Activity Recognition and Resident Identification in Smart Home Environment

A dissertation presented by

Venkatesh Subramanya Kashyap

To
The School of Computing & Information Technology

In Partial fulfillment of the requirements
For the Degree of

Master of Computing

Unitec Institute of Technology
Auckland, New Zealand
Febrauary 2020

Activity Recognition and Resident Identification in Smart Home Environment

Advisors

Principal	Associate	Associate
Dr. Bashar Barmada	Dr. Guillermo Ramirez Prado	Dr. Veronica Leisaputra

Abstract

World's population is ageing rapidly. There have been various efforts to improve the quality of life for elderly. Ambient assisted living is one possible solution which enables elderly or disabled people to live a better lifestyle. Currently there are smart home systems that utilize a wide range of sensors to predict our everyday activities. However, research into activity recognition and resident identification using ultrasonic sensors are limited.

This work introduces machine learning techniques with ultrasonic sensors to predict the activities of one and two person in the smart home environment. The proposed system is capable of recognising the activities and identifying the residents without the need to manually label the prior activities. Our evaluation demonstartes that the proposed approach can predict resident's activities with high accuracy. The trained model could be used to predict other resident's activities and also identify resident's from each other.

This research enables the smart home system to be widely adopted in people's houses with minimal training and also enable people who need support, to live independently with less interference from caregivers which in turn enables caregivers to manage more people at the same time.

Publications

Resilient Activities Tracking in a Smart Home using Ultrasonic Sensors.

in *IEEE International Conference on Big Data, Los Angeles, United States* - by Kashyap Venkatesh, Bashar Barmada, Guillermo Ramirez-Prado & Veronica Liesaputra (December 2019).

Robust Features for Activities Recognition.

in *IEEE International Conference on Big Data, Los Angeles, United States* - by Kashyap Venkatesh, Bashar Barmada, Guillermo Ramirez-Prado & Veronica Liesaputra (December 2019).

Ongoing

- **Multi-Resident Activity Recognition in a smart Home using Ultrasonic Sensors**
- **Resident Identification using Ultrasonic Sensors**

Contents

List of Figures

List of Tables

For My Mother.

Acknowledgements

To my advisor, Bashar Barmada – for his guidance, faith in my abilities, and for the opportunity to work on this project,

To my co-advisors, Veronica and Guillermo - for invaluable insights and assistance,

Special thanks to Hamid and Iman, for there complete support as acamedic leaders from Computing Department.

To my parents, Suma and Shivaswamy – for investing on my education and supporting throughout the journey.

Finally, I would like to thank all my friends for there support and encouragement.

Thank you.

Chapter 1

Introduction

World governments are facing a significant financial burden in health system expenditure due to the increase in the aging population and chronic disease, it is expected to be worse in the coming future [1]. As per New Zealand statistics 2016, there are about 711,200 people who are aged above 65, there are about 169,000 people who are aged over 80 and 5,800 people aged over 95+ residing in New Zeland. According to a survey in 1997, around 81% of people 65 years and older own their own home and are mortgage free, and staying in their own home is considered to be their most cost effective living option. By 2026 it is expected that over 23 percent of the population will be aged over 65 compared to 14 percent in 2006 [2]. In addition, the percentage of population living alone in New Zealand will reach 11% by 2038 [3].

The advancement in technology can support people who need support to live independently in their own house using a smart home platform [4][5]. The "smart home" concept came to light in the 1980's [6] and identified as the technology which can be used to support people in many different areas such as healthcare, comfort, and security. The smart home should be capable of tracking different activities of a resident in the house by using different types of sensors as shown in figure 1.1 and notify the caregiver if there are any emergencies and make sure that the resident is safe. Everyday activities that a normal person performs can be either stationary or continuous motion with time stamps associated with them [7][8][9]. Applications based on human activity recognition are increasing rapidly due to highly efficient low-cost detecting systems [10] and growth of pervasive computing technologies [11]. Falls in the elderly occur one in three people of which most of them end up in the hospital and suffer from a hip fracture [12]. One can identify the behaviour of a person based on the activity performed over time and detect if there are any emergency.

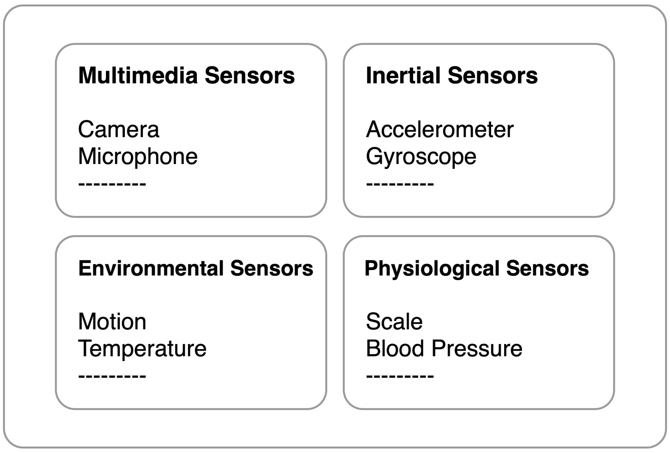

Figure 1.1: Types of sensors used for Tracking location and Activity Recognition

Figure 1.1 shows different types of sensors which can be used to track and recognise the activity of a person in a smart home environment. Multimedia sensors use audio and video to monitor the activities of an individual. Attaching low-cost microphone sensors at different locations of the house such as the sink, toilets, showers, etc. one can track and monitor the person by analyzing the water usage [13]. Due to privacy concerns, a large number of individuals are unfavourable to embedding multimedia sensors in their personal spaces [14]. Vision-based activity recognition provides high accuracy results, but they are expensive and require high computational power to process the captured images or videos [15]. Studies have also shown that if an individual realizes that if he/she is being monitored, then their conduct will change, a phenomenon known as the Hawthorne effect [16] [17]. Environmental sensors such as humidity, temperature, power, etc. can be used to monitor a person in a house by recognizing the changes in the environment but cannot solely be used to identify the person's activities [18]. Wearable sensors such as gyroscopes, accelerometers, blood pressure, skin temperature, etc can be used to track the different activities of a person and monitor their status. The problem with such sensors is that they are impractical solutions as an individual might decide or forget to wear them [19] [20]. Non-intrusive sensors can be embedded in the smart home to identify the location of the occupant and monitor their activities. Examples of such sensors are water flow sensors to measure water consumption, electrical current flow sensors to detect when people turn on a room light or electrical appliances, and ultrasonic sensors to detect the distance of objects of interest. PIR (Pyroelectric Infrared) sensors can also be used for tracking and activity recognition [21]. The main problem with PIR sensors is that they are insensitive to slow motion and have narrower sensor field view when the body is in a standing position.

1.0.1 Proposed System

The proposed system in this work uses Ultrasonic sensors to identify the activty of a person by measuring the distance from the ceiling to the person. Using machine learning classifiers on the distance readings, on/off state and duration, the system can predict the types of activities the house occupant is performing. We Identify the resident of the house from other people based on the actvities performed by the resident. It is possible to have multiple people in the house. In this work, we consider single resident and multi resident (two people) at the same time in the test environment to predict the activities of the resident. By using machine learning algorithms the system is trained with the occupant's activities for one day and then manages to predict the activities of the same occupant on other days with great confidence, and predicts the activities of a different person on a different day with good confidence. Based on the activities performed, the system is trained with who the occupant is to identify the resident of the house from other person. We choose to use Ultrasonic sensors as the method of collecting data to preserve the privacy of the occupant as there are no image or video recordings. Also, in a real environment, these sensors can be hidden and integrated easily with the features of the house, so the occupant will not feel that they are being monitored and their behaviour will not change.

Contribution

We believe that this work has the following research contribution:

- In this research we identified that several researcher's used ultrasonic sensors for tracking location and activity recognition using only distance measurement as the feature. In this work we introduce different features such as sensor ON/OFF state and sensor duration identify which are the robust feature for activity recognition.

- Data labelling is tedious and time consuming. To address this issue, we trained our machine learning model with one resident and tested it on other residents and managed to obtain good accuracy results.

- The proposed system is capable of identifying residents by solely using ultrasonic sensors.

Used Technology

Hardware

- Ultrasonic sensors - Maxbotix MV MaxSonar EZ0
- Arduino Mega 2560 Microcontroller
- Computer with windows 10 installed
- Breadboard

Software

- Arduino IDE 1.8.11
- Python 3.7.2
- Microsoft Excel 2018
- Weka 3.9.3

The rest of the dissertation is organised as follows:

- In chapter 2, we review the ambient sensors used for tracking location, Activity Recognition, Behaviour Identification and Identifying a person, with special focus on machine learning approaches used with ultrasonic sensors to predict the human activities.
- Chapter 3, introduces the environment setup for collecting the data, filtering the acquired data, labelling the dataset and finally processing the dataset to train and evaluate the model.
- Chapter 4, presents the core of our work - Building different models and evaluating them to find robust features and classifiers for activity recognition.
- In Chapter 5, we label the data with resident name and identify the resident from others based on the activities such as cooking, eating, standing, laying on couch and walking around.
- Finally, in chapter 6 we summarise the ideas of this thesis and point a direction for future work.

Chapter 2

Related Work

To Identify a person from we need to know (i) who the person is (ii) where the person is (iii) what are his/her daily activities (iv) what is the behaviour of the person. By answering these questions one can identify the person and distinguish from others. In this chapter, we discuss the previous research conducted on (i) Tracking location (ii) Activity Recognition (iii) Behaviour Identification and (iv) Identifying people.

2.1 Tracking Location

Ultrasonic systems are popular for locating a person in an indoor environment [22] [23] [24] [25]. Priyantha [22] proposed a location-support system that combined radio and ultrasonic signals to decide on the location of a mobile device (called Cricket) inside a building. They managed to obtain high accuracy location measurements with less than 10 centimetres error. The problem with such system is, since there were many moving objects in the environment, if an object blocks the line of sight (LoS) signal for the Cricket, the system fails to give the accurate distance measurements. Another drawback is that, the user must carry their mobile device with them all the time. Similar systems can be found [23], which rely on Bluetooth and Ultrasonic sensors to decide on the location of a mobile in a room. Beside the complexity, the user must carry their mobile device all the time. [25] proposed an algorithm to track the ultrasonic signals in order to overcome the limitation of the line of sight signal being blocked. The system could also estimate the direction of the target and could achieve accuracy as low as 35cm for the location, even with frequent cut offs of the LoS signals.

Another interesting tracking system was proposed in [24] where they used an array of 6x6 ultrasonic sensors mounted to the ceiling to cover around 4×4 m^2

room. The aim was to track a child moving towards his mother and away from a stranger by using only the distance reading from the sensors and applying Kalman filter for tracking. The mean square error for locating the target was 1.3m^2 .[7] proposed a tracking system by mounting ultrasonic sensors on several doorways inside a house to measure the height of a passing person. The readings from these doorways were fed to an algorithm to predict the direction of walking for that person and to distinguish between two different people with 90% accuracy. However, the assumption here is that the person was walking in the most natural way possible and with an acceptable constant speed. Another major assumption was that the people under the experiments were of different heights.

2.2 Activity Recognition

Ultrasonic sensors can be used to recognize different activities in an indoor environment. Many researcher's concentrate on how and where to position the sensors in a room, how many sensors are needed to cover the area of interest and what are the environmental factors that might affect the accuracy or readings and so on. In [8] an array of sensors was used to detect whether a person in a room had fallen down. The proposed concept was tested in a 60cm $\times$ 60cm box equipped with 20 sensors on the ceiling and 20 sensors around the wall, which were arranged in octagonal shape. The human figure used for the testing was 30cm tall. Under such conditions, the system claimed to give 94% accuracy. The system was not tested under real conditions and not to mention the large number of sensors used in a such small area.

Nadee [26] used an array of ultrasonic sensors on the top and side of the room of size 30x30cm which consists of 16 ultrasonic sensors on each side. The proposed system consists of two parts namely hardware and fall detection system. The system was tested with a human model of 20cm tall and 2.5cm wide. The change in distance measurement from the sensors placed in top and side walls are used to recognise various activities such as sitting, standing and falling. Sensitivity of the sound depends on the temperature and pressure of the surrounding environment. The sensitivity of sound is said to increase by 1% when the temperature is increased by 6 degree celsius. Hence constant temperature has to be maintained in the indoor environment to reduce the noise from ultrasonic sensors . A digital filter was used to remove the noise caused by ultrasonic sensors due to the variation of temperature in the room and the system was able to achieve an accuracy of 92%.

Using ultrasonic sensors to detect the activity of an individual is privacy preserving approach. The key idea is, a sound wave is emitted from the sensor and the sound wave is obstructed from an individual which reflects back to the sensor. The time taken by the sound wave to reflect back is accounted and the distance is measured to predict the activities. Biswas [27] perform activity analysis using the distance obtained from first-reflection echolocation i.e the distance is computed to the first reflected object only and all the other reflected signals are ignored. This leads to simple time series signal and low computational power to process the signal. Data was collected from six individuals performing sitting, standing and walking activity in which the EAD (Echolocation based Activity Detector) unit was placed approximately 3 to 10 feet. The time series signal between the subject and EAD was collected and various machine learning algorithms such as SVM, Naiive Bayes and Sequential Minimal Optimizer(SMO) were applied on the data to differentiate between sit, stand and walk actvity. All the classfiers were able to achieve more than 80% accuary. A more realistic system can be found in [4] where the proposed system used 8 ultrasonic sensors mounted to the bathroom ceiling to detect whether an elder person has fallen down based on the sensor distance readings and for how long the sensor captured this reading. Once a fall is detected, the system send a tweet and an email to the caregivers account. However, they did not show any results on how accurate the system was. A combination of PIR sensors and ultrasonic sensors were used in [9] to detect the walking activity of a person regarding the direction and speed, classified as slow, normal and fast. The system used 2 PIR and 8 ultrasonic sensors with Bluetooth device as a communication module.

With advancement in sensor technology and persavise-computing human activity recognition using sensor data has increased research interest. The task is not just simple but also complex because the activities performed by an individual in real life can be one after the other or multi-tasking. Activity recognition can be seen as a classification problem where several machine learning algorithms can be applied to build a model. Most of the research work covers predicting sequential activities using different sensors [28] [29] [30] but there is a lot of work to be done to address complex issues in interleaved and concurrent activities. Tao. Gu [5] proposed a pattern minining approach for sensor based data to recognise both simple and complex activities. A wireless platform was built to obtain four kind of data from sensors such as target location, movement, environmetal information and object interaction. Data was collected in a smart home by four volunteers performing the activities as normally as there everday life. 26 sequential activities such as using phone, eating meal, making tea etc , 15 interleaved activities and 16 concurrent activities such as using phone and eating meal, listening to music while reading book etc. were recorded. The data collected were trained and tested using different machine algorithms with 10 fold cross validation. epSICAR algorithm a custom made algorithm achieved more than 80% accuracy in recognising sequential, interleaved and concureent activities compared to Naiive Bayes and Hidden Markov Model(HMM).

Three machine learning algorithms, such as support vector machine (SVM) with linear kernel, k-nearest neighbors (KNN) and decision tree, were used in [31] to detect the movement of a person and to classify the activities of a person into three basic postures; sitting, standing and falling. The system prototype consisted of 5 ultrasonic sensors distributed on 55cm $\times$ 55cm board, which was mounted on the ceiling. The system utilized the distance of a person from the sensors in this small area to determine the person's activity, and the sequence of the sensors being activated to decide on the movement. The accuracy of the results was claimed to be between 81% and 90% depending on the complexity of the movement, with the decision tree giving the highest results followed by SVM and KNN. [32] comparing the performance of 3 classifiers, namely, PART, Lazy IBK and Lazy KStar, to predict different activities of a person based on the sensor readings: standing, lying, sitting, walking and running. It utlized the time factor to further classify these activities into normal and abnormal behaviours.

