $k$ -th trace, derived from the combination of basic activities and ambient sensor activations. -  $ik$  is the itemset assigned to the same trace, representing the combination of low-level actions and contextual information. -  $An_k$  represents the label of the derived activity, indicating whether it is classified as an anomaly or non-anomaly based on user context.

The classification of activities as anomaly or non-anomaly can be determined using an algorithm or predefined rules with equations that consider the user's context and typical behaviour patterns. For example, let's define the anomaly classification equation as follows:

$$An_k = \text{Anomaly Classification}(tr_k)$$

where

$$An_k(tr_k) = \begin{cases} \text{Anomaly} & \text{if anomaly conditions are satisfied for } tr_k \\ \text{Non-Anomaly} & \text{otherwise} \end{cases}$$

The anomaly conditions in the equation above can be based on various fac-

tors, such as the frequency of certain activities, the locations where activities are performed, the temporal patterns, and user-provided manual inputs. Considering these factors and using appropriate thresholds, the activity recognition system can identify unusual or abnormal behaviour patterns and classify the derived activities accordingly.

For example, consider the activity trace  $tr_1$ , where the derived activity is "Lying in Bedroom" with the itemset  $ik = ["Lying", "Bedroom"]$ . The algorithm or predefined rules would evaluate the conditions and classify it as a non-anomaly by setting  $An_1$  to "Non-Anomaly." On the other hand, for the activity trace  $tr_4$ , where the derived activity is "Jogging in Bedroom" with the itemset  $ik = ["Jogging", "Bedroom"]$ , the algorithm would evaluate the conditions and classify it as an "Unnatural" anomaly by setting  $An_4$  to "Anomaly."

The classification process helps identify abnormal activities that may require further investigation or attention, such as potential health concerns or safety issues, while allowing the system to recognise and categorise typical behaviour patterns as non-anomalies for more accurate activity recognition and understanding of user behaviour.

Classifying activities as anomaly or non-anomaly is based on predefined rules and thresholds considering the user's context and typical behaviour patterns. These rules may consider factors such as the frequency of certain activities, the locations where activities are performed, the temporal patterns of activities, and user-provided manual inputs. Considering these factors, the activity recognition system can identify unusual or abnormal behaviour patterns and classify the derived activities accordingly.

There are two types of activities recorded: Basic Activities and Derived Activities. Basic Activities are Standing, Sitting, Walking, Jogging, Upstairs and Downstairs. Derived Activities are derived by combining the Basic Activities, Ambient sensors like TV Switch ON/OFF, Mirror Bulb ON/OFF and considering the user's manual input. All the Activities are listed in Table 4.10.

Table 4.10 presents the derived activities and their labels based on the combination of basic activities and ambient sensor data collected every 2 minutes. The "Basic Activity" column contains activities such as lying, sitting, standing, jogging, and sleeping. The "Ambient Sensor" column indicates the room or condition where the activity is detected, such as the bedroom, kitchen, hall, or outside. The "Derived Activity" column shows the resulting derived activity based on basic activity and ambient sensor data. Activities like "Lying in Bedroom," "Sitting in Kitchen," and "Watching TV Lying" are examples of derived activi-

Table 4.10: Derived activities using basic activity and ambient sensors.

Basic Activity	Ambient Sensor	Derived Activity	Anomaly/ Non-Anomaly
Lying	Bedroom	Lying in Bedroom	Non-Anomaly
Sitting	Bedroom	Sitting in Bedroom	Non-Anomaly
Standing	Bedroom	Standing in Bedroom	Non-Anomaly
Jogging	Bedroom	Jogging in Bedroom	Unnatural
Sleeping	Kitchen	Sleeping in Kitchen	Non-Anomaly
Lying	Kitchen	Lying in Kitchen	Non-Anomaly
Sitting	Kitchen	Sitting in Kitchen	Non-Anomaly
Standing	Kitchen	Standing in Kitchen	Non-Anomaly
Jogging	Kitchen	Jogging in Kitchen	Unnatural
Lying	Hall	Lying in Hall	Non-Anomaly
Sitting	Hall	Sitting in Hall	Non-Anomaly
Standing	Hall	Standing in Hall	Non-Anomaly
Jogging	Hall	Jogging in Hall	Non-Anomaly
Sleeping	Worship	Lying in Worship	Non-Anomaly
Sitting	Worship	Sitting in Worship	Non-Anomaly
Standing	Worship	Standing in Worship	Non-Anomaly
Jogging	Worship	Jogging in Worship	Non-Anomaly
Sleeping	Hall + TV	Sleeping TV ON	Non-Anomaly
Lying	Hall + TV	Watching TV Lying	Non-Anomaly
Sitting	Hall + TV	Watching TV Sitting	Non-Anomaly
Standing	Hall + TV	Watching TV Standing	Non-Anomaly
Jogging	Hall + TV	Watching TV Jogging	Unnatural
Stair Up	Stairs	Stair Up	Non-Anomaly
Stair Down	Stairs	Stair Down	Non-Anomaly
Walking	Outside	Outside Activity	Non-Anomaly
	Water Bottle Use	Drinking Activity	Non-Anomaly
	Mirror Switch ON	Grooming	Non-Anomaly
	Bathroom Switch ON	Bathroom Activity	Non-Anomaly
Lying/Sleeping	Kitchen	Sleeping in Kitchen	Anomaly
Lying/Sleeping	Bathroom	Sleeping in Bathroom	Anomaly
Lying/Sleeping	Worship	Sleeping in Worship room	Anomaly
Lying/Sleeping	Stairs	Sleeping on Stairs	Anomaly

ties. The "Anomaly/Non-Anomaly" column categorises the derived activities as either "Non-Anomaly" (normal activities) or "Anomaly" (unusual or abnormal

activities). The table provides valuable insights into the combinations of activities and sensor data, aiding in understanding user behaviour and room occupancy patterns.

Table 4.11: Description of Activity Codes

<b>Activity Code</b>	<b>Description</b>
SLP_BED	Sleeping in the bedroom
LYE_BED	Lying down in the bedroom
BTH_ACT	Bathroom activity
GROOM	Grooming activities
WAT_TV	Watching TV in the living room
SIT_HAL	Sitting in the living room
WAK_OUT	Waking up and moving outdoors
OUT_ACT	Engaging in outdoor activities
WLK_KIC	Walking in the kitchen
BRK_FST	Having breakfast in the kitchen
SIT_KIC	Sitting in the kitchen
SIT_BED	Sitting in the bedroom
LYE_HAL	Lying down in the living room
LUNCH	Having lunch in the kitchen
LYE_TV	Lying down and watching TV
STD_HAL	Standing in the living room
KIC_ACT	Engaging in activities in the kitchen
WLK_HAL	Walking in the living room
DINNER	Having dinner in the kitchen
DRN_WTR	Drinking water in the kitchen
WORSHIP	Engaging in worship activities

We define the codes for each from the derived activities to help analyse them quickly in the implementation. Table 4.11 presented a description of the activity codes, outlining the specific activities associated with each code.

## 4.4 Proposed Priority Labelling

The activities recognised from the mobile app are stored in the file and the ambient sensor data. We took every 2 minutes of data and labelled the activity by following the algorithm 4.3. The significance of accomplishing the two-minute profiling is reducing the number of activities for a long-hour dataset for human activity pattern recognition and not compromising the crucial activities considering the entire day of activity data. The window size of the profiling can be varied to Five-Minute profiling or Ten-Minute profiling, depending on the application that is developed using human patterns. Table 4.15 shows the sample instance of ten-minute labelling. The labels are recorded every 5 seconds in the smartphone-based basic activity recognition. There is a total of 24 entries of labels in the 2 minutes of data. Now, we represent one label every 2 minutes using the priority-based algorithm. Segmenting the data to every 2 minutes reduces the data size as we are analysing the long-hour data for a week. Considering every 5 seconds of data will increase the computation cost. Moreover, when analysing human activities, we are not considering the small actions but the larger context of human activities. Also, the segmentation is complex but considers the priority of the activity during segmentation. The segmented activity labelling retains the user-centric context of the human activity.

In this section, we are explaining the priority-based labelling for ten-minute profiling for simplicity. The 2 minutes of data is passed to the algorithm to get one label every 10 minutes. The priority list will consist of the list of activities in priority. The significance of the priority list is to label the activity of importance, not just by the frequency. For example, consider a scenario where walking activity has the highest frequency in a two-minute window. However, the drinking activity has a lesser frequency in that two-minute window. We must pay attention to drinking activity just because the frequency is less. So, the drinking activity's priority will be more than the walking activity in the two-minute profiling. The Algorithm 4.3 checks the activities from the priority list and the frequency of all the activities. The priority of the activity is higher than that activity is labelled, or the activity with the higher frequency is labelled. The simple case of a priority setting for the activities is shown in Table 4.14.

The priority is set while labelling the activity for every interval. For example, in 5-minute profiling, taking the maximum occurred activity as a label is not a good criterion. This is because some activities rarely happen in 5 minutes. For example, the Drinking water activity is just for a maximum of 30 seconds, but this

is a crucial activity to be considered while creating the Five-minute labelling for a day. So, the priority needs to be set for the drinking activity while labelling the 5-minute, and the rarely-occurring fall activity must be addressed. The priority is set according to the following equation. All the notations used in this section are summarised in Table 4.12.

$$\text{let } s = \{A_1, A_2, A_3, \dots, A_k\} \text{ where } A_i \text{ are the activity list}$$
$$\text{Priority of } A_i \geq A_j \text{ where } \text{Freq}(A_i) \leq \text{Freq}(A_j)$$

The priorities for the lesser frequency equations are set at a higher frequency. However, depending on the requirement, the user-centric priority list can be manually set.

We tried sampling the activities by taking the maximum frequency and the average of the activities in 10-minute time intervals. However, the problem with this type of sampling is that it removed some important activities while sampling. For example, drinking or using the bathroom occurs less frequently and with less duration than other activities like walking outside and watching TV. Nevertheless, drinking activities must be addressed in the sampling as it adds more value in calculating the user's activity pattern. That is the reason we have proposed priority-based labelling. Here, the priorities are set for the activities as shown in the proposed Algorithm 4.3.

The *CalculatePriorities* function is added to calculate the activity priorities based on durations. This function takes *ActivityDuration* as input, a dictionary containing each activity's duration in minutes. The *minDuration* and *maxDuration* variables are used to determine the range of priorities for activities, and the priority scale is used to scale the priorities within that range. Finally, the *priorities* dictionary is created, where each activity is assigned a priority based on its duration and the scaling factor.

After determining the label based on frequency and priority, the *getUserContext* function is called to retrieve the user's context, which can be used to adjust the activity priorities. The individual user may have prioritised differently from the generated. These priorities can be adjusted with the *adjustPriority*. The function adjusts the label's priority based on the user context. The *adjustPriority* function takes two inputs, the label and the *userContext* dictionary, which maps each activity to an adjusted priority.

The *PriorityList* input lets the user list priority activities, overriding the calculated priority. If the *PriorityList* is empty or does not include any of the activities in the *FrequencyList*, the label is determined based on the calculated priority. The

Table 4.12: Notation Table

<b>Notation</b>	<b>Description</b>
<i>PriorityList</i>	A user-centric list of activities in priority order, used in the Priority-Based Labelling Strategy.
$A_i, A_j$	Activity categories in the priority list.
$Freq(A_i), Freq(A_j)$	The occurrence frequencies of activities $A_i$ and $A_j$ within the fixed interval.
<i>ActivityDuration</i>	A set of durations for each activity category in the fixed interval.
<i>priorities</i>	A dictionary containing the priority values for each activity category based on their duration in the fixed interval.
<i>getFrequency()</i>	A function that calculates the occurrence frequencies of activities in the fixed interval from timestamped labels.
<i>calculatePriorities()</i>	A function that calculates the priority values for each activity category based on their duration in the fixed interval.
<i>adjustPriority()</i>	A function that adjusts the derived activity label based on user context, if available.
<i>userContext</i>	User-specific information or context that can influence the derived activity label adjustment in the Priority-Based Labelling Strategy.
<i>label</i>	The derived activity label assigned to an activity trace in the Priority-Based Labelling Strategy.
<i>minDuration, maxDuration</i>	The minimum and maximum durations of activities in the fixed interval.
<i>priorityScale</i>	The scaling factor used to determine the priority values for each activity category in the fixed interval.

*getUserContext()* function retrieves any user context that may influence the label selection. If user context is available and the default label is in the user context, the label is adjusted based on the user context. We have also experimented with

---

**Algorithm 4.3** Labelling Strategy with Activity Priorities

---

**Require:** Data (timestamp, labels), PriorityList, ActivityDuration**Ensure:** label

```
1: FrequencyList[] = getFrequency(labels)
2: Priorities[] = calculatePriorities(ActivityDuration)
3: if set(PriorityList).isdisjoint(FrequencyList) then
4:   if count(frequency(label)) > 1 then
5:     label = last(FrequencyList)
6:   else
7:     label = first.FrequencyList
8:   end if
9: else
10:  label = first(PriorityList)
11: end if
12: userContext = getUserContext()
13: if userContext is not None then
14:  label = adjustPriority(label, userContext)
15: end if
16: return label
17: function CALCULATEPRIORITIES(ActivityDuration)
18:  minDuration = min(ActivityDuration)
19:  maxDuration = max(ActivityDuration)
20:  priorityScale = maxDuration/minDuration
21:  for activity in ActivityDuration do
22:    duration = ActivityDuration[activity]
23:    priority = priorityScale/duration
24:    priorities[activity] = priority
25:  end for
26:  return priorities
27: end function
28: function ADJUSTPRIORITY(label, userContext)
29:  adjustedLabel = label
30:  if label in userContext.keys() then
31:    adjustedLabel = userContext[label]
32:  end if
33:  return adjustedLabel
34: end function
```

---

the 2-minute and 5- 5-minute labelling based on priority. The smaller the granularity gave the accurate activities of a human but increased the computation time while drawing the inferences about the patterns. For example, the activity is labelled every 2 minutes in the considered dataset. We have a sequence of activities recorded at specific timestamps. The sequence starts with the activity "Drinking" at 10:00 AM, followed by multiple instances of "SIT\_HAL" at 10:02

AM, 10:04 AM, 10:06 AM, and 10:08 AM. The algorithm considers a priority list with "DRINK" as the first activity and "SIT\_HAL" as the second activity. Additionally, the duration of "DRINK" is shorter (2 minutes) compared to "SIT\_HAL" (8 minutes). The user can modify this priority depending on his context.

Applying the priority-based labelling strategy, the algorithm assigns higher priority to "DRINK" despite the higher frequency of "SIT\_HAL" in the data. This decision is based on the activity priorities defined in the priority list. As a result, the algorithm selects "DRINK" as the label for this sequence. This showcases the effectiveness of the priority-based labelling approach in considering activity priorities and making informed labelling decisions. By prioritising activities based on importance, the algorithm ensures that the most significant activities are identified and labelled accordingly, even if they occur less frequently in the data.

#### 4.4.1 Effect of Priority Labelling

The experiment aimed to compare the effectiveness of two labelling strategies: Max Frequency Labeling and Priority Labeling. Table 4.13 has a dataset consisting of 3000 instances of 2-minute data, and the objective was to assign labels to every 10-minute window using these strategies. The 3000 instances are selected from the data with various intervals over a day. The Max Frequency Labeling strategy was initially applied, which involved selecting the label with the highest frequency within each window. Subsequently, the Priority Labeling strategy was implemented, where labels were assigned based on predetermined priorities that reflected the importance of each activity from a user context.

After labelling the data using both strategies, it is manually validated to evaluate the correctness of the assigned labels. During the manual validation, the assigned labels were compared with ground truth or verified with the assistance of a user or expert. The results revealed that the Max Frequency Labeling strategy achieved a correct label assignment rate of 60%, while the Priority Labeling strategy had a higher rate of 92%. This indicated that Priority Labeling was more effective in assigning accurate labels and reducing incorrect assignments.

The experiment demonstrated the advantages of Priority Labeling in achieving higher accuracy in the assigned labels. By prioritising activities based on their importance or urgency, the Priority Labeling strategy ensured that the most relevant labels were assigned to the corresponding time windows. The crosscheck process further validated the effectiveness of Priority Labeling, as it resulted in significantly fewer incorrect label assignments than the Max Frequency Labeling

strategy.

Overall, the experiment highlighted the significance of considering priorities when assigning labels and the impact it can have on the accuracy of the labelling process. The findings provided valuable insights into the effectiveness of different labelling strategies and can inform future endeavours in activity labelling or similar applications.

Table 4.13: Experiment Results: Labelling Comparison

Scenario	Correct Label Assignments	Incorrect Label Assignments
Max Frequency Labeling	1800 instances	1200 instances
Priority Labeling	2760 instances	240 instances

The provided Table 4.14 showcases a simple case of priority setting for different activities. Each activity is assigned a priority value, with the smallest number indicating the highest priority. In this scenario, "Drinking Activity" holds the highest priority, with one value indicating its significance or urgency. The activities "Sitting in Hall" and "Kitchen Activity" share the second-highest priority, denoted by a priority value of 2. Finally, "Walking Outside" is assigned a priority of 3, indicating a lower priority than the other activities. Table 4.14 is a reference for understanding different activities' relative importance or prioritisation. It can be utilised in various contexts, such as task management, scheduling, or resource allocation, to ensure that activities with higher priorities receive appropriate attention or resources. It is worth noting that the priority values and activity names in the table can be adjusted based on specific requirements or domain-specific considerations.

Table 4.14: Simple case of priority setting for the activities. The smallest number indicates the highest priority.

Activity	Priority
Drinking Activity	1
Sitting in Hall	2
Kitchen Activity	2
Walking Outside	3

In the given example, the algorithm is applied to a specific scenario where three different strategies are used to determine the label for a 10-minute window. The

Table 4.15: Sample instance of labelling for a 10-minute window using 2-minute segmented data using the priority.

Cases, Operation	[Priority] Input	10-minute labelling
no priority, Max()	[3] Walking outside [3] Walking outside [3] Walking outside [3] Walking outside [3] Walking outside	Walking Outside
priority, HighestPriority()	[2] Sitting in Hall [2] Sitting in Hall [2] Sitting in Hall [2] Kitchen Activity [1] Drinking water	Drinking Water
equal frequency, equal priority, ChoseOccurredFirst()	[2] Sitting in Hall [2] Sitting in Hall [2] Kitchen Activity [2] Kitchen Activity [3] Walking outside	Sitting in Hall

results highlight different algorithms' impact on the label assignments within that time frame. Consider the Table 4.15 in the "No priority, Max()" scenario, where no priority is assigned; the algorithm selects the label with the highest frequency within the 10-minute window. As a result, all instances in the example are labelled as "Walking outside" since it is the most frequently occurring label.

In the "Priority, HighestPriority()" scenario, labels are assigned priorities, and the HighestPriority() algorithm selects the label with the highest priority within the 10-minute window. The label assignments in this scenario are based on the assigned priorities. In the example, "Sitting in Hall" is the highest priority, which is the label for the first four instances. The fifth instance has "Drinking water" with the highest priority, leading to its assignment as the label.

In the "Equal frequency, equal priority, ChoseOccurredFirst()" scenario, labels have equal frequency and priority. The ChoseOccurredFirst() algorithm selects the label that occurred first within the 10-minute window. As a result, the label

assignments in this scenario are determined by the order of occurrence. In the example, the first instance is labelled as "Kitchen Activity" since it occurred first. The following three instances are labelled "Sitting in Hall" as they occurred before "Kitchen Activity" within the window. The last instance is labelled "Walking outside" since it happened before the other options.

These results demonstrate how different algorithms and criteria can lead to varying label assignments within the 10-minute window. The specific algorithm chosen depends on the priority assignment rules and the desired behaviour for label selection in the given context. It is important to note that these results are specific to the example provided and may differ based on the actual data and the implementation of the algorithms.

## 4.5 Comparison with Existing Datasets

The table in Table 4.16 provides a comprehensive overview and comparison of various studies in activity recognition and ambient assisted living. Each study is evaluated based on multiple criteria, including Personalization (Per), Auto Labelling (AL), Ambient or Context Information (ACI), Low Level Activities (LLA), High Level Activities (HLA), and the employed model. The comparison highlights key aspects of each study's methodology and approach. Noteworthy among these is the proposed approach, which stands out with its implementation of Personalization and Auto Labelling, enhancing the accuracy and automation of the recognition process. Additionally, the incorporation of both Location and Appliances information further enhances the sophistication of the proposed model. Moreover, LLA and a notable number of 21 HLA illustrate the robustness of the proposed method. This model encompasses a combination of CNN, GRU, Activity Graph (AG), and CAR techniques, contributing to its superior performance compared to the other studies. This comparative analysis aids in understanding the strengths and uniqueness of each approach, with the proposed model demonstrating significant advancements in context-aware activity recognition and assisted living systems.

The proposed dataset stands out in several aspects compared to other datasets. It covers a longer duration of 3 months, providing a more extensive and comprehensive understanding of daily activities over an extended period. Captures a wide range of daily activities, with 21 activities detected, ensuring a comprehensive representation of human behaviour in the smart home environment. The proposed dataset involves 12 subjects, providing a diverse and varied sample popu-

Table 4.16: Summary and Comparison of various Studies.

Study	Per	AL	ACI	LLA	HLA	Model
Wen <i>et al.</i> (2015)	No	No	No	No	8	CAR
Rashidi <i>et al.</i> (2010)	No	No	Time	No	8	Clustering ARM
Lüühr <i>et al.</i> (2007)	No	No	Time	No	13	ARM
Riboni and Bettini (2011)	No	No	Location	4	6	MLR, SI
Hong <i>et al.</i> (2015)	Yes	No	No	7	No	HBN, SVM
Filippoupolitis <i>et al.</i> (2017b)	No	No	Location	No	8	KNN, LR, RF, SVM
Gong <i>et al.</i> (2018)	No	No	Location	8	No	XGBoost
Khowaja <i>et al.</i> (2018)	No	No	Location	6	10	HCA, ELM
Zhang <i>et al.</i> (2019)	No	No	Time	No	18	CNN
Aminikhanghahi and Cook (2019)	No	No	Time	No	10	RF
Khowaja <i>et al.</i> (2020)	No	No	Intervals	7	8	CAR, AL, RNN
Proposed	Yes	Yes	Location, Appliances	6	21	CNN, GRU, AG, ARM

Abbreviations: PER - Personalization, AL-Auto Labelling, ACI - Ambient or Context Information, LLA - Low-Level Activities, HLA - High-Level Activity, CAR - Contextual Activity Recognition, ARM - Ambient Assisted Living, HBN - Hybrid Bayesian Network, SVM - Support Vector Machine, KNN - K-Nearest Neighbors, LR - Logistic Regression, RF - Random Forest, XGBoost - Extreme Gradient Boosting, HCA - Hierarchical Clustering Analysis, ELM - Extreme Learning Machine, CNN - Convolutional Neural Network, RNN - Recurrent Neural Network, GRU - Gated Recurrent Unit, AG - Activity Graph.

lation, contributing to the generalizability of the activity recognition model. Also, the dataset utilises smartphone sensors (Accelerometer, Gyroscope) along with ambient sensors, enhancing the richness of contextual information for activity recognition. The dataset employs advanced LSTM and GRU models for activity recognition, leveraging the power of deep learning techniques for accurate predictions. The proposed dataset achieves an accuracy of 87.3%, demonstrating the effectiveness of the proposed activity recognition approach.

The combination of a diverse dataset, extended duration, and advanced activity recognition models makes the proposed dataset a promising choice for human activity recognition research, enabling a deeper understanding of user behaviour, preferences, and lifestyle patterns in a smart home environment.

