Employing multi-modal sensors for personalised smart home health monitoring.

FORBES, G.

2022

Computing

PhD

Employing Multi-modal Sensors for Personalised Smart Home Health Monitoring

Glenn Forbes

A thesis submitted in partial fulfilment
of the requirements of
Robert Gordon University
for the degree of Doctor of Philosophy

October 2022

Abstract

Smart home systems are employed worldwide for a variety of automated monitoring tasks. FITsense is a system which performs personalised smart home health monitoring using sensor data. In this thesis, we expand upon this system by identifying the limits of health monitoring using simple IoT sensors, and establishing deployable solutions for new rich sensing technologies.

The FITsense system collects data from FitHomes and generates behavioural insights for health monitoring. To allow the system to expand to arbitrary home layouts sensing applications must be delivered while relying on sparse ground truth data. An enhanced data representation was tested for improving activity recognition performance by encoding observed temporal dependencies. Experiments showed an improvement in activity recognition accuracy over baseline data representations with standard classifiers.

Channel State Information (CSI) was chosen as our rich sensing technology for its ambient nature and potential deployability. We developed a novel Python toolkit called CSIKit to handle various CSI software implementations, including automatic detection for off-the-shelf CSI formats. Previous researchers proposed a method to address AGC effects on COTS CSI hardware, which we tested and found to improve correlation with a baseline without AGC. This implementation was included in the public release of CSIKit.

Two sensing applications were delivered using CSIKit to demonstrate its functionality. Our statistical approach to motion detection with CSI data showed a 32% increase in accuracy over an infrared sensor-based solution using data from 2 unique environments. We also demonstrated the first CSI activity recognition application on a Raspberry Pi 4, which achieved an accuracy of 92% with 11 activity classes.

An application was then trained to support movement detection using data from all COTS CSI hardware. This was combined with our signal divider implementation to compare CSI wireless and sensing performance characteristics. The IWL5300 exhibited the most consistent wireless performance, while the ESP32 was found to produce viable CSI data for sensing applications. This establishes the ESP32 as a low-cost high-value hardware solution for CSI sensing.

To complete this work, an in-home study was performed using real-world sensor data. An ESP32-based CSI sensor was developed to be integrated into our IoT network. This sensor was tested in a FitHome environment to identify how the data from our existing simple sensors could aid sensor development. We performed an experiment to demonstrate that annotations for CSI data could be gathered with infrared motion sensors. Results showed our new CSI sensor collected real-world data of similar utility to that collected manually in a controlled environment.

Acknowledgments

First, I would like to thank my supervisory team for their support throughout my academic career. Dr Stewart Massie and Professor Susan Craw have both encouraged and supported my growth as a researcher since we met in 2017. Stewart has adapted well to the divergent trajectory of this thesis, while ensuring I remained focused on delivering meaningful progress on our research goals. He has continually listened to my meandering thoughts and plans, and made sense of them when others would rightfully pay no heed. Without his support and guidance, I could not have delivered the contributions in this thesis. Susan has continually provided granular feedback and taught me vital lessons in performing research and communicating the process and outcomes. Perhaps the most important of these: no matter how much you like a piece of writing, it must be fit for purpose. Feedback was delivered surgically at many stages in my career, and the works I have produced improved significantly as a result.

Second, I'd like to thank the research community within the School of Computing, and the Graduate school. I have benefit massively from opportunities provided through the AI group, both in attending conferences, and sharing my work with like minded researchers.

Next, I'd like to thank Chris Clare for their support and feedback throughout this PhD. Without their influence, I would not have considered Radio both for sensing applications, and as a general field of interest. Their unconventional suggestions and constant questioning of details have elevated the quality of contributions in this thesis beyond my best expectations. I look forward to collaborating with them on our 47th "dumb idea".

I would like to thank my partner, Ahman, for their continued supportive presence. They have listened to countless one-sided discussions on the current state of my work. Their presence motivates me most. I'd also like to thank my parents who provided me with the opportunities throughout my life to explore my potential. They gave me space and I'm grateful for their support. Thanks also to Vladimir Damyanovski for continued support throughout the project, and assistance with structure and design of CSIKit.