Tim Van [33] uses a radio frequency module wireless network and bluetooth headset to detect the activities. An individual who lives in a three bedroom apartment was considered for data collection. Sensors were placed in different locations of the house such as doors, refrigerators, cupboard and toilet flush. The data was manually annoted by the individual while performing the activities using a bluetooth headset with speech recognition software connected to the wireless network. Seven different activities were chosen based on Katz index [34] a tool used to determine the physical capabilities of elderly people. The activities chosen were "Preparing Breakfast", "Preparing Dinner", "Leave house", "Showering", "Sleeping", "Preparing Beverage" and "Idle". The data was collected for over 28 days and temporal probalistic model such as HMM and Conditional Random Field's (CRF) were applied to compare the performance of the model. By comparing the confusion matrix for both the probabilistic models it was seen that CRF provided higher accuracy for "Idle" state and "HMM" provided significantly higher accuracy for all other activities. An individual can perform different types of activities in everyday life which can be stationary or dynamic. Wearable gadgets are characterized as smaller than usual electronic sensor-based gadgets that are worn by the individual under, with or over apparel [35]. Nandy [36] proposed to use smartphone embedded sensors such as accelerometer and heart rate sensor to determine six static and dynamic activities such as walking, walking with weight, sitting, standing etc. Accelerometer sensor data was collected when the smartphone is placed in the pocket and heart rate sensor data is collected by attaching the smartphone to the target's chest. The data collected are fused to obtain meaningful information. Average age of 22 years with average weight of 64 kgs was chosen to perform the activities and collect the data. Heart rate was the key feature used to distinguish between "standing" and "standing with weight" activity. Several classifers such as decision tree, K-nearest neighbour, bagged trees, Linear Regression, Multi layer Perceptron,Gaussian Naiive Bayes, Support Vector Machine was applied using 10 fold cross validations and KNN was found to be the best classifier with 94% accuracy.

Inertial sensors such as accelerometer, gyroscope and other sensors provide body parts motion data and physical accleration data which can be used for human activity recognition [37]. Isha.A [38] used a fusion of six different sensors such as accelerometer, gyroscope, GPS, light, magnetometer and audio attached to different body parts such as chest, waist, arm, thighs etc. The motion signals obtained from the experimentation were converted to image sequence. Convolution Neural Network (CNN) was applied to recognise activities such as climbing down, climbing up, jumping, running and walking with more than 80% accuracy. Wearable sensors are capable of providing high level and low level activities such as standing, sitting, laying, walking, eating, etc, of single or multiple residents in an smart home environment. RFID, Bluetooth, and Radio Frequency label based wearable sensors are commonly used by the researcher's since they provide higher accuracy results. Since the tag-based sensors has to be worn by the individual throughout the day and keep it charged it is not a feasible approach.[39].

Junji Yamato [40] proposes a vision based approach for activity recognition. A time-sequential images are captured and transformed to image feature vector squence and then to symbol sequence using vector quantisation. Hidden Markov Model is applied to the symbol sequence dataset to train and test the model. Three people performed the activities with six different actions for ten times. Five sequence were used for training and the rest five was used to test the performance of the model. Experiments were carried out using datasets of different subjects as train and test. The HMM trained with dataset of two subjects and tested with different subject was able to achieve 70.8% accuracy. James Fogarty [13] introduced a new method to overcome the problem with vision based and wearable sensors by using low-cost microphone sensors which can be easily mounted in the existing infrastructure. The key idea was, fresh waster enters the house at a single point and wastewater leaves the house at a handful location. Attaching microphones outside the existing waste water pipes to detect the waterflow at different areas such as sinks, toilets, shower, dishwasher and other appliances can be used to detect activities. Sensors were placed away from systematic noise sources to avoid the environmental noise. Data was collected for six weeks and support vector machine (SVM) classifier was used to classify the activities. The system was capable to detecting dish washing, cloth washing with 100% accuarcy, toileting, showering with more than 90% accuarcy. Detection of falls using a pressure sensor on the floor is an alternative approach for activity recognition. Henry Rimminen [41] proposes a floor sensor-based approach using near field imaging floor sensors. The test floor of 19m^2 had a resolution of 9x16 sensors. The test was conducted on 10 people with 650 events to get more than 90% sensitivity and specificity. The pressure sensors need to be pre installed on the floor before the construction and the number of sensors required to cover the area of interest is high which inturn makes it expensive to install. Table 2.1 summarize different types of sensors used and the focus of study of other researcher's on human activity recognition with accuracy achieved and cost involved.

Author	Sensor	Focus of Study	Accuracy	Cost
Nadee [8]	Ultrasonic	- Fall posture recognition. - Minimizing the dead zone using US in octagonal form	High	Low
Biswas [27]	Ultrasonic	- Only one transmitter is active at any given time to avoid signal overlapping. - Fall detection algorithm is used for decision making.	High	Low
Ghosh [31]	Ultrasonic	- Using heterogeneous ultrasonic sensors to cover more area. - Detecting human movement direction. - Comparing the results from decision tree, KNN, SVM for activity recognition.	High	Low
James Fogarty [13]	Microphone	- Placing microphone in waste waster outlet pipeline to detect the water flow. - SVM classifier is used to obtain more than 90% accuracy.	Low	Low
Henry Rimminen [41]	Pressure Sensor	- Detect fall based on pressure sensivity. - A group of 10 people were used to simulate the fall and more than 90% specificity and sensitivity was observed.	Low	High
Junji Yamato [40]	Camera	- Time-sequential images is transformed into symbol sequence using vector quantisation. - HMM is applied to recognise the activities and achieve accuracy rates higher than 90%.	High	High

Table 2.1: Summary of Human Activity Recognition using different sensors

2.3 Behaviour Identification

Every individual has a unique behaviour and follows a pattern in performing the activities such as eating, sleeping, cooking, traveling during the course of the day. Using time with activity data one can identify the behaviour pattern [42]. Roland

Cheng [24] proposed to use kalman filter to track the movements of mother and child to monitor the behaviour. An array of ultrasonic sensors is placed on the ceiling of the observation area. Mother is said to be "secure-base" for the child. Under any abnormal or stressful condition, the child returns to the mother for protection. The distance between child-mother and child-stranger is the critical factor for deciding the behaviour [43].Due to the poor lateral resolution of ultrasonic sensors, multiple people appear to be one single peak. To overcome this problem a tracking filter is used. Kalman filter is chosen over other linear filters because of its simplicity, ability to get the squared difference between the actual value and estimated value and also capable of handling non-stationary targets [44]. A total of 168 experiments were conducted using different socio-economic backgrounds. Ultrasonic sensors are placed on the cieling of an observation room of 4.45 X 2.23m. The distance between child and parent is measured in real time by knowing the position of child and parent from ultrasonic sensors. This experimentation is based on quality of parent-child attachment bond. The distance between parent-stanger and parent-child has been assessed over 30 years by trained human coders. Video camera was placed to record the ground truth to measure the experimental error. Since the child behaviour cannot be scripted more errors were recorded in estimating the child behaviour compared to mother-stranger which leads to noisy data.

Due to comapct size, increase in processing capabilities and the types of sensors embedded to smartphones, it has contributed to applications in different domains including healthcare (fall detection [45], behaviour monitoring [46]), security etc. Rich data is available by exploring the smartphone log. In [47], contact list data in a smartphone is collected and analysed based on number of features such as intensity of calls, the medium used for calling, regularity etc. The proposed work was able to provide more than 90% accuracy to classify the type of relation with the contact such as Family/Friend/ Work by applying machine learning techniques. Rischan Mafrur [48] proposed a smartphone log data for discovering human behaviour. A andriod smartphone was chosen to collect 19 kinds of sensor data. A total of 47 students smartphone log such as whatsApp, SMS, bluetooth, screentime, BatteryProbe and several other data was collected. Based on the features extracted a human behaviour model was built to predict the similar data patterns. Except for four students the system was able to achieve more than 80% accuracy. In [49] uses a multi camera video data to recognise high level human activities. Abstract Hidden Markov Model (AHHM) is used to identify state dependent and context free behaviours. The experiment is carried out placing the video cameras in a real complex indoor environment to recognise the complex behaviour of people. The behaviour recognition model takes the co-ordinates obtained from tracking module and decides the behaviour of the person based on the highest probability match. For example, if a person enters dining area and approaches dining table then the system can predict most likely the person is eating. The system was able to achieve high level human behaviour recognition based on multi camera surveillance system.

Markov model is instrumental in human behaviour analysis [50] [51]. Kosuke Hara [52], introduces a Markov model to detect the unusual behaviour of a person in an intelligent house. A two storied house with small motion detectors, CCD cameras and open/close door sensors were used to gather state transition and daily activities of a person in the house. Since the activities performed by an individual vary from another, the system should modify the services to adapt to an individual. Activities are performed on morning and evening basis and the daily pattern is recorded for over two months. If there is a shift identified in the regular pattern then the system reocgnizes it as abnormal behaviour.

2.4 Identifying People

Various sensing technologies such as facial recognition [53], fingerprint [54], hand geometry [55], Iris [56], biometrics such as height [57] and weight [58] are used to identify a person in last few decades. Facial recognition technique involves geometrical features and the gray-level information to identify a person. Andreas Lanitis [53] built a model which can identify a person based on different variations such as expressions, 3D pose, lighting and apperance. A total of 30 individual face images with variations in expressions and environment lighting is collected for training and testing the model. Shape model, shape-free gray model, local gray-level models were used for classification and found more than 90% accuracy by combining the models. Fingerprint, hand geometry, and iris associated systems are highly efficient and reliable, but these systems require a high level of interaction with the user which raise privacy concerns and requires the user to authenticate. The authentication requirement is difficult to implement, and the identification process fails if the user does not validate.

Chasity DeLoney [59] introduces a person identification system based on the classifying the sound of the footstep. The shoe worn by each individual and footsteps form a unique shoeprint. 9 different people wearing 3 kinds of shoes on 2 floors were used for data collection. A microphone was fixed to the ground and connected to a laptop to collect the footstep sounds. A high pass filter was applied to the collected data to remove the noise. 90% of the data is used for training Linear SVM classifer and the rest 10% data was used to test the performance. The results were higher when the train and test dataset are from the same shoes and different floors but the accuracy depreciated to even 18% when the train and test dataset were chosen to be from different shoes.

Robert J [60] uses a smart floor mechanism to track and identify a person. The house floor was outfitted with force measuring instruments. The force applied on the floor while walking, running or laying is unique to a person and the results are much promising than biometric technologies. The reaction produced by the measuring instrument due to the weight and inertia of the body when a person is in contact with that device is called Ground Reaction Force (GRF) [61] [62]. GRF profile is created for each user's footstep and Nearest neighbour is used to classify and identify the unknown footsteps. The system was able to achieve 93% accuracy in identifying the correct user.

IBM Smart surveillance systems [63] can track the location, identify people based on facial recognition and trigger an event based alarm if there is any emergency. Khalil [64] proposes a sonic door system which collects contextual information of a person walking through the door to identify the person. Data used to differentiate between people are whether the walker is using the phone, holding a handbag or wearing a backpack. Three ultrasonic sensors are installed for a doorframe one on each side and one on the top. The ultrasonic sensor placed on top is used to measure the height of the person and record time. The sensors placed on the side frames determine the width of the person. Using a Markov chain model for decision making the system was able to achieve 90.2 percent accuracy. An innovative approach to identifying a person using WiFi was introduced by Yunze Zeng [65]. The proposed method is effortless and device-free since the user is not required to wear/carry any smart devices. A framework was built and named as Whiwho which can identify a person using WiFi. The WiFi signals reflected from the body can be used to determine the activities of a person. Similar to gait analysis using accelerometer, CSI based gait analysis is performed for a group of people to identify a person. 20 volunteers were considered for the experimentation conducted in multiple locations and found that the system can achieve an accuracy of 80-90% in identifying a person in a group of 2 to 6 people.

Conclusion

Research on human activity recognition and Identifying the resident from others

Author	Sensor Used	Population	Accuracy
Chasity DeLoney [59]	Camera	9	60%
Robert J [60]	Floor Sensor	15	93%
Khalil [64]	Ultrasonic	170	90%
Zeng [65]	WiFi	20	80-90%

Table 2.2: Summary of Person Identification using different sensors

using ultrasonic sensors is very limited. This area of study creates a major impact on elderly people who is willing to live independently in there own house as well as allows caregivers to take care of more people by letting them know if there are any emergencies.
The first step in identifying the activities of a person is by data collection. The challenge of data collection and data labelling may be the reason for limited research in this field.

In the next chapter we introduce the experimental setup created for data collection in-house - by performing everday activities.

Chapter 3

Data Collection & Processing

Human activity recognition and identifying a person using machine learning techniques, requires meaningful data. As discussed in chapter 2, there are different types of ambient sensors such as pressure, motion, temperature, etc. which can be used to extract information for activity classification. There are several Activity of Daily Living (ADL) datasets readily available online such as CASAS [66], Placelab datasets [67], Orange4home [68] that other researchers have used in smart home activity recognition. These datasets are all combination of output from different sensors such as motion, water, ultrasonic and electricity which is not useful for us - since we are solely interested in data from ultrasonic sensors. We believe three participants are enough to prove the proposed method, as we are trying to distinguish between two people. There can be more number of participants considered which will increase the confidence level of the results. We choose to use an ultrasonic sensor over other ambient sensors in our research because of its reliability, long-range detection, ease of installation and its cost-effectiveness [69]. There are several ultrasonic sensors available in the market for different applications and environments [70]. We use MB1004 LV EZ0 Maxsonar [71] high-performance ultrasonic sensors which are designed for object and people detection in an incredibly small size.

3.1 Environment setup

We designed a test environment in a living room of dimension 4m x 4m x 2.5m in length, width and height respectively. Test room was equipped with a sofa, a kitchen bench, a dining table with two chairs and a TV table as shown in figure 3.1. 12 ultrasonic sensors are mounted on the ceiling 70cm apart from each other as shown in figure 3.2, to measure the distance from the ceiling to the object (person, furniture, floor). The actual test room looks as shown in figure 3.3.