## 4.6 Constraints and Limitations

This section discusses the constraints and limitations of the long-hour dataset collection using multimodal sensors. The assumptions and limitations are important to assess our methodology, and we have tried to achieve the broader objective.

Table 4.17: Comparison of Different Datasets and Activity Recognition Models

Dataset	Duration	Activities	Subjects	Sensors	Ambient	ARM	Accuracy (%)	Year
<b>Proposed Dataset</b>	3 months	Daily activities (21)	12	Smartphone (Accelerometer, Gyroscope)	Yes	LSTM, GRU	87.3	N/A
CASAS	15 days	Daily activities	10	Multimodal (Accelerometer, PIR sensors, etc.)	Yes	HMMs, CRFs	91.2	2018
ExtraSensory	Variable (2 - 25 days)	Various activities	60	Smartphone (Accelerometer, Gyroscope), Wearable sensors	Yes	DNN	83.6	2015
SisFall	Within a day	19 activities	22	Smartphone accelerometer	No	Naive Bayes	95.8	2014
UCI-HAR	6 hours	6 activities	30	Accelerometer, Gyroscope	No	Deep Neural Network	95.6	2012
WISDM	2 weeks	5 activities	29	Accelerometer	No	Decision Tree	89.4	2010
OPPORTUNITY	8 hours	19 activities	4	Accelerometer, Gyroscope, etc.	No	Conditional Random Fields	90.1 F1-score	2012
PAMAP2	5 hours	12 activities	9	Accelerometer, Gyroscope, etc.	No	Random Forest	82.4	2012
MobiFall	Variable	17 activities	26	Accelerometer	No	Support Vector Machine	85.6 F1-score	2013

Considering the motivation and objectives we have defined in Section 1.8 and 1.9 following constraints and limitations are mentioned.

### 4.6.1 Sensors and the Activities

Our research uses smartphone-based sensors such as accelerometers and gyroscopes to detect human activities. While this limits the range of activities we can recognise, our broader goal is to develop a comprehensive framework for human behaviour pattern recognition using this data. We have focused on a minimum set of activities as a starting point, but there is potential to expand and improve upon them in future iterations.

It is reasonable to state that variations in smartphone sensor configurations can impact the system's accuracy (Kuhlmann *et al.* 2021). However, the potential effects of sensor variations are minimised by using the same sampling rate of 50 Hz and a similar preprocessing technique. Even in our research, we experimented with multiple devices from different companies, and the accuracies remained consistent. We also implemented post-processing scripts to address missing values, nan, and calibration errors and performed manual verification and validation to ensure data integrity.

The proposed system has been designed specifically to recognise and annotate basic daily human activities in an indoor smart home environment. The selected activities include walking, sleeping, stairs up, stairs down, and sitting. While these activities are fundamental to understanding an individual's daily routines, it is important to acknowledge that the system may not recognise or capture more diverse or specific activities beyond this predefined set. Our primary target audience for this system is elderly people within the context of smart homes and their activities of daily living. As such, recognising basic activities is particularly relevant for improving their quality of life and promoting independent living. To ensure the accuracy and reliability of the system, we have focused on developing robust algorithms for detecting the selected basic activities. However, recognising more complex activities such as cycling, specific kitchen activities, or mountain climbing would require specialised sensors or tailored algorithms to achieve accurate classification. It is important to highlight that the limitations in recognising diverse or specific activities maintain the value of the proposed system. On the contrary, the system's focus on basic activities provides a solid foundation for further research and development. By combining data from multiple sensors and understanding the relationships between basic activities, we can explore the possibility of deriving

more complex activities in the future.

Indeed, variations in smartphone sensor configurations can have an impact on the system’s accuracy. In our study, we acknowledge the potential challenges posed by different smartphone models having varying sensor characteristics. We have taken several steps in the preprocessing stage to mitigate the impact of sensor variations. Firstly, the data from smartphone sensors are processed using consistent preprocessing techniques applied to the WISDM dataset. These techniques include filtering and sampling rate standardisation. Precisely, accelerometer sensors measure acceleration forces, and gyroscope sensors measure angular velocity. The system records movement in the x, y, and z directions by combining data from these sensors. To ensure consistent data representation, noise filters are applied, and a fixed sampling rate of 2.56 seconds is used for all devices. Secondly, we utilise Butterworth low-pass filters to remove noise from the sensor data. This filtering process aids in improving the signal-to-noise ratio and helps in capturing relevant activity patterns accurately. These are the techniques used in the base paper of the existing WISDM dataset (Janarthanan *et al.* 2020). Our experiments used multiple devices from different manufacturers, totalling three different smartphones. Despite the variations in sensor configurations among these devices, we observed that the system’s accuracy remained consistent. However, we could have performed consistency and other analyses over multiple smartphones. To ensure data quality, we conducted a thorough data cleaning process, which involved handling missing values, NaN entries, and other calibration errors. Python post-processing scripts were utilised for data verification and validation.

Additionally, we manually reviewed the data to ensure its integrity. We aim to enhance the system’s robustness and generalizability across different smartphone models by addressing sensor variations and employing rigorous preprocessing techniques. However, future work could explore additional strategies, such as transfer learning or model adaptation, to improve the system’s performance on a broader range of smartphone sensors and configurations. Overall, we are confident that our approach has effectively minimised the impact of smartphone sensor variations, ensuring reliable and accurate human activity recognition in diverse real-world scenarios.

This work assumes the presence of a single person in the room during activity detection. This motivation stems from our aim to develop applications for the remote monitoring of elderly individuals who live alone. By focusing on single-person occupancy, we can tailor our system to address the specific needs and challenges associated with ensuring their safety and well-being.

### 4.6.2 Privacy

We completely acknowledge the significance of privacy concerns and the potential implications of collecting sensor data, including location information. While our current implementation does not include these privacy safeguards, we are acutely aware of their importance. We have a well-defined action plan to address these concerns and uphold user privacy. It is important to highlight that our approach does not involve GPS location tracking but centres on room-level location data. Even though our current system does not incorporate specific privacy measures, our roadmap includes a comprehensive strategy to ensure user privacy remains a top priority. As part of our commitment to safeguarding privacy, we are actively working on implementing appropriate measures to protect user data. This may involve anonymising or encrypting the collected data, establishing secure storage and transmission protocols, and obtaining explicit consent from users. These measures are crucial to adhere to data protection regulations and ensure user privacy remains intact. We assure you that the privacy of our users is of paramount importance, and we are dedicated to implementing the necessary safeguards to address any privacy concerns that may arise due to the collection and processing of sensor data. Our approach underscores our commitment to responsible and ethical data handling as we continue to enhance our system's capabilities.

### 4.6.3 Adaptability

The system's sustainability requires regular updates, maintenance, and continuous improvements. Once the application is developed and deployed, it is imperative to recognise that the technology ecosystem is dynamic. Emerging challenges, shifts in smartphone models and capabilities, and advancements in deep learning and sensor technologies necessitate a proactive approach. Regular updates and maintenance will ensure the system remains compatible with new smartphone models and operating systems. This includes adapting the application to changes in hardware specifications, software interfaces, and security protocols. Furthermore, our commitment to incorporating these improvements will be critical as advancements in deep learning and sensor technologies unfold. This may involve refining the models, optimising algorithms, and harnessing new sensor capabilities to enhance the accuracy and performance of the activity recognition system. We aim to provide users with a relevant, reliable, and effective solution over time by approaching our system's life cycle with an eye toward ongoing refinement and enhancement. This perspective underscores our dedication to staying abreast of technological progress

and ensuring our system delivers value in a rapidly evolving environment.

## 4.7 Summary

This chapter presents a framework for collecting extensive, long-hour data using multimodal sensors. The subsequent sections will explore analysing human behavioural patterns based on this data. The framework addresses challenges in various modules, encompassing basic activities recognition using smartphones, higher-level activities recognition via the fusion of basic activities, and user context recognition utilising ambient sensors. The employed labelling strategy enables the derivation of user-centric activities, aiding in the comprehensive analysis of human behaviour patterns. This approach ultimately offers insights into human activities, contributing to a more profound comprehension of human behaviour.

## Chapter 5

# Human Behavioural Activity Pattern Recognition

With the rise of smart homes and IoT devices, collecting vast amounts of data on human activities within the smart home environment has become possible (Eagle and Pentland 2006). These activities range from simple tasks like cooking and cleaning to more complex activities like exercising and working from home. Analysing these activity patterns can provide insights into individuals' daily routines and behaviours, which can be helpful in various applications, such as healthcare, security, and energy management. In recent years, advances in data collection technologies, such as smartphones and wearable, have enabled the continuous monitoring and recording of ADL of elderly individuals (Kiukkonen *et al.* 2010). Understanding the daily activities and behavioural patterns of elderly individuals is crucial for providing better care and support to help them maintain their independence and quality of life. This knowledge can offer valuable insights into their physical, cognitive health, social and emotional well-being (Rawassizadeh *et al.* 2013). Researchers have explored various approaches to analyse and make sense of the vast data collected from these sources. For example, the Reality Mining approach has been used to sense complex social systems Eagle and Pentland (2006). Other studies have focused on leveraging mobile phone datasets for rich data collection and analysis Kiukkonen *et al.* (2010); Wagner *et al.* (2014). Habit mining approaches have also been utilised to discover similar mobile users and understand their behaviour Ma *et al.* (2012a). Additionally, research has been conducted on developing frameworks and tools for mobile data collection and analysis, such as Ubiqlog for mobile phone-based life-logging Rawassizadeh *et al.* (2013) and Device Analyzer for large-scale mobile data collection Wagner *et al.* (2014). Ensuring the

security and privacy of the collected data has also been a topic of interest Rawassizadeh and Tjoa (2010). Detecting human activity patterns has been a research topic in various domains, including healthcare, education, computer science, and engineering. Previous studies have used sensors and data collection methods to gather activity data from the sensors, such as accelerometer sensors, wearable device-based sensors, and cameras Falah Rad *et al.* (2022); Newman *et al.* (2023); Biswas *et al.* (2023); Alam *et al.* (2019). However, these studies often require individuals to wear or carry devices or may invade their privacy, which can be a barrier to adoption in real-world settings.

To address this challenge, various methods for identifying and summarising frequent behavioural patterns of elderly individuals, using techniques such as clustering, association rule mining, and sequence mining Vanahalli and Patil (2022); Potluri *et al.* (2019), can be proposed. These methods aim to group similar activities within a specific time interval, share similar characteristics, and identify frequent and consistent activities over time. These patterns can then be used to create profiles of elderly individuals, which can help care providers and family members better understand their needs, preferences, and challenges and provide more personalised and effective care and support. Adaptive granularity Hassan *et al.* (2020) refers to the ability to dynamically adjust the level of detail or resolution in capturing and analysing data based on specific needs or context. In the context of the Smart Home User Activity Profiler, adaptive granularity means that the system can adaptively adjust the level of temporal granularity (time intervals) used for analysing user activities based on the characteristics of the activities and the requirements of the application. For example, the system can use smaller time intervals (e.g., minutes) for analysing activities that require finer temporal resolution, such as detecting short-term movements or interactions, while using larger time intervals (e.g., hours) for activities that require coarser temporal resolution, such as identifying patterns in longer-term routines or behaviours (Rawassizadeh *et al.* 2016b).

The following contributions are significant because they address the challenges and requirements of creating personalised daily activity schedules for elderly individuals.

Deep learning techniques and multimodal sensor data allow for accurate activity recognition, crucial in generating personalised schedules. The promising results achieved by GRU models demonstrate the effectiveness of these approaches in accurately identifying and classifying activities. The introduction of the human-centric temporal granularity method acknowledges the dynamic nature of human

behaviour and provides a more flexible analysis approach. By considering time intervals instead of rigid timestamps, the framework can adapt to variations in activity patterns and better capture the nuances of daily routines. The novel priority-based labelling technique adds a layer of relevance and importance to activity classification. By assigning priority levels based on impact on well-being and daily functioning, the framework can prioritise activities that significantly influence an individual's overall health and quality of life. This enhances the accuracy and personalisation of the generated activity schedules.

Furthermore, the comprehensive user profiles created in this research go beyond activity classification and scheduling. The framework captures a holistic view of the individual by integrating sensor data with lifestyle information, health conditions, and personal preferences. This tailored approach ensures that activity schedules align with the individual's needs and promotes better health outcomes and engagement.

These contributions address the complex requirements of personalising activity schedules for elderly individuals by leveraging advanced techniques in sensor data analysis, temporal granularity, activity classification, and user profiling. This research aims to improve the well-being, independence, and quality of life of elderly individuals by providing tailored activity schedules that consider their unique characteristics and requirements.

The literature survey highlights the significance and need for data collection using multimodal sensors, particularly smartphones and ambient sensors, for activity recognition. Combining these sensors allows the model to derive a broader range of activities than individual sensors alone. The existing approaches in the literature labelling need to consider the user context and be more effective. The survey also emphasises the need and challenge for pattern recognition and the development of generalised models for activity analysis in various intervals.

## 5.1 Proposed Work on Human Behavioural Pattern Recognition using Adaptive Granularity

This section formally defines the problem of Frequent Behaviour Pattern (FBP). The concept of *Adaptive Granularity* (Rawassizadeh *et al.* 2016a) within frequent behaviour patterns involves dynamic adjustment of the level of detail or resolution for identifying and analysing recurring activities or behaviours. Unlike the

conventional approach of considering behaviour patterns at a static, predefined granularity, adaptive granularity empowers the system to make finer or coarser distinctions based on varying contexts, user objectives, or preferences.

The Adaptive Granularity concept lays the foundation for our methodology and algorithms in addressing the [FBP](#) challenge. The intricacies of integrating this adaptability into recognising and analysing behaviour patterns are described. The algorithms developed dynamically adapt the level of granularity based on real-time context, user preferences, and system requirements. Our approach accounts for activities' varying complexity and significance, enabling the system to focus on specific behaviours when necessary while still capturing the broader picture.

A comprehensive explanation of each algorithm's functioning is provided, highlighting how adaptive granularity plays a pivotal role in enhancing the accuracy, relevance, and efficiency of the [FBP](#) recognition process.

Subsequently, the results of the experiments and analyses were presented. Also, it showcases how the application of adaptive granularity influences the identification and understanding of frequent behaviour patterns. Quantitative metrics that demonstrate the system's adaptability and effectiveness in capturing behaviours at varying levels of detail are provided. Various findings of the proposed method are discussed and compared with existing approaches.

### 5.1.1 Methodology

This section formally defines the problem and is subdivided into definitions. The main focus of the problem is to provide a model that systematically identifies frequent behavioural patterns and user profiles.

*Problem:* With the timestamped activities of elderly individuals, the goal is to efficiently identify frequent behavioural patterns of the individuals and create profiles that summarise these patterns.

The profiles should include groups of similar activities that occur within a specified time interval and share identical data characteristics. The frequency of these groups should meet or exceed a minimum threshold. The similarity and dissimilarity between these profiles should also be measured to identify commonalities and differences in the activities of elderly individuals. The resulting profiles and insights can then be used to provide better care and support to elderly individuals.

**Definition 1:** Entity  $e$  is defined as a tuple of three elements (Table 1),  $e = \langle A, D, T \rangle$ , where  $A$  represents the activity code or category,  $D$  represents the data associated with the activity, and  $T$  represents the timestamp (time interval)

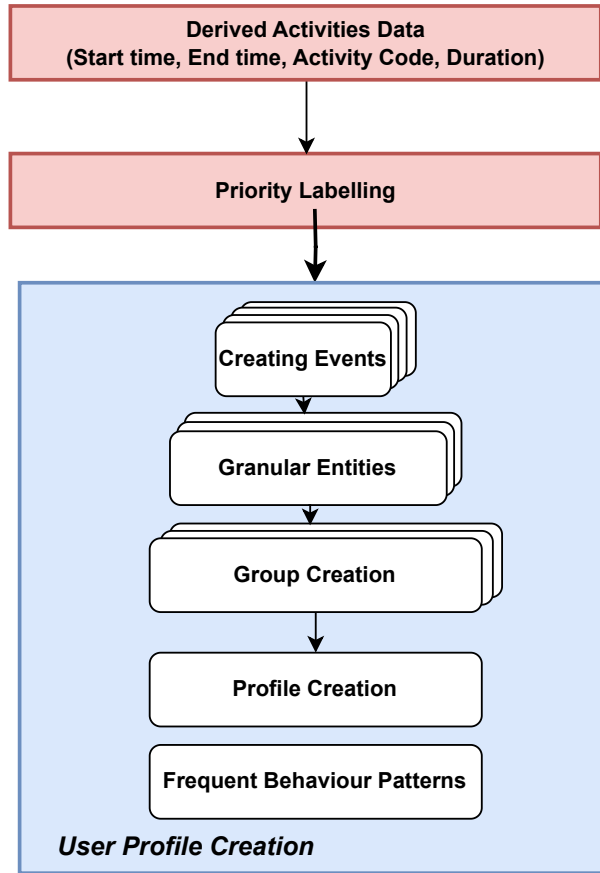


Figure 5.1: Proposed Work on Human Behavioural Pattern Recognition using Adaptive Granularity

of the activity. The entity  $e$  can be defined as follows:  $e = \langle A, D, T \rangle$ .

For example, two records might indicate that the individual had 8 hours of sleep at 11:00 PM on April 4, 2020, and went for a 30-minute walk at 10:00 AM on April 5, 2023. Definition 1 can represent these records as entities  $e_1$  and  $e_4$  with the following tuples:

$$e_1 = \langle \text{SLP\_BED}, 8 \text{ hours}, 2023-05-15 \ 22:00:00 \rangle$$

$$e_4 = \langle \text{WALK\_OUT}, 30 \text{ minutes}, 2023-05-15 \ 07:00:00 \rangle$$

Here,  $A = \text{SLP\_BED}$  in  $e_1$  and  $A = \text{WALK\_OUT}$  in  $e_2$  represent the activity codes of sleeping in the Bedroom and walking outside the house activities. Representing the data into such events helped us to analyse the activities as events.

**Definition 2:** A group of entities  $g$  is defined as entities with similar activity codes or categories, occurring within a specified time interval and sharing similar data characteristics. Group of entities  $g$  can be defined as follows:  $g = e_1, e_2, \dots, e_n$ , where  $e_i = \langle A_i, D_i, T_i \rangle$  and  $A_i = A_j$  for all  $1 \leq i \leq n$  and  $1 \leq j \leq n$ , and  $(T_{i+1} - T_i) \leq \delta$  for all  $1 \leq i < n$ , where  $\delta$  represents the maximum time interval

Table 5.1: The Notation and their description used in the Definitions

Notation	Description
$e$	Defined as a tuple of three elements, $e = \langle A, D, T \rangle$ .
$A$	Activity code or category.
$D$	Data associated with the activity.
$T$	Represent the timestamp (time interval) of the activity.
$g$	set of entities with similar activity codes or categories, occurring within a specified time interval and sharing similar data characteristics.
$\delta$	Maximum time interval between two consecutive entities in the group.
$\lambda$	number of times the $g$ is repeated in an interval.

between two consecutive entities in the group.

For example, consider the following events,

$e_1 = \langle \text{"WORSHIP"}, \text{"10 minutes"}, \text{"2023-05-15 06:00:00"} \rangle$

$e_2 = \langle \text{"BATHING"}, \text{"15 minutes"}, \text{"2023-05-15 06:15:00"} \rangle$

$e_3 = \langle \text{"BREAKFAST"}, \text{"10 minutes"}, \text{"2023-05-15 06:30:00"} \rangle$

In this example, a group of entities  $g$  that includes these three events as they occur within a specified time interval. Let's assume the maximum time interval between consecutive events in the group, denoted by  $\delta$ , is set to 30 minutes. Therefore, the group of entities can be defined as follows:

$$g = e_1, e_2, e_3$$

In this group, each entity represents a different event, namely "WORSHIP," "BATHING," and "BREAKFAST." The duration of the events is 10 minutes, 15 minutes, and 10 minutes, respectively. The time intervals between consecutive events within the group are less than or equal to the specified maximum interval of 30 minutes.

Here, the durations and specific timestamps used in this example are for illustrative purposes, and in practice, the actual durations and timestamps may vary based on the specific activities and individuals being considered.

**Definition 3:** A profile  $P$  is a set of repeated and similar groups  $g$  that have been identified more than or equal to  $\lambda$  times, where  $\lambda$  represents the minimum

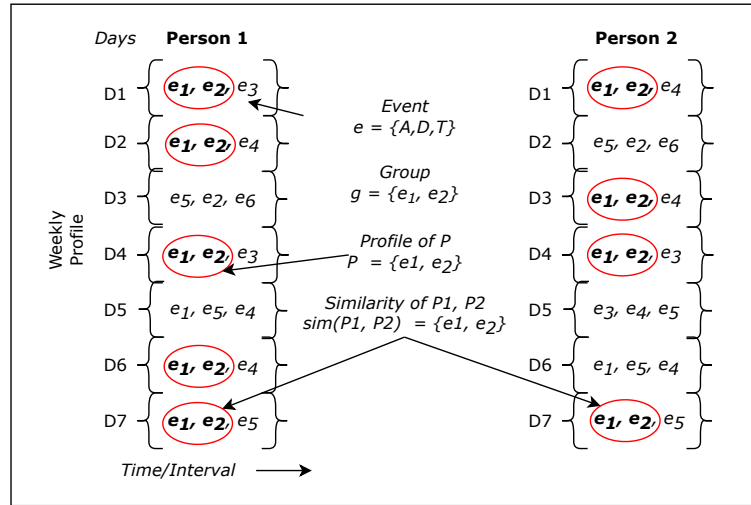


Figure 5.2: The Weekly Profile of a Person based on the Events and Groups. The Group (e1, e2) is Repeating on Five Days ( D1, D2, D4, D6, D7 ). The same Group is Common Between Person1 and Person2 Four Times a Week.

frequency threshold. Profile  $P$  can be defined as follows:  $P = g_1, g_2, \dots, g_k$ , where  $g_i = e_1, e_2, \dots, e_n$  and the frequency of  $g_i$  is greater than or equal to  $\lambda$ .

For example,  $g_1 = e_1, e_2, e_3$ ;  $g_2 = e_4, e_5$ . Here, two groups ( $g_1$  and  $g_2$ ) are identified based on similar activity codes and shared data characteristics occurring within a specified time interval.

$g_1$  represents a group of entities with activities "WORSHIP," "BATHING," and "BREAKFAST" occurring consecutively. The frequency of  $g_1$  is equal to or greater than the minimum frequency threshold.  $g_2$  represents a group of entities with activities "EXERCISE" and "LUNCH" occurring consecutively. The frequency of  $g_2$  is equal to or greater than the minimum frequency threshold. These groups capture repeated and similar patterns of activities, where the entities within each group share similar activity codes, occur within a specified time interval, and have similar data characteristics.

**Definition 4:** The similarity between two profiles  $P_1$  and  $P_2$  is defined as the Jaccard similarity coefficient between their corresponding group sets, where the Jaccard coefficient measures the degree of overlap between two sets. The similarity between two profiles  $P_1$  and  $P_2$  can be defined as follows:

$$\text{sim}(P_1, P_2) = \frac{|P_1 \cap P_2|}{|P_1 \cup P_2|}, \text{ where } |\cdot| \text{ represents the cardinality of a set.}$$

Definition 4 represents how the pattern of activities of the multiple individuals match. This will give an about the two personalities. Figure 5.2 showing the event (e), group (g), profile (P), and similarity (sim) of profile generation for D1 to D7 days (a week).

The methodology involves a multi-step data collection and analysis approach, as depicted in Figure 5.1. A comprehensive dataset is initially constructed by amalgamating data from diverse sensor sources, including smartphones and ambient sensors like PIR, switches, and pressure sensors. This combination facilitates the capture of a wide spectrum of activity-related information non-intrusively.