Finally I thank my cat, Baby, for her companionship and insistence on my accountability.

Contents

List of Figures

List of Tables

Abbreviations

ADC	Analogue to Digital Converter	4.3.1
ADL	Activity of Daily Living	3.2
AGC	Automatic Gain Control	4.3.1
AP	Access Point	5.2.1
CNN	Convolutional Neural Network	2.4.3
COTS	Commercial Off-the-shelf	1
CSI	Channel State Information	2.4
CSMA	Carrier-sense Multiple Access	4.1
CSV	Comma Separated Values	2.4
DSP	Digital Signal Processing	4.1
EVM	Error Vector Magnitude	6.1.1
FIR	Finite Impulse Response	4.2
HAR	Human Activity Recognition	3.5
LiPo	Lithium-Ion Polymer	7.1.1
LOS	Line of Sight	5.2

SDR	Software-defined Radio	4.3.5
SNR	Signal-to-Noise Ratio	4.3.2
SPI	Serial Peripheral Interface	7.1
SVM	Support Vector Machine	2.4.2

Chapter 1

Introduction

Sensor engineering brings together the complementary fields of hardware development and data science. Engineers develop novel sensor technologies and identify their sensing capabilities through data analysis. Research in sensor engineering may focus on establishing new modalities, refining sensor technologies, or identifying new capabilities with existing hardware. This supports the development of new sensing applications. The rise of the Internet of Things (IoT) and Smart Houses has lead the way in reducing sensor cost. Researchers have now been looking towards ubiquitous computing to utilise dense deployments of low-cost sensor hardware. We considered their use in ambient health monitoring in modern smart homes. In this thesis, we aimed to identify how multiple sensing modalities could be combined to enhance capabilities and speed up research and development for new sensing modalities.

Sensors are primarily categorised by their sensory modality. This refers to the stimulus which provides input for the device. A sensor's modality defines both its capabilities and limitations. The correct modality must be used for a desired application. For instance, a door sensor reports whether the magnetic switch is open or closed, but it cannot tell temperature. However we are not limited to a single sensor to deliver a complete application. Multiple sensing modalities can be used at once to provide complementary functionality. For instance, the sensors in a modern smartphone are used simultaneously to provide a cohesive experience. The accelerometer measures changes in relative velocity as you lift the phone from a desk, which informs the device the screen should be turned on.

A photodiode sensor measures ambient light conditions so the device can increase screen brightness for readability. Both sensors have radically different modalities. Together they provide additional knowledge used to improve the user's experience.

Intrusive sensing modalities are those which require user interaction or which deprive the user of basic privacy. This presents a unique problem in smart housing. Sensors using these modalities are perceived to be intrusive when deployed in sensitive environments. While a user may be happy to use a camera on their phone at will, they might object to having one constantly monitor them in their home. Sensors lie on a sliding scale ranging from intrusive to ambient. Cameras are typically seen to be an intrusive form of sensing. The data they collect can be used to identify the user and their behaviours which can be easily interpreted. Wearable accelerometers are slightly less intrusive as their data is more difficult to interpret but the user must physically interact with and maintain it. The aforementioned door sensor is almost entirely ambient, as it blends into the resident's environment. It does however generate far less data than either of the previous sensors. Less intrusive sensing modalities are typically capable of less fidelity in sensing applications. The more intrusive a sensor is, the more data you can collect. However no matter how high fidelity your sensors are, if they are too intrusive then the layman will never want them installed in their home. Uptake is massively important in sensor development. Users must feel comfortable using sensors for their designated purpose, otherwise they will not catch on.