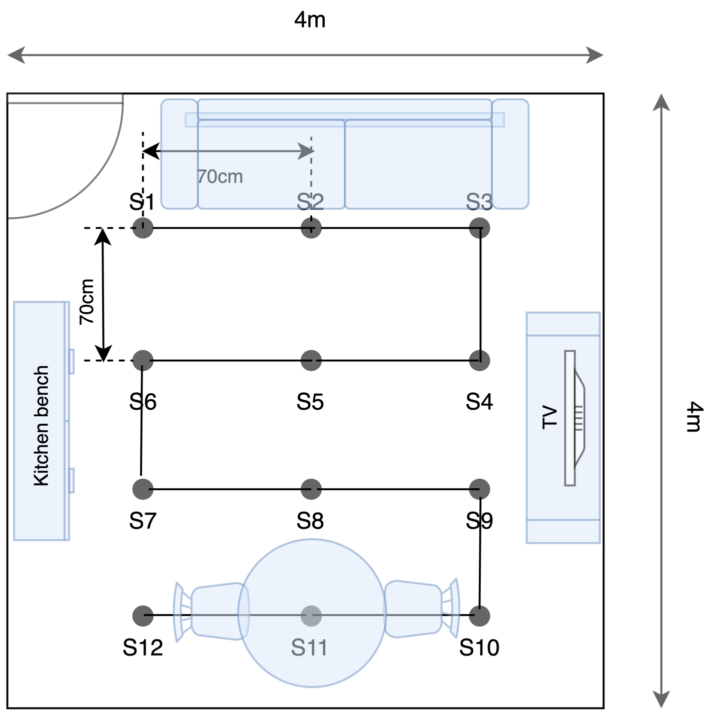

Figure 3.1: Environment setup for data collection with 12 ultrasonic sensors

Figure 3.2: Sensors placed on cieling of test room

Figure 3.3: Test Room

3.1.1 Setting Up Ultrasonic Sensors

Since we are using multiple ultrasonic sensors in the same space, we need to connect them in such a way that there is no signal interference, referred to as crosstalk or noise. Interference happens when two sensors are placed nearby because one sensor receive the signals from other causing incorrect value to be read. There are multiple ways ultrasonic sensors can be connected to avoid crosstalk, namely (i) range-simultaneously (ii) sequentially read each sensor and (iii) continuous looping. According to maxbotix [71] the most reliable way of connecting multiple sensors in the same environment is by sequentially reading each sensor. This method allows the sensors to range only after the previous one has finished ranging. There will be no signal interference between the sensors since there is only one sensor active at any given time. The first sensor is pulled high (ON) for 96ms, while the rest 11 sensors are in OFF state. Once the first sensor finishes ranging, it is switched off and the second sensor is switched ON. The process is carried out in sequence until sensor 12 is switched ON and finished ranging. The analog pin of each sensor is connected to the microcontroller as shown in figure 3.4 which will send all the readings to a central unit for processing and decision making. Four steps that are followed to get the distance measurement from ultrasonic sensors are (i) connect the ultrasonic sensors with Arduino microcontroller (ii) install Arduino sketch software on the PC (iii) Setup the sensor with Arduino and finally (iv) compile and run the program.

Since the furniture is part of the layout in our test environment, the distance readings from each sensor might vary depending on the location of the sensor with the furniture. The sensors are placed perpendicular to the ceiling as much as possible and the readings are obtained in centimetres. Table shows the average values of the sensor readings when there are no activities in the test environment.

Sensors	Average Value (cm)	Description
S1, S2, S3	180	Hit the edge of the sofa
S4, S9	220	Hit the edge of TV Table
S5, S6, S7	250	Hit the floor (No obstacle)
S8	210	Hit the edge of dining table
S10, S12	180	Hit the chair
S11	180	Hit the dining table

Table 3.1: Average sensor readings with no activity

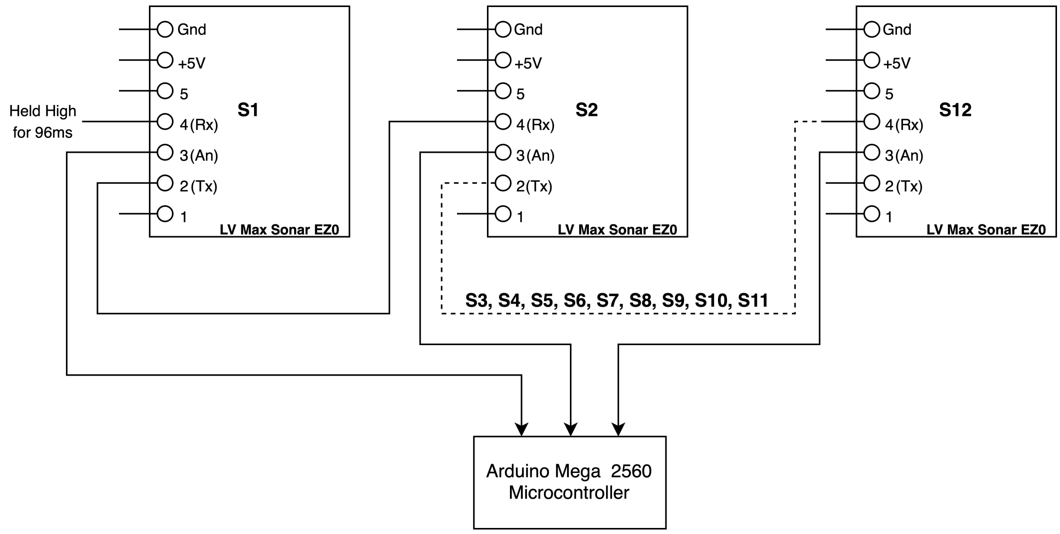

Figure 3.4: Ultrasonic sensors connected for sequential read operation

3.1.2 Experiment setup

Myself and two of my supervisors who vary in height, width and weight are chosen as subjects for data collection. Only one of us will live in the test room and perform the actvities to collect single-resident actvity data. For collecting multi-resident activity data, two people will perform the activities in the test room at the same time. The type of activities to be performed is informed to the resident but the order in which they need to be performed is not prescribed. Simple activities a normal human performs in everyday basis were chosen, such as "cooking", "eating", "sitting on couch", "walking around" and "laying on couch". The residents were asked to act normally as living in a house. The sensor readings with timestamp are collected every second and stored in an excel file using a computer. The sample raw data obtained from the ultrasonic sensors is as shown in table 3.2. The raw sensor data needs to be processed before applying machine learning algorithms inorder to remove the noise in data and also add additional features to increase the accuracy of activity prediction by the model. The steps involoved in converting the raw dataset to processed dataset is as shown in the figure 3.5.

3.2 Data Filtering

From the previous studies, we know that ultrasonic sensors are sensitive to environment and generate noise due to external shocks and reflection from furniture in the

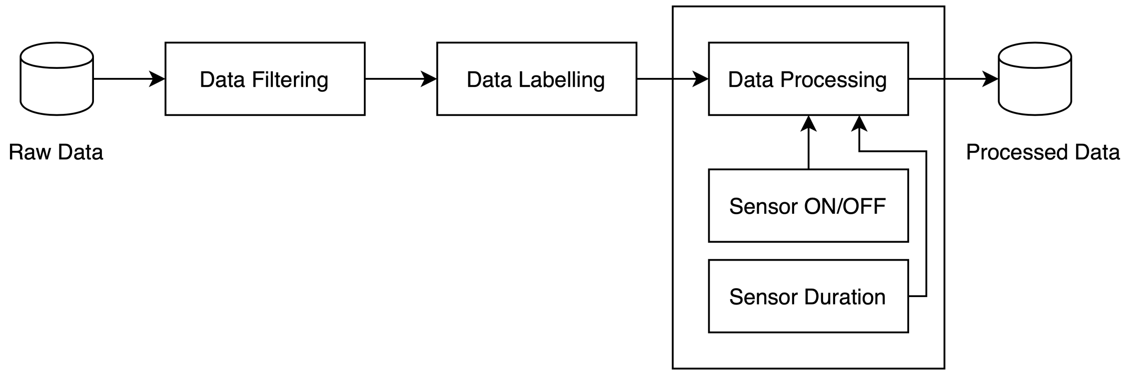

Figure 3.5: Process of converting raw dataset to processed dataset

Time	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
9:00:01	181	180	180	220	249	246	251	248	252	184	183	110
9:00:02	180	181	182	219	250	246	251	249	252	184	183	103
9:00:03	181	180	183	219	249	246	101	248	252	184	184	182
9:00:04	181	180	182	220	250	98	251	248	252	184	183	182
9:00:05	180	181	183	221	250	247	105	248	251	184	183	182
9:00:06	181	180	182	220	249	116	103	248	252	184	183	182
9:00:07	181	181	182	220	250	247	105	248	252	184	184	182
9:00:08	180	181	182	220	250	97	104	248	252	183	184	182
9:00:09	180	183	183	221	250	123	102	248	252	184	183	182
9:00:10	181	182	182	220	250	114	108	248	251	184	183	182
9:00:11	180	182	182	220	251	116	250	248	252	184	183	183
9:00:12	181	183	181	221	250	116	104	248	252	184	183	182
9:00:13	181	183	182	220	250	96	101	249	251	184	183	182
9:00:14	180	183	183	220	248	154	105	247	252	183	183	182
9:00:15	181	182	182	220	250	108	105	249	251	184	183	182

Table 3.2: Raw sensor data with timestamp from 12 ultrasonic sensors

environment [72]. The sensors are turned ON and the readings are recorded for 30 minutes without anyone in the test room. The maximum value, minimum value, mode and standard deviation are calculated as shown in the table 3.3, for each of the sensor to determine how noisy is the data.

From table 3.3 the standard deviation was found high and a maxi distance difference between maximum value and minimum value obtained was found to be 5cm, which indicates that there is noise in the data collected. There are several differ-

Sensor	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
Max Value	183	185	182	222	250	251	251	212	225	182	182	181
Min Value	179	180	179	219	249	246	248	208	220	180	180	178
Mode	180	181	181	220	250	250	249	211	221	181	181	179
St.dev	1.640139	0.82323	0.694327	0.973328	1.66123	1.78688	1.30774	0.501788	0.781039	0.257592	1.47032	0.870103

Table 3.3: Max, min, mode and standard deviation of sensors

ent types of filters, such as kalman filter [73], particle filter [74], adaptive filtering [75], median filter [76] used for data filtering. We choose to use three different filters, namely (i) First-order Filter (ii) Moving Average Filter and (iii) Impulse filter beacuse of its simplicity, ease of implementation and provide good results [77].

3.2.1 First-order Filter

An input signal is transformed into an ouput sequence using a function or operation. The signal flow graph is as shown in the figure 3.6. The symbol z $^{-1}$ indicates the delay in one sample i.e z $^{-1}$ x(n) = x(n-1). The impulse response for a first-order filter is given by

$$y[n] = \frac{1}{2}(x[n] + x[n-1]) \tag{3.1}$$

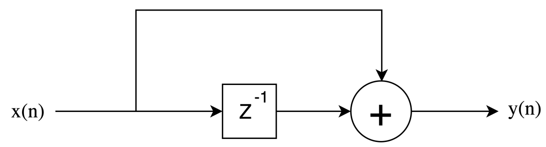

Figure 3.6: System diagram for first order filter

3.2.2 Moving Average Filter

Moving average filter also known as running average filter uses the current output and previous output reading as input to calculate the present output. The system diagram for moving average filter is as shown in the figure 3.7 and the impulse response of moving average filter is given by

$$y[n] = \frac{1}{3}(x[n] + x[n-1] + x[n-2]) \tag{3.2}$$

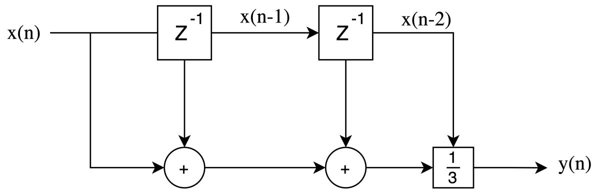

Figure 3.7: System diagram for first order filter

3.2.3 Impulse Filter

A unit impulse sequence is a set of numbers having a non zero value of one only, when its argument is zero i.e

$$\delta[n] = \begin{cases} 1, n = 0 \\ 0, n \neq 0 \end{cases} \tag{3.3}$$

By time shifting the amplitude scale of impulse sequence, we can form linear combination of them to form any sequence. For example

$$x[n] = 2\delta[n] + 4\delta[n-1] + 6\delta[n-2] + 4\delta[n-3] + 2\delta[n-4] \tag{3.4}$$

The current output of impulse filter depends on the past four readings obtained from the sensors. For example, if the readings obtained from the sensors are 180, 179, 181, 181 and 183 for five seconds then the output of impulse filter is calculated as

$$x[0] = 2 * [180] + 4 * [179] + 6 * [181] + 4 * [181] + 2 * [183]$$

$$x[0] = 178.56$$

The filters are implemented in the spreadsheet using formulas in the cell to calculate the output sequence. The filtered values from all the three filters are rounded up to the nearest value and the standard deviation was calculated for each sensor. In table 3.4 raw data refers to the values before applying the filters and the results obtained by applying first order, moving average and impulse filter respectively. From the table, it is observed that standard deviation for moving average filter is low compared to other two filters. Hence moving average filter was chosen over first order and impulse filter to remove the noise in data collected. Figure 3.8 and 3.9 shows the sample plot drawn for Sensor 1 and Sensor 4 readings obtained before and after applying the moving average filter.

Sensor	Values	Raw Data	First Order Filter	Moving Average Filter	Impulse Filter
	Max	185	184	184	184
S1	Min	180	180	180	180
	St. dev	1.640139	1.131034	**0.978527**	1.043185
	Max	223	223	222	222
S4	Min	220	220	220	220
	St. dev	0.973328	0.645497	**0.45573**	0.535927

Table 3.4: Min, max and standard deviation before and after applying the filters

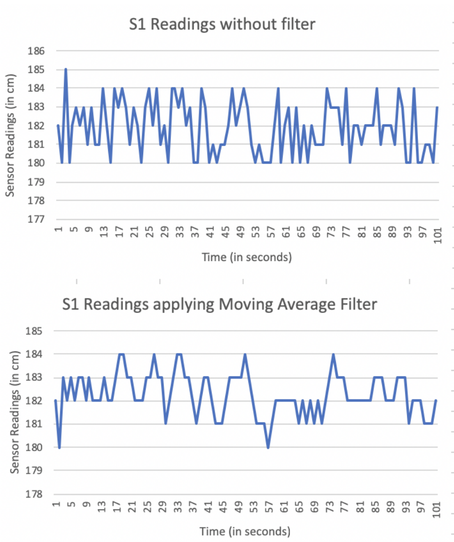

Figure 3.8: Sensor 1 readings before and after applying moving average filter

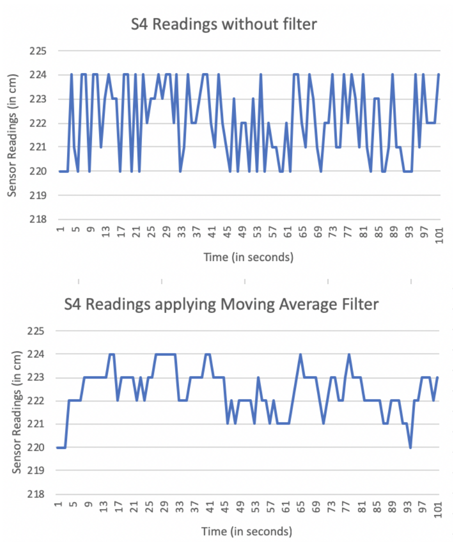

Figure 3.9: Sensor 4 readings before and after applying moving average filter

3.3 Data Labelling

Once we have the raw data in the spreadsheet it is time to add some meaning to it. The data was collected for approximately one hour and ten minutes and the sensor readings were recorded every second. For instance, the spreadsheet for Resident A contains 4,267 rows and 13 columns of data which represents resident A - performing the activities over 4,267 seconds. The first column contains the timestamp and the other 12 columns contain the sensor readings. To classify the activities using machine learning we need to train the models with ground truth. A video camera was placed in the test environment to cover the entire area of interest and the video was captured every second when the activity was performed by the resident. A new column is inserted in the spreadsheet and named as activity. We label the data manually, looking at the video for each row of sensor readings in the activity column of the spreadsheet. The sample labelled data looks as shown in table 3.5.

Time	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	Activity
9:01:01	181	182	182	227	250	98	251	210	220	180	180	180	cooking
9:30:22	181	181	182	225	248	247	251	211	220	181	180	97	Walking Around
9:45:53	181	182	182	226	249	246	251	210	219	181	138	180	Eating
9:55:04	182	180	177	226	250	247	251	210	221	181	181	181	Sitting on Couch
10:05:05	169	172	175	220	250	247	250	211	220	180	181	181	Laying on Couch

Table 3.5: Data labelled manually for sensor readings using ground truth.