Deep learning models are then utilised to classify and recognise recorded activities. Integrating sensor-based data from smartphones and ambient sensors allows the extraction of additional valuable activities that might not be discernible from individual sensor data. This fusion of information enhances the overall comprehension of user activities.

A prioritised labelling technique is implemented to assign meaningful and informative labels to activities. This technique involves assigning priority levels to activities based on their significance to well-being and daily functioning. Activity classification accuracy and relevance are improved by considering the relative importance of activities, resulting in more personalised and meaningful activity profiles.

Lastly, user profiles are generated by creating events, granular entities, and groups based on activity data. These profiles encapsulate individual traits, preferences, and requirements, giving deeper insight into user behaviour patterns. The analysis and interpretation of these profiles yield valuable insights into users' daily routines, paving the way for personalised recommendations and interventions.

## 5.2 Implementation

The concepts of entities, granularity, groups, profiles, and similarities have been established to facilitate the analysis of human behavioural patterns. This section delves into the practical implementation of these definitions using specific algorithms. These algorithms comprehensively explain how the formally defined concepts are put into action. The data employed for this implementation is structured in CSV format and encompasses essential columns, including *Activity\_Code*, *Start*, *Finish*, and *Duration* of the activities. Moreover, additional columns such as the day of the week, activity categories, and other pertinent information have been derived to enhance the contextual richness of the dataset. This comprehensive dataset serves as the fundamental basis for executing the algorithms, which give tangible shape to the theoretical constructs delineated earlier.

### 5.2.1 Granular Entities

The main reason to generate granular entities based on time is to model how human behaviour is perceived. If this is not done, inferring from the timestamps will not be easy. The activities are defined in the events rather than just timestamps. These events are perceived by humans better than the timestamps.

---

#### Algorithm 5.1 Creating Granular Entities

---

**Require:** Activity dataset  $D$ , minimum activity duration  $min_{dur}$

**Ensure:** set of granular entities  $E$

```

1: Initialize an empty set  $E$ 
2: for all activity record  $r$  in  $D$  do
3:   Extract the activity code  $A$ , duration  $D$ , and timestamp  $T$  from  $r$ 
4:   if  $D < min_{dur}$  then
5:     Split the activity into granular entities of duration  $D$ 
6:     for all granular entity  $e$  in the split activity do
7:       Create a new entity tuple  $e = \langle A, D_e, T_e \rangle$ 
8:       Add  $e$  to the set  $E$ 
9:     end for
10:  else
11:    Create a new entity tuple  $e = \langle A, D, T \rangle$ 
12:    Add  $e$  to the set  $E$ 
13:  end if
14: end for
15: return  $E$ 

```

---

The Algorithm 5.1 for creating granular entities from an activity dataset. The algorithm takes as input the activity dataset  $D$ , which contains information about each activity, such as the activity code, duration, and timestamp. It also takes a minimum activity duration  $min_{dur}$  as input, which is used to split activities shorter than the specified minimum duration. The algorithm's output is a set of granular entities  $E$ , where each entity represents a single activity and includes information about the activity code, duration, and timestamp. The algorithm iterates over each activity record in the dataset  $D$ . For each activity record  $r$ , the algorithm extracts the activity code  $A$ , duration  $D$ , and timestamp  $T$ . If the duration of the activity is less than the specified minimum duration, the activity is split into granular entities of duration  $D$ . For each granular entity  $e$  in the split activity, a new entity tuple  $e = \langle A, D_e, T_e \rangle$  is created and added to the set  $E$ . If the duration of the activity is greater than or equal to the specified minimum duration, a new entity tuple  $e = \langle A, D, T \rangle$  is created and added to the set  $E$ . This helps generate more accurate and granular data, allowing us to capture more fine-grained details

about user activities. From the experiments and results shown, it is observed that by breaking down activities into smaller units can be more accurately model the transitions between activities.

## 5.2.2 Groups Creation

---

**Algorithm 5.2** Grouping Entities

---

**Require:** *entities*: a list of entities

**Ensure:** *groups*: a list of groups of entities with similar activity code, data, and timestamp

```
1: Sort entities in ascending order of timestamp
2: Initialize an empty list groups
3: Initialize an empty dictionary temp to store temporary groups
4: for e in entities do
5:   key = (e.A, e.D)
6:   if key is in temp and  $e.T - temp[key][-1].T \leq \delta$  then
7:     Add e to temp[key]
8:   else
9:     if key is in temp and  $e.T - temp[key][-1].T > \delta$  then
10:      Add temp[key] to groups
11:      Remove temp[key] from temp
12:     end if
13:     Add e as a new key to temp with value [e]
14:   end if
15: end for
16: for key in temp do
17:   Add temp[key] to groups
18: end for
19: return groups
```

---

Here, *entities* is a list of entities defined in Definition 1 (Section III). The Algorithm 5.2 first sorts the entities in ascending timestamp order to ensure they are processed in a temporal order. It then initialises an empty list *groups* to store the resulting groups of entities. It also initialises an empty dictionary *temp* to store temporary groups of entities. For each entity *e* in *entities*, the algorithm checks if a group in *temp* has the same activity code and data as *e* (i.e., if the key (*e.A*, *e.D*) is in *temp*). If such a group exists and the time difference between *e* and the last entity is less than or equal to the maximum time interval  $\delta$ , then *e* is added. Otherwise, if the time difference is more significant than  $\delta$ , the temporary group is added to *groups* and removed from *temp*, and a new group is created for *e*. If there is no group in *temp* with the same key as *e*, a new group is created for

*e.* After all entities have been processed, any remaining groups in *temp* are added to *groups*. The algorithm then returns the resulting list of groups.

---

**Algorithm 5.3** Creating Profile
 

---

```

1: Initialize an empty dictionary  $G$  to store the groups of entities
2: for each entity  $e_i$  in the dataset do
3:   Find a group  $g_j$  in  $G$  such that  $e_i$  can be added to  $g_j$  based on the time
   interval constraint and similarity of activity codes and data
4:   if such a group  $g_j$  exists then
5:     Add  $e_i$  to  $g_j$  and increment the frequency of  $g_j$  by 1
6:   else
7:     Create a new group  $g_k$  with  $e_i$  as its first element and frequency 1, and
     add  $g_k$  to  $G$ 
8:   end if
9: end for
10: Filter out the groups with frequency less than  $\lambda$  from  $G$  to obtain the set of
    candidate groups  $C$ 
11: Initialize an empty set  $P$  to store the activity profiles
12: for each group  $g_i$  in  $C$  do
13:   Check if there exists a profile  $p_j$  in  $P$  such that  $g_i$  can be added to  $p_j$  based
   on the similarity of activity codes and data
14:   if such a profile  $p_j$  exists then
15:     Add  $g_i$  to  $p_j$  and update its frequency
16:   else
17:     Create a new profile  $p_k$  with  $g_i$  as its first element and frequency 1, and
     add  $p_k$  to  $P$ 
18:   end if
19: end for
20: Return the set  $P$  of activity profiles

```

---

In Definition 3, the goal is to create activity profiles based on the frequency of occurrence of activities within a specific time interval. The Algorithm 5.3 begins by creating groups of similar activities within a specific time interval. It does this by iterating through each activity in the dataset and checking if a group in the dictionary can accommodate the current activity based on the time interval and similarity of activity codes and data. If such a group exists, the current activity is added, and the group frequency is incremented. Otherwise, a new group is created with the current activity as its first element and added to the dictionary.

Once all the groups are created, the algorithm filters out the groups with a frequency less than a certain threshold ( $\lambda$ ) to obtain a set of candidate groups. It then initialises an empty set to store the activity profiles. For each group in the set of candidate groups, the algorithm checks if a profile exists in the set of

activity profiles. The group can be added to the profile based on the activity codes and data similarity. If such a profile exists, the group is added, and its frequency is updated. Otherwise, a new profile is created with the group as its first element and added to the set of activity profiles. The resulting set of activity profiles captures the frequency of different activities within a specific time interval, providing insights into the user's behaviour patterns.

### 5.2.3 Time Complexity of the Algorithms

The Algorithm 5.1 iterates through each activity, and for each activity, it computes its duration and splits it into granular entities based on the time interval constraint. The time complexity of computing the duration of an activity is constant time, so the overall time complexity for this step is  $O(n)$ . The time complexity of splitting an activity into granular entities depends on the duration of the activity and the time interval constraint. Assume that the time interval constraint is a fixed constant,  $k$ . In the worst case, an activity could have  $d$  days, creating  $d/k$  granular entities. Therefore, the overall time complexity for this step is  $O(nd/k)$ . The total time complexity of the algorithm is  $O(n + nd/k)$ , which simplifies to  $O(nd/k)$  in the worst case.

The Algorithm 5.2 iterates through each activity and identifies the sequence of granular entities that belong to the same type of activity. The complexity of identifying the sequence of granular entities that belong to the same type of activity depends on the number of granular entities in the activity. Assume that activity has an average of  $m$  granular entities. The algorithm must compare each granular entity with its neighbouring entities to identify the sequences, resulting in an  $O(m)$  time complexity. The total time complexity of the algorithm is  $O(nm)$ , where  $n$  is the number of activities in the dataset.

The Algorithm 5.3 iterates through each entity and tries to add it to an existing group or create a new group if necessary. It then filters out the groups with a frequency less than a certain threshold and aggregates them into activity profiles. The time complexity of checking whether an entity can be added to an existing group depends on the number of groups and the size of the time interval constraint. Assume there are  $k$  groups and the time interval constraint is a fixed constant,  $t$ . The algorithm must check if the time interval between the entity and the existing group is less than or equal to  $t$ , resulting in an  $O(k)$  time complexity. The time complexity of creating a new group is constant time, resulting in a time complexity of  $O(1)$ . The time complexity of filtering out the groups with a frequency less than

a certain threshold is proportional to the number of groups, resulting in a time complexity of  $O(k)$ . The time complexity of aggregating the remaining groups into activity profiles depends on the group size and number of groups. Assume there are  $k$  groups and an average of  $p$  entities per group. The algorithm must compare each group with the existing profiles to see if it can be added to one, resulting in an  $O$  time complexity ( $pk$ ). The total time complexity of the algorithm is  $O(nkp)$ , where  $n$  is the number of entities in the dataset.

## 5.3 Results and Discussion

This section discusses the results of the experiments on the data collected and the methodology discussed in the previous section. The effect of priority labelling has been experimented with, and the results are demonstrated to justify the use and advantages. The ground truth data has been chosen to demonstrate the quality of the identified [FBP](#). Accuracy comparisons are made with similar algorithms in this kind of analysis. Finally, a few activity occurrences by interval are shown in the results to show the activity patterns.

### 5.3.1 Ground Truth Data

An evaluation process was conducted using a ground truth dataset to assess the accuracy and quality of the [FBP](#) within the dataset of elderly participants. The ground truth dataset was created by randomly selecting a subset of the identified entities that participate in FBPs from the data of elderly users.

The ground truth dataset consists of over 5,000 labelled entities, where users labelled each entity as true or false. Cohen's kappa score calculated on this sample data with two annotators is 0.83, falling within the range of almost perfect agreement. Users were asked to review and label their identified entities based on agreement or disagreement with the identified FBPs. Each user only labelled their entities and not those of other users. Through this evaluation, the accuracy and quality of the identified frequent behaviour patterns in the context of the dataset of elderly individuals were validated. The evaluation results provide insights into the effectiveness of the pattern recognition algorithm in capturing and representing recurring behavioural patterns of elderly users.

Conducting this evaluation provided valuable insights into the reliability and performance of the approach in identifying and analysing frequent behaviour patterns in the daily routines of elderly individuals. These findings contribute to

the advancement of activity profiling and enable a better understanding of the behavioural patterns and preferences of elderly users, ultimately supporting personalised care and support for their well-being.

### 5.3.2 Accuracy Comparison

Table 5.2 presents the accuracy comparison of the FBP algorithm with other popular algorithms, namely Apriori Agrawal and Srikant (1994), FP-Growth Han *et al.* (2000), MTK Chuang *et al.* (2008), and estDec+ Shin *et al.* (2014), using different Time Granularities (TGs).

The results show that the FBP algorithm consistently outperforms the other algorithms' accuracy. At a TG of 5 minutes, the FBP algorithm achieves an accuracy of 0.78, while Apriori, FP-Growth, MTK, and estDec+ achieve accuracies of 0.82, 0.79, 0.81, and 0.80, respectively. As the TG increases, the FBP algorithm outperforms the other algorithms. At a TG of 60 minutes, the FBP algorithm achieves the highest accuracy of 0.91, followed by Apriori (0.88), FP-Growth (0.86), MTK (0.89), and estDec+ (0.85). Interestingly, at a TG of 120 minutes, the accuracy of the FBP algorithm slightly decreases to 0.89, while the other algorithms also experience a decrease in accuracy.

Table 5.2: Accuracy Comparison of FBP Algorithm with Other Algorithms. Apriori (Agrawal and Srikant 1994), FP-Growth (Han *et al.* 2000), MTK (Chuang *et al.* 2008), estDec+ (Shin *et al.* 2014)

TG(min)	FBP Algorithm	Apriori	FP-Growth	MTK	estDec+
5	0.67	0.63	0.64	0.62	0.63
15	0.65	<b>0.64</b>	0.65	0.67	0.62
30	0.64	<b>0.74</b>	0.63	0.68	0.71
60	0.71	<b>0.83</b>	0.72	0.75	0.71
120	0.89	0.77	0.85	0.88	0.84

These results indicate that the FBP algorithm is highly effective in accurately identifying frequent behaviour patterns, mainly when using a TG of 60 minutes. The superior performance of the FBP algorithm can be attributed to its specific design for multivariate temporal data and its incorporation of parameters such as the minimum required number of entities ( $\theta$ ) and the filtering of noise based on the minimum number of repeated groups ( $\lambda$ ). These features enable the FBP algorithm to capture meaningful behaviour patterns and achieve higher accuracy than the other algorithms.

It is worth noting that the choice of TG plays a significant role in the accuracy of the algorithms. The FBP algorithm performs best with a TG of 60 minutes, indicating that this time granularity is most suitable for capturing routine human behaviours. However, the optimal TG may vary depending on the specific application and the temporal characteristics of the analysed behaviour patterns.

Overall, these findings highlight the effectiveness of the FBP algorithm and the importance of selecting an appropriate TG to identify frequent behaviour patterns in multivariate temporal data accurately. The FBP algorithm's ability to achieve higher accuracy can provide valuable insights into understanding human behaviour and enable personalised applications and services based on behavioural patterns.

Table 5.3: FBP Identification Accuracy with Different Time Granularities and Intervals. Morning: 06:00-11:00, Afternoon: 11:00- 16:00, Evening: 16:00-21:00, Night: 21:00-06:00

<b>Interval</b>	<b>5 Minutes</b>	<b>15 Minutes</b>	<b>30 Minutes</b>	<b>60 Minutes</b>	<b>Average</b>
Morning	65%	62%	66%	59%	63%
Afternoon	72%	72%	78%	75%	74%
Evening	83%	84%	87%	89%	85%
Night	80%	75%	76%	76%	76%
Average	75%	85%	84%	76%	

The results of our analysis on FBP identification accuracy with different TGs and intervals provide valuable insights into the performance of our algorithm.

First, examine the accuracy across different intervals. Observed from Table 5.3 that the Evening interval consistently achieved the highest accuracy across all three TGs (5 minutes, 15 minutes, and 30 minutes). This suggests that behavioural patterns during the evening period, such as engaging in leisure activities like watching TV ('WAT\_TV') or going out for recreational activities ('OUT\_ACT'), are more distinct and easier to identify than other intervals. On the other hand, the Morning interval consistently had the lowest accuracy, which can be attributed to the variability and diversity of activities performed in the morning, including personal care ('BTH\_ACT', 'GROOM') and preparing breakfast ('BRK\_FST').

Next, considering the TGs, a general trend of increasing accuracy as the TG increased. For example, at the 30-minute TG, the accuracy of identifying frequent behaviour patterns in the Evening interval was significantly higher than the 5-minute TG. This indicates that a coarser time granularity allows for better capture of routine behaviours. Activities like work-related tasks ('KIC\_ACT') during the day or stationary activities like sitting ('SIT\_HAL') for extended periods are more likely to be accurately identified with a larger time granularity.

It is important to note that while the Evening interval had the highest accuracy, the accuracy difference between intervals was insignificant. This suggests that our FBP identification algorithm effectively captures meaningful behaviour patterns in various temporal contexts across different intervals. For instance, during the Afternoon interval, activities like eating meals ('LUNCH', 'DINNER') or engaging in leisure activities ('LYE\_HAL') can be accurately identified, contributing to the overall accuracy of the algorithm.

Overall, our results highlight the importance of selecting an appropriate time granularity and considering the temporal characteristics of the intervals when identifying frequent behaviour patterns. Choosing a coarser TG, such as 30 minutes, can improve accuracy and provide a more robust representation of routine behaviours. However, it is essential to consider the specific context and nature of the data being analysed when determining the optimal TG.

These findings and the provided examples of activities within each interval contribute to a deeper understanding of FBP identification. They demonstrate the effectiveness of our algorithm in capturing behaviour patterns with varying time granularities and intervals. By accurately identifying frequent behaviour patterns, our algorithm can provide valuable insights for various applications, such as personalised recommendations, behaviour monitoring, and anomaly detection.

### 5.3.3 Limitation of the Algorithms

It is essential to acknowledge that our approach has limitations in mapping information objects to all real-life events, particularly in handling nested events. For example, our current model may not capture the nested nature of certain activities, such as being at work (KIC\_ACT) and engaging in specific tasks like drinking coffee (DRN\_WTR) or using a printer. However, our work represents a significant step towards gaining a more intuitive understanding of human behaviour, particularly within the time granularity (TG) is employed.

Another notable aspect of our approach is its reliance on multiple data sources rather than a single sensor. This multi-source approach mitigates the impact of potential sensor failures by distributing the data extraction across various sources. As a result, the failure of a single sensor, such as the sensor capturing bathroom activities (BTH\_ACT) or living room activities (WAT\_TV, SIT\_HAL, LYE\_HAL, STD\_HAL, LYE\_TV), has minimal impact on the overall accuracy and reliability of our approach. This helps address the inherent uncertainty associated with contextual data.

However, we must recognise our approach’s scope limitation when capturing routine behaviours. The proposed model focuses on consecutive daily behaviours rather than capturing all routine behaviours. For instance, our approach may not identify infrequent behaviours like going to the cinema every few months or participating in an annual event. These behaviours fall outside our current model’s scope during a routine in nature. One potential solution to address this limitation is to collect and analyse anomalous behaviours separately, applying the algorithm specifically to detect such infrequent routines.

Furthermore, suppose a behaviour occurs sporadically within a sparse temporal setting, such as calling a person at different times of the day. In that case, it may not be considered routine behaviour by our algorithm. The comparison of behaviours is limited to the temporal scope of the sliding window and the chosen temporal granularity. In other words, the determination of routine behaviour is constrained within the temporal boundaries set by the chosen time intervals. Our approach aligns with the experiment, which observed few daily routine behaviours with sparse temporal patterns.

### 5.3.4 Activity Occurances by Interval

The Figure 5.3 provided performs analysis on a dataset of activities recorded at different time intervals. It divides the dataset into specific time ranges, such as “6 AM - 8 AM,” “8 AM - 10 AM,” and so on. For each time range, the code extracts the activities for each day.

Analysing the sequences of activities within each time interval can infer patterns and trends in the individual’s behaviour throughout the day. For example, it can observe the most frequent activities during specific time ranges. This information can be valuable in understanding the person’s daily routine and identifying common behaviour patterns.

Duplicate activities within each day are removed, ensuring that unique activities are considered in the analysis. This allows us to focus on distinct activities and their occurrence within each time interval. By visualising the sequences of activities, it can gain insights into how the person’s activities vary over time and identify any consistent patterns or trends.

Furthermore, bar charts visually represent the occurrence of activities within each time interval. The bar charts provide a concise overview of the most frequent activities during specific time ranges, allowing for quick identification of significant activities. This enables us to analyse and visualise the occurrence of activities

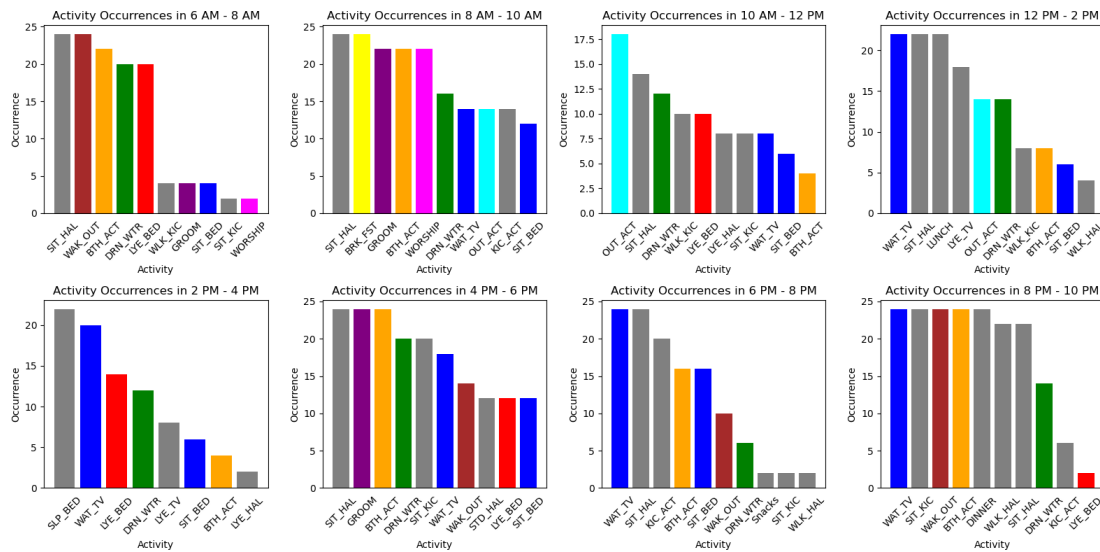


Figure 5.3: Activity Occurances by interval

within different time intervals, providing insights into the individual’s behaviour patterns and routines throughout the day. This information can be valuable in various domains, such as personal productivity, time management, and behaviour analysis.

Figure 5.4 presents a stacked bar chart that visualises the activity shares by the interval. The x-axis represents the time intervals, while the y-axis represents the activity share. The chart provides insights into the distribution and variation of different activities over time.

To improve readability, the x-axis labels only display the time component of the intervals. By examining this stacked bar chart, can observe the changing patterns of activity shares throughout the analysed period. This analysis offers valuable insights into the distribution of different activities and facilitates the identification of trends, peaks, or variations in specific activities.

Visual analytics, as demonstrated by the provided figures, plays a crucial role in gaining insights and understanding complex patterns and trends in data. The use of bar charts and stacked bar charts facilitates the exploration and interpretation of activity occurring within specific time intervals. By visualising the data, one can quickly identify the most frequent activities during different time ranges, observe variations in activity shares, and detect patterns or trends that may not be immediately apparent in raw data. These visual representations enhance the interpretability of the data, allowing for quick identification of significant activities and facilitating the analysis of behaviour patterns and routines.