The socially acceptable approach to sensing in smart housing is the fully ambient one. Simple IoT sensors such as magnetic door switches and passive infrared motion sensors (PIRs) are familiar to most users as home security equipment. As such, they easily blend into the background of the resident's home environment. They also monitor events such that a data point is only generated when the sensor's state changes. These state changes are usable for coarse sensing applications. A PIR can only tell us whether movement is occurring in its line of sight. This limitation of the modality limits their sensing fidelity. Instead of focusing on the limits of their low level observations, their utility is in high level monitoring. Their data can be used more effectively when combined with domain knowledge and temporal context. In our initial works using these sensors, the FitHomes and FITsense projects, high level activities could be observed and tracked over a long term. However this was predicated by manually encoded knowledge of the home and its layout. In this thesis, we developed novel methodologies for data sequencing and temporal representation with simple IoT sensor data. This allows us to map unseen home environments and improve activity detection performance, representing the limit of our sensing capability with this equipment.

After we observed the limitations faced in using simple IoT sensors, we investigated modern approaches to ambient sensing. Radio Frequency (RF) sensing modalities drew particular interest due to the invisible nature of their stimuli. RF-based sensors rely on the behaviour of radio waves for their input. Radios are sensitive equipment and can monitor minute changes in the signals they detect. Many RF solutions can be used for sensing, but they typically require expensive enterprise equipment. Channel State Information (CSI) is a form of RF sensing which can be used to monitor the quality of a wireless link between two devices. It can also be used to observe the disturbances in the link such as those caused by a human. Some commercial off-the-shelf (COTS) WiFi equipment can be used to collect CSI data. This presented an opportunity to investigate low-cost sensors using CSI, which may be capable of greater sensing fidelity than our simple IoT sensors.

To establish whether this was possible, first the available CSI sensing hardware was identified. They each differed in specification, performance, and price. A comparison of the available hardware was then necessary to identify whether the low-cost options are less capable in CSI sensing applications. Tools for analysing CSI data are publicly available, however they are also limited to their specific hardware platforms. These tools are also aimed at RF engineers with experience in the field. A clear gap in the field existed for CSI tools geared towards data scientists, of whom many are familiar with the data processing methods used. We developed CSIKit: a Python library for parsing, processing, and visualising CSI data from any hardware for research and development. CSIKit was then used to develop sensing applications for generic CSI hardware, demonstrating the strength of low-cost accessible hardware. Combined with additional novel methodologies for CSI data analysis, CSIKit also enabled us to perform a comparison of sensor hardware to establish the viable hardware entrypoint for low-cost CSI sensing.

Finally, we performed an in-home study using real-world data from CSI sensors in an occupied house from the FitHomes project. This achieved two goals: demonstrating deployable CSI sensors, and producing a proof of concept for the FitHomes CSI Platform. Off-the-shelf CSI sensor solutions for in-home deployment do not currently exist. We implemented and tested a novel hardware design for low-cost CSI sensors which can be integrated with other IoT sensors. These were then deployed in a FitHome, allowing us to observe useful insights into sensing performance. This deployment also highlighted challenges which did not arise in controlled experiments. The data from these new sensors was then collated with that from the existing basic IoT sensors. An experiment was then performed to demonstrate that the data from both sensing modalities could be used to annotate CSI data for supervised training. This provides us with the capability to automatically generate CSI datasets from real-world sensor data for targeted behavioural sensing applications.

This thesis represents the culmination of our works in sensor engineering and data analysis. We have demonstrated a method of combining data from multiple modalities to enhance health monitoring capabilities and support research and development with CSI sensors.

1.1 Background

Long-term health conditions such as risk of falling and dementia are prevalent in modern society. In over 65s falls are the most common cause of death the UK, and are an ongoing problem in the rest of the developed world, with an average of 35,848 fall-related deaths occurring annually between 2010 and 2012 (Tian, Thompson, Buck & Sonola 2013, Turner, Kisser & Rogmans 2015). Falls account for over 4 million bed days in the UK (NHS 2017). A hospital visit is typically needed to diagnose a long-term health condition. However these visits are not possible for many due to inaccessibility or inconvenience. A primary concern for these people is their independence, such that they can continue to live at home. Early indications of long-term conditions often present over time in the patient's natural environment. The easiest way to observe these indicators is through natural behaviour in the home. Through regular monitoring a preventative approach can be employed. Early indicators highlight increased risk to the resident so they can prioritise a hospital visit. This could lower general morbidity in these patients, while also reducing costs and load on health services.