We labeled the transition of activity as "Walking Around" activity to keep it simple for activity classification. For example, the resident prepares his breakfast in the kitchen and walks towards the dining table to eat. This transition is labeled as "walking around" activity. Data can be labeled in two levels namely (i) High-level activity (ii) Low-level activity [78]. The high-level activity includes cooking, Walking Around, Eating, etc. and low-level activity includes the postures such as sitting, standing, laying, etc. Since the low-level activity was common throughout the experiment for all the residents it did not add value for decision making. For example, all the residents were in standing posture while preparing breakfast in the kitchen, all the residents were in sitting posture while having breakfast on the dining table and in the standing posture while walking around. Hence only high-level activity was considered for data labeling.

3.4 Data Processing

Data processing is a technique of extracting more features from the raw data collected which can add-in value for decision making. From the previous studies on activity recognition, most of the researchers use only sensor reading as a feature to classify the activities using machine learning algorithms. But, in our proposed work we were eager to improve the accuracy of the machine learning models by including features such as sensor ON/OFF state and sensor duration. We train the models with different combination of features such as (i) sensor reading, (ii) sensor ON/OFF and (iii) sensor duration to find the robust features for recognising the activities using ultrasonic sensors.

3.4.1 Sensor ON/OFF

When there is no activity in the test environment the sensors measure the distance from the ceiling to the obstacle i.e furniture or floor. Since there is no variation in the sensor readings this state is considered as Sensor OFF state. When there is a person in the test environment performing activities, the sensor readings change depending on the height of the person and activities performed. One or more sensor readings can be changed at the same time depending on the activity performed. Since the difference between the maximum value and minimum value obtained from the sensors is not greater than 5cm, it was considered as threshold value. If the sensor value drops below 5cm of the average value, then the sensor is said to be in ON state. Sensor OFF state is represented by "0" and ON state is represented by "1". 12 columns were inserted on the spreadsheet naming PS1, PS2 PS3,...., PS12. Each column was governed with the threshold value using if else statement in the spreadsheet. Sample data of sensor ON/OFF state for different activities performed by the resident is as shown in the table 3.6. There can be one or more sensors always on the same state for different activities however the duration of sensors is dependent on sensor ON/OFF state. For example, the state of all the sensors in walking Around activity and Eating activity are same but the duration of sensor ON/OFF changes.

PS1	PS2	PS3	PS4	PS5	PS6	PS7	PS8	PS9	PS10	PS11	PS12	Activity
0	0	0	0	0	1	0	0	0	0	0	0	cooking
0	0	0	0	0	1	0	0	0	0	0	0	cooking
0	0	0	0	0	1	0	0	0	0	0	0	cooking
0	0	0	0	0	0	0	0	0	0	0	1	Walking Around
0	0	0	0	0	0	0	0	0	0	1	0	Walking Around
0	0	0	0	0	0	0	0	0	0	1	0	Walking Around
0	0	0	0	0	0	0	0	0	0	1	0	Eating
0	0	0	0	0	0	0	0	0	0	1	0	Eating
0	0	0	0	0	0	0	0	0	0	1	0	Eating
0	0	1	0	0	0	0	0	0	0	0	0	Sitting on Couch
0	0	1	0	0	0	0	0	0	0	0	0	Sitting on Couch
0	0	1	0	0	0	0	0	0	0	0	0	Sitting on Couch

Table 3.6: Sensor ON/OFF state for different activities performed by resident

3.4.2 Sensor Duration

Sensor duration can be defined as the amount of time the sensor is turned ON i.e. the number of consecutive 1's seen in a column. A python script was written to calculate the sensor duration and store this data in an excel file. An excel file in CSV format with sensor ON/OFF state and activity label as shown in table 3.6 was uploaded to python library. Sample data after the script was executed is as shown in table 3.7.

DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9	DS10	DS11	DS12	Activity
0	0	0	0	0	1	0	0	0	0	0	0	cooking
0	0	0	0	0	2	0	0	0	0	0	0	cooking
0	0	0	0	0	3	0	0	0	0	0	0	cooking
0	0	0	0	0	0	0	0	0	0	0	1	Walking Around
0	0	0	0	0	0	0	0	0	0	1	0	Walking Around
0	0	0	0	0	0	0	0	0	0	2	0	Walking Around
0	0	0	0	0	0	0	0	0	0	3	0	Eating
0	0	0	0	0	0	0	0	0	0	4	0	Eating
0	0	0	0	0	0	0	0	0	0	5	0	Eating
0	0	1	0	0	0	0	0	0	0	0	0	Sitting on Couch
0	0	2	0	0	0	0	0	0	0	0	0	Sitting on Couch
0	0	3	0	0	0	0	0	0	0	0	0	Sitting on Couch

Table 3.7: Sensor duration for different activities performed by Resident

From table 3.7 we can see that sensor 6 was ON for 3 seconds and then turned OFF while "cooking", sensor 12 was ON for 1 second while the resident was "walking around", sensor 11 was ON for 5 seconds while "walking around" and "eating breakfast", sensor 3 was ON for 3 seconds while "sitting on couch" and all the sensors were switched OFF while "No activity" was performed.

Conclusion

By having the experiment setup, we can perform the activities and collect the data. Now we know that moving average filter can be applied to reduce the noise and manually label the data. In the next chapter, we discuss how different types of activities can be recognised by applying different machine learning algorithms and also evaluate what are the robust features and algorithms for recognising the activities of single and multiple residents.

Chapter 4

Activity Recognition

Usually ultrasonic sensors are used to find the distance between object and sensor node. We have used the same principal to detect the activities of a person in the test environment. Once the ultrasonic sensors are placed on the cieling of the test room we obtain the height measurement of the person based on the activities performed. The raw sensor readings obtained from the ultrasonic sensors are filtered, labelled and processed as discussed in chapter 3. In this chapter we discuss how the processed dataset is trained and tested on different classifiers for single resident, and also two resident's performing the activities at the same time in the test room.
To the best of our knowlege, for single resident activity recognition the performance of all the proposed smart home systems for predicting a resident's activities, including the ones mentioned in the previous section, have only been tested based on the prior activities of the same occupant only using sensor reading as the feature. Although, those research showed the possibility of having a smart home system that could help people, its utilization on real world scenarios is limited because manual human labelling is cumbersome and time consuming [79]. Furthermore, to accurately label those activities requires researchers to monitor the resident's activities through multimedia system - defeating the purpose of using ambient sensors. For a smart home system to be widely - used in real life, the system must be able to predict the resident's activities without any training on the resident's prior activities. In our proposed research for single resident activity recognition, we create a system which is able to predict the activities regardless of who the person is by using ultrasonic sensor's reading, on/off state and duration. The activities considered are cooking, eating, sitting on the couch and laying on the couch. Our proposed system is trained with activity dataset of one person and then tested with the same person and with two different people. For multi resident activity recognition, two

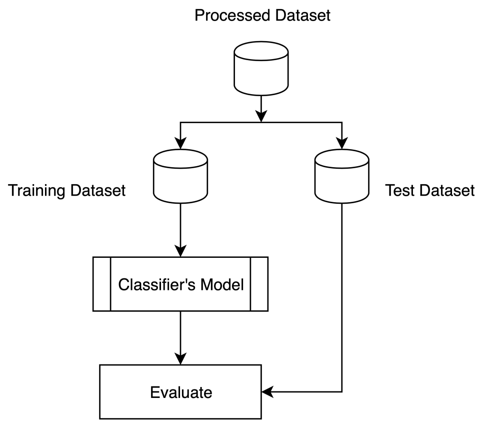

Figure 4.1: Proposed system for activity recognition

individuals will perform the activities in the test room at the same time and make sure that they are doing as much concurrent, interleaved and parallel activities. The system is trained and tested with same people as well as different people to evaluate the accuracy of the system. Figure 4.1 shows the block diagram of the proposed system. The processed data is split into train and test dataset to train and evaluate different classifiers.

4.1 Single-Resident Activity Recognition

4.1.1 Data Collection & Preparation

For single resident activity data collection, only one person performs the activities in the test room at all times. Myself, considered as Resident A live in the test room for approximately 1.5 hours everyday and perform the activities for three consecutive days. While my supervisors refered as Resident B and Resident C perform the activities for one day for approximately two hours individually. We provide the occupants with the set of activities which has to be performed but will not prescribe the order or duration of the activity to be performed. Since there are no timers used, the duration of activity being performed is left to individual choice. The sensor readings are collected for each of the individuals performing the activities and stored in an excel file in the computer. Five different activities were performed by each individual namely (i) eating (ii) cooking (iii) sitting on the couch (iv) Laying on the couch and (v) walking around.

The sensor readings obtained from each of the individuals are stored separately.

The data is filtered using moving average filter and the activities are labelled manually using the video footage captured. Table 4.1 shows the total observations or instances of data collected for each activity for an individual with total duration in hours.

Activity	Resident A			Resident B	Resident C
	Day 1	Day 2	Day 3		
eating	739	808	977	857	3686
cooking	692	647	641	525	213
sitting on couch	1060	1893	1466	1905	1126
laying on couch	717	851	633	1529	10
walking around	479	268	430	525	281
Total duration	1:28	1:41	1:32	1:52	2:18

Table 4.1: Number of instances collected for each activity performed by the resident

Once the sensor readings are filtered and labelled, the data is processed to extract the features such as sensor on/off and sensor duration. Each feature and all possible combination of features are saved in different spreadsheet in csv format. We split the dataset collected into 60% train and 40% test dataset to train and evaluate the classifier. An open source machine learning analysis software called WEKA *(Waikato Environment for Knowledge Analysis)* is used to split the dataset and run the machine learning algorithms. The csv*(comma seperated value)* files are converted to arff *(attribute related file format)* using arff viewer in WEKA. A resample filter is applied with no replacement to uniformly distribute the class values into train and test dataset.

4.1.2 Experiments

To investigate the efficacy of our proposed features and classifier models at predicting the resident's activities, we performed our experiments in three phases.

In first phase, we split each dataset collected from each resident into 60% train and 40% test to find what are the best features and classifiers to predict the activities of each resident. We believe that this would give us the best performance because the model is tailored made for each resident based on the resident's prior activities.

In the second phase, we combined the dataset collected from all of the residents i.e resident A day 1, 2, 3 , resident B and resident C into a single dataset and split into 60% train and 40% test set. This phase emulates the way the other researchers evaluate the performance of their proposed systems.

In the third phase, we set the data collected from resident A - day 1 as the training dataset and the rest of the dataset i.e resident A - day 2, 3, resident B and resident C are used as test dataset. We believe that the performace of this method will be lowest compared to other two phases because the model is not trained to detect the activity pattern for different person. Since we are using sensor ON/OFF state feature, which is less volatile it will be accurate enough that it outweighs the manual labelling cost of the other two phases.

In all the three phases, we will compare the performance of the model trained with DT *(Decision Tree)*, RF *(Random Forest)*, KNN *(K-Nearest Neighbours)*, NB *(Naiive Bayes)*, SVM *(Support Vector Machine)*, LR *(Logistic Regression)* and BN *(Bayesian Network)* using Weka to predict the following activities: cooking, eating, sitting on couch, walking and Laying on couch.

4.1.3 Results & Evaluation

To identify which set of features and classifiers are the best for each phase, we compare the performance of the model trained by each classifier given only the duration of each sensor being ON at any given moment (Duration), the on/off state of each sensor (ON_OFF), the distance values read by each sensor (Reading), and any possible combinations of those three basic features (ON_OFF_Duration, Reading_Duration, Reading_ON_OFF and Reading_ON_OFF_Duration).
Accuracy is the best metric for evaluating the classification models when there is a class balanced dataset [80]. Formally accuracy is defined as

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{4.1}$$

For binary classification accuracy can be calculated in terms of positives and negatives as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2}$$

where TP is*(True Positive)*, TN is*(True Negative)*, FP is*(False Positive)*, FN is*(False Negative)*. Since the class values are distributed uniformly in our dataset we use accuracy (in %) as our evaluation criteria.For each classifier's result, we have bolded the highest accuracy value that we can get from any of the input combinations. The highest accuracy for each experiment setup is bolded and italicized.

First Phase Evaluation

Resident A

The activities performed by resident A on day 1, 2 and 3 are combined and split into 60% train dataset and 40% test dataset. Table 4.2 illustrates the results of our first phase evaluation on Resident A. The best feature and classifier for resident A are Reading_ON_OFF_Duration and Random Forest with 98.51% accuracy.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	93.4986	91.345	93.6611	96.6477	97.3182	96.6477	97.501
RF	92.8484	91.6497	92.95	97.7042	98.395	97.8058	***98.5169***
KNN	91.5685	91.6091	92.3811	96.2007	97.501	96.1804	97.2978
NB	85.2093	90.2885	87.8098	91.284	93.1329	92.503	93.336
SVM	82.4868	91.1621	93.0313	91.9545	93.2954	94.5144	95.1849
LR	90.4917	91.4059	93.7221	93.844	95.449	94.7989	95.3068
BN	93.5798	90.7355	93.3564	92.4827	94.9208	93.6814	94.8395

Table 4.2: Classification results when resident A dataset is split into train and test dataset

Table 4.3 shows the confusion matrix i.e the number of correct and incorrect predictions made by Random forest classifier in predicting each activity. The best activity that can be predicted by random forest classier for resident A is eating activity with 100% accuracy and lowest accuarcy is obtained for walking around activity with 96.179%.

a	b	c	d	e	classified as
1010	0	0	0	0	**a = Eating**
0	865	0	13	3	b = Laying on couch
0	0	788	0	4	c = Cooking
1	33	0	1733	1	d = Sitting on couch
9	1	2	6	453	e = Walking Around

Table 4.3: Confusion Matrix for Resident A dataset split into train and test dataset by Random forest classifier

Resident B

Table 4.4 shows the results obtained from the classfiers for resident B dataset split into train and test set. The best feature and classifier for resident B are Reading_ON_OFF_Duration and Random Forest with 97.70% accuracy.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	92.5129	81.2822	91.8577	94.1507	95.0866	93.402	94.2911
RF	94.8058	81.4693	94.759	96.1161	97.7071	96.3032	***97.7071***
KNN	94.993	81.4693	94.4314	94.0103	96.9584	94.1507	95.8353
NB	85.4001	80.1591	88.5353	91.1558	91.2494	91.109	92.0917
SVM	85.4937	78.7085	89.4712	90.2199	92.9808	92.1385	93.7763
LR	90.9686	81.2822	91.2026	92.1853	93.5891	91.8109	92.934
BN	90.0328	80.5803	89.8456	93.168	93.3084	92.3257	92.7

Table 4.4: Classification results when resident B dataset is split into train and test dataset

Random forest classifier can classify the eating activity with 99.4% accuracy as shown in the confusion matrix table 4.5. Sitting on couch activity provides the lowest accuracy of 96% in which 13 instances are classfied as laying on couch and 1 instance is classified as walking around activity.

a	**b**	**c**	**d**	**e**	**classified as**
341	0	0	1	1	a = **Eating**
0	589	0	23	0	b = Laying on couch
0	0	208	0	2	c = Cooking
0	13	0	748	1	d = Sitting on couch
0	0	2	6	202	e = Walking Around

Table 4.5: Confusion Matrix for Resident B dataset split into train and test dataset by Random forest classifier

Resident C

Table 4.6 shows the results obtained for resident C dataset split into train and test set. The best feature and classifier for resident C are Reading_ON_OFF_Duration and Random Forest with 99.01% accuracy.