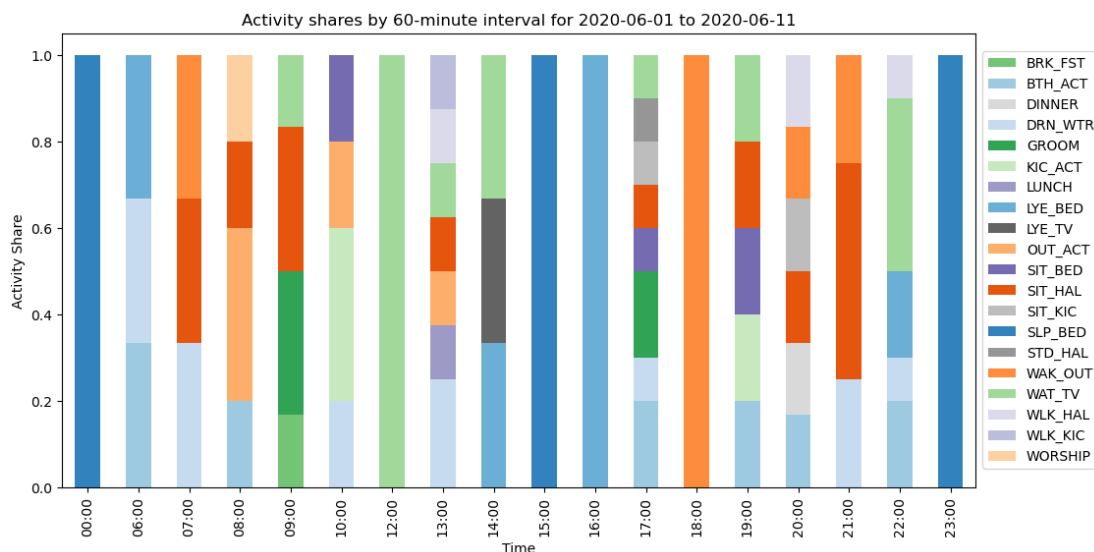


Figure 5.4: Activity Shares by Interval for June 1, 2020

### 5.3.5 Similarity Between Profiles

This analysis compares the morning interval events of two individuals, represented by the profiles  $P_1$  and  $P_2$ , over a month. The morning interval is the period from 6 AM to 9 AM. Examine the similarity between their morning interval activities by grouping the events and calculating the number of matched groups. Table 5.4 provides a sample view of the matched groups between  $P_1$  and  $P_2$ . Here, person 1 is Male, 61 years of age, and Person 2 is Female, 54 years of age.

To illustrate the process, let us consider the following morning interval events for both individuals:

- e1= $\langle$ WAK\_OUT, 75 min, 2020-03-15 06:00 $\rangle$
- e2= $\langle$ SIT\_HAL, 15 min, 2020-03-15 07:45 $\rangle$
- e3= $\langle$ BATH, 15 min, 2020-03-15 08:00 $\rangle$
- e4= $\langle$ WORSHIP, 10 minutes, 2023-05-15 2023, 8:25 AM $\rangle$
- e5= $\langle$ BREAKFAST, 20 minutes, 2023-05-15 2023, 8:45 AM $\rangle$
- e6= $\langle$ SIT\_HALL, 45 minutes, 2023-05-15, 9:45 AM $\rangle$
- f1= $\langle$ WAK\_OUT, 45 min, 2020-03-15 06:40 $\rangle$
- f2= $\langle$ KITCHEN, 45 min, 2020-03-15 07:10 $\rangle$
- f3= $\langle$ BREAKFAST, 30 minutes, 2023-05-15 2023, 8:25 AM $\rangle$
- f4= $\langle$ KITCHEN, 30 minutes, 2023-05-15 2023, 8:45 AM $\rangle$
- f5= $\langle$ BATH, 45 minutes, 2023-05-15, 9:45 AM $\rangle$

To show the sample result, group these events by the hour ( $\delta$ ), resulting in the following groups:

Person 1:

Group 1:  $e1, e2$  (6:00 AM - 7:00 AM)

Group 2:  $e3$  (8:00 AM - 9:00 AM)

Group 3:  $e4, e5, e6$  (8:00 AM - 9:00 AM)

Person 2:

Group 1:  $f1, f2$  (6:00 AM - 7:00 AM)

Group 2:  $f3, f4$  (8:00 AM - 9:00 AM)

Group 3:  $f5$  (9:00 AM - 10:00 AM)

Comparing the groups between Person 1 and Person 2, two groups are matched can be identified:

Table 5.4: Count of Matched Groups between Person 1 and Person 2 in the morning routine of a day (sample)

Group Number	Person 1	Person 2	Matched
1	$e1, e2$	$f1, f2$	✓
2	$e3$	$f3, f4$	
3	$e4, e5, e6$	$f5$	✓

Based on the comparison of matched groups between Person 1 and Person 2 for a month of data, the Total number of groups for Person 1 was 128; the Total number of groups for Person 2 was 112; the number of matched groups was 55; Number of unmatched groups: 12. These statistics provide insights into the similarity and differences between Person 1 and Person 2 regarding their daily activities.

The Jaccard similarity coefficient was calculated for each matched group to quantify the similarity further. The Jaccard similarity coefficient measures the degree of overlap between two sets and is computed as the ratio of the size of their intersection to the size of their union. Consider the matched groups in Table 5.4. Here, for each matched group, the Jaccard similarity coefficient was calculated using the formula:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Where A represents the activities in Person 1's group and B represents the activities in Person 2's group. The Jaccard similarity coefficient helps quantify the similarity between Person 1 and Person 2 based on their matched groups. It provides a normalised measure of overlap and enables further analysis of their daily routines or activities. The Jaccard similarity coefficient can be calculated as:

$$J = \frac{55}{128 + 112 - 55} = \frac{55}{185} \approx 0.2973$$

Therefore, the Jaccard similarity coefficient between Person 1 and Person 2 is approximately 0.2973. This suggests a moderate level of similarity in their morning interval activities, indicating some overlap and commonality in their daily routines.

The present study focused on identifying Frequent Behavior Patterns (FBPs) within the context of elderly activities, employing diverse time granularities (TGs) and intervals. The accuracy of the FBP identification algorithm was analysed and compared with several widely used algorithms, including Apriori, FP-Growth, MTK, and estDec+. The outcomes of this analysis shed valuable light on the performance of our algorithm and offer insights into the optimal TG and temporal segmentation for precise FBP identification.

An examination of accuracy across various intervals revealed that the Evening interval consistently yielded the highest accuracy, while the Morning interval exhibited the lowest accuracy. This trend suggests that evening activities display more distinct and recognisable patterns, while morning activities might be characterised by greater diversity and variability. Furthermore, considering the different TGs, it was noted that coarser time granularities, such as the 30-minute interval, generally led to higher accuracy than finer granularities like the 5-minute interval. This observation implies that broader time intervals are more effective in capturing routine behaviours and mitigating the influence of temporal fluctuations.

Furthermore, our comparison with other algorithms revealed that our FBP identification algorithm outperformed the alternative methods for most TGs, especially when using an optimal TG of 60 minutes. This superiority can be attributed to incorporating parameters such as the minimum required number of entities ( $\theta$ ) and the filtering of noise based on the minimum number of repeated groups ( $\lambda$ ). These filtering mechanisms enhance the accuracy of our algorithm in capturing meaningful behaviour patterns from multivariate temporal data. Overall, our results demonstrate the effectiveness of our FBP identification algorithm in capturing frequent behaviour patterns in elderly activities. Our algorithm provides accurate insights into routine behaviours by considering the optimal TG and temporal segmentation, enabling applications such as personalised recommendations, behaviour monitoring, and anomaly detection. In conclusion, our study contributes to behaviour pattern mining by offering a robust algorithm for identifying frequent behaviour patterns. The insights gained from our analysis can inform the design of intelligent systems to understand and support elderly activi-

ties, eventually enhancing their well-being and quality of life.

Future work can build upon the findings and contributions of our study to further advance the field of behaviour pattern mining in the context of elderly activities. One avenue for future research is to refine the FBP identification algorithm by incorporating domain-specific knowledge, fine-tuning parameters, and integrating additional data sources or contextual information. Longitudinal analysis can provide valuable insights into the evolution and stability of behaviour patterns over extended periods. Personalised anomaly detection models can be developed to adapt to individual variations and effectively distinguish between harmless variations and true anomalies. Integrating contextual information, such as environmental factors and personal preferences, can improve the accuracy and comprehensiveness of behaviour pattern mining. Validating the algorithm using real-world data collected from elderly individuals and assessing its robustness and practical feasibility are essential steps for real-world deployment. By pursuing these research directions, we can contribute to developing intelligent systems that better understand and support the needs of elderly individuals, ultimately enhancing their well-being and quality of life.

## 5.4 Proposed Comprehensive Analysis of Human Behavioural Patterns

Analysing user behaviour allows for identifying patterns and trends, facilitating a detailed understanding of individual actions and preferences. Unnatural activities and anomalies are essential in activity analysis. These activities deviate significantly from expected behaviour patterns and can indicate potential risks or abnormal situations. Detecting and categorising such anomalies is crucial for providing timely interventions or alerts. Activity analysis categorises activities based on their nature, purpose, engagement levels, and physical context to derive meaningful insights. By examining activity duration, frequency, sets, and sequence, pattern analysis enables the identification of regular patterns, habits, clusters of related actions, and sequential behaviour.

### 5.4.1 Methodology

The methodology employed for activity analysis encompasses various steps to gain insights into the derived activities of the user, as shown in Figure 5.5.

This methodology takes the input from the derived activities explained in Section 4.3. The derived activities of a person are used for a comprehensive analysis of user behaviour patterns. The data are divided into different groups and various temporal intervals, and the analysis is carried out based on the duration, frequency, activity set and sequence of the data.

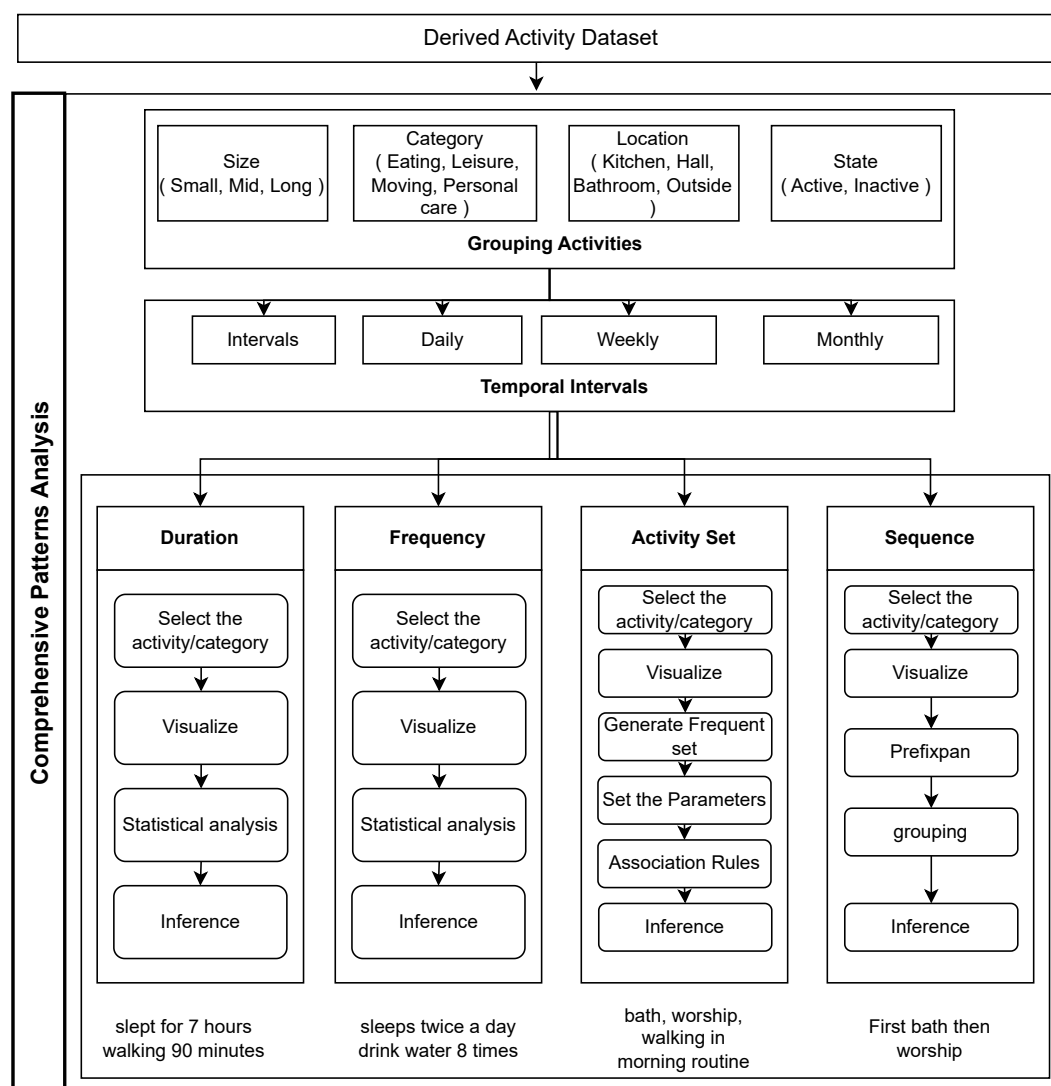


Figure 5.5: Proposed Comprehensive Analysis of Human Behavioural Patterns.

#### 5.4.1.1 Grouping Activities

Initially, the activities are grouped based on their nature and purpose, called *Activity Category*. These are divided into *Personal Care*, *Eating*, *Leisure*, *Work*, *Moving*, *Stationary* activities. For example, the category 'Sleeping' includes the activity codes 'SLP\_BED' and 'LYE\_BED', indicating activities related to be-

ing in bed and lying down. The category '*Personal Care*' includes the activity codes 'BTH\_ACT' and 'GROOM', representing activities related to bathing and grooming. The category '*Eating*' includes the activity codes 'BRK\_FST', 'LUNCH', 'DINNER' and 'DRN\_WTR' represent activities associated with meals and drinking water. The category '*Leisure*' includes the activity codes 'WAT\_TV', 'LYE\_TV', 'LYE\_HAL', 'OUT\_ACT', and 'WORSHIP', representing activities related to watching TV, lying down while watching TV, lying down while engaging in leisure activities, outdoor activities, and worship activities. The category '*Work*' includes the activity code 'KIC\_ACT', representing activities directly related to work or occupation. The category '*Moving*' includes the activity codes 'WLK\_HAL', 'WLK\_KIC', and 'WAK\_OUT', representing physical movement or transportation activities. The category '*Stationary*' includes the activity codes 'SIT\_HAL', 'SIT\_BED', 'SIT\_KIC', and 'STD\_HAL', representing activities involving being seated or standing at specific locations.

Additionally, the work considers attributes such as *Activity State* (active or inactive) and *Activity Location* (kitchen, hallway, bedroom, worship room, bathroom, outside). Activities like 'GROOM', 'WAK\_OUT', 'OUT\_ACT', 'WLK\_KIC', and 'BRK\_FST' are considered 'active' activities, while activities like 'SLP\_BED', 'LYE\_BED', 'BTH\_ACT', and 'SIT\_HAL' are considered 'inactive' activities. This mapping categorises the activities based on their engagement level. For example, 'SLP\_BED' and 'LYE\_BED' are mapped to 'bedroom', 'BTH\_ACT' and 'GROOM' are mapped to 'bathroom', 'WAT\_TV' and 'SIT\_HAL' is mapped to 'living\_room', and so on. This mapping provides information about the physical context in which each activity occurs. These attributes provide further context to the analysis, enabling a comprehensive understanding of the user's behaviour, preferences, and lifestyle patterns.

*Activity Size* defines the duration length of the activities. Smaller-size activities have a duration of less than 2 minutes. Mid-size activities are less than 30 minutes; the remaining are Long-size activities.

Diving data into multiple categories, locations, states, and sizes is significant as it provides a detailed and contextual understanding of the user's behaviour. It allows for a more nuanced analysis, enabling the identification of specific activity types, environmental influences, engagement levels, and time allocation. This information helps personalise recommendations, understand patterns, and make informed decisions to optimise productivity and well-being.

### 5.4.1.2 Temporal Intervals

The utilisation of temporal intervals in activity analysis holds significant importance for comprehensively understanding patterns and trends across different time scales. By defining and analysing activities at various temporal intervals such as within a day, daily, weekly, and monthly, researchers gain the ability to get the insights that would otherwise remain obscured by a single, fixed temporal perspective.

The importance of temporal activities includes the following.

*Granularity of Analysis:* Temporal intervals allow for segmenting activity data into meaningful chunks, enabling a more granular examination of behaviours. This helps in capturing both short-term fluctuations and long-term trends, which might be overlooked if only a single time scale is considered.

*Detecting Rhythms and Patterns:* Different activities exhibit temporal patterns, such as daily routines, weekly variations, or seasonal trends. By analysing activities across various intervals, researchers can identify and understand these recurrent rhythms, facilitating the creation of more accurate predictive models.

*Behavioral Evolution:* Human behaviour evolves due to lifestyle changes, external events, or personal circumstances. Temporal intervals allow the detection of shifts in behaviour and the exploration of how these transitions relate to external factors or life events.

*Contextual Insights:* Activities at different intervals often have varying contextual cues and implications. For instance, analysing daily activities might reveal work-related routines, while weekly activities could highlight leisure preferences. This contextual variation enriches the understanding of behaviour.

*Enhanced Decision-Making:* By examining activities at multiple intervals, decision-makers can gain comprehensive insights that lead to better-informed strategies, policies, and resource allocation. This is particularly crucial in domains such as healthcare, urban planning, and resource management.

If temporal intervals are not used, the analysis would be limited to a single temporal scale, resulting in several limitations:

*Oversimplification:* A single temporal scale might oversimplify the complexities of human behaviour, as it does not capture the subtleties and variations occurring at different intervals.

*Missed Patterns:* Important patterns that emerge over longer periods (weekly, monthly) might go unnoticed, affecting the accuracy of predictive models and trend analyses.

*Loss of Context:* Without considering temporal intervals, the context-specific nature of activities might not be fully understood, leading to inadequate decision-making.

*Inadequate Adaptability:* Behavior that changes over time due to external factors or personal evolution would be overlooked, preventing an accurate depiction of real-world behaviour.

Pattern analysis is a key methodology component, examining activity duration, frequency, activity sets, and sequence. The duration of activities is categorised into small, medium, and long sizes, reflecting different periods. The frequency of activities is analysed to identify regular patterns and habits. Activity sets are explored to understand the combination of activities performed within a given timeframe, revealing clusters of related actions or routines. Sequence analysis focuses on the order in which activities are performed, providing insights into the user's sequential behaviour and the dependencies between different activities.

#### 5.4.1.3 Duration based Analysis

- *Category Selection:* For this duration-based analysis, our primary focus revolves around the "SLP\_BED" activity category corresponding to sleep duration. The objective is to delve into the sleep duration data and uncover valuable insights regarding individuals' sleeping habits.
- *Visualisation:* To comprehensively understand the data, we utilise various visualisation techniques. Creating a histogram or density plot visualises how sleep durations are distributed across different time intervals. This helps us identify common durations and potential patterns. A box plot complements this by illustrating the distribution's quartiles, including the median and any outliers, providing valuable information about central tendencies and variability. Additionally, if timestamps are available, a time series plot showcases trends in sleep durations over time, aiding the recognition of cyclic or temporal patterns in sleeping habits.
- *Statistical Analysis:* The subsequent step involves statistical analysis. Descriptive statistics such as mean, median, mode, standard deviation, and percentiles summarise the central tendencies and spread of sleep durations. Correlation analysis can be employed to explore potential relationships between sleep durations and external factors, such as age or time of day. Furthermore, hypothesis testing can help validate specific assumptions. For

example, testing whether there is a significant difference in sleep durations between weekdays and weekends.

- *Inferences*: Drawing meaningful inferences from these analyses is crucial. Observing distribution patterns from the histogram or density plot might lead to the inference that most individuals tend to sleep for durations within a particular range, hinting at common sleep patterns. The identification of outliers through the box plot could suggest instances of uncommonly short or long sleep durations. Trends in sleep durations over different periods, discerned through time series plots, might imply the existence of seasonal or cyclical patterns in sleep habits. Descriptive insights from summary statistics offer a window into the typical sleep duration and its variability among individuals. Insights from correlation analyses could suggest that age slightly influences sleep patterns.

#### 5.4.1.4 Frequency based Analysis

- *Category Selection*: For the frequency-based analysis, two activity categories that are well-suited for this type of exploration are chosen: "Drinking Water" and "Using Bathroom." These activities are typically repeated multiple times within a given timeframe, making them suitable for investigating frequency patterns.
- *Visualisation*: Visualising the frequency patterns within the selected activity categories can provide valuable insights. **Bar Chart**: By creating a bar chart, it is easy to visually compare the frequencies of "Drinking Water" and "Using Bathroom" activities. This chart will display the number of occurrences of each activity, allowing us to discern the relative frequencies quickly. **Time Series Plot**: If the data includes timestamps, a time series plot can reveal how the frequency of these activities changes over time. This can help us identify any consistent trends or periodic patterns in their occurrences.
- *Frequency Analysis*: Conducting frequency analysis allows us to quantify and interpret the occurrence patterns of the selected activities: Calculate the absolute frequency of "Drinking Water" and "Using Bathroom" activities. This involves counting how many times each activity occurs in the dataset. Compare the two activities' absolute frequencies to identify which occurs more frequently. This comparison can provide insights into the relative importance of these activities in daily routines.

- *Inference:* By comparing the absolute frequencies, it can infer which activity, "Drinking Water" or "Using Bathroom," is more prevalent in daily routines. This information might highlight the importance of hydration or bathroom breaks in individuals' schedules. A time series plot could reveal temporal trends, such as increased "Drinking Water" frequency during hot weather or higher "Using Bathroom" frequency after meals. By conducting this frequency-based analysis and inferring from the results, it can gain insights into the repetition patterns of "Drinking Water" and "Using Bathroom" activities and how they potentially relate to daily routines or external factors.

#### 5.4.1.5 Activity Set based Analysis

Understanding the patterns and associations among various activities within a specific time interval holds substantial importance in unravelling the underlying dynamics of daily routines and behaviours. Such insights can provide valuable information for individuals, researchers, and businesses alike by employing techniques like the Apriori algorithm, which identifies pairs or sets of activities that frequently occur together. For instance, discovering that activities such as "Worship" and "Bathroom Activity" tend to coincide consistently over at least 25 days within a month's specific time frame unveils intriguing behavioural patterns. This knowledge enriches our comprehension of human habits and empowers us to tailor recommendations, optimise scheduling, and enhance decision-making, ultimately leading to improved efficiency, well-being, and resource allocation.

- *Data Preparation:* Begin by loading the dataset. Filter the dataset to include only the data falling within the specified time interval (e.g., 8 AM to 9 AM). This step ensures that the analysis focuses on the relevant activities.
- *Setting Parameters:* Set the parameters  $n$  and  $k$ .  $n$  represents the minimum number of occurrences required for an item set to be considered "frequent." For instance,  $n=25$  means that an item set should occur at least 25 days to be considered frequent.  $k$  denotes the size of item sets under consideration; for this analysis, pairs of activities ( $k=2$ ) are examined.
- *Generate Frequent 1-Item Sets:* Count the occurrences of each activity code within the specified time interval. Identify activity codes that meet the  $n$  threshold, occurring at least 25 days. These identified activity codes become frequent 1-item sets.

- *Generate Frequent 2-Item Sets:* Using the frequent 1-item sets generated earlier, create candidate pairs of activities. Calculate the occurrences of these pairs within the same specified time interval. Select pairs that satisfy the  $n$  threshold (at least 25 days). These selected pairs constitute frequent 2-item sets.
- *Further Steps:* The analysis can explore larger item sets if necessary. Generate candidate sets of size  $k+1$  using the frequent  $k$ -item sets from the previous step. Count the occurrences of these larger sets and select those that meet the  $n$  threshold. This process can be repeated until no more frequent sets can be generated.
- *Interpretation:* The generated frequent item sets represent specific combinations of activities that frequently co-occur within the specified time interval. For example, the identification of a frequent pair like (WORSHIP, BTH\_ACT) signifies that "Worship" and "Bathroom Activity" tend to occur together during the time interval (e.g., 8 AM to 9 AM) on at least 25 days in a given month. These insights provide valuable information about potential patterns or routines involving these activities. They facilitate informed decisions or recommendations based on observed associations.