As pervasive computing technologies have become more accessible, continuous monitoring in the home has become possible using low-cost off-the-shelf sensors. Perhaps most synonymous with IoT is the smart home, a concept in which a house is fitted with many sensors and devices to deliver a networked home environment. The underlying technologies in basic IoT sensors have not radically changed over the last decade with infrared (IR), magnetic switches, temperature sensors, and other passive binary sensor technologies largely retaining their core designs. However, the technology required to allow even simple sensors to communicate over a network have become smaller and cheaper, making them easier to integrate into a smart home environment. WiFi devices were previously power hungry and expensive, however the past decade of IoT research has drastically improved this. Cheap low-power microcontrollers now contain WiFi radios on their System on Chip (SoC), ensuring WiFi functionality is accessible in all form factors. A single storey 3 bedroom home can be outfitted with sensors in each room at a reasonable cost.

We consider the high level behavioural monitoring capabilities of basic IoT sensors, because they cannot achieve the accuracy required for lab-based testing. Continuous monitoring in a Smart Home provides an opportunity for the resident's activities to be tracked over time, and hence changes in behaviours can be identified (Juarez, Ochotorena, Campos & Combi 2015). Activities of Daily Living (ADLs) are everyday tasks performed in daily life, and they can be identified using basic IoT sensor data. Identification of ADLs, their frequency and change over time can aid recognition of precursors to long-term health conditions (Wilson & Atkeson 2005). This understanding of the relationship between ADLs and long-term health conditions underpinned proposals for the FitHomes and FITsense projects.

FitHomes is an ongoing initiative lead by Albyn Housing Society in partnership with Robert Gordon University that aims to support independent living, with the supply of custom-built Smart Homes fitted with ambient basic IoT sensors. The project has been funded by the Inverness City Region Deal. An initial development of 16 homes was built in Alness, Scotland aimed at housing residents at risk of mobility-based health conditions. Residents began living in these homes from July 2018. Additional developments have since been completed in Nairn and Dingwall, with more planned in the future. FITsense was a complementary one year project funded by The Data Lab which was aimed at using sensor data to establish behavioural patterns and support the development of a prototype fall prediction system. The prototype FITsense system was implemented once FitHomes data collection began in August 2018.

FITsense produces high level behavioural annotations using basic IoT sensor data to generate profiles of resident fall risk. Annotated sensor data can be collated into behavioural sequences, some of which are classified as ADLs. Regular ADL performances are then tracked to identify frequency and variability. Health risk factors relevant to risk of falling were identified from literature and approximated from ADL features. These were then used to generate a resident's fall risk profile. A "risky" profile was generated for each resident using relative thresholds of each factor. This data could be viewed on a web-based dashboard, representing two stages of knowledge: raw sensor data, and high level behaviours. All raw sensor data could be viewed which allows for real-time monitoring of low level behaviours. High level behaviours were produced in daily batches, which allowed residents to view their activity performances and fall risk profiles. This data is owned entirely the residents and can only be used with their consent. They could also share access to this data privately with family, carers, and healthcare professionals.

Feedback from the FitHomes and FITsense projects has been generally positive. Albyn Housing confirmed no residents had to move from their FitHome to a hospital or long-term care. Residents reported they felt comfortable in FitHomes and did not have concerns about monitoring or the security of their data. Our ambient approach to sensing does not disrupt residents' lives and allows them to continue living independently in their homes like normal.