Random forest classifier achieved 100% accuracy to classify the laying on couch and sitting on couch activity as shown in the confusion matrix table 4.7. Eating activity is the second best correctly classified activity with 99.8% accuracy. 13 instances of walking around activity was classified as other activities which lead to the lowest accuracy of 88.5%.

From the first phase evaluation we can see that there is not much significant difference in terms of accuracy performance between the features Reading_Duration, Reading_ON_OFF and Reading_ON_OFF_Duration but the other input features

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	97.1831	93.2394	97.6526	96.4319	97.9343	96.5258	97.1831
RF	98.169	93.615	98.0282	97.277	98.8732	97.277	***99.0141***
KNN	96.9484	93.5211	97.6056	96.2441	98.4038	96.1927	97.7465
NB	93.2864	91.5493	93.3803	93.3333	94.6948	93.5211	95.1643
SVM	92.8169	93.0047	96.0094	93.1925	96.1033	95.7277	97.5587
LR	96.5728	93.1925	97.5117	95.3991	97.4648	96.5728	97.5587
BN	97.8404	92.8169	97.0423	95.6808	98.2629	95.1643	97.9812

Table 4.6: Classification results when resident C dataset is split into train and test dataset

a	b	c	d	e	classified as
1472	0	0	2	1	a = Eating
0	5	0	0	0	b = **Laying on couch**
0	0	81	0	5	c = Cooking
0	0	0	451	0	d = **Sitting on couch**
7	0	1	5	100	e = Walking Around

Table 4.7: Confusion Matrix for Resident C dataset split into train and test dataset by Random forest classifier

does create a significant impact. This suggests that using features such as sensor ON/OFF state and sensor duration in combination with sensor reading provides good accuracy in activity prediction.

4.1.4 Second Phase Evaluation

Table 4.8 shows the result of our second phase evaluation. The best features and classifiers for all the resident's data combined together are Reading_Duration and Random Forest with 98.7% accuracy. Similar to results that we got from our first phase evaluation, the best performance can only be obtained by combining the sensor readings with the sensor duration and/or sensor ON/OFF state features. We can observe that Reading_Duration, Reading_ON_OFF and Reading_ON_OFF_Duration features provide similar performance.

From table 4.8 we can see that cooking activity is classified with 99.8% accuracy, eating activity with 99.7% accuracy, sitting on couch activity with 98.7% accuracy, laying on couch activity with 97.7% accuracy and sitting on walking around activity with 95.8% accuracy.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	92.6301	86.708	92.7172	96.0266	97.3329	96.3096	97.3873
RF	92.4995	86.7842	92.2817	97.5724	***98.7154***	97.6268	98.563
KNN	91.683	86.7407	91.9443	95.983	97.6268	95.6129	96.8757
NB	79.3817	82.7564	85.3582	83.366	86.2726	88.7873	89.6255
SVM	74.472	84.2151	87.7966	89.3534	91.2149	92.3362	93.3268
LR	82.9741	85.3473	88.6458	89.6691	93.8276	92.924	94.4045
BN	89.5384	83.6055	88.8308	90.3222	93.3486	92.2491	93.3812

Table 4.8: Classification results when resident A+B+C dataset is split into train and test dataset

a	b	c	d	e	classified as
2818	2	0	4	3	a = Eating
0	1463	0	34	0	b = Laying on couch
0	0	1086	0	2	c = **Cooking**
3	36	0	2940	1	d = Sitting on couch
14	1	5	13	761	e = Walking Around

Table 4.9: Confusion Matrix for Resident A+B+C dataset split into train and test dataset by Random forest classifier

Resilient Evaluation

From the results of previous two phases, prior researchers in this area would usually conclude that the best classifier would be Random Forest and the best input combination would be Reading_ON_OFF_Duration. Although our accuracy is much higher than the accuracy performance that other researchers have reported, as we mentioned in the previous section, it is expensive to continuously label each resident's prior activities and of course defeats the purpose of our original research to provide smart home with the smallest intrusion to the resident's privacy.

To evaluate whether it is valid to assume that the pattern extracted from previous activities can reliably predict future activities, in our resilient evaluation, we use Resident A Day 1 data as our training dataset and Resident A Day 2 and Day 3 data as our testing dataset. The best features and classifiers for Resident A at Day 2 are Reading_ON_OFF_Duration and Support Vector Machine with 95.38% accuracy as shown in table 4.10; for Resident A at Day 3, are ON_OFF and any of the classifiers with 91.17% accuracy as shown in table 4.11. From this result, we can see that there is a 4.21% decay in performance from Day 2 to Day 3. In fact for some classifiers, the performance could dramatically drop to 30%. Thus the original assumptions that researchers have done in this area are not valid. Detecting and improving this concept drift is outside the scope of our research. However, our evaluation highlights the importance of creating a generic system that could be used for any resident with minimal training.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	81.285	89.1202	81.1059	32.6617	36.0197	35.7287	36.0197
RF	90.9335	88.9859	91.0007	68.7591	84.6877	70.0696	91.9633
KNN	80.6805	89.1874	84.5758	69.7784	88.9635	87.7994	94.0004
NB	83.0759	89.2993	85.3145	71.3279	76.5486	71.9495	77.1079
SVM	75.6212	88.4934	84.2848	84.4638	92.0528	93.4408	***95.3884***
LR	89.2993	88.5158	72.7782	59.3015	68.7591	50.839	49.6789
BN	91.3812	88.3815	91.6723	58.4421	80.982	72.1981	83.2401

Table 4.10: Classification results when resident A day 1 dataset is used as training and day 2 dataset is used as testing

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	88.0395	89.6552	88.0878	38.4615	44.7552	37.2576	44.7552
RF	88.353	91.1743	88.3771	70.337	74.2706	63.4917	88.6424
KNN	72.5585	90.1857	79.9373	71.8592	74.8734	90.668	85.6282
NB	76.7784	88.4254	78.9486	81.6944	72.8532	74.4961	73.4765
SVM	71.9067	89.3658	80.6848	76.055	82.8792	87.1473	88.3048
LR	86.0863	***91.1743***	67.7357	67.7285	73.7304	51.0157	46.2604
BN	87.6537	90.2098	88.1119	58.4026	72.8763	72.7839	77.4931

Table 4.11: Classification results when resident A day 1 dataset is used as training and day 3 dataset is used as testing

To investigate which input combination and classifiers are more resilient, we calculate the difference ($\mathrm{diff}_{\mathrm{c,f}}$) between accuracy that we got for Day 2 and Day 3 for a classifier (c) and an input combination (f):

$$diff_c, f = |accuracy_{c,f}^{day2} - accuracy_{c,f}^{day3}| \tag{4.3}$$

We define the maximum difference for classifier, c, regardless of features as the sum of the average of all the accuracy difference for that classifier and the standard deviation of all the accuracy difference for the classifier (i.e. $\max_{\mathrm{diff}}(\mathrm{c}) = \overline{\mathrm{diff}_\mathrm{c}} + \sigma_\mathrm{c}$). Similarly, the maximum difference for input combination, f, regardless of the classifiers is defined as $\max_{\mathrm{diff}}(\mathrm{f}) = \overline{\mathrm{diff}_\mathrm{f}} + \sigma_\mathrm{f}$.

From table 4.12, we can see that the three classifiers that give maximum accuracy difference of at most 5.9% are Decision Tree, K-Nearest Neighbour and Support

Algorithms	$\overline{\text{diff}_c}$	σ_c	Features	$\overline{\text{diff}_f}$	σ_f
DT	5.58	3.29	**Duration**	4.91	2.11
RF	4.18	3.19	**ON_OFF**	**1.42**	**0.80**
KNN	5.88	4.61	**ON_OFF_Duration**	4.68	1.57
NB	4.82	3.12	**Reading**	5.24	4.02
SVM	5.59	2.97	**Reading_Duration**	8.45	3.44
LR	**3.98**	**2.54**	**Reading_ON_OFF**	2.93	2.57
BN	3.37	2.87	**Reading_ON_OFF_Duration**	5.75	2.35

Table 4.12: Accuracy difference between day 2 and day 3

Vector Machine. The input combination that gives maximum accuracy difference of less than 5.9% are ON_OFF, ON_OFF_Duration and Reading_ON_OFF. Naiive Bayes is the most consistent classifier when we are just comparing the accuracy results from ON_OFF, ON_OFF_Duration, Reading_ON_OFF and Reading_ON_OFF_Duration.

In the first phase and second phase evaluation, we observe that Reading and Duration play a significant factor at improving the classifier's performance. However, a closer resilient evaluation seems to suggest that the ON_OFF feature is less volatile than the Reading or Duration features and can be utilized to predict the activities of the other residents.

Third Phase Evaluation

Sensor readings vary based on height of the person performing the activities and sensor duration vary based on the amount of time the person is doing the activities. Sensor ON/OFF is the only feature which is less volatile compared to other two. Resident A Day 1 dataset is used as training dataset and resident B and C dataset are used as test dataset. Table 9 and 10 display the results of our third phase evaluation on Resident B and Resident C, respectively. The best features and classifiers for Resident B are Duration and Logistic Regression with 73.07% accuracy; and for Resident C are Reading_ON_OFF and Support Vector Machine with 82.72% accuracy. As expected their accuracy is significantly lower than when we tested on Resident A Day 2 and 3 dataset. Similar to our findings during our resilient evaluation, the input features combination and classifiers chosen in either the first or second phase evaluation do not perform well in this evaluation. Naiive Bayes is the classifier that can produce a consistent performance. ON_OFF feature is much more resilient than Reading or Duration features. Based on the resilient and third

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	36.5676	33.5961	36.5676	23.2868	23.2868	23.2868	23.2868
RF	45.1991	60.1577	38.2050	50.0101	54.6594	54.5381	59.6725
KNN	46.3918	**70.1233**	58.0756	64.4027	58.0352	68.8700	60.2183
NB	54.8615	67.2327	56.5797	72.1447	64.8878	72.9129	65.3325
SVM	64.4835	48.8983	38.8518	51.8496	62.9877	52.6986	49.8686
LR	***73.0746***	51.2634	**70.2446**	24.2773	**72.1649**	30.1799	37.6188
BN	56.8830	**68.0412**	57.8128	26.3796	59.3087	59.1267	67.0507

Table 4.13: Classification results when resident A day 1 dataset is used as training and resident B dataset is used as testing

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	31.9857	33.4366	31.9857	60.0913	60.0913	60.0913	60.0913
RF	29.8174	33.5018	29.4751	69.6120	**72.2041**	**72.0900**	69.3675
KNN	45.7124	65.6994	54.6462	**71.1933**	52.4291	**71.5194**	61.8194
NB	48.8425	66.3352	51.4998	**76.1004**	**70.8999**	**75.8885**	**70.4108**
SVM	39.6642	**71.0629**	69.0903	**71.6009**	69.9054	***82.7193***	**75.8396**
LR	33.9909	69.9217	68.6501	49.5435	29.524	69.1718	45.1092
BN	60.4336	65.8461	65.0147	60.0913	69.5142	69.7913	67.4438

Table 4.14: Classification results when resident A day 1 dataset is used as training and resident C dataset is used as testing

phase evaluations, we can conclude that Reading_ON_OFF and Naiive Bayes are the best input features combination and classifier to be used. The accuracy of this model across Resident A Day 2, Resident A Day 3, Resident B and and Resident C, when the model is trained only based on Resident A Day 1 activities, is consistent. It is within 72% to 76%. Although this accuracy seems low, this model has the benefit of not requiring continuous human labelling or retraining for each resident. The system can be trained only on a single resident and can be used to predict the activities of other residents. The system provides higher accuracy results when "No Activity" Dataset is included in training and testing the model.

4.2 Multi-Resident Activity Recognition

4.2.1 Data Collection & Preparation

For multi resident activity data collection, two individuals will perform the activities in the test room at the same time. Resident A and B, resident A and C, will live in the test room for approximately 2 hours and make sure that they are doing as much concurrent, interleaved and parallel activities. For example, while resident

A is preparing food in the kitchen, resident B is sitting on the couch watching TV in the living room. B helps A to prepare food in kitchen for a bit and goes back to living room to watch TV. A brings his cooked food to the dining area to eat together with B. Several different scenarios were chosen for activities such as (i) cooking (ii) eating (iii) sitting on couch (iv) laying on couch and (v) walking around. Similar to single resident activity the residents were asked to perform the activities as normally as they would in the real world. The sensor readings obtained from the activities performed by two individuals at the same time are stored separately and the collected sensor readings are filtered using moving average filter.

Since there are two people performing the activities and the activities performed by each individual at a given point of time can be different the data has to be labelled based on the activity performed by each individual. For example if Resident A is sitting on the couch and Resident B is cooking then the data is labelled as "A_sitting_B_cooking" for that instance. The activities are labelled manually by looking at the video footage captured. Table 4.15 shows different activities performed by resident A and resident B living in the test room at the same time and also the number of instances captured for each of the activities performed and total duration the activities being performed in hours.

The labelled data is processed to get sensor on/off state and sensor duaration. The processed data is captured in different spreadsheets and saved in csv format similar to single resident activity dataset. All possible combinations of sensor reading, on/off state and sensor duration are considered for activities performed by resident A-B and resident A-C. The csv files are converted into arff files and the acquired dataset is split into 60% train and 40% test dataset by applying resample filter in weka.

4.2.2 Experiments

The experiments have been conducted in two phases to determine the accuracy of classifier models at predicting multi resident's activities.

In the first phase, we split the dataset that was acquired from resident A-B and resident A-C performing the activities into train and test sets, to predict the activities performed by two residents at the same time.

In the second phase we combine all the dataset i.e resident A-B and resident A-C dataset and split into train and test set. In this phase, resident B and resident C are labelled as others i.e "A_eating_O_sitting" to evaluate how well the model

Resident A and B		Resident A and C	
Activities	**Number of instances**	**Activities**	**Number of instances**
A_eating_B_eating	400	A_eating_C_eating	555
A_eating_B_cooking	348	A_eating_C_cooking	4
A_eating_B_sitting	583	A_eating_C_sitting	519
A_eating_B_walking_around	288	A_eating_C_walking_around	212
A_no_activity_B_sitting	47	A_no_activity_C_sitting	416
A_no_activity_B_walking_around	110	A_no_activity_C_walking_around	227
A_cooking_B_eating	237	A_cooking_C_eating	74
A_cooking_B_sitting	118	A_cooking_C_sitting	723
A_cooking_B_walking_around	8	A_cooking_C_walking_around	139
A_sitting_B_eating	380	A_sitting_C_eating	19
A_sitting_B_cooking	480	A_sitting_C_cooking	611
A_sitting_B_sitting	362	A_sitting_C_sitting	161
A_sitting_B_walking_around	427	A_sitting_C_walking_around	4
A_walking_around_B_eating	303	A_walking_around_C_eating	521
A_walking_around_B_laying	24	A_walking_around_C_laying	743
A_walking_around_B_cooking	8	A_walking_around_C_cooking	37
A_walking_around_B_sitting	384	A_walking_around_C_sitting	12
A_walking_around_B_walking_around	234	A_walking_around_C_walking_around	45
Total duration	**2:00**	**Total duration**	**1:40**

Table 4.15: Number of instances collected for Multi Resident Activity performed by resident A, B and C

can predict the activities of resident A and other two people when it is trained and tested with all the three at the same time.
In both the phases, we will compare the performance of the model trained with DT *(Decision Tree)*, RF *(Random Forest)*, KNN *(K- Nearest Neighbours)*, NB *(Naiive Bayes)*, SVM *(Support Vector Machine)* and BN *(Bayes Net)* to determine different combination of activities such as eating, cooking, sitting on couch, laying on couch and walking around performed by two individulas in the test room at the same time.