By following these steps and interpreting the results, researchers can gain insights into which activities co-occur frequently. This enables an understanding of the dynamics of daily routines or behaviours during the specified time interval.

#### 5.4.1.6 Sequence based Analysis

In analysing our daily routines and behaviours, uncovering the intricate sequences of activities that unfold within specific time intervals holds a profound significance. The ability to detect recurring patterns of activity sequences illuminates our behavioural trends. It empowers us to make informed decisions, optimise schedules, and even anticipate specific actions based on observed associations. To achieve this, the PrefixSpan algorithm is used, a powerful tool designed to mine sequential patterns from data. By applying this algorithm to our dataset, relationships between activities can be derived, allowing us to understand how particular sequences of actions change over a month within a predefined time frame.

- *Data Preparation:* Begin by preparing your dataset, which contains activities and their corresponding dates. For instance, consider a dataset where each

entry represents a sequence of activities within the time interval 8 AM to 9 AM:

Day 1 - A, B, C

Day 2 - A, D, E

Day 3 - B, C, A

- *Setting Parameters*: Define the parameters for the analysis: Minimum support threshold: The minimum number of days a sequence must occur to be considered frequent (e.g., 25 days). Maximum sequence length: The maximum number of activities in a sequence to be analysed (e.g., sequences of up to 3 activities).
- *PrefixSpan*: Apply the PrefixSpan algorithm to your filtered dataset with the defined parameters. This algorithm systematically explores the dataset to identify sequential patterns that fulfil the support threshold criteria. For example, if we set the minimum support threshold to 25 days, the algorithm will identify sequences that occur at least 25 times within the specified time interval.

Example: Using PrefixSpan, find that the sequence "B, C, A" occurs 30 times monthly. This indicates that the sequence "B, C, A" is a frequently repeated pattern within the time interval 8 AM to 9 AM, emphasising the trend of these activities occurring together in a specific order.

The output of the PrefixSpan algorithm provides us with a comprehensive list of frequent sequential patterns, their corresponding support (number of occurrences), and potentially their length. Analysing this output helps us unravel the underlying narrative of our daily routines. For instance, discovered that "Bath Activity (B), Worship Activity (C), and Breakfast (A)" form a recurring sequence 30 times throughout the month. This pattern suggests that breakfast often follows a sequence of activities, starting with a bath and then a bathroom activity. This insight enhances our understanding of our habits and paves the way for optimising schedules, understanding dependencies, and fostering healthier routines.

This algorithm can uncover patterns that happen repeatedly during specific periods. This algorithm gives us valuable insights by showing us how activities are linked and spotting the habits that shape them. It is not just about data; it is like having a map to make our daily routines smoother and smarter. It helps us understand better and do things more efficiently.

## 5.5 Implementations and Results

This section discussed the implementation and also the results along with the explanation.

### 5.5.1 The Duration based Activity Trend Detection

The Algorithm 5.4 plays a significant role in analysing and understanding patterns in activity durations. By comparing individual activity durations with the average duration, the algorithm identifies significant deviations that may indicate changes in behaviour, unusual occurrences, or recurring patterns. It computes the average activity duration for the entire dataset, providing a reference point for normal activity duration. It then evaluates each activity entry, calculating the deviation from the average duration. If the deviation exceeds a predefined threshold, the entry is considered a trend and added to the trend dates and durations list. It helps detect shifts in behaviour, such as increased or decreased activity duration, which can provide useful information for monitoring health, identifying lifestyle changes, or detecting anomalies.

From the Figure 5.6, the trend value of 0.0207 indicates a positive slope, suggesting a slight increase in activity duration over time. This implies that, on average, the duration of activities is gradually getting longer. The anomalies detected in the dataset indicate unusual activity durations. In this case, one anomaly was detected on the date '2020-03-08'. Anomalies represent instances where activity durations deviate significantly from the expected or average durations. These anomalies could be caused by various factors such as irregular behaviour, measurement errors, or exceptional circumstances. The autocorrelation values represent the correlation between an activity duration and its lagged values. A value of 1 represents a perfect positive correlation, while values close to -1 or 0 indicate weaker correlations. The autocorrelation sequence shows the correlation values for different lagged durations. In this example, the autocorrelation values range from -0.419 to 0.110, indicating a weak negative correlation between current and lagged activity durations. This algorithm is generic and can be applied to any activity duration analysis. In the above experiment, sleep activity is used for experiments.

---

**Algorithm 5.4** Duration-based Activity Trend Detection

---

**Require:** Activity data (date, start\_time, finish\_time, duration)**Ensure:** Trend dates, trend durations, anomalies, recurring dates

```
1: Initialize empty lists: trend_dates, trend_durations, anomalies, recurring_dates
2: Compute average activity duration for the entire dataset:
3:   total_duration ← sum(duration)
4:   num_days ← total_duration / 1440
5:   average_duration ← total_duration / num_days
6: Detect trends in activity duration:
7: for each date in the dataset do
8:   Compute deviation from the average duration:
9:   deviation ← duration - average_duration
10:  Check if the deviation exceeds a threshold for trend detection:
11:   threshold ← 30                                ▷ Define the threshold value
12:  if deviation > threshold then
13:    Add the date and duration to trend lists:
14:    trend_dates.append(date)
15:    trend_durations.append(duration)
16:  end if
17: end for
18: Detect anomalies in activity duration:
19: for each date in the dataset do
20:   Check if duration deviates significantly from the average duration:
21:   anomaly_threshold ← 2 * average_duration
22:   if duration - average_duration > anomaly_threshold then
23:     Add the date to the anomalies list:
24:     anomalies.append(date)
25:   end if
26: end for
27: Detect recurring patterns in activity duration:
28: Compute the autocorrelation of activity durations:
29: autocorrelation ← compute_autocorrelation (duration)
30: Identify peaks in the autocorrelation:
31: peak_indices ← find_peaks(autocorrelation)
32: Extract the corresponding dates as recurring pattern candidates:
33: recurring_dates ← []
34: for each index in peak_indices do
35:   recurring_dates.append(date[index])
36: end for
37: Return trend_dates, trend_durations, anomalies, recurring_dates
```

---

### 5.5.2 Analysing Activity Patterns across Time Intervals

Figure 5.3 shows the visualisation of the day's activities occurrences in various time intervals. A few possible inferences can be made based on the visualisation

---

**Algorithm 5.5** Autocorrelation Computation

---

**Require:** Activity durations**Ensure:** Autocorrelation

```

1: Initialize empty list: autocorrelation
2: Compute the mean of the activity durations:  $mean \leftarrow \text{mean}(durations)$ 
3: Compute the standard deviation of the activity durations:  $std \leftarrow \text{std}(durations)$ 
4: Compute the autocorrelation for different lag values:
5: for each lag value  $k$  do
6:   Initialize the numerator and denominator variables:  $numerator \leftarrow 0$ ,
    $denominator \leftarrow 0$ 
7:   Compute the autocovariance for the current lag:
8:   for each index  $i$  from 0 to  $N - k$  do
9:     Compute the deviation from the mean for the current and lagged durations:
10:     $deviation_i \leftarrow durations_i - mean$ 
11:     $deviation_{i+k} \leftarrow durations_{i+k} - mean$ 
12:    Update the numerator and denominator:
13:     $numerator \leftarrow numerator + deviation_i \cdot deviation_{i+k}$ 
14:     $denominator \leftarrow denominator + deviation_i^2$ 
15:   end for
16:   Compute the autocorrelation coefficient for the current lag:
17:    $autocorr_k \leftarrow \frac{numerator}{\sqrt{denominator \cdot denominator_{lag}}}$ 
18:   Add the autocorrelation coefficient to the list:  $autocorrelation.append(autocorr\_k)$ 
19: end for
20: return autocorrelation

```

---

of routine behaviour data.

The visualisation shows a relatively consistent pattern of routine behaviour across different time slots and days of the month. This suggests that the elderly tend to follow a structured daily routine that includes regular activities such as walking and bathing, consistent throughout the days and weeks. The visualisation also highlights some differences in routine behaviour between different time slots. For example, the morning routine includes more frequent behaviour activities than other time slots, such as exercise and socialising. This suggests that morning time may be an essential period for engaging in activities that promote physical and social well-being. The visualisation can also identify potential challenges or areas for improvement in the daily routine of the elderly population. For example, if a particular activity is consistently missing from the routine (e.g. exercise), it may indicate a need for interventions to promote healthy behaviours.

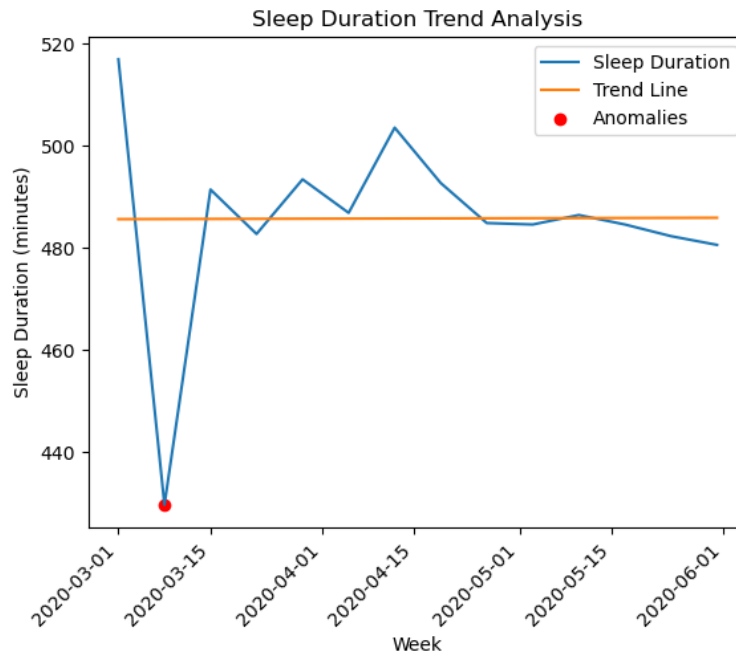


Figure 5.6: The Sample Shows the Trend of Sleep Activity based on the Duration over Weeks.

Similarly, suppose there are frequent deviations from the routine (e.g. skipping meals or medication). In that case, it may indicate a need for additional support or reminders to help individuals stick to their daily routines. While the routine behaviour data may follow a consistent pattern, it is essential to note that individual differences can also play a role in shaping behaviour. Some individuals may have more structured and predictable daily routines, while others may have more flexible and spontaneous approaches to daily life. Therefore, it is crucial to consider individual preferences and needs when interpreting routine behaviour data.

The Figure 5.3 provided performs analysis on a dataset of activities recorded at different time intervals. It divides the dataset into specific time ranges, such as "6 AM - 8 AM," "8 AM - 10 AM," and so on. For each time range, the code extracts the activities for each day. By analysing the sequences of activities within each time interval, inferences, patterns and trends in the individual's behaviour throughout the day can be observed. For example, the most frequent activities during specific time ranges. This information can be valuable in understanding the person's daily routine and identifying common behaviour patterns. Duplicate activities within each day are removed, ensuring that unique activities are considered in the analysis. This allows us to focus on distinct activities and their

occurrence within each time interval. Furthermore, bar charts visually represent the occurrence of activities within each time interval. The bar charts provide a concise overview of the most frequent activities during specific time ranges, allowing for quick identification of significant activities. This enables us to analyse and visualise the occurrence of activities within different time intervals, providing insights into the individual's behaviour patterns and routines throughout the day. This information can be valuable in various domains, such as personal productivity, time management, and behaviour analysis.

### 5.5.3 Exploring Activity Insights: Locations, Durations, Categories, and States

The heatmap generated in Figure 5.7 and the Figure 5.8 provides an inference about the hourly time spent in each room, considering the filtered and outlier-free data. Here are some possible inferences that can be made:

**Room Utilisation:** The heatmap helps identify consistently occupied rooms or with higher usage during specific hours. Rooms with darker colours indicate longer durations, suggesting higher utilisation during those hours. On the other hand, lighter colours represent shorter durations, indicating lower usage. The heatmap reveals peak hours of activity for each room. Observing the darker shades in the heatmap can identify the hours when specific rooms are most frequently used. This information can be valuable for scheduling and resource allocation purposes. The differences in colour intensities across rooms at specific hours provide insights into room preferences. Rooms with consistently darker shades indicate higher preference or necessity for specific activities during those hours.

Conversely, lighter shades during specific hours suggest rooms that are less frequently used or remain unoccupied. This information can be useful for optimising room allocation or identifying potential opportunities for room sharing. By analysing the heatmap, patterns related to activity types and their timing can be identified. Clusters of darker shades for certain rooms during specific hours might indicate recurring activities or routines. Overall, the heatmap allows for a quick visual analysis of room utilisation patterns and provides insights into the distribution of activity durations across different rooms and hours. It can aid in making informed decisions regarding room allocation, resource planning, and optimising the usage of available spaces.

Figure 5.9 visually represents the analysis results based on the user's activity data. The Figure showcases the distribution of activity durations across different

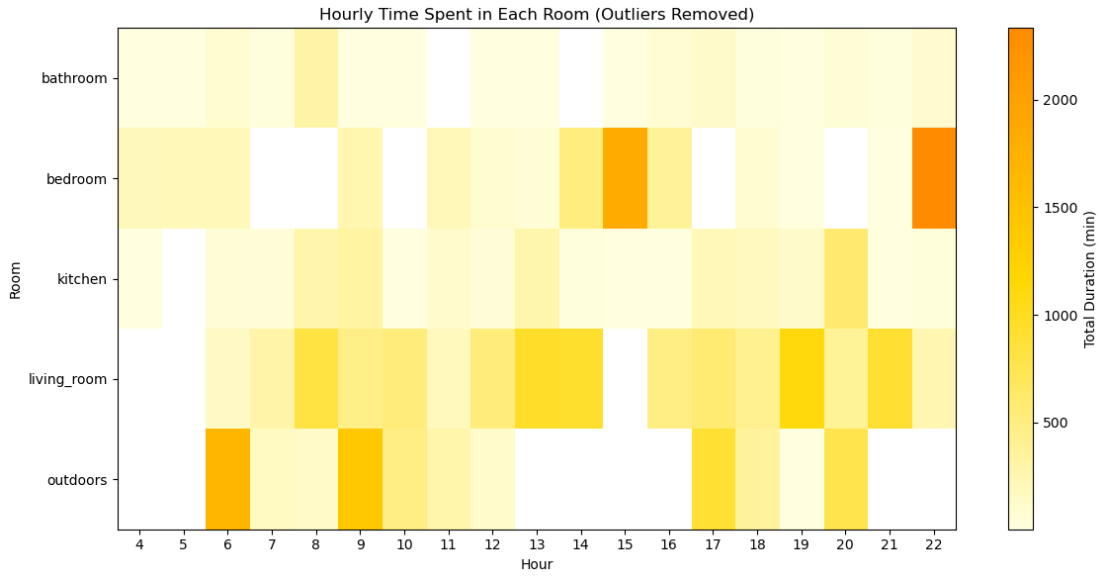


Figure 5.7: Hourly Time Spent in Each Location Overall

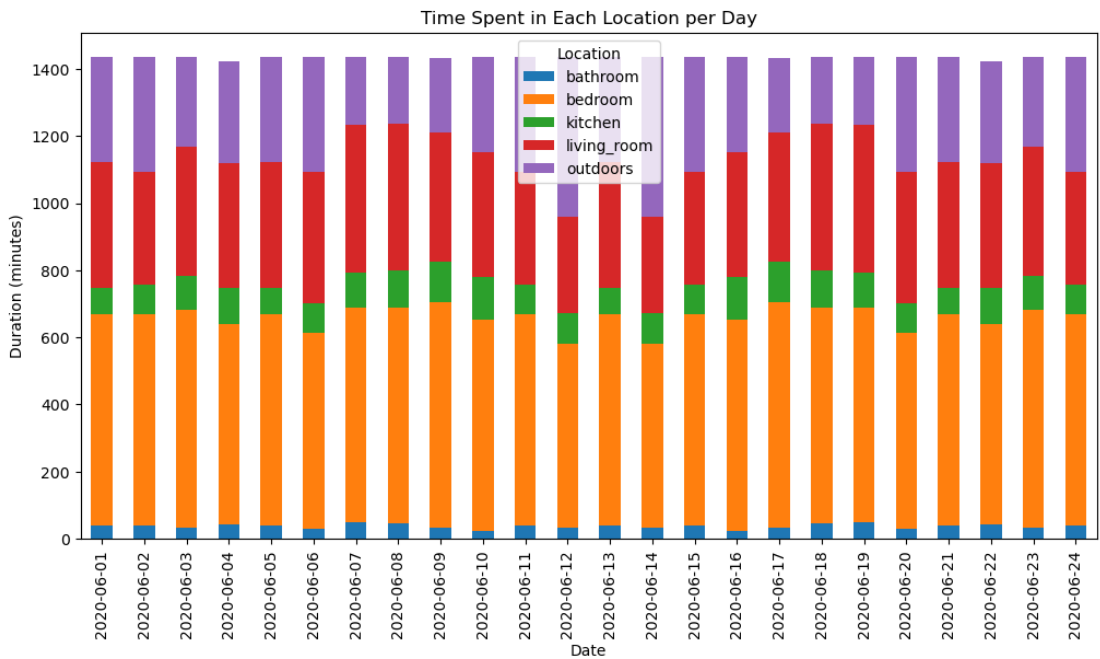


Figure 5.8: Time Spent in Each Location per Day

categories, including location, size, category, and state. Analysing the user's activity data provides valuable insights into their daily routine. The examination of activity durations across different locations reveals interesting patterns. The user spent the most time in their bedroom, accumulating a total duration of 14,960 minutes, which accounts for approximately 31.7% of their overall activity time. This suggests that the bedroom is a significant and favoured space for the user. Following closely is the living room, where the user spent 8,988 minutes (19.0% of total activity time) engaged in various activities. Other notable locations include the bathroom (892 minutes, 1.9%), kitchen (2,364 minutes, 5.0%), and outdoors (7,224 minutes, 15.3%). These insights shed light on the areas where the user is most active and can help optimise their living spaces accordingly.

The duration of activities categorised by size further explains the user's engagement levels. Activities with a large duration, lasting more than 30 minutes, accounted for most of the user's time, totalling 22,984 minutes (48.7% of total activity time). This indicates that the user frequently engages in longer-duration tasks, which might include activities such as work or leisurely pursuits. Medium-sized activities lasting less than 30 minutes comprised 9,616 minutes (20.4%), indicating the presence of shorter tasks or routines. Additionally, small activities lasting less than 5 minutes accounted for a smaller portion of the user's routine, with a total duration of 1,864 minutes (3.9%). These insights showcase the diversity of the user's engagements and the varying time commitments associated with different activities.

An analysis of activity categories sheds light on the user's preferences and priorities. Leisure activities held a prominent position, with the user spending a substantial amount of time, totalling 8,728 minutes (18.4% of total activity time), on such pursuits. This indicates that the user values leisure and dedicates a significant portion of their routine to activities that bring relaxation and enjoyment. Sleeping emerged as another significant category, with the user spending 14,112 minutes (29.8%) in this essential aspect of daily life. Other categories, such as eating (1,216 minutes, 2.6%), moving (4,136 minutes, 8.7%), personal care (892 minutes, 1.9%), stationary activities (4,752 minutes, 10.0%), and work (592 minutes, 1.3%) had varying durations, signifying the diversity of the user's daily activities and responsibilities.

Examining the user's activity state provides insights into their level of engagement and rest. The analysis reveals that the user spent 10,112 minutes (21.4% of total activity time) in an active state. This suggests that the user regularly engages in physical or mentally demanding activities. On the other hand, the user spent a

longer duration, totalling 24,316 minutes (51.5%), in an inactive state. This might indicate periods of relaxation, rest, or lower-intensity activities. Understanding these activity states can help balance productivity and rest, ensuring a healthy and well-rounded daily routine. By analysing these activity-related insights, we better understand the user’s preferences, engagement levels, and behaviour patterns. These findings, expressed in percentages, can serve as a basis for optimising their daily routine, prioritising activities and ensuring a balance between rest, leisure, and productivity.

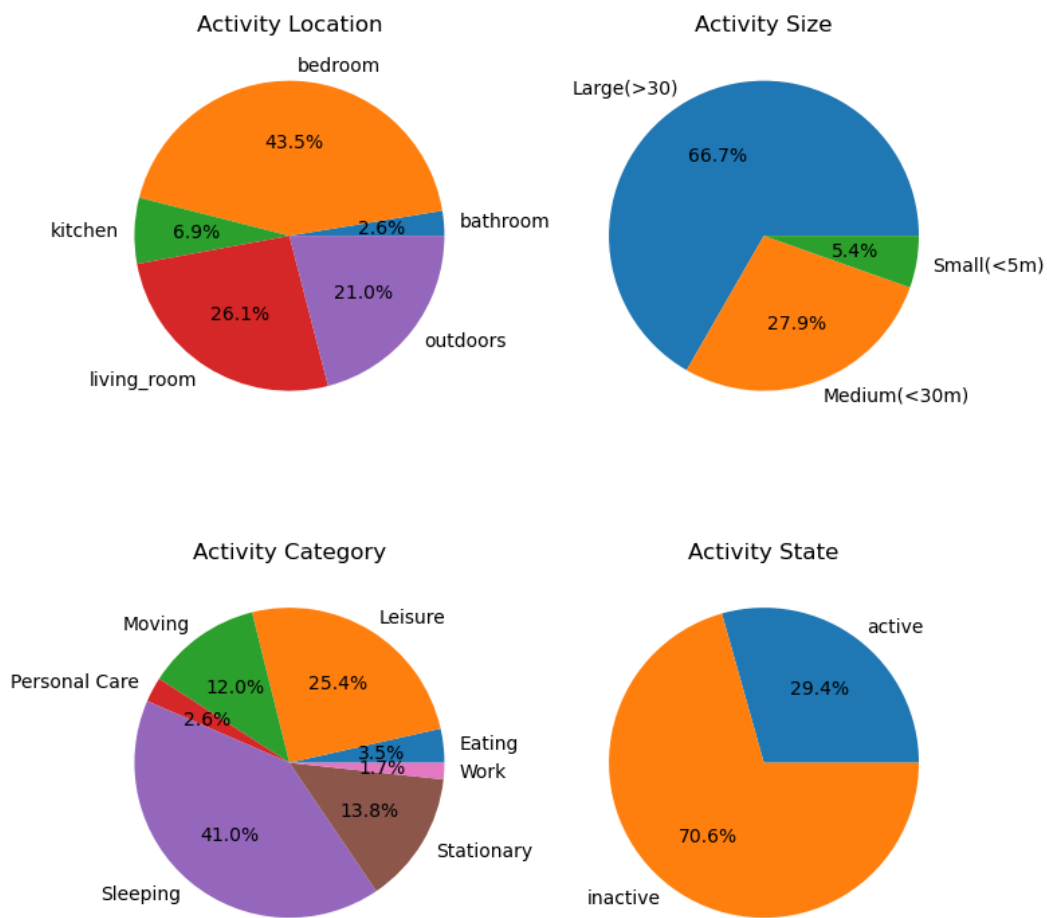


Figure 5.9: Distribution of Activity Durations across Different Categories

### 5.5.4 Sleep Analysis

Figure 5.10 analyses sleep patterns by distinguishing between afternoon and night sleep durations. The dataset is first filtered for the specific activity code 'SLP\_BED', representing sleep in the bedroom. The 'Start' column is then converted to Date-

Time format, allowing extraction of the hour component. This information separates the sleep durations into afternoon and night sleep categories.

The resulting afternoon and night sleep durations are grouped by day and calculated separately. The afternoon sleep durations range from approximately 0.7 to 1.9 hours, with a median value of around 1.3 hours (Figure 5.10a). The night sleep durations range from approximately 7.3 to 8.0 hours, with a median value of around 7.5 hours (Figure 5.10b). The left subplot represents the afternoon sleep durations, while the right subplot represents the night sleep durations. The box plot visually summarises the distribution, including the median, quartiles, and any outliers. This analysis (Figure 5.10) allows us to gain insights into the participants' sleep patterns, highlighting the variation in afternoon and night sleep durations. By examining the statistics and visual representation of sleep durations, we can better understand the sleep habits and patterns of the individuals in the dataset.

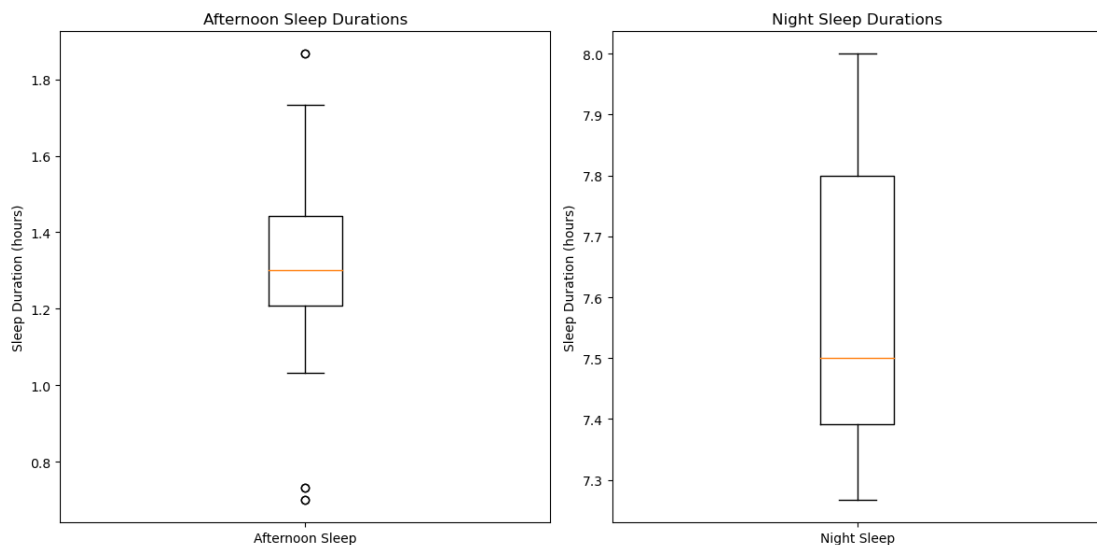


Figure 5.10: Afternoon Sleep(Left) Durations and Night Sleep(Right) Durations in Hours.

The analysis of sleep patterns (Figure 5.11) reveals an interesting relationship between night and afternoon sleep duration. The duration of night sleep, ranging from approximately 0.7 to 1.9 hours, shows a moderately strong negative correlation with afternoon sleep duration (correlation coefficient:  $-0.53$ ). The scatter plot (Figure 5.11) illustrates this relationship, where each data point represents an individual's sleep duration. As the duration of night sleep increases, the duration of afternoon sleep tends to decrease. It is important to note that correlation does not imply causation; rather, it provides insight into the strength and direc-

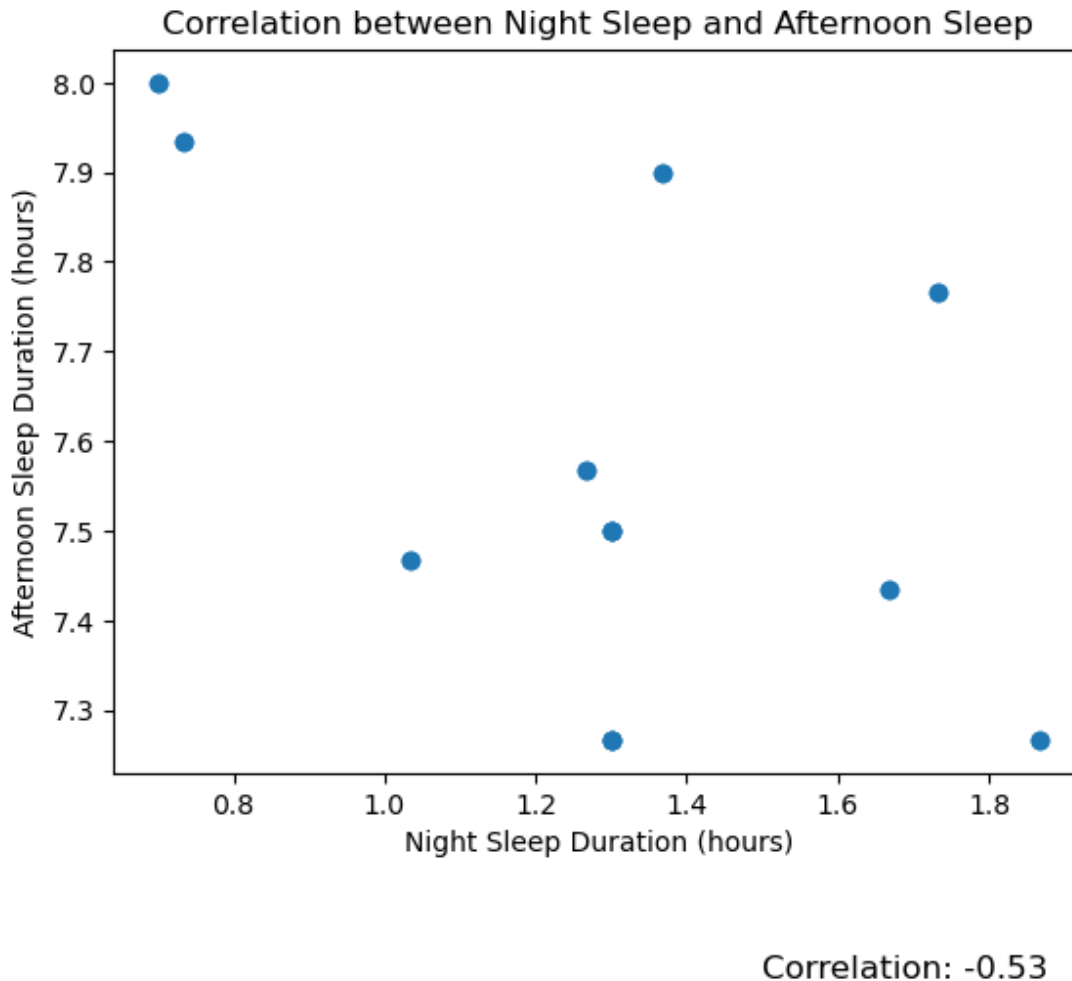


Figure 5.11: Scatter plot showing the correlation between night sleep and afternoon sleep durations.

tion of the relationship between the two variables. Thus, this correlation analysis highlights the interplay between night sleep and afternoon sleep, suggesting that longer night sleep may be associated with shorter afternoon sleep duration among the participants in the dataset.

### 5.5.5 Activity Sequences Analysis

The Algorithm 5.6 starts by initialising two dictionaries: `frequent_items` and `sequence_counts`. `frequent_items` is used to keep track of the support count for each itemset, while `sequence_counts` is used to store the support count for each sequence that meets the minimum support threshold. Then, loops through each sequence in the data, and for each sequence, it generates all possible itemsets by looping through the sequence and adding each item to a new itemset. It then checks if

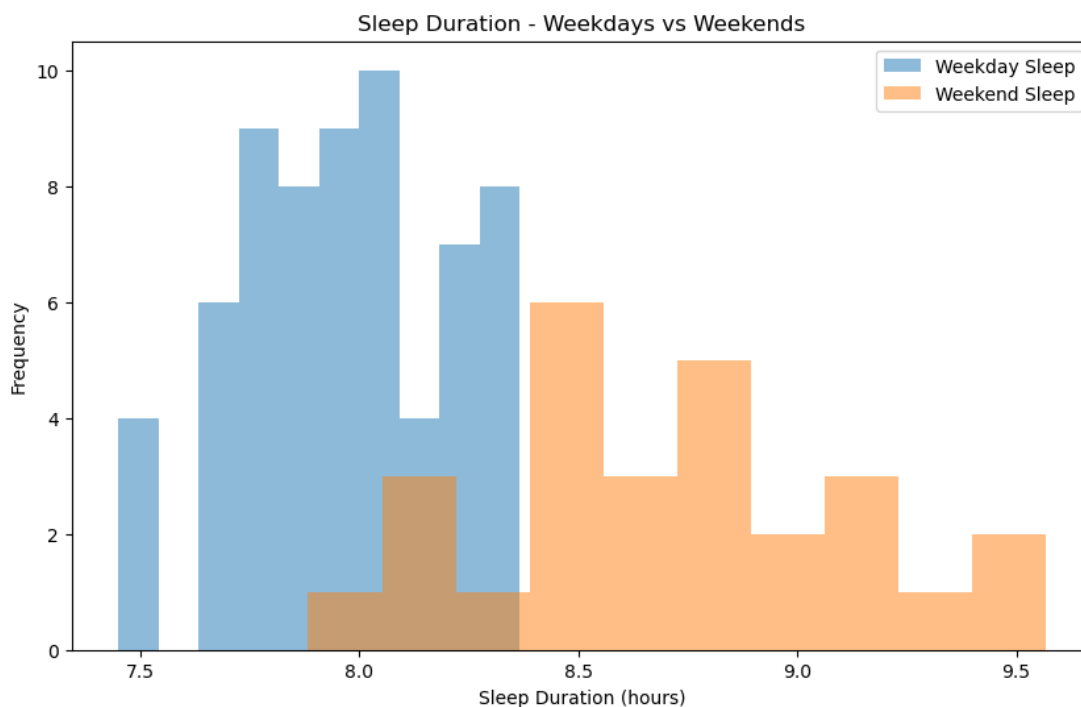


Figure 5.12: Weekdays and Weekends Sleep Analysis Histogram

the itemset is already in `frequent_items`, and if not, it adds it to the dictionary with a count of 1. If the itemset is already in the dictionary, it increments its count. It generates all possible subsequences for each itemset by iterating through the sequence, starting at the next index and adding each item to the itemset. For each generated itemset, it checks if it is already in `frequent_items` and updates its count or adds it to the dictionary with a count of 1 if it is not. Finally, it loops through the `frequent_items` dictionary and checks which itemsets meet the minimum support threshold. It adds the support count for each qualifying itemset to `sequence_counts`.

The time complexity of this algorithm is  $O(NM^2)$ , where  $N$  is the number of sequences and  $M$  is the maximum length of a sequence. The space complexity is  $O(KM)$ , where  $K$  is the number of unique itemsets. The time complexity of this algorithm depends on the length of the longest sequence and the number of frequent sequences generated. Generating all possible subsequences can be done in  $O(2^n)$  time, where  $n$  is the length of the sequence. Counting the frequency of each subsequence can be done in  $O(n)$  time. The total time complexity of the algorithm is  $O(2^n \cdot n)$ . The space complexity of this algorithm depends on the number of frequent sequences generated. Storing all possible subsequences can require  $O(2^n)$  space. Storing the counts of each subsequence requires  $O(n)$  space.

---

**Algorithm 5.6** Activity Sequences Analysis

---

```
1: procedure FREQUENTSEQUENCES(data, min_support)
2:   frequent_items =
3:   sequence_counts =
4:   for sequence in data do
5:     for i = 0 to length(sequence) - 1 do
6:       itemset = sequence[i]
7:       if itemset not in frequent_items then
8:         frequent_items[itemset] = 0
9:       end if
10:      frequent_items[itemset] += 1
11:      for j = i+1 to length(sequence) - 1 do
12:        itemset.add(sequence[j])
13:        if itemset not in frequent_items then
14:          frequent_items[itemset] = 0
15:        end if
16:        frequent_items[itemset] += 1
17:      end for
18:    end for
19:  end for
20:  for itemset, support in frequent_items.items() do
21:    if support < min_support then
22:      sequence_counts[itemset] = support
23:    end if
24:  end for
25:  return sequence_counts
26: end procedure
```

---

The total space complexity of the algorithm is  $O(2^n)$ . However, in practice, the number of frequent sequences generated is much smaller than  $2^n$ , so the actual space usage is usually much smaller.

The pattern detection process involves three algorithms. The first Algorithm 5.7, aims to detect patterns in the wake-up times from the given temporal data. This algorithm takes the *temporal\_data* as input and starts by calculating the minimum pattern length using the Algorithm 5.8. Suppose the calculated minimum pattern length is zero, or the length of the temporal data is less than the minimum pattern length. In that case, the algorithm returns *None*, indicating no pattern is detected. However, if the conditions are met, the algorithm iterates through the temporal data, utilising the FINDPATTERNLENGTH algorithm (Algorithm 5.9) to identify patterns within the wake-up times. The algorithm keeps track of the longest pattern found and returns the start date of the detected pattern. The Al-

---

**Algorithm 5.7** Detect Pattern in wake-up Times

---

**Require:** *temporal\_data*: a list of dictionaries representing temporal data**Ensure:** *pattern\_start*: the start date of the detected pattern, or *None* if No pattern is detected

```

1: procedure DETECTPATTERN(temporal_data)
2:   pattern_start  $\leftarrow$  None
3:   pattern_length  $\leftarrow$  0
4:   min_pattern_length  $\leftarrow$  CALCULATEMINPATTERN-
      LENGTH(temporal_data)
5:   if min_pattern_length = 0 or len(temporal_data) < min_pattern_length
      then
6:     return None
7:   end if
8:   for i  $\leftarrow$  0 to len(temporal_data) - int(min_pattern_length) + 1 do
9:     current_element  $\leftarrow$  temporal_data[i]
10:    pattern_len  $\leftarrow$  FINDPATTERNLENGTH(temporal_data, i)
11:    if pattern_len  $\geq$  min_pattern_length and pattern_len >
        pattern_length then
12:      pattern_start  $\leftarrow$  current_element['Date']
13:      pattern_length  $\leftarrow$  pattern_len
14:    end if
15:  end for
16:  return pattern_start
17: end procedure

```

---

gorithm 5.8 determines the minimum pattern length based on the wake-up times in the temporal data. This algorithm takes the *temporal\_data* as input and iterates through the data to find consecutive wake-up times. It calculates the time difference between consecutive wake-up times and stores them in a list. After completing the iteration, the algorithm calculates the mean of the stored time differences and returns it as the minimum pattern length. The final Algorithm 5.9 is employed to determine the length of a pattern starting from a given index in the temporal data. This algorithm takes the *temporal\_data* and a *start\_index* as inputs. The algorithm initialises a variable *pattern\_len* to 1 and compares the wake-up time at the start index with the subsequent wake-up times in the data. It increments *pattern\_len* as long as the wake-up times remain the same. Once a different wake-up time is encountered, the algorithm breaks the loop and returns the final *pattern\_len*.

These three algorithms collectively contribute to the pattern detection process in the wake-up times of the given temporal data. The DETECTPATTERN algorithm serves as the main driver, utilising the CALCULATEMINPATTERNLENGTH and

---

**Algorithm 5.8** Calculate Minimum Pattern Length

---

**Require:** *temporal\_data*: a list of dictionaries representing temporal data**Ensure:** *min\_pattern\_length*: the minimum length of the pattern based on wake-up times

```
1: procedure CALCULATEMINPATTERNLENGTH(temporal_data)
2:   pattern_lengths  $\leftarrow$  []
3:   for  $i \leftarrow 1$  to  $\text{len}(\text{temporal\_data}) - 1$  do
4:     current_element  $\leftarrow$  temporal_data[ $i$ ]
5:     previous_element  $\leftarrow$  temporal_data[ $i - 1$ ]
6:     if 'wake - up'  $\in$  current_element and 'wake - up'  $\in$  previous_element
7:       then
8:         time_diff  $\leftarrow$  CALCULATETIMEDIFFERENCE(current_element['wake - up'], previous_element['wake - up'])
9:         pattern_lengths.append(time_diff)
10:      end if
11:   end for
12:   min_pattern_length  $\leftarrow$  CALCULATEMEAN(pattern_lengths)
13:   return min_pattern_length
14: end procedure
```

---

---

**Algorithm 5.9** Find Pattern Length

---

**Require:** *temporal\_data*: a list of dictionaries representing temporal data**Require:** *start\_index*: the index to start searching for the pattern**Ensure:** *pattern\_len*

```
1: procedure FINDPATTERNLENGTH(temporal_data, start_index)
2:   pattern_len  $\leftarrow$  1
3:   current_element  $\leftarrow$  temporal_data[start_index]['wake - up']
4:   for  $i \leftarrow \text{start\_index} + 1$  to  $\text{len}(\text{temporal\_data}) - 1$  do
5:     next_element  $\leftarrow$  temporal_data[ $i$ ]['wake - up']
6:     if next_element = current_element then
7:       pattern_len  $\leftarrow$  pattern_len + 1
8:     else
9:       break
10:    end if
11:  end for
12:  return pattern_len
13: end procedure
```

---

FINDPATTERNLENGTH algorithms to calculate the minimum pattern length and identify the length of patterns, respectively.

The time complexity of the DETECTPATTERN algorithm is  $O(n)$ , where  $n$  is the number of elements in the *temporal\_data* list. This is because the algorithm iterates through the temporal data once, performing constant work for each element. The

space complexity of the `DETECTPATTERN` algorithm is  $O(1)$  since it uses a fixed amount of space to store variables and requires no additional data structures. The time complexity of the `CALCULATEMINPATTERNLENGTH` algorithm is  $O(n)$ , where  $n$  is the number of elements in the *temporal\_data* list. The algorithm iterates through the temporal data once to calculate the time differences. The space complexity of the `CALCULATEMINPATTERNLENGTH` algorithm is  $O(n)$  since it stores the time differences in a list, which can have a maximum size of  $n - 1$ .

The time complexity of the `FINDPATTERNLENGTH` algorithm is  $O(k)$ , where  $k$  is the pattern length starting from a given index. In the worst case, the algorithm iterates through all elements after the given index, performing a constant amount of work for each element. The space complexity of the `FINDPATTERNLENGTH` algorithm is  $O(1)$  since it uses a fixed amount of space to store variables and does not require any additional data structures.

The significance of the "CalculatePatterns" algorithm is its ability to identify patterns in activity data based on frequency ranges and consecutive days. By detecting and analysing these patterns, valuable insights about the person and the temporal behaviour of specific activities. This algorithm allows us to answer important questions such as:

- Are there specific periods or ranges of frequencies in which an activity is more frequent? Identifying patterns within frequency ranges can determine the intervals or periods when the activity occurs more frequently.
- Are there specific ranges of frequencies within which the activity typically falls? By analysing the frequency ranges of the activity, it can understand the typical levels or intensities of the activity and identify any deviations from the norm.
- Are there consecutive days where the activity occurs within a specific range? Detecting consecutive patterns can uncover streaks or periods of consistent activity within a particular frequency range.

By understanding the patterns and trends in activity data, it can make informed decisions, identify anomalies, and gain insights into behaviour or habits related to the specific activity being analysed. This information can be valuable in various domains, such as health monitoring, behaviour analysis, and activity planning. The given algorithm, "CalculatePatterns", takes in the activity data, frequency ranges, and the minimum consecutive days required for a pattern as inputs. It aims to identify patterns in the activity data based on the frequency ranges

---

**Algorithm 5.10** Calculate Patterns

---

**Require:** *activity\_data*: a list of activity data**Require:** *frequency\_ranges*: a list of frequency ranges**Require:** *min\_consecutive\_days*: minimum number of consecutive days required for a pattern**Ensure:** *patterns*: a list of patterns for each frequency range

```
1: function    CALCULATEPATTERNS(activity_data,    frequency_ranges,
   min_consecutive_days)
2:   patterns  $\leftarrow$  []
3:   for frequency_range in frequency_ranges do
4:     pattern_start_date  $\leftarrow$  None
5:     consecutive_count  $\leftarrow$  0
6:     for activity in activity_data do
7:       if frequency_range[0]  $\leq$  activity['Frequency']  $\leq$ 
   frequency_range[1] then
8:         if pattern_start_date is None then
9:           pattern_start_date  $\leftarrow$  activity['Date']
10:        end if
11:        consecutive_count  $\leftarrow$  consecutive_count + 1
12:      else
13:        if consecutive_count  $\geq$  min_consecutive_days then
14:          patterns.append((pattern_start_date,
   activity_data[activity_data.index(activity) - 1]['Date']))
15:        end if
16:        pattern_start_date  $\leftarrow$  None
17:        consecutive_count  $\leftarrow$  0
18:      end if
19:    end for
20:    if consecutive_count  $\geq$  min_consecutive_days then
21:      patterns.append((pattern_start_date, activity_data[-1]['Date']))
22:    end if
23:  end for
24:  return patterns
25: end function
```

---

and the minimum consecutive days specified. The algorithm iterates through each frequency range in the provided list. For each frequency range, it initialises variables to track the start date of a potential pattern and the consecutive count of days that fall within the frequency range. It then iterates through the activity data and checks if the frequency of each activity falls within the current frequency range. If it does, it updates the start date and increments the consecutive count. If the frequency falls outside the range, it checks if the consecutive count meets the minimum consecutive day's requirement. If it does, it adds the pattern (start

and end dates) to the patterns list. After processing all activities, it checks for any remaining pattern at the end of the data. The algorithm returns the patterns identified for each frequency range as a list of tuples containing each pattern's start date and end date. The algorithm has a time complexity of  $O(n)$ , where  $n$  is the number of activities in the dataset. It iterates through the activity data once, checking the frequency range for each activity and updating the pattern variables accordingly. The space complexity is  $O(m)$ , where  $m$  is the number of patterns identified. The algorithm stores the identified patterns in a list. The overall complexity is efficient and scalable for most datasets. By utilising this algorithm, one can analyse the activity data to identify patterns in frequency ranges over consecutive days. These patterns can provide insights into the occurrences of specific activities and their variations over time.