4.2.3 Results & Evaluation

To identify which set of features and classifiers are the best for predicting multi resident activities we compare the model trained with sensor duration, on/off state, reading and any possible combinations of these three features. Similar to single resident evaluation, we use accuracy (in%) as our evaluation criteria.

First Phase Evaluation

Table 4.16 illustrates the results of our first phase evaluation on resident A-B. The best feature and classifier for resident A-B are Reading_Duration and RF *(Random Forest)* with 97% accuracy. However, from the table we can see that there is not much accuracy difference between Reading_Duration and Reading_ON _ OFF_Duration. But the accuracy has dropped by more than 10% if only ON_OFF feature is used. This suggests that we must use duration, ON/OFF and reading features in combination to predict the activities of multiple people with good accuracy.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	93.2281	83.1741	93.4774	92.0233	94.059	91.8571	94.1421
RF	95.6377	83.548	95.5546	96.3855	***97.8812***	95.8039	97.6735
KNN	93.3112	82.177	91.2339	89.3228	93.4358	89.6136	91.8155
NB	88.5334	79.4765	89.4059	85.2098	90.7769	85.7914	90.6107
SVM	73.9925	82.7586	90.86	86.5808	90.86	91.3585	94.1005
BN	91.774	79.5596	90.0706	94.3498	96.3855	93.0619	94.8899

Table 4.16: Classification results with Resident A-B dataset split into training and testing set

Random forest classifier achieves 100% accuracy in predicting four different combination of activities performed by resident A and resident B. These activities are bolded in the confusion matrix table 4.18. The lowest accuracy was found in predicting walking around activity of 55%.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	93.3502	78.5688	93.6393	89.0857	94.1814	88.7243	93.82
RF	96.0969	77.7015	95.3741	95.1572	***97.5425***	95.1934	97.1449
KNN	95.4463	77.4846	91.0372	85.0018	94.3621	86.0137	91.4348
NB	85.6162	75.7499	87.64	82.1467	89.5193	84.785	89.9892
SVM	73.4369	78.1352	89.4471	79.6169	91.0011	87.4232	92.9888
BN	93.1695	76.798	90.9288	93.567	95.88	93.4948	94.7597

Table 4.17: Classification results with Resident A-C dataset split into training and testing set

Table 4.17, shows the results obtained from the classifiers for resident A-C dataset split into train and test set. The best feature and classifier for resident A-C are Reading_Duration and RF *(Random Forest)* with 97.88% accuracy.

Random forest classifier can predict A_eating_C_eating, A_sitting_C_eating, A_sitting_C_cooking, A_noactivity_C_sitting with 100% accuracy as shown in

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	classified as
85	0	0	0	1	4	0	0	0	1	0	0	1	0	0	0	2	0	a = A_walking_around_B_walking_around
0	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	b = A_walking_around_B_cooking
0	0	192	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**c = A_sitting_B_cooking**
0	0	3	145	0	1	0	0	0	0	3	0	0	0	0	0	0	0	d = A_sitting_B_eating
0	0	0	2	169	0	0	0	0	0	0	0	0	0	0	0	0	0	e = A_sitting_B_walking_around
0	0	0	0	0	121	0	0	0	1	0	0	0	0	0	0	0	0	f = A_walking_around_B_eating
0	0	0	0	0	0	95	0	0	0	0	0	0	0	0	0	0	0	**g = A_cooking_B_eating**
0	0	0	0	0	0	3	1	0	0	0	0	0	0	0	0	0	0	h = A_cooking_B_walking_around
0	0	0	0	0	0	0	0	48	0	0	0	0	0	0	0	0	0	**i = A_cooking_B_sitting**
3	0	0	0	0	0	0	0	3	145	3	0	0	0	0	0	0	0	j = A_walking_around_B_sitting
0	0	2	0	0	0	0	0	0	0	231	1	0	0	0	0	0	0	k = A_eating_B_sitting
0	0	0	0	0	0	0	0	0	1	1	113	1	0	0	0	0	0	l = A_eating_B_walking_around
0	0	0	0	0	0	0	0	0	0	0	0	160	0	0	0	0	0	**m = A_eating_B_eating**
0	0	0	0	0	0	0	0	0	7	0	0	0	3	0	0	0	0	n = A_walking_around_B_laying
1	0	0	0	0	0	0	0	0	0	0	0	0	0	139	0	0	0	o = A_eating_B_cooking
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	145	0	0	p = A_sitting_B_sitting
1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	42	0	q = A_no_activity_B_walking_around
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	**r = A_no_activity_B_sitting**

Table 4.18: Confusion matrix for multi resident activity of resident A-B dataset split into train and test set by Random Forest classifier

the table 4.19.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	classified as
10	1	0	0	0	2	1	0	0	0	0	1	1	0	0	0	0	2	a = A_walking_around_C_walking_around
0	54	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	b = A_cooking_C_walking_around
0	0	286	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	c = A_cooking_C_sitting
0	0	2	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	d = A_walking_around_C_sitting
0	0	0	0	201	0	0	0	0	0	0	0	0	0	7	0	0	0	e = A_eating_C_sitting
0	0	0	0	0	83	0	0	0	0	0	0	0	0	0	0	0	2	f = A_eating_C_walking_around
0	3	0	0	0	1	79	0	1	0	0	0	0	0	0	0	0	7	g = A_no_activity_C_walking_around
0	0	0	0	0	0	0	222	0	0	0	0	0	0	0	0	0	0	**h = A_eating_C_eating**
0	0	0	0	0	0	2	2	204	0	0	0	0	0	0	0	0	1	i = A_walking_around_C_eating
0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	**j = A_sitting_C_eating**
0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	k = A_sitting_C_walking_around
0	0	0	0	0	0	0	0	0	0	0	245	0	0	0	0	0	0	**l = A_sitting_C_cooking**
1	1	0	0	0	0	0	0	0	0	0	3	9	0	0	0	0	1	m = A_walking_around_C_cooking
0	0	0	0	0	0	0	0	1	0	0	0	0	29	0	0	0	0	n = A_cooking_C_eating
0	0	0	0	0	0	0	0	0	0	0	0	0	0	167	0	0	0	**o = A_no_activity_C_sitting**
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	64	0	0	p = A_sitting_C_sitting
0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	q = A_eating_C_cooking
0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	294	r = A_walking_around_C_laying

Table 4.19: Confusion matrix for multi resident activity of resident A-C dataset split into train and test set by Random Forest classifier

Second Phase Evaluation

Table 4.20 shows the classification results for resident A-B and resident A-C dataset combined together and split into 60% train and 40% test set. In this phase we consider resident B and C as others to determine how accurately the model can predict if it is trained with all the residents. Similar to the results obtained from our first phase evaluation, the best classifier and feature for predicting the activities of resident's A-B and resident A-C are Reading_Duration and RF *(Random Forest)*.

Algorithm	Duration	ON_OFF	ON_OFF_Duration	Reading	Reading_Duration	Reading_ON_OFF	Reading_ON_OFF_Duration
DT	90.951	75.0173	90.8358	89.2931	94.0364	90.1451	94.1515
RF	94.0824	75.1784	93.2075	95.0495	***97.4672***	94.7962	96.7994
KNN	92.1483	74.5798	88.8326	86.1847	92.5167	86.3689	90.5595
NB	76.7212	71.1259	79.323	77.2047	86.6222	80.175	86.7603
SVM	65.761	73.6357	85.7011	79.4152	88.1418	86.8524	92.5858
BN	88.4412	71.7016	85.7472	87.9576	93.2996	88.5102	92.0562

Table 4.20: Classification results with Resident A-B+ Resident A-C dataset split into training and testing set

Random Forest classifier achieved 100% accuracy in predicting Resident A sitting and other resident cooking. The second best activity which could be predicted with 99.7% accuracy was eating activity as shown in the confusion matrix table 4.21. From our evaluation we can conclude that Reading_Duration and Random Forest

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	classified as
381	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	a = A_eating_O_eating
0	140	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	b = A_eating_O_cooking
0	0	439	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	c = A_eating_O_sitting
0	0	5	183	0	6	0	0	0	0	0	0	0	0	2	0	1	3	d = A_eating_O_walking_around
0	0	0	0	184	1	0	0	0	0	0	0	0	0	1	0	0	0	e = A_no_activity_O_sitting
0	0	0	2	0	126	0	0	0	0	0	0	1	1	2	0	1	2	f = A_no_activity_O_walking_around
0	0	0	0	0	0	123	0	1	0	0	0	0	1	0	0	0	0	g = A_cooking_O_eating
0	0	0	0	0	0	0	336	0	0	0	0	0	0	1	0	0	0	h = A_cooking_O_sitting
0	0	0	0	0	0	2	1	56	0	0	0	0	0	0	0	0	0	i = A_cooking_O_walking_around
0	0	0	0	0	0	0	0	0	156	2	0	1	1	0	0	0	0	j = A_sitting_O_eating
0	0	0	0	0	0	0	0	0	0	437	0	0	0	0	0	0	0	**k = A_sitting_O_cooking**
0	0	0	0	1	0	0	0	0	0	0	207	1	0	1	0	0	0	l = A_sitting_O_sitting
0	0	1	0	0	0	0	0	0	2	0	0	168	0	0	0	1	1	m = A_sitting_O_walking_around
6	0	0	0	0	1	0	0	0	0	0	0	0	322	1	0	0	0	n = A_walking_around_O_eating
0	0	1	0	0	2	0	0	0	0	0	0	0	0	296	0	8	0	o = A_walking_around_O_laying
0	1	0	0	0	0	1	0	1	0	3	0	0	0	0	12	0	0	p = A_walking_around_O_cooking
0	0	0	1	1	0	0	2	0	0	1	0	1	0	3	0	146	4	q = A_walking_around_O_sitting
0	0	0	1	0	4	0	0	0	0	1	0	0	1	1	3	3	98	r = A_walking_around_O_walking_around

Table 4.21: Confusion matrix for multi resident activity of resident A-B+A-C dataset split into train and test set by Random Forest classifier

is the best feature and classifier for predicting the activities of two people performing the activities in the test room at the same time.

Conclusion

By knowing what is the activity being performed by a Resident can we identify the the Resident? In this chapter, we were successful in predicting the activities performed Single and Multiple Residents with more than 90% accuracy results. In the next chapter, we discuss how to identify the Resident of the house by considering the activities performed by the resident.

Chapter 5

Resident Identification

Resident identification using non intrusive sensors enables numerous applications in smart home environment such as personalization of climate, lighting and security. In our proposed research, we identify the resident of the house using the activities performed such as cooking, eating, sitting on couch, laying on couch and walking around. To the best of our knowledge, research into resident identification using ultrasonic sensors is very limited. As discussed in the earlier chapter there can be single resident or multiple residents in a house. We identify the resident of the house in a single resident environment considering three individuals performing the same activities using machine learning algorithms.

5.1 Data Preparation

The data acquired for single resident activity recognition is considered for resident identification. Previously, the sensor readings acquired from the ultrasonic sensors were labelled with the activity being performed. Now, we consider each activity and label the data with resident himself. The sample labelled data for eating activity is as shown in the table 5.1 Each activity performed by the resident is considered as a separate csv file and labelled with the resident himself. Since we are eager to find what are the best features and classifiers to identify the resident we consider sensor on/off state and duration.

5.2 Experiments

To investigate the efficacy of our proposed features and classifier models in identifying the resident, we performed our experiments in two phases.

Eating												
S1	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**	**S11**	**S12**	**Resident**
182	181	183	225	250	216	202	167	255	182	159	153	Resident A
182	182	192	226	250	246	248	216	248	181	132	176	Resident B
182	185	185	219	251	250	252	235	251	183	155	157	Resident C

Table 5.1: Data labelling for resident identification

In the first phase, we combine each activity performed by all the residents into a single file and split them into 60% train and 40% test dataset. For example, the eating activity performed by all the residents is considered in one file and divided into train and test set and run the classifiers to identify the resident. This is repeated for sitting on couch, cooking, walking around and laying on couch activity by considering sensor on/off, duration and all possible combinations of reading, on/off and duration.

In the second phase, we combine the activities performed by all residents similar to first phase but we label Resident B and C as others to check how well the classifier can predict Resident A considering Resident B and C as others.

In both phases, we will compare the performance of the model trained with DT *(Decision Tree)*, RF *(Random Forest)*, KNN *(K-Nearest Neighbours)*, NB *(Naiive Bayes)*, SVM *(Support Vector Machine)*, LR *(Logistic Regression)* and BN *(Bayes Network)* using WEKA to Identify the resident.

5.3 Results & Evaluation

To identify which set of fetaures and classiers are best for identifying the resident based on the activities we compare the model trained with sensor reading, duration, on/off and all possibble combinations of these three features. We use accuracy (in%) as our evaluation criteria.

First Phase

Eating Activity

Table 5.2 shows the results obtained from the classifiers from combining resident A, B and C eating activity dataset and splitting into training and testing set. The best feature and classifier for identifying the resident's are Reading and Logistic Regression with 61.59% accuracy. Logistic Regression classifier can identify Resident

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	58.7513	60.5382	60.366	53.9074	59.2465	59.1173	54.4948
RF	37.761	37.5888	58.3423	37.761	37.5673	37.6103	35.452
KNN	37.7395	37.5242	58.4069	37.6749	37.5673	37.6319	35.5027
NB	60.4952	41.1625	60.5382	61.3132	57.0075	48.0301	60.1165
SVM	60.8611	60.3875	60.5597	60.5382	60.3875	60.2368	59.5087
LR	61.5931	60.5813	59.9569	60.9688	61.1841	61.0118	61.1041
BN	61.4855	60.3875	60.1507	61.5285	61.5285	60.6889	61.408

Table 5.2: Classifier results for eating activity with Resident A,B,C dataset split into train and test set

A with 75% accuracy compared to Resident B with 38% accuracy and Resident C with 40% accuracy as shown in the confusion matrix table 5.3.

a	b	c	classified as
2135	135	557	a = Resident A
210	133	0	b = Resident B
882	0	593	c = Resident C

Table 5.3: Confusion Matrix for Eating Activity with Resident A,B,C dataset by Logistic Regression Classifier.

Laying on couch Activity

Table 5.4 shows the results obtained from the classifers by considering laying on couch activity performed by the residents and splitting them into train and test set. The best feature and classifier for identifying the resident while laying on couch are Duration and Logistic Regression with 71% accuracy.

From the confusion matrix table 5.5, logistic regression classifier can identify Resident A with 91.7% accuracy. Since resident C did not perform the laying on couch activity for a longer duration the classifier does not have sufficient dataset to train and identify Resident C leading to lower accuracy.

Cooking Activity

Table 5.6 shows the results obtained from the classifers by considering cooking activity performed by the residents and splitting them into train and test set. The best

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	66.6982	70.8136	70.1514	63.1031	66.6982	70.8136	65.2948
RF	53.264	52.7436	69.7729	53.264	53.122	52.649	51.7798
KNN	53.3586	52.5544	69.7729	53.2167	53.2167	52.7436	51.8354
NB	70.2933	67.3132	70.719	70.1987	70.2933	70.3879	70.1891
SVM	69.2526	70.4352	70.246	68.9215	69.7256	70.4352	67.9088
LR	69.7729	***71.2394***	70.3879	69.3472	69.3945	70.6717	68.02
BN	70.246	70.3879	70.3879	70.1987	70.0095	70.246	69.4661

Table 5.4: Classifier results for Laying on couch activity with Resident A,B,C dataset split into train and test set

a	b	c	classified as
1373	124	0	a = Resident A
479	133	0	b = Resident B
3	0	2	c = Resident C

Table 5.5: Confusion Matrix for Laying on couch Activity with Resident A,B,C dataset by Logistic Regression Classifier.

feature and classifier for identifying the resident while cooking Reading_Duration and Bayes Network with 79% accuracy.