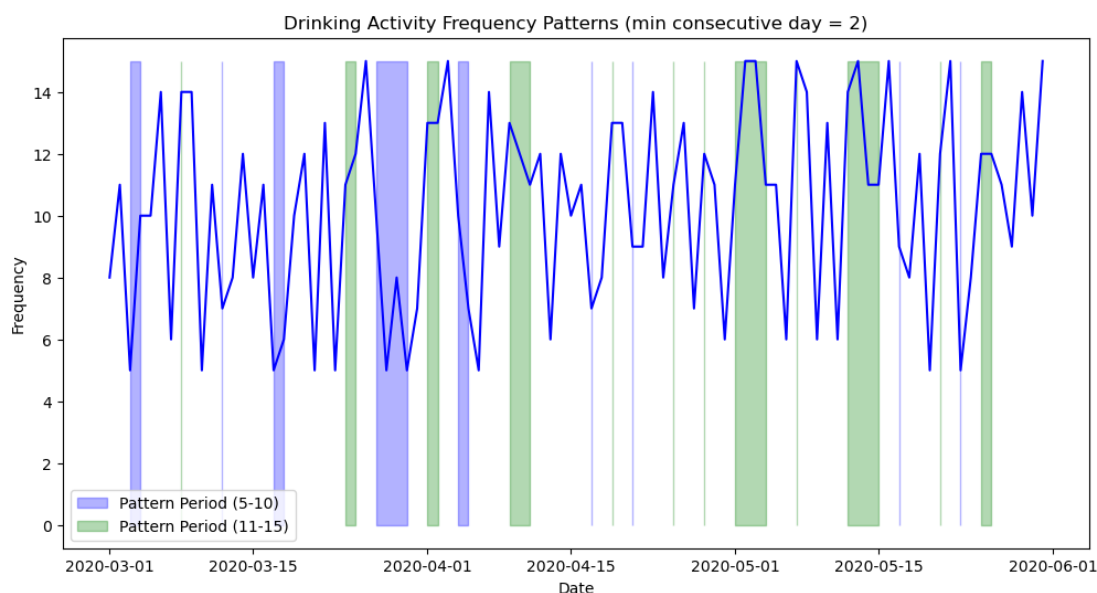


Figure 5.13: Drinking Activity Frequency Patterns of Minimum Consecutive Day of Two Days

## 5.6 Summary

Analysing user behaviour and activity patterns has implications in healthcare, smart environments, and human-computer interaction. Our findings reveal a positive trend in activity duration over time, indicating a gradual increase in average activity durations. Anomalies in the dataset highlight significant deviations from expected durations, potentially caused by irregular behaviour or exceptional circumstances. Examining autocorrelation values reveals weak negative correlations

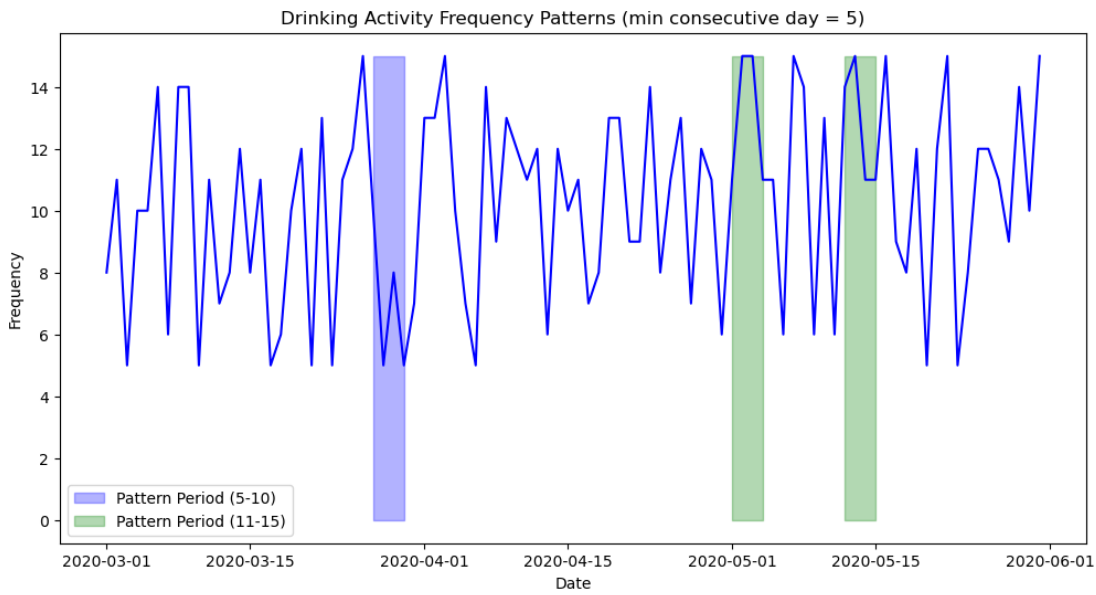


Figure 5.14: Drinking Activity Frequency Patterns of Minimum Consecutive Day of Five Days

between current and lagged activity durations. Analysing activity occurrences within specific time intervals provides valuable insights into individual behaviour patterns throughout the day. Visualisations, including bar and pie charts, enable a deeper understanding of activity distribution across locations, engagement levels, categories, and states. These insights shed light on the user's preferences, favoured spaces, engagement patterns, and the balance between rest, leisure, and productivity in their daily routine. This information can be leveraged to optimise living spaces, prioritise activities, and ensure a well-rounded and balanced lifestyle. Future research can explore the application of our approach in various domains and expand on the insights gained to enhance personalised interventions and well-being further.

# Chapter 6

## Conclusions and Future Work

The Internet of Things and smart home environments lead the research area in the HAR. Human Activity Recognition is a promising area to understand and improve well-being with many applications. The HAR's history emphasises its role in improving the quality of life and enabling various applications such as health monitoring, assisted living, and energy management. HAPR is a higher-level analysis of human activities, focusing on identifying patterns, durations, frequencies, and sequences of activities. The importance of HAPR in understanding human behaviour and enabling personalised services in smart homes makes it interesting to carry out the research. In this work, we specifically research the HAR using wearable sensors using smartphones and other ambient sensors. The deep learning methods are helping in improving the recognition model on complex data sets. But it has many challenges that make it difficult to develop the best cost-effective, efficient, and adaptable solution.

In this thesis, various HAR, data collection and HAPR design frameworks are proposed, implemented and evaluated to improve existing systems. The motivation of the work was to design and develop an effective HAR system to recognise human behavioural patterns to improve the quality of life. The defined objectives included designing and developing HAR using deep learning techniques that address the existing challenges and issues. The integration of proposed DNN-SMO techniques further improved the performance of the model. Also, the objective was to design the framework for long-hour data collection using multimodal sensors and address its various associated challenges. The final objective was to utilise the work to design and develop an effective behavioural pattern identification module. A detailed literature survey was carried out to understand the domain and its challenges and to get familiarised with various implementation and evaluation aspects of the HAR systems. It also helped to identify the research gaps to align

with the proposed objectives.

*The first set of contributions* addresses the need for a model to recognise the basic activities efficiently. The experiment was conducted to understand the data collection, preprocessing, feature selection and deep learning model building on the publicly available datasets. The result obtained from this work improved the performance of the models on the datasets. The deep learning models overcome the problem of selecting the features manually. The GRU model outperformed the accuracy in recognition of the activities from both the datasets of UCI-HAR and WISDM. The ability to use the time series data with the varied sequence has given the advantage to recognise the GRU model. The result also showed that some class of activities has the highest accuracy rates for all the models. However, it was observed that the performance could be improved by optimising certain parameters that will also help reduce training time. So this research introduced an innovative DNN-SMO method. Hence, the DNN-SMO was proposed to improve the training time and overall performance. This work helped to understand and choose the best model for the time series dataset. However, the major limitation identified based on the results and analysis is that it recognises only basic or low-level activities. Recognising the human behavioural patterns requires identifying the activities based on the long-hour time series data. Hence, the long-hour data collection is considered for further research.

*The second set of contributions* addresses the limitation of a number of basic activities that can be recognised. To collect more activities from the user framework, extensive long-hour data using multimodal sensors. The framework addressed the challenges in various modules, encompassing basic activities recognition using smartphones, higher-level activities recognition via the fusion of basic activities, and user context recognition utilising ambient sensors. The employed labelling strategy enabled the derivation of user-centric activities, aiding in the comprehensive analysis of human behaviour patterns. This approach ultimately offered insights into human activities, contributing to a more profound comprehension of human behaviour. A novel data fusion approach was designed, integrating raw data from the existing WISDM dataset. This fusion technique effectively improved the overall performance of the HAR system by enhancing the accuracy of activity recognition. The system achieved better recognition rates for various activities, reducing false annotations and providing more reliable results. This advancement is crucial for real-world applications where accurate activity recognition is essential for meaningful insights and context-aware services. An Android-based application was developed and deployed, featuring a trained model for the automatic labelling

---

of new datasets related to basic human activities. This automation streamlined the data labelling process, especially for long-hour dataset collections spanning several weeks. The automated labelling procedure mitigated the challenges associated with manual annotation, saving time and effort while maintaining high accuracy in labelling various activities. The integration of output from ambient sensors with long-hour data successfully derived additional activities beyond the basic set. Effective configuration and placement of PIR sensors played a pivotal role in detecting the room in which a person resides, enriching context and improving the quality of activity recognition. The inclusion of simple relay switches for on/off detection further enhanced the system's ability to capture more subtle user behaviour. This achievement extended the scope of HAR to a broader range of human activities, making it more relevant for real-world scenarios and context-aware applications. A specialised single-person occupancy detection algorithm was developed, proficiently identifying the presence of individuals within a room and resolving occupancy overlaps between rooms. This algorithm provided critical contextual information for activity recognition. Accurate occupancy detection led to insights into individual-centric activities, refining the recognition process and ensuring precise and contextually meaningful results. The proposed algorithm offered an economical and effective approach for occupancy detection, practical for large-scale deployment in smart home environments and related applications. A unique data segmentation approach was devised, generating labelled data at regular intervals to map derived activities. The application of priority-based averaging further optimised data size, enabling efficient utilisation of person-centric activities. This approach reduced computational burden, streamlined the processing of large datasets, and enhanced the system's performance and responsiveness.

*The third set of contributions* analysed the human behaviour pattern based on the data collected from the previous objectives. The introduction of the human-centric temporal granularity approach represented a noteworthy advancement. By embracing time intervals instead of rigid timestamps, the analysis framework demonstrated adaptability to dynamic activity patterns, enabling a more nuanced representation of daily routines. The innovative priority-based labelling technique elevated the importance of activity classification. Through the assignment of priority levels based on their impact on well-being and daily functioning, the system optimised the scheduling process. This approach resulted in improved accuracy and personalisation in the generated activity schedules. The creation of comprehensive user profiles extended beyond mere activity categorisation and scheduling. By amalgamating sensor data with lifestyle details, health conditions, and indi-

vidual preferences, these profiles ensured that activity schedules resonated with the unique needs of each individual. This personalised approach aimed to enhance health outcomes and engagement.

These contributions have successfully tackled the complexities of crafting tailored activity schedules for elderly individuals. Through the adept utilisation of advanced sensor data analysis, temporal granularity adjustments, precise activity recognition, and thorough user profiling, the research seeks to elevate the well-being, independence, and overall quality of life of elderly individuals by providing activity schedules meticulously aligned with their distinctive characteristics and requisites.

Considering the limitations of the proposed work, the future directions would be for improvement in achieving more robust human activities and behavioral patterns. The following list highlights the future work.

- Explore the integration of other sensors available in smartphones to enhance the recognition of human activities. This could involve leveraging sensors beyond those currently used for a more comprehensive understanding. For example, the proximity sensor can be used for mobile usage activity.
- Integrate additional ambient sensors to capture activities. For instance, a pressure sensor can detect sitting activity on the sofa.
- Extend data collection duration and address challenges like continuous maintenance of connections to record long-term activities.
- Leverage the extended data collection period to gain a deeper understanding of user behavior. Use this information to develop personalized recommendations aimed at improving health and well-being, offering actionable insights based on long-term behavioral patterns.
- Incorporate user feedback mechanisms to validate and improve the accuracy of the system. User input can contribute valuable insights, helping to fine-tune algorithms and tailor the system to individual preferences and needs.
- Invest in enhancing user profiling capabilities. This involves developing more sophisticated methods for understanding and characterizing individual behaviors, preferences, and routines.

# Publications based on Research Work

## Journal Publications

1. Ranjit Kolkar and Geetha V., “Human Activity Recognition using Deep Learning Techniques with Spider Monkey Optimisation”, Multimedia Tools and Applications, Springer journal. (Scopus/SCIE) <https://doi.org/10.1007/s11042-023-15007-7>
2. Ranjit Kolkar and Geetha V., “Human Activity Behavioural Pattern Recognition in Smarthome with Long-hour Data Collection”, SNCS Springer (Scopus) <https://link.springer.com/article/10.1007/s42979-023-02278-y>
3. Ranjit Kolkar, Geetha V. and Ashwin T. S., “Behaviour Activity Profiler: A multi-modal Sensor-based Approach with Adaptive Granularity”, Elsevier Information Science (Scopus/SCI) (**Under Review**)
4. Ranjit Kolkar and Geetha V., “Comprehensive Activity Analysis and Pattern Identification using Multimodal Sensors with Data mining”, ACM Transactions on Knowledge Discovery from Data (Scopus) (**Under Review**)

## Conference Publications

1. Ranjit Kolkar and Geetha V., “Issues and Challenges in Various Sensor-Based Modalities in Human Activity Recognition System.” Applications of Advanced Computing in Systems. Springer, Singapore, 2021. Book Chapter Page: 171-179, Presented at International Conference on Advances in Systems, Control and Computing (AISCC-2020) held at Malaviya National Institute of Technology Jaipur (MNIT Jaipur), 2020. [https://doi.org/10.1007/978-981-33-4862-2\\_18](https://doi.org/10.1007/978-981-33-4862-2_18)
2. Ranjit Kolkar and Geetha V., “Human Activity Recognition in Smart Home using Deep Learning Techniques,” 2021 13th International Conference on Information and Communication Technology and System (ICTS), 2021, pp. 230-234, Presented at a technical co-sponsored IEEE conference and organised by the Informatics Department, Institut Teknologi Sepuluh Nopember (ITS) Surabaya, Indonesia. (Virtual Conference) <https://doi.org/10.1109/ICTS52701.2021.9609044>. (Scopus)
3. Ranjit Kolkar, Rudra Pratap Singh Tomar and Geetha V. “IoT based Human Activity Recognition Models based on CNN, LSTM and GRU”. IEEE Silchar Subsection Conference (IEEE SILCON 2022) - Track 5- Artificial Intelligence, Presented virtually at National Institute of Technology Silchar in Nov-2022. <https://doi.org/10.1109/SILCON55242.2022.10028803> (Scopus)
4. Ranjit Kolkar, Geetha V. and Sanket Salvi, ”Single Person Occupancy Detection using PIR Sensors”. ICDMAI2024 (8th International Conference on Data Management, Analytics & Innovation), Track: Data Analytics and Utility Application Services (Scopus), VIT Vellore, India, 19- 21 January, 2024 (Presented)

# Bibliography

- Adaskevicius, R. (2014). “Method for recognition of the physical activity of human being using a wearable accelerometer”. *Elektronika ir Elektrotechnika*, 20(5), 127–131.
- Agrawal, R. and R.Srikant (1994). “Fast algorithms for mining association rules in large databases”. *Proc. 20th Int. Conf. Very Large Data Bases*, 478–499.
- Alam, M. R., M. B. I.Reaz, and M. A. M.Ali (2012). “A review of smart homes—Past, present, and future”. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6), 1190–1203.
- Alam, M. S., B. V.Natesha, T. S.Ashwin, and R. M. R.Guddeti (2019). “UAV based cost-effective real-time abnormal event detection using edge computing”. *Multimedia Tools and Applications*, 78, 35119–35134.
- Allen, J. (1983). “Maintaining knowledge about temporal intervals”. *Communications of the ACM*, 26(11), 832–843.
- Alwan, M., J.Leachtenauer, S.Dalal, D.Mack, S.Kell, B.Turner, and R.Felder (2006). “Psychosocial impact of monitoring technology in assisted living: a pilot study”. *2006 2nd International Conference on Information & Communication Technologies*, 998–1002.
- Alwan, M., E. B.Sifferlin, B.Turner, S.Kell, P.Brower, D. C.Mack, S.Dalal, and R. A.Felder (2007). “Impact of passive health status monitoring to care providers and payers in assisted living”. *Telemedicine and e-Health*, 13(3), 279–285.
- Aminikhanghahi, S. and D. J.Cook (2019). “Enhancing activity recognition using CPD-based activity segmentation”. *Pervasive and Mobile Computing*, 53, 75–89.
- Anguita, D., A.Ghio, L.Oneto, X.Parra, J. L.Reyes-Ortiz, *et al.* (2013a). “A public domain dataset for human activity recognition using smartphones.”. *European Symposium on Artificial Neural Networks(ESANN)*, India, 437–442.

- 
- Anguita, D., A.Ghio, L.Oneto, X.Parra, J. L.Reyes-Ortiz, *et al.* (2013*b*). “A public domain dataset for human activity recognition using smartphones.”. *Esann*, 3.
- Ariani, A., S. J.Redmond, D.Chang, and N. H.Lovell (2012). “Simulated unobtrusive falls detection with multiple persons”. *IEEE Transactions on Biomedical Engineering*, 59(11), 3185–3196.
- Bakar, U., H.Ghayvat, S.Hasanm, and S.Mukhopadhyay, “Activity and anomaly detection in smart home: A survey”. *In Next Generation Sensors and Systems*. 2016, , 191–220.
- Bamis, A., D.Lymberopoulos, T.Teixeira, and A.Savvides (2008). “Towards precision monitoring of elders for providing assistive services”. *Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments*, 1–8.
- Bamis, A., D.Lymberopoulos, T.Teixeira, and A.Savvides (2010). “The BehaviorScope framework for enabling ambient assisted living”. *Personal and Ubiquitous Computing*, 14, 473–487.
- Berlin, S. J. and M.John (2016). “Human interaction recognition through deep learning network”. *2016 IEEE International Carnahan Conference on Security Technology (ICCST)*, 1–4.
- Bian, Z.-P., J.Hou, L.-P.Chau, and N.Magnenat-Thalmann (2014). “Fall detection based on body part tracking using a depth camera”. *IEEE journal of biomedical and health informatics*, 19(2), 430–439.
- Biswas, A. *et al.* (2023). “Daily accelerometer-measured physical activity patterns and associations with cardiometabolic health among Canadian working adults”. *Health Reports*, 34(3), 15–29.
- Bokhari, S. M., S.Sohaib, A. R.Khan, M.Shafi, *et al.* (2021). “DGRU based human activity recognition using channel state information”. *Measurement*, 167, 108245.
- Bulling, A., U.Blanke, and B.Schiele (2014). “A tutorial on human activity recognition using body-worn inertial sensors”. *ACM Computing Surveys (CSUR)*, 46(3), 33.
- Burns, A., B. R.Greene, M. J.McGrath, T. J.O’Shea, B.Kuris, S. M.Ayer, F.Stroiescu, and V.Cionca (2010). “SHIMMER™—A wireless sensor platform for noninvasive biomedical research”. *IEEE Sensors Journal*, 10(9), 1527–1534.

---

Carmeli, E., B.Imam, and J.Merrick, “Assistive technology and older adults”. *In Health Care for People with Intellectual and Developmental Disabilities across the Lifespan*. 2016, , 1465–1471.

Celler, B., E.Ilsar, and W.Earnshaw (1996). “Preliminary results of a pilot project on remote monitoring of functional health status in the home”. *Proceedings of 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 63–64.

Chaqfeh, M. A., H.Ghasemzadeh, O.Amft, J.Hong, N.Berthouze, and A.Helal (2017). “Smart health: A context-aware health paradigm within smart homes”. *IEEE Communications Magazine*, 55(1), 46–53.

Chattopadhyay, A. and M.Maitra (2022). “MRI-based Brain Tumor Image Detection Using CNN based Deep Learning Method”. *Neuroscience Informatics*, 100060.

Chen, H., G.Wang, J.-H.Xue, and L.He (2016). “A novel hierarchical framework for human action recognition”. *Pattern Recognition*, 55, 148–159.

Cheng, L., Y.Guan, K.Zhu, and Y.Li (2017). “Recognition of human activities using machine learning methods with wearable sensors”. *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, 1–7.

Chernbumroong, S., S.Cang, A.Atkins, and H.Yu (2013). “Elderly activities recognition and classification for applications in assisted living”. *Expert Systems with Applications*, 40(5), 1662–1674.

Chuang, K., J.-L.Huang, and M.-S.Chen (2008). “Mining top-K frequent patterns in the presence of the memory constraint”. *Proc. 20th Int. Conf. Very Large Data Bases J.*, 1321–1344.

Cook, A. J., G.Gargiulo, T.Lehmann, and T. J.Hamilton (2015). “Open platform, eight-channel, portable bio-potential and activity data logger for wearable medical device development”. *Electronics Letters*, 51(21), 1641–1643.

Cook, D. J. and S. K.Das (2009). “How smart are our environments? An updated look at the state of the art”. *Pervasive and Mobile Computing*, 5(2), 107–131.

Cook, D. J., M.Schmitter-Edgecombe, A. S.Crandall, C.Sanders, and B. L.Thomas (2012). “Collecting and disseminating smart home sensor data in the CASAS project”. *AI Magazine*, 33(1), 43–56.

---

Cornacchia, M., K.Ozcan, Y.Zheng, and S.Velipasalar (2016). “A survey on activity detection and classification using wearable sensors”. *IEEE Sensors Journal*, 17(2), 386–403.

Davidson, C., T.Rezwana, and M. A.Hoque (2019). “Smart home security application enabled by iot: Using arduino, raspberry pi, nodejs, and mongodb”. *Smart Grid and Internet of Things: Second EAI International Conference, SGIoT 2018, Niagara Falls, ON, Canada, July 11, 2018, Proceedings 2*, 46–56.

Deep, S. and X.Zheng (2019). “Hybrid model featuring CNN and LSTM architecture for human activity recognition on smartphone sensor data”. *2019 20th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, 259–264.

Dempsey, P. G., R. W.McGorry, and W. S.Maynard (2005). “A survey of tools and methods used by certified professional ergonomists”. *Applied ergonomics*, 36(4), 489–503.

Dhiman, C. and D. K.Vishwakarma (2019). “A Robust Framework for Abnormal Human Action Recognition Using R-Transform and Zernike Moments in Depth Videos”. *IEEE Sensors Journal*, 19(13), 5195–5203.

Diethe, T., N.Twomey, M.Kull, P.Flach, and I.Craddock (2017). “Probabilistic sensor fusion for ambient assisted living”. *arXiv preprint arXiv:1702.01209*.

Dua, N., S. N.Singh, and V. B.Semwal (2021). “Multi-input CNN-GRU based human activity recognition using wearable sensors”. *Computing*, 1–18.

Eagle, N. and A.Pentland (2006). “Reality mining: Sensing complex social systems”. *Personal and Ubiquitous Computing*, 10(4), 255–268.

Fahad, L. G. and M.Rajarajan (2015). “Anomalies detection in smart-home activities”. *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, 419–422.

Falah Rad, M. *et al.* (2022). “Probabilistic elderly person’s mood analysis based on its activities of daily living using smart facilities”. *Pattern Analysis and Applications*, 1–14.

Fanti, M. P., G.Faraut, J.-J.Lesage, and M.Rocotelli (2016). “An integrated framework for binary sensor placement and inhabitants location tracking”. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(1), 154–160.

- 
- Farah, J. D., N.Baddour, and E. D.Lemaire (2019). “Design, development, and evaluation of a local sensor-based gait phase recognition system using a logistic model decision tree for orthosis-control”. *Journal of neuroengineering and rehabilitation*, 16(1), 22.
- Fernandez, D., V.Vielzeuf, and J. L.Crowley (2015). “Associative mining for crowd behavior analysis”. *Computer Vision and Image Understanding*, 141, 83–91.
- Feurer, M. and F.Hutter, “Hyperparameter optimization”. In *Automated machine learning*. 2019, , 3–33.
- Filippoupolitis, A., W.Oliff, B.Takand, and G.Loukas (2017a). “Location-enhanced activity recognition in indoor environments using off the shelf smart watch technology and BLE beacons”. *Sensors*, 17(6), 1230.
- Filippoupolitis, A., W.Oliff, B.Takand, and G.Loukas (2017b). “Location-enhanced activity recognition in indoor environments using off the shelf smart watch technology and BLE beacons”. *Sensors*, 17(6), 1230.
- Georgi, M., C.Amma, and T.Schultz (2015). “Recognizing Hand and Finger Gestures with IMU based Motion and EMG based Muscle Activity Sensing.”. *Biosignals*, 99–108.
- Gong, H., K.Xing, and W.Du (2018). “A user activity pattern mining system based on human activity recognition and location service”. *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 1–2.
- Gu, T., Z.Wu, X.Tao, H. K.Pung, and J.Lu (2009). “epsicar: An emerging patterns based approach to sequential, interleaved and concurrent activity recognition”. *2009 IEEE International Conference on Pervasive Computing and Communications*, 1–9.
- Guan, Q., X.Yin, X.Guo, and G.Wang (2016). “A novel infrared motion sensing system for compressive classification of physical activity”. *IEEE Sensors Journal*, 16(8), 2251–2259.
- Guo, J., X.Zhou, Y.Sun, G.Ping, G.Zhao, and Z.Li (2016). “Smartphone-based patients’ activity recognition by using a self-learning scheme for medical monitoring”. *Journal of medical systems*, 40(6), 140.

---

Gutchess, D., N.Checka, and M. S.Snorrason (2007). “Learning patterns of human activity for anomaly detection”. *Intelligent Computing: Theory and Applications V*, 65600Y.

Han, J., J.Pei, and Y.Yin (2000). “Mining frequent patterns without candidate generation”. *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 1–12.

Hassan, M. M., M. Z.Uddin, A.Mohamed, and A.Almogren (2018). “A robust human activity recognition system using smartphone sensors and deep learning”. *Future Generation Computer Systems*, 81, 307–313.

Hassan, M. M., S.Ullah, M. S.Hossain, and A.Alelaiwi (2021). “An end-to-end deep learning model for human activity recognition from highly sparse body sensor data in internet of medical things environment”. *The Journal of Supercomputing*, 77, 2237–2250.

Hassan, S., R.Bahsoon, and R.Kazman (2020). “Microservice transition and its granularity problem: A systematic mapping study”. *Software: Practice and Experience*, 50(9), 1651–1681.

Hayashi, T., M.Nishida, N.Kitaoka, and K.Takeda (2015). “Daily activity recognition based on DNN using environmental sound and acceleration signals”. *2015 23rd European Signal Processing Conference (EUSIPCO)*, 2306–2310.

Heierman, E. O. and D. J.Cook (2003). “Improving home automation by discovering regularly occurring device usage patterns”. *Third IEEE International conference on data mining*, 537–540.

Hong, J.-H., J.Ramos, and A. K.Dey (2015). “Toward personalized activity recognition systems with a semipopulation approach”. *IEEE Transactions on Human-Machine Systems*, 46(1), 101–112.

Jalal, A., S.Kamal, and D.Kim (2017). “A Depth Video-based Human Detection and Activity Recognition using Multi-features and Embedded Hidden Markov Models for Health Care Monitoring Systems.”. *International Journal of Interactive Multimedia & Artificial Intelligence*, 4(4).

Janarthanan, R., S.Doss, and S.Baskar (2020). “Optimized unsupervised deep learning assisted reconstructed coder in the on-nodule wearable sensor for human activity recognition”. *Measurement*, 164, 108050.

- 
- Kang, D.-H., M.-S.Park, H.-S.Kim, D.-y.Kim, S.-H.Kim, H.-J.Son, and S.-G.Lee (2017). “Room temperature control and fire alarm/suppression IoT service using MQTT on AWS”. *2017 International Conference on Platform Technology and Service (PlatCon)*, 1–5.
- Kashimoto, Y., K.Hata, H.Suwa, M.Fujimoto, Y.Arakawa, T.Shigezumi, K.Komiya, K.Konishi, and K.Yasumoto (2016). “Low-cost and device-free activity recognition system with energy harvesting PIR and door sensors”. *Adjunct Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing Networking and Services*, 6–11.
- Kashyap, B., C. K.Sestok, A. G.Dabak, S.Ramaswamy, and R.Kumar (2019). “Ultra-precision liquid level sensing using impedance spectroscopy and data analytics”. *IEEE Sensors Journal*, 19(20), 9468–9478.
- Keogh, E. and S.Kasetty (2003). “On the need for time series data mining benchmarks: A survey and empirical demonstration”. *Data Mining and Knowledge Discovery*, 7(4), 349–371.
- Khan, S., S.Nazir, and H. U.Khan (2021). “Smart object detection and home appliances control system in smart cities”.
- Khan, Z. A. and W.Sohn (2011). “Abnormal human activity recognition system based on R-transform and kernel discriminant technique for elderly home care”. *IEEE Transactions on Consumer Electronics*, 57(4), 1843–1850.
- Khowaja, S. A., A. G.Prabono, F.Setiawan, B. N.Yahya, and S.-L.Lee (2018). “Contextual activity based Healthcare Internet of Things, Services, and People (HIoTSP): An architectural framework for healthcare monitoring using wearable sensors”. *Computer Networks*, 145, 190–206.
- Khowaja, S. A., B. N.Yahya, and S.-L.Lee (2020). “CAPHAR: context-aware personalized human activity recognition using associative learning in smart environments”. *Human-centric Computing and Information Sciences*, 10(1), 1–35.
- Kim, E. (2020). “Interpretable and accurate convolutional neural networks for human activity recognition”. *IEEE Transactions on Industrial Informatics*, 16(11), 7190–7198.
- Kinney, J. M., C. S.Kart, L. D.Murdoch, and C. J.Conley (2004). “Striving to provide safety assistance for families of elders: the SAFE house project”. *Dementia*, 3(3), 351–370.

- 
- Kiukkonen, N., J.Blom, O.Dousse, G.-P.Daniel, and J.Laurila (2010). “Towards rich mobile phone Datasets: Lausanne data collection campaign”. *Proceedings of the ACM International Conference on Pervasive Services*.
- Kuhlmann, T., P.Garaizar, and U.-D.Reips (2021). “Smartphone sensor accuracy varies from device to device in mobile research: The case of spatial orientation”. *Behavior research methods*, 53, 22–33.
- Kwapisz, J. R., G. M.Weiss, and S. A.Moore (2011a). “Activity recognition using cell phone accelerometers”. *ACM SigKDD Explorations Newsletter*, 12(2), 74–82.
- Kwapisz, J. R., G. M.Weiss, and S. A.Moore (2011b). “Activity recognition using cell phone accelerometers”. *ACM SigKDD Explorations Newsletter*, 12(2), 74–82.
- Laudanski, A., B.Brouwer, and Q.Li (2015). “Activity classification in persons with stroke based on frequency features”. *Medical engineering & physics*, 37(2), 180–186.
- Levinas, D., G.Perelman, and A.Ostfeld (2021). “Water leak localization using high-resolution pressure sensors”. *Water*, 13(5), 591.
- Li, Y., W.Li, V.Mahadevan, and N.Vasconcelos (2016). “Vlad3: Encoding dynamics of deep features for action recognition”. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1951–1960.
- Liu, A.-A., N.Xu, W.-Z.Nie, Y.-T.Su, Y.Wong, and M.Kankanhalli (2016). “Benchmarking a multimodal and multiview and interactive dataset for human action recognition”. *IEEE Transactions on cybernetics*, 47(7), 1781–1794.
- Liu, R., A. A.Ramli, H.Zhang, E.Henricson, and X.Liu (2021). “An overview of human activity recognition using wearable sensors: Healthcare and artificial intelligence”. *International Conference on Internet of Things*, 1–14.
- Lorussi, F., N.Carbonaro, D.De Rossi, R.Paradiso, P.Veltink, and A.Tognetti (2016). “Wearable textile platform for assessing stroke patient treatment in daily life conditions”. *Frontiers in bioengineering and biotechnology*, 4, 28.
- Lotfi, A., C.Langensiepen, S. M.Mahmoud, and M. J.Akhlaghinia (2012). “Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour”. *Journal of ambient intelligence and humanized computing*, 3, 205–218.

- 
- Lühr, S., G.West, and S.Venkatesh (2007). “Recognition of emergent human behaviour in a smart home: A data mining approach”. *Pervasive and Mobile Computing*, 3(2), 95–116.
- Luo, X., Q.Guan, H.Tan, L.Gao, Z.Wang, and X.Luo (2017). “Simultaneous indoor tracking and activity recognition using pyroelectric infrared sensors”. *Sensors*, 17(8), 1738.
- Ma, H., H.Cao, Q.Yang, E.Chen, and J.Tian (2012a). “A habit mining approach for discovering similar mobile users”. *Proc. 21st Int. Conf. World Wide Web*, 231–240.
- Ma, H., H.Cao, Q.Yang, E.Chen, and J.Tian (2012b). “A habit mining approach for discovering similar mobile users”. *Proc. 21st Int. Conf. World Wide Web*, 231–240.
- Ma, R., F.Hu, and Q.Hao (2016). “Active compressive sensing via pyroelectric infrared sensor for human situation recognition”. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(12), 3340–3350.
- Mehr, H. D. and H.Polat (2019). “Human activity recognition in smart home with deep learning approach”. *2019 7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, 149–153.
- Mehr, H. D., H.Polat, and A.Cetin (2016). “Resident activity recognition in smart homes by using artificial neural networks”. *2016 4th international istanbul smart grid congress and fair (ICSG)*, 1–5.
- Mekruksavanich, S. and A.Jitpattanakul (2021). “Lstm networks using smart-phone data for sensor-based human activity recognition in smart homes”. *Sensors*, 21(5), 1636.
- Ministry of Statistics, G. o. I. (2016). “Elderly In India”. *National Sample Survey Office*, 1–95.
- Mo, L., F.Li, Y.Zhu, and A.Huang (2016). “Human physical activity recognition based on computer vision with deep learning model”. *2016 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, 1–6.
- Mohamed, A. B. H., T.Val, L.Andrieux, and A.Kachouri (2012). “Using a Kinect WSN for home monitoring: principle, network and application evaluation”. *2012*

---

*International Conference on Wireless Communications in Underground and Confined Areas*, 1–5.

Montalto, F., C.Guerra, V.Bianchi, I.De Munari, and P.Ciampolini (2015). “MuSA: Wearable multi sensor assistant for human activity recognition and indoor localization”. *Ambient Assisted Living: Italian Forum 2014*, 81–92.

Mörchen, F. and A.Ultsch (2007). “Efficient mining of understandable patterns from multivariate interval time series”. *Data Mining and Knowledge Discovery*, 15(2), 181–215.

Nakamura, M., J.Nakamura, M.Shuzo, S.Warisawa, and I.Yamada (2010). “Collaborative processing of wearable and ambient sensor system for health monitoring application”. *2010 3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL 2010)*, 1–5.

Nandy, A., J.Saha, and C.Chowdhury (2020). “Novel features for intensive human activity recognition based on wearable and smartphone sensors”. *Microsystem Technologies*, 1–15.

Nazir, S. and M.Kaleem (2019). “Reliable image notifications for smart home security with MQTT”. *2019 International Conference on Information Science and Communication Technology (ICISCT)*, 1–5.

Newman, D. B., J. B.Nezlek, and T. M.Thrash (2023). “The dynamics of prayer in daily life and implications for well-being”. *Journal of Personality and Social Psychology*.

Ni, Q., A. B.Garcia Hernando, D.laCruz, and I.Pau (2015). “The elderly’s independent living in smart homes: A characterization of activities and sensing infrastructure survey to facilitate services development”. *Sensors*, 15(5), 11312–11362.

Pansiot, J., D.Stoyanov, D.McIlwraith, B. P.Lo, and G.-Z.Yang (2007). “Ambient and wearable sensor fusion for activity recognition in healthcare monitoring systems”. *4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007) March 26–28, 2007 RWTH Aachen University, Germany*, 208–212.

Pham, M., D.Yang, and W.Sheng (2018). “A sensor fusion approach to indoor human localization based on environmental and wearable sensors”. *IEEE Transactions on Automation Science and Engineering*, 16(1), 339–350.

- 
- Phillips, L. J., C. B.DeRoche, M.Rantz, G. L.Alexander, M.Skubic, L.Despins, C.Abbott, B. H.Harris, C.Galambos, and R. J.Koopman (2017). “Using embedded sensors in independent living to predict gait changes and falls”. *Western journal of nursing research*, 39(1), 78–94.
- Poh, S.-C., Y.-F.Tan, X.Guo, S.-N.Cheong, C.-P.Ooi, and W.-H.Tan (2019). “LSTM and HMM Comparison for Home Activity Anomaly Detection”. *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, 1564–1568.
- Potluri, S. *et al.* (2019). “Machine learning based human gait segmentation with wearable sensor platform”. *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*.
- Qian, H., Y.Mao, W.Xiang, and Z.Wang (2010). “Recognition of human activities using SVM multi-class classifier”. *Pattern Recognition Letters*, 31(2), 100–111.
- Ramanujam, E., T.Perumal, and S.Padmavathi (2021). “Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review”. *IEEE Sensors Journal*, 21(12), 13029–13040.
- Rantz, M., M.Skubic, S.Miller, and J.Krampe (2008). “Using technology to enhance aging in place”. *International Conference On Smart homes and health Telematics*, 169–176.
- Rashidi, P., D. J.Cook, L. B.Holder, and M.Schmitter-Edgecombe (2010). “Discovering activities to recognize and track in a smart environment”. *IEEE transactions on knowledge and data engineering*, 23(4), 527–539.
- Rashidi, P. and A.Mihailidis (2012). “A survey on ambient-assisted living tools for older adults”. *IEEE journal of biomedical and health informatics*, 17(3), 579–590.
- Rawassizadeh, R., E.Momeni, C.Dobbins, J.Gharibshah, and M.Pazzani (2016*a*). “Scalable daily human behavioral pattern mining from multivariate temporal data”. *IEEE Transactions on Knowledge and Data Engineering*, 28(11), 3098–3112.
- Rawassizadeh, R. and A.Tjoa (2010). “Securing shareable life-logs”. *Proceedings of the IEEE 2nd International Conference on Social Computing*, 1105–1110.

---

Rawassizadeh, R., M.Tomitsch, K.Wac, and A. M.Tjoa (2013). “Ubiqlog: A generic mobile phone-based life-log framework”. *Personal Ubiquitous Computing*, 17(4), 621–637.

Rawassizadeh, R. *et al.* (2016b). “Scalable daily human behavioral pattern mining from multivariate temporal data”. *IEEE Transactions on Knowledge and Data Engineering*, 28(11), 3098–3112.

Reiss, A. and D.Stricker (2012). “Introducing a new benchmarked dataset for activity monitoring”. *2012 16th international symposium on wearable computers*, 108–109.

Riboni, D. and C.Bettini (2011). “COSAR: hybrid reasoning for context-aware activity recognition”. *Personal and Ubiquitous Computing*, 15, 271–289.

Roggen, C. A. N.-D. L.-V. C. R., Daniel and Sagha (2012). “OPPORTUNITY Activity Recognition”. UCI Machine Learning Repository.

Roy, N., A.Misra, and D.Cook (2016). “Ambient and smartphone sensor assisted ADL recognition in multi-inhabitant smart environments”. *Journal of ambient intelligence and humanized computing*, 7(1), 1–19.

Salih, A. and A.Abraham (2013). “A review of ambient intelligence assisted health-care monitoring”. *International Journal of Computer Information Systems and Industrial Management (IJCISIM)*, 5, 741–750.

Schenato, L., J. P.Aguilar-Lopez, A.Galtarossa, A.Pasuto, T.Bogaard, and L.Palmieri (2021). “A rugged FBG-based pressure sensor for water level monitoring in dikes”. *IEEE Sensors Journal*, 21(12), 13263–13271.

Schuldt, C., I.Laptev, and B.Caputo (2004). “Recognizing human actions: a local SVM approach”. *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 32–36.

Shankar, K., Y.Zhang, Y.Liu, L.Wu, and C.-H.Chen (2020). “Hyperparameter tuning deep learning for diabetic retinopathy fundus image classification”. *IEEE Access*, 8, 118164–118173.

Sharma, B., V. K.Sharma, and S.Kumar, “Sigmoidal spider monkey optimization algorithm”. *In Soft Computing: Theories and Applications*. 2020, , 109–117.

- 
- Shi, Y., Y.Tian, Y.Wang, and T.Huang (2017). “Sequential deep trajectory descriptor for action recognition with three-stream CNN”. *IEEE Transactions on Multimedia*, 19(7), 1510–1520.
- Shin, S., D. S.Lee, and W. S.Lee (2014). “CP-tree: An adaptive synopsis structure for compressing frequent itemsets over online data streams”. *Inf. Sci.*, 278, 559–576.
- Siddiqi, M. H., N.Almashfi, A.Ali, M.Alruwaili, Y.Alhwaiti, S.Alanazi, and M.Kamruzzaman (2021). “A unified approach for patient Activity recognition in healthcare using depth camera”. *IEEE Access*, 9, 92300–92317.
- Stikic, M., K.Van Laerhoven, and B.Schiele (2008). “Exploring semi-supervised and active learning for activity recognition”. *2008 12th IEEE International Symposium on Wearable Computers*, 81–88.
- Stisen, A., H.Blunck, S.Bhattacharya, T. S.Prentow, M. B.Kjærgaard, A. K.Dey, *et al.* (2015). “Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition”. *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 127–140.
- Subasi, A., K.Khateeb, T.Brahimi, and A.Sarirete, “Human activity recognition using machine learning methods in a smart healthcare environment”. *In Innovation in health informatics*. 2020, , 123–144.
- Sztyler, T., H.Stuckenschmidt, and W.Petrich (2017). “Position-aware activity recognition with wearable devices”. *Pervasive and mobile computing*, 38, 281–295.
- Tanberk, S., Z. H.Kilimci, D. B.Tükel, M.Uysal, and S.Akyokuş (2020). “A hybrid deep model using deep learning and dense optical flow approaches for human activity recognition”. *IEEE Access*, 8, 19799–19809.
- Tapia, E. M., S. S.Intille, W.Haskell, K.Larson, J.Wright, A.King, ..., and R. H.Friedman (2007). “Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor”. *Proceedings of the 4th international conference on embedded networked sensor systems*, 1–12.
- Teng, M.-C., J.-P.Chen, T.-H.Lin, P.Huang, C.-L.Huang, C.-H.Chen, and H.-S.Chen (2005). “Emergency alarm system: Prototype and experience”. *Proceedings of 7th International Workshop on Enterprise networking and Computing in Healthcare Industry, 2005. HEALTHCOM 2005.*, 73–76.

---

Terada, T. and K.Tanaka (2010). “A framework for constructing entertainment contents using flash and wearable sensors”. *International Conference on Entertainment Computing*, 334–341.

Tunca, C., H.Alemdar, H.Ertan, O.Incel, and C.Ersoy (2014). “Multimodal wireless sensor network-based ambient assisted living in real homes with multiple residents”. *Sensors*, 14(6), 9692–9719.

Uddin, M., W.Khaksar, J.Torresen, *et al.* (2018). “Ambient sensors for elderly care and independent living: A survey”. *Sensors*, 18(7), 2027.

Vaizman, Y., K.Ellis, and G.Lanckriet (2017). “Recognizing detailed human context in the wild from smartphones and smartwatches”. *IEEE pervasive computing*, 16(4), 62–74.

Vanahalli, M. K. and N.Patil (2022). “An efficient colossal closed itemset mining algorithm for a dataset with high dimensionality”. *Journal of King Saud University-Computer and Information Sciences*, 34(6), 2798–2808.

Wagner, D., A.Rice, and A. R.Beresford (2014). “Device analyzer: Large-scale mobile data collection”. *SIGMETRICS Performance Evaluation Review*, 41(4), 53–56.

Wang, S.-W., C.-C.Chen, C.-M.Wu, and C.-M.Huang (2018). “A continuous water-level sensor based on load cell and floating pipe”. *2018 IEEE International Conference on Applied System Invention (ICASI)*, 151–154.

Wang, W. and C.Miao (2018). “Activity Recognition in New Smart Home Environments”. *Proceedings of the 3rd International Workshop on Multimedia for Personal Health and Health Care*, 29–37.

Wang, Y., S.Cang, and H.Yu (2019). “A survey on wearable sensor modality centred human activity recognition in health care”. *Expert Systems with Applications*.

Wen, J., M.Zhong, and Z.Wang (2015). “Activity recognition with weighted frequent patterns mining in smart environments”. *Expert Systems with Applications*, 42(17-18), 6423–6432.

Wood, W., L.Tam, and M. G.Witt (2005). “Changing circumstances, disrupting habits.”. *Journal of personality and social psychology*, 88(6), 918.

- 
- Yang, B. and M.Zhang (2016). “Credit-based multiple human location for passive binary pyroelectric infrared sensor tracking system: Free from region partition and classifier”. *IEEE Sensors Journal*, 17(1), 37–45.
- Ye, Y., Y.Zheng, Y.Chen, J.Feng, and X.Xie (2009). “Mining individual life pattern based on location history”. *Proc. 10th Int. Conf. Mobile Data Manage.: Syst. Serv. Middleware*, 1–10.
- Yin, J., Q.Yang, and J. J.Pan (2008). “Sensor-based abnormal human-activity detection”. *IEEE Transactions on Knowledge and Data Engineering*, 20(8), 1082–1090.
- Yoo, H., H. J.Lee, and Y. S.Lim (2014). “Mining frequent activity patterns of elderly people from smart home sensor data”. *Sensors*, 14(10), 18620–18640.
- Zaidi, S., B.Jagadeesh, K.Sudheesh, and A. A.Audre (2017). “Video Anomaly Detection and Classification for Human Activity Recognition”. *2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC)*, 544–548.
- Zela, A., A.Klein, S.Falkner, and F.Hutter (2018). “Towards automated deep learning: Efficient joint neural architecture and hyperparameter search”. *arXiv preprint arXiv:1807.06906*.
- Zhang, L., X.Wu, and D.Luo (2015). “Recognizing human activities from raw accelerometer data using deep neural networks”. *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, 865–870.
- Zhang, S., Z.Weil, J.Nie, L.Huang, S.Wang, and Z.Li (2017). “A review on human activity recognition using vision-based method”. *Journal of healthcare engineering*, 2017.
- Zhang, W., L.Qin, W.Zhong, X.Guo, and G.Wang (2019). “Framework of sequence chunking for human activity recognition using wearables”. *Proceedings of the 2019 International Conference on Image, Video and Signal Processing*, 93–98.
- Zhihao, J.Chen, and S. C.Hoi (2020). “Deep learning for image super-resolution: A survey”. *IEEE transactions on pattern analysis and machine intelligence*, 43(10), 3365–3387.

---

Zhou, X., W.Liang, I.Kevin, K.Wang, H.Wang, L. T.Yang, and Q.Jin (2020). “Deep-learning-enhanced human activity recognition for Internet of healthcare things”. *IEEE Internet of Things Journal*, 7(7), 6429–6438.

Zhu, W., C.Lan, J.Xing, W.Zeng, Y.Li, L.Shen, and X.Xie (2016). “Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks”. *Thirtieth AAAI Conference on Artificial Intelligence*.

Zouba, N., F.Bremond, and M.Thonnat (2009). “Multisensor fusion for monitoring elderly activities at home”. *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, 98–103.

## Bio-data

**Name:** Ranjit Kolkar

**Address:** Research Scholar  
Veer-Seeta Niwas, Yadav Colony,  
Udyambag, Belgaum 590008,  
Karnataka, India

**Email:** ranjit.kolkar@gmail.com

**Mobile No:** +91 8618879217

**Qualification:** Ph.D. in Information Technology,  
Department of Information Technology,  
National Institute of Technology Karnataka,  
Surathkal, Mangalore, India

M.Tech. in Computer Networks Engineering,  
Department of Information Technology,  
National Institute of Technology Karnataka,  
Surathkal, Mangalore, India

B.E. in Information Science Engineering,  
Department of Information Science Engineering,  
GIT, Belgaum, Karnataka

**Research Areas:** Human Activity Recognition, Human Behaviour  
Pattern Analysis, Internet of Things, Sensors