Bayes Network classifier can identify Resident B and Resident C with 100% accu-

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	75.2168	78.1069	76.5173	73.1936	75.289	75	75.2761
RF	66.0405	66.0405	75.289	66.0405	66.0405	65.9682	65.2506
KNN	66.0405	65.4624	75.289	65.8237	66.1127	65.5347	65.2506
NB	78.3237	37.211	78.1792	78.3237	77.3121	67.052	78.3347
SVM	77.4566	78.0347	77.6012	76.8064	77.3844	77.3121	76.4656
LR	78.3237	77.7457	77.3121	76.5173	77.3121	76.7341	75.7009
BN	78.685	77.4566	77.9624	78.685	***78.8572***	77.6734	78.8445

Table 5.6: Classifier results for cooking activity with Resident A,B,C dataset split into train and test set

racy while cooking activity is being performed by the Residents. 202 instances of Resident B is identified as Resident A and 92 instances of Resident C is identified as Resident A leading to 73% accuarcy in Identifying Resident A as shown in confusion matrix table 5.7

a	b	c	classified as
794	202	92	a = Resident A
0	210	0	b = Resident B
0	0	86	c = Resident C

Table 5.7: Confusion Matrix for Cooking Activity with Resident A,B,C dataset by Bayes Network Classifier.

Sitting on couch Activity

Table 5.8 shows the results obtained from the classifers by considering sitting on couch activity performed by the residents and splitting them into train and test set. The best feature and classifier for identifying the resident while sitting on couch activity being performed are Duration and Logistic Regression with 71% accuracy.

Logistic Regression classifier can identify Resident A with 94% accuracy while

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	65.8717	71.0708	71.0708	65.2516	65.6332	71.0708	66.6105
RF	53.2077	54.9726	70.2361	53.2077	53.0885	54.9726	51.3187
KNN	53.2554	55.7119	70.2361	53.2554	53.0169	55.7358	51.3187
NB	70.3792	46.6015	69.4968	70.3077	70.3077	46.2437	70.3984
SVM	69.783	71.0231	70.9993	69.7591	69.5683	70.8323	69.9214
LR	70.5461	***71.4286***	70.9516	70.26	69.6637	70.9277	69.3603
BN	70.3792	70.9754	71.047	70.3792	70.3554	69.6637	70.3704

Table 5.8: Classifier results for sitting on couch activity with Resident A,B,C dataset split into train and test set

sitting on couch activity is being performed by the Residents. Majority of the observations from Resident B and C is identified as Resident A as shown in confusion matrix table 5.9.

a	b	c	classified as
2806	170	4	a = Resident A
581	181	0	b = Resident B
443	0	8	c = Resident C

Table 5.9: Confusion Matrix for Sitting on couch Activity with Resident A,B,C dataset by Logistic Regression Classifier.

Walking Around Activity

Table 5.10 shows the results obtained from the classifers by considering walking around activity performed by the residents and splitting them into train and test set. The best feature and classifier for identifying the resident while walking around activity being performed are Reading_ON_OFF_Duration and Logistic Regression with 72% accuracy.

Logistic Regression classifier can identify Resident A with 87% accuracy while

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	71.0833	71.0833	71.0833	65.085	71.0833	68.2184	66.1053
RF	54.7001	54.8791	63.4736	54.7896	54.7896	54.6106	53.4737
KNN	52.2829	53.7153	64.1003	52.8201	52.8201	52.9096	51.5789
NB	61.1459	45.1209	70.5461	66.3384	54.7896	61.0564	61.4737
SVM	71.0833	70.8147	71.0833	71.0833	70.8147	70.8147	70.7368
LR	70.2775	69.7404	71.0833	69.7404	69.6509	70.0833	70.5263
BN	72.3366	71.3518	71.2623	72.4261	72.5157	69.7404	***72.5263***

Table 5.10: Classifier results for walking around activity with Resident A,B,C dataset split into train and test set

walking around activity is being performed by the Residents. 138 observations of Resident B is identified as Resident A and 58 instances of Resident C is identified as Resident A as shown in confusion matrix table5.9.

a	b	c	classified as
445	151	79	a = Resident A
26	153	0	b = Resident B
5	0	91	c = Resident C

Table 5.11: Confusion Matrix for walking around Activity with Resident A,B,C dataset by Bayes Network Classifier.

From the first phase evaluation we can say that the best fetaures and classfiers for identifying a Resident varies based on the activity. But Sensor Duration and Logistic Regression was found to be the best feature and classifier for identifying the Resident's with more than 70% accuracy considering any activity.

Second Phase Evaluation

Since the accuracy of the classifiers in identifying the Resident's B and C was found to be low in the first phase, we labelled Resident B and C as others in second phase evaluation to evaluate how well can the classifier identify Resident B and C labelled as others.

Eating Activity

Table 5.12 shows the results obtained from the classifers by considering eating activity performed by the residents, splitting them into train and test set and labelling Resident B and C as others. The best feature and classifier while eating activity being performed are Reading_ON_OFF_Duration and Bayes Network with 62% accuracy.

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	60.8611	60.8611	59.1604	56.2971	60.8611	58.9666	55.3698
RF	38.1485	37.9978	58.084	38.1485	37.9763	37.9548	36.1196
KNN	38.127	37.9117	58.084	38.1485	37.9763	37.9763	36.1196
NB	59.1173	60.8827	60.2368	60.6889	60.3445	60.9042	61.1449
SVM	60.8611	60.732	60.8181	60.366	60.7535	60.7966	60.1064
LR	61.0334	60.8611	60.6674	60.3875	61.1625	60.6674	61.0689
BN	60.6459	61.141	60.2153	60.9042	60.7535	60.4736	***61.8541***

Table 5.12: Classifier results for eating activity with Resident A,B,C dataset split into train and test set while Resident B,C labelled as others

Bayes Network classifier can identify others with 88% accuracy while eating activity is being performed by the Residents. 1335 observations of Resident A is classified as others leading to 45% accuracy in identifying Resident A as shown in confusion matrix table 5.9.

a	b	classified as
1068	1335	a = Resident A
171	1374	b = others

Table 5.13: Confusion Matrix for Eating Activity with Resident A,B,C dataset labelled as Resident A and others by Bayes Network Classifier.

Laying on couch Activity

Table 5.14 shows the results obtained from the classifers by considering laying on couch activity performed by the residents, splitting them into train and test set and labelling Resident B and C as others. The best feature and classifier while eating activity being performed are Reading_ON_OFF and Naiive Bayes with 71% accuracy.

Naiive Bayes classifier can identify others with 98% accuracy while laying on couch

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	67.0293	70.5298	69.8675	63.245	66.7455	69.8675	65.3868
RF	52.9328	52.4598	69.0634	52.9328	52.8382	52.4598	50.8625
KNN	52.8382	52.5071	69.0634	53.0747	52.8855	52.649	50.8625
NB	70.9555	69.8675	70.5298	***71.0501***	70.9555	70.9555	70.1169
SVM	69.0634	70.0095	70.1514	69.3472	69.8675	69.9149	67.557
LR	70.9082	70.6244	70.1987	69.7729	69.1107	70.1041	68.2805
BN	70.8609	70.6244	70.0568	70.6717	70.7663	70.1987	69.7273

Table 5.14: Classifier results for laying on couch activity with Resident A,B,C dataset split into train and test set while Resident B,C labelled as others

activity is being performed by the Residents. 604 observations of others is classified as Resident A leading to 59.7% accuracy in identifying Resident A as shown in confusion matrix table 5.15.

a	b	classified as
893	604	a = Resident A
8	609	b = others

Table 5.15: Confusion Matrix for Laying on couch Activity with Resident A,B,C dataset labelled as Resident A and others by Bayes Network Classifier.

Cooking Activity

Table 5.16 shows the results obtained from the classifers by considering cooking activity performed by the residents, splitting them into train and test set and labelling Resident B and C as others. The best feature and classifier while cooking activity being performed are Reading_ON_OFF_Duration and Naiive Bayes with 79% accuracy.

Naiive Bayes classifier can identify Resident A with 84% accuracy while cooking activity is being performed by the Residents. 100 observations of others is classified as Resident A leading to 60% accuracy in identifying others as shown in confusion matrix table 5.17.

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	74.7832	78.6127	76.3728	75.289	74.4942	76.0838	71.6837
RF	66.3295	66.2572	75.5058	66.3295	66.2572	66.2572	64.5408
KNN	66.4017	65.7514	75.5058	66.2572	66.185	65.7514	64.5408
NB	78.396	78.0347	78.2514	78.2514	78.5405	77.8179	***79.1667***
SVM	78.1792	77.6012	78.3237	77.3121	77.5289	77.7457	77.0408
LR	78.396	78.1792	77.9624	77.8902	77.3844	77.3121	75.4252
BN	78.7572	78.3237	78.6127	78.9017	78.8295	78.1069	79.0816

Table 5.16: Classifier results for cooking activity with Resident A,B,C dataset split into train and test set while Resident B,C labelled as others

a	b	classified as
780	145	a = Resident A
100	151	b = others

Table 5.17: Confusion Matrix for Cooking Activity with Resident A,B,C dataset labelled as Resident A and others by Bayes Network Classifier.

Sitting on couch Activity

Table 5.18 shows the results obtained from the classifers by considering sitting on couch activity performed by the residents, splitting them into train and test set and labelling Resident B and C as others. The best feature and classifier while sitting on couch activity being performed are Duration and Support Vector Machine with 71% accuracy.

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	70.4031	70.7131	70.6177	68.2328	70.4031	70.5223	68.4343
RF	53.2554	54.6864	69.8545	53.2554	53.0408	54.6387	51.0943
KNN	53.4462	55.5211	69.7353	53.4224	52.9931	55.5211	51.0101
NB	70.4031	69.2583	70.8085	70.4269	70.4031	70.4746	70.3984
SVM	69.9738	***70.9039***	70.7131	70.1407	69.9261	70.7131	69.3322
LR	70.6654	70.6415	70.5938	70.6415	70.1407	70.2361	70.1178
BN	70.4031	70.7846	70.6415	70.4984	70.4031	70.4508	70.5107

Table 5.18: Classifier results for sitting on couch activity with Resident A,B,C dataset split into train and test set while Resident B,C labelled as others

SVM classifier can identify Resident A with 95% accuracy while sitting on couch activity is being performed by the Residents. 1070 observations of others is classified as Resident A leading to 12% accuracy in identifying others as shown in confusion matrix table 5.19.

a	b	classified as
2830	150	a = Resident A
1070	143	b = others

Table 5.19: Confusion Matrix for Sitting on couch Activity with Resident A,B,C dataset labelled as Resident A and others by Bayes Network Classifier.

Walking Around Activity

Table 5.20 shows the results obtained from the classifers by considering walking around activity performed by the residents, splitting them into train and test set and labelling Resident B and C as others. The best feature and classifier while walking around activity being performed are Reading_Duration and Bayes Network with 73% accuracy.

Algorithm	Reading	Duration	ON_OFF	Reading_ON_OFF	Reading_Duration	ON_OFF_Duration	Reading_ON_OFF_Duration
DT	71.0833	71.0833	71.0833	65.8908	71.0833	71.0833	64.5263
RF	55.2372	55.3268	65.1746	55.3268	55.2372	52.3268	54.3158
KNN	53.1782	53.6258	65.4432	54.2525	53.2677	53.5363	52.7368
NB	68.1289	70.0985	69.8299	70.8147	70.6356	70.9042	70.5263
SVM	71.0833	71.0833	71.0833	71.0833	71.0833	71.0833	70
LR	71.0833	70.8147	71.7995	71.5309	70.4566	70.9042	69.5789
BN	72.4261	71.0833	70.7252	**73.2319**	73.2319	71.3518	72.7368

Table 5.20: Classifier results for walking around activity with Resident A,B,C dataset split into train and test set while Resident B,C labelled as others

Bayes Network classifier can identify others with 93% accuracy while walking around activity is being performed by the Residents. 277 observations of Resident A is classified as others leading to 65% accuracy in identifying Resident A as shown in confusion matrix table 5.21. From the two phases of evaluation, we can conclude

a	b	classified as
517	277	a = Resident A
22	301	b = others

Table 5.21: Confusion Matrix for walking around Activity with Resident A,B,C dataset labelled as Resident A and others by Bayes Network Classifier.

that Resident can be identified with high accuracy of more than 75% while cooking activity is being performed and using Sensor features such as on/off and duration in combination with reading the classifier can provide better results in identifying the resident.

Conclusuion

This work adds to the scant literature on resident identification using non intrusive ultrasonic sensors. Although the features such as reading, on/off and duration provide lower accuracy this work presents how a person can be identified by just using the activity data.

Chapter 6

Conclusion

Previous research conducted in smart home systems using ultrasonic sensors completely rely on sensor readings to predict the activities. Although sensor reading's provide good results our research shows that using additional features such as sensor ON/OFF state and Duration in combination with reading improve the accuracy of activity recognition.

Collecting activity data in a smart home setup and accuractely labelling those activities require researchers to monitor the residents activities through a multi media system - defeating the purpose of using ambient sensors. For single resident activity recognition, we create a system that is able to predict the different activities a person can do in their daily life regardless of the person who is doing the activities. Our tesing evaluation of training a system based on participant's activities on day 1 and testing the system on the same participant's activities performed on day 2 and 3 showed performance decay and some features/ classifier's are much more volatile. on the other hand, we were able to identify what are the best features and classifiers that could give consistent performance without the need for retraining the system with other participants.

Random Forest and Sensor Reading_ON_OFF_Duration are the best classifier and features to recognise the activities of an individual if the system is trained and tested with the same individual. Naiive Bayes and Sensor_ON_OFF are found to be best classifier and feature for recognizing the activities of any person by training the system with only one person. For predicting multi-resident activities considering two residents performing the activities in the test room at the same time we found that sensor Reading_Duration and Random Forest are the best feature and classifier with more than 95% accuracy.

Finally we try to identify the resident's based on the activities, by training the ma-

chine learning models using features such as sensor reading, on/off and duration and also identify what are the best features and classfiers for resident identification. We found that while residents are performing cooking activity our system is able to identify the resident with more than 75% accuracy using Reading and Duration.
In the future we would like to perform the experiment by five factors namely (i) Sensor placement (ii) Features (iii) Furnitures (iv) Activities and (v) Behaviour .

- **Sensor placement** - Finding the best possible distance between the sensors which will not interfere the ultrasound signals and also cover more area which in turn- reduces the number of sensors required for area coverage.
  Placing the sensors on the side walls with sensors placed on cieling can improve the accuracy in activity recognition.

- **Features** - Adding more features to the dataset such as number of sensors ON/OFF , combination of sensors being ON/OFF, total number of sensors switched ON for an activity etc. can be used to improve the accuracy of the system.

- **Furnitures** - In our test environment, furnitures are considered to be part of the environment and are placed in the same positions. If the couch is moved to a different position in the same test room then our system fails to predict the activities. Training the machine learning model with all possible furniture positions in the test room is our longer timeline.

- **Activities** - The activities considered in our proposed system are limited to sitting, eating, cooking and walking around. There are high level activities and transition activities which need to be considered for activity recognition.

- **Behaviour** - Adding timestamp to the activity data and collecting it for over a period of time to detect the behaviour of the resident which can add more value to identify the resident.

Bibliography

[1] G. Hugo and A. SA, "The demographic facts of ageing in australia," *available at the website to the DSS website: `www.dss.gov.au`*, 2017.

[2] M. Khawaja and B. Boddington, "Informing the debate on population ageing in new zealand: the role of statistics new zealand," *New Zealand Population Review*, vol. 36, p. 117, 2010.

[3] H. O. W. I. an Organisation, "Ageing workforce in the new zealand crown entity sector survey report 2014,"

[4] Y. Huang and K. Newman, "Improve quality of care with remote activity and fall detection using ultrasonic sensors," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5854–5857, IEEE, 2012.

[5] T. Gu, L. Wang, Z. Wu, X. Tao, and J. Lu, "A pattern mining approach to sensor-based human activity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 9, pp. 1359–1372, 2010.

[6] S. Hussain, S. Schaffner, and D. Moseychuck, "Applications of wireless sensor networks and rfid in a smart home environment," in *2009 Seventh Annual Communication Networks and Services Research Conference*, pp. 153–157, IEEE, 2009.

[7] T. W. Hnat, E. Griffiths, R. Dawson, and K. Whitehouse, "Doorjamb: unobtrusive room-level tracking of people in homes using doorway sensors," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, pp. 309–322, ACM, 2012.

[8] C. Nadee and K. Chamnongthai, "Octagonal formation of ultrasonic array sensors for fall detection," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–4, IEEE, 2016.

[9] G. Mokhtari, Q. Zhang, G. Nourbakhsh, S. Ball, and M. Karunanithi, “Bluesound: A new resident identification sensor—using ultrasound array and ble technology for smart home platform,” *IEEE Sensors Journal*, vol. 17, no. 5, pp. 1503–1512, 2017.

[10] T. Hori, “Overview of digital human modeling,” in *IROS2001 Workshop Tutorial Note, Oct.*, pp. 1–14, 2001.

[11] U. Hansmann, L. Merk, M. S. Nicklous, and T. Stober, *Pervasive computing handbook*. Springer- Verlag Telos, 2001.

[12] G. F. Fuller, “Falls in the elderly.,” *American family physician*, vol. 61, no. 7, pp. 2159–68, 2000.

[13] J. Fogarty, C. Au, and S. E. Hudson, “Sensing from the basement: a feasibility study of unobtrusive and low-cost home activity recognition,” in *Proceedings of the 19th annual ACM symposium on User interface software and technology*, pp. 91–100, ACM, 2006.

[14] S. S. Intille, “Designing a home of the future,” *IEEE pervasive computing*, vol. 1, no. 2, pp. 76–82, 2002.

[15] S.-R. Ke, H. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, “A review on video-based human activity recognition,” *Computers*, vol. 2, no. 2, pp. 88–131, 2013.

[16] J. Steinmetz, Q. Xu, A. Fishbach, and Y. Zhang, “Being observed magnifies action.,” *Journal of personality and social psychology*, vol. 111, no. 6, p. 852, 2016.

[17] J. G. Adair, “The hawthorne effect: a reconsideration of the methodological artifact.,” *Journal of applied psychology*, vol. 69, no. 2, p. 334, 1984.

[18] G. N. G. Mokhtari, Q. Zhang, “Bluesound: A new resident identification sensor - using ultrasound array and ble technology for smart home platform,” *Sensor Journal IEEE*, vol. 17, no. 5, pp. 1503–1512, 2017.

[19] K. Culhane, G. Lyons, D. Hilton, P. Grace, and D. Lyons, “Long-term mobility monitoring of older adults using accelerometers in a clinical environment,” *Clinical rehabilitation*, vol. 18, no. 3, pp. 335–343, 2004.

[20] Y. Jang, S. Shin, J. W. Lee, and S. Kim, "A preliminary study for portable walking distance measurement system using ultrasonic sensors," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5290–5293, IEEE, 2007.

[21] T. Ishrat, M. A. Rahaman, and A. Ahammad, "Smart fan for human tracking," in *2014 9th International Forum on Strategic Technology (IFOST)*, pp. 124–127, IEEE, 2014.

[22] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, pp. 32–43, ACM, 2000.

[23] P. Lazik, N. Rajagopal, O. Shih, B. Sinopoli, and A. Rowe, "Alps: A bluetooth and ultrasound platform for mapping and localization," in *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pp. 73–84, ACM, 2015.

[24] R. Cheng, W. Heinzelman, M. Sturge-Apple, and Z. Ignjatovic, "A motion-tracking ultrasonic sensor array for behavioral monitoring," *IEEE sensors journal*, vol. 12, no. 3, pp. 707–712, 2011.

[25] K.-W. Kim, J. Kwon, C.-G. Lee, and J. Han, "Accurate indoor location tracking exploiting ultrasonic reflections," *IEEE Sensors Journal*, vol. 16, no. 24, pp. 9075–9088, 2016.

[26] C. Nadee and K. Chamnongthai, "Ultrasonic array sensors for monitoring of human fall detection," in *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1–4, IEEE, 2015.

[27] S. Biswas, B. Harrington, F. Hajiaghajani, and R. Wang, "Contact-less indoor activity analysis using first-reflection echolocation," in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2016.

[28] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *International conference on pervasive computing*, pp. 1–17, Springer, 2004.

[29] E. M. Tapia, S. S. Intille, and K. Larson, "Activity recognition in the home using simple and ubiquitous sensors," in *International conference on pervasive computing*, pp. 158–175, Springer, 2004.

[30] B. Logan, J. Healey, M. Philipose, E. M. Tapia, and S. Intille, "A long-term evaluation of sensing modalities for activity recognition," in *International conference on Ubiquitous computing*, pp. 483–500, Springer, 2007.

[31] A. Ghosh, A. Sanyal, A. Chakraborty, P. K. Sharma, M. Saha, S. Nandi, and S. Saha, "On automatizing recognition of multiple human activities using ultrasonic sensor grid," in *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, pp. 488–491, IEEE, 2017.

[32] G. Chand, M. Ali, B. Barmada, V. Liesaputra, and G. Ramirez-Prado, "Tracking a person's behaviour in a smart house," in *International Conference on Service-Oriented Computing*, pp. 241–252, Springer, 2018.

[33] T. Van Kasteren, A. Noulas, G. Englebienne, and B. Kröse, "Accurate activity recognition in a home setting," in *Proceedings of the 10th international conference on Ubiquitous computing*, pp. 1–9, ACM, 2008.

[34] S. Katz, T. D. Downs, H. R. Cash, and R. C. Grotz, "Progress in development of the index of adl," *The gerontologist*, vol. 10, no. 1_Part_1, pp. 20–30, 1970.

[35] O. Amft and K. Van Laerhoven, "What will we wear after smartphones?," *IEEE Pervasive Computing*, vol. 16, no. 4, pp. 80–85, 2017.

[36] A. Nandy, J. Saha, C. Chowdhury, and K. P. Singh, "Detailed human activity recognition using wearable sensor and smartphones," in *2019 International Conference on Opto-Electronics and Applied Optics (Optronix)*, pp. 1–6, IEEE, 2019.

[37] G. Ogbuabor and R. La, "Human activity recognition for healthcare using smartphones," in *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, pp. 41–46, ACM, 2018.

[38] I. A. Lawal and S. Bano, "Deep human activity recognition using wearable sensors," in *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 45–48, 2019.

[39] C.-H. Lu and L.-C. Fu, "Robust location-aware activity recognition using wireless sensor network in an attentive home," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 4, pp. 598–609, 2009.

[40] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," in *Proceedings 1992 IEEE Computer Society conference on computer vision and pattern recognition*, pp. 379–385, IEEE, 1992.

[41] H. Rimminen, J. Lindström, M. Linnavuo, and R. Sepponen, "Detection of falls among the elderly by a floor sensor using the electric near field," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 6, pp. 1475–1476, 2010.

[42] G. Vagni and B. Cornwell, "Patterns of everyday activities across social contexts," *Proceedings of the National Academy of Sciences*, vol. 115, no. 24, pp. 6183–6188, 2018.

[43] M. D. S. Ainsworth, M. C. Blehar, E. Waters, and S. N. Wall, *Patterns of attachment: A psychological study of the strange situation.* Psychology Press, 2015.

[44] D. Simon and T. L. Chia, "Kalman filtering with state equality constraints," *IEEE transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 128–136, 2002.

[45] L. Tong, Q. Song, Y. Ge, and M. Liu, "Hmm-based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849–1856, 2013.

[46] V. K. Singh, L. Freeman, B. Lepri, and A. S. Pentland, "Predicting spending behavior using socio-mobile features," in *2013 International Conference on Social Computing*, pp. 174–179, IEEE, 2013.

[47] J.-K. Min, J. Wiese, J. I. Hong, and J. Zimmerman, "Mining smartphone data to classify life-facets of social relationships," in *Proceedings of the 2013 conference on Computer supported cooperative work*, pp. 285–294, ACM, 2013.

[48] R. Mafrur, I. G. D. Nugraha, and D. Choi, "Modeling and discovering human behavior from smartphone sensing life-log data for identification purpose," *Human-centric Computing and Information Sciences*, vol. 5, no. 1, p. 31, 2015.

[49] S. Luhr, S. Venkatesh, G. West, and H. H. Bui, "Duration abnormality detection in sequences of human activity," *Curtin University of Technology, Perth*, 2004.

[50] S. Lühr, S. Venkatesh, G. West, and H. H. Bui, "Explicit state duration hmm for abnormality detection in sequences of human activity," in *Pacific rim international conference on artificial intelligence*, pp. 983–984, Springer, 2004.

[51] K. Khoshhal, H. Aliakbarpour, K. Mekhnacha, J. Ros, J. Quintas, and J. Dias, "Lma-based human behaviour analysis using hmm," in *Doctoral Conference on Computing, Electrical and Industrial Systems*, pp. 189–196, Springer, 2011.

[52] K. Hara, T. Omori, and R. Ueno, "Detection of unusual human behavior in intelligent house.," in *NNSP*, pp. 697–706, 2002.

[53] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Automatic interpretation and coding of face images using flexible models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, p. 743–756, July 1997.

[54] Andrew.K.Hrechak and J. A.McHugh, "Automated fingerprint recognition using structural matching," *Pattern Recognition*, vol. 23, no. 8, pp. 893 – 904, 1990.

[55] S. Pan, A. Chen, and P. Zhang, "Securitas: user identification through rgb-nir camera pair on mobile devices," in *SPSM '13*, 2013.

[56] C.-L. Tisse, L. Martin, L. Torres, and M. Robert, "Person identification technique using human iris recognition," in *Proc. of Vision Interface*, pp. 294–299, 2002.

[57] V. Srinivasan, J. Stankovic, and K. Whitehouse, "Using height sensors for biometric identification in multi-resident homes," in *International Conference on Pervasive Computing*, pp. 337–354, Springer, 2010.

[58] J. Jenkins and C. Ellis, "Using ground reaction forces from gait analysis: Body mass as a weak biometric," in *International conference on pervasive computing*, pp. 251–267, Springer, 2007.

[59] C. DeLoney, "Person identification and gender recognition from footstep sound using modulation analysis," 2008.

[60] R. J. Orr and G. D. Abowd, "The smart floor: A mechanism for natural user identification and tracking," in *CHI'00 extended abstracts on Human factors in computing systems*, pp. 275–276, ACM, 2000.

[61] A. Pedotti, "Simple equipment used in clinical practice for evaluation of locomotion," *IEEE Transactions on Biomedical Engineering*, no. 5, pp. 456–461, 1977.

[62] G. C. Santambrogio, "Procedure for quantitative comparison of ground reaction data," *IEEE transactions on biomedical engineering*, vol. 36, no. 2, pp. 247–255, 1989.

[63] Y.-l. Tian, L. Brown, A. Hampapur, M. Lu, A. Senior, and C.-f. Shu, "Ibm smart surveillance system (s3): event based video surveillance system with an open and extensible framework," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 315–327, 2008.

[64] N. Khalil, D. Benhaddou, O. Gnawali, and J. Subhlok, "Sonicdoor: scaling person identification with ultrasonic sensors by novel modeling of shape, behavior and walking patterns," in *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments*, p. 3, ACM, 2017.

[65] Y. Zeng, P. H. Pathak, and P. Mohapatra, "Wiwho: wifi-based person identification in smart spaces," in *Proceedings of the 15th International Conference on Information Processing in Sensor Networks*, p. 4, IEEE Press, 2016.

[66] J. D. Diane Cook, Aaron Crandall, "Center for advanced studies in adaptive systems (casas)," 2018 (accessed September 4 , 2019). `http://casas.wsu.edu/datasets`.

[67] N. S. Foundation, "Placelab datasets," 2005 (accessed September 4 , 2019). `http://web.mit.edu/cron/group/house_n/data/PlaceLab/PlaceLab`.

[68] J. Cumin, "Orange4home: A dataset of routine daily activities in an instrumented home," 2017 (accessed September 4 , 2019). `http://amiqual4home.inria.fr/orange4home/`.

[69] B. Mustapha, A. Zayegh, and R. K. Begg, "Ultrasonic and infrared sensors performance in a wireless obstacle detection system," in *2013 1st International Conference on Artificial Intelligence, Modelling and Simulation*, pp. 487–492, Dec 2013.

[70] L. Alonso, V. Milanés, C. Torre-Ferrero, J. Godoy, J. P. Oria, and T. De Pedro, "Ultrasonic sensors in urban traffic driving-aid systems," *Sensors*, vol. 11, no. 1, pp. 661–673, 2011.

[71] Maxbotix, "Lv-maxsonar®-eztm series high performance sonar range finder," 2015 (accessed June 6 , 2019). `https://www.maxbotix.com/documents/LV-MaxSonar-EZ_Datasheet.pdf`.

[72] L. Weidman, A. Chen, and F. Chenghao, "Method and apparatus for noise control in ultrasonic sensors," Google Patents, Jan. 8 2019. US Patent 10,175,371.

[73] A. R. Al Tahtawi, "Kalman filter algorithm design for hc-sr04 ultrasonic sensor data acquisition system," *IJITEE (International Journal of Information Technology and Electrical Engineering)*, vol. 2, no. 1, pp. 15–19.

[74] J. Hightower and G. Borriello, "Particle filters for location estimation in ubiquitous computing: A case study," in *International conference on ubiquitous computing*, pp. 88–106, Springer, 2004.

[75] B. Wirnitzer, W. M. Grimm, and H. Schmidt, "Interference cancelling in ultrasonic sensor arrays by stochastic coding and adaptive filtering," in *IEEE International Conference on Intelligent Vehicles*, Citeseer, 1998.

[76] Y. Jo and I. Jung, "Analysis of vehicle detection with wsn-based ultrasonic sensors," *Sensors*, vol. 14, no. 8, pp. 14050–14069, 2014.

[77] A. Media, "The scientist and engineer's guide to digital signal processing," 2016 (accessed March 24 , 2020). `https://www.analog.com/media/en/technical-documentation/dsp-book/dsp_book_Ch15.pdf`.

[78] R. Liu, T. Chen, and L. Huang, "Research on human activity recognition based on active learning," in *2010 International Conference on Machine Learning and Cybernetics*, vol. 1, pp. 285–290, IEEE, 2010.

[79] J. Yang *et al.*, "Automatically labeling video data using multi-class active learning," in *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 516–523, IEEE, 2003.

[80] M. Hossin and M. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, p. 1, 2015.