After FITsense was completed, a evaluation of the project's outcomes was performed. Through this, we observed challenges that highlighted inherent limitations in our approach and sensing hardware. FITsense required manually-encoded mappings of the home environment, which limited our ability to deploy the system in arbitrary home layouts. Also, our sensor hardware limits us to coarse sensing, restricted to detecting movement within a room or triggering an appliance-related event. We considered options to address this. The density of our current sensing installation could be increased, with multiple coarse sensors used in one room. However this would lead to increased noise from each sensor, and massively increase the cost of our solution. Alternatively we could investigate other sensing modalities capable of higher fidelity sensing applications. Regardless we also found the process of collecting manual labels for smart home sensor data to be slow, expensive, and low yield. Research and development of supervised machine learning models requires far more labelled data than we could get residents to generate. This PhD was proposed to investigate and address these issues.

1.2 Research Question

This thesis explores the primary research question:

"How can multi-modal ambient sensor technologies be used to monitor resident health in a network of smart houses?"

This question breaks down into 4 key sub-questions:

- To what extent can **simple IoT sensors** be used to **monitor long-term resident behaviour and ongoing health** in a smart home?
- How can **CSI-based sensors** be used to **supplant PIRs** based on **sensing functionality and value**?
- How does **COTS CSI hardware compare** in wireless and sensing performance, and is a **viable hardware entrypoint** identifiable?
- How can we use **behavioural annotations** from simple IoT sensors to **aid the development of supervised CSI sensing applications**?

1.3 Aims and Objectives

5 aims were formulated to address our research questions. These aims were realised through their associated objectives.

- **Expand upon the FITsense system to improve scalability for new and retrofit home environments**

  - Remove noise from event-based sensor data in unknown environments by inferring room links
  - Improve the representation of temporal dependencies for activity classification to improve the value of manually annotated data

- **Develop a homogeneous approach for handling data from all COTS CSI hardware platforms**
  - Develop a Python library for parsing, processing, and visualisation of serialised CSI data using accessible data science methodologies
  - Demonstrate AGC reversal for CSI recovery and measure the efficacy of this approach

- **Show CSI is capable of higher fidelity sensing applications than basic IoT sensors**
  - Develop sensing applications using CSIKit to demonstrate the capabilities of CSI-based device-free wireless sensing

- **Identify differences in CSI collection performance across COTS hardware**
  - Use CSIKit to implement a sensing application for a comparison of COTS hardware, thus identifying the best value option

- **Perform an in-home study using the selected best value CSI sensing hardware to demonstrate real-world sensing capabilities**
  - Design and test a hardware implementation for deployable CSI sensors to identify data insights and challenges in real-world applications

1.4 Contributions and Practical Developments

The original contributions in this thesis relate directly to our objectives. Each is an original piece of work which contributes new knowledge in the field of sensor engineering. They are supported by experimental results, measured impact, and publications.

- A **temporally enhanced representation** for basic IoT sensor data used in **activity classification tasks**
- A **homogenised data representation** for **all COTS CSI hardware** platforms
- Novel implementations of **CSI sensing applications** for **movement detection** and **activity recognition**
- **Comparative evaluation of COTS WiFi devices** for **CSI sensing** performance
- Development and testing of a **deployable CSI sensor** for use in smart homes
- An approach to **CSI data annotation** using **PIR sensors** to aid sensing application development
- Open source **code** and **datasets** from our experiments

1.5 Ethical Considerations

This study makes use of data obtained from actual FitHomes residents, who are currently living as tenants in Albyn Housing Society Ltd. (shortened to Albyn in this section) properties. Each FitHome generates data points throughout the course of the resident's daily actions. These data points belong to the resident and may only be used by other parties as granted through a data sharing agreement. This data sharing agreement is produced by Albyn and details data retention, usage, and destruction policies. Residents can choose to enter into a data sharing agreement, which includes an on-boarding process through which the FitHomes network, the sensors, and data dashboards are explained. They are also informed as to which external bodies may make use of the data for research purposes. The purposes for data usage may only be changed alongside the data sharing agreement. The amount of data generated by each home is expected to increase as additional functionality is offered through the FitHomes network. Albyn must update the agreement when additional FitHomes functionality is developed to ensure residents remain informed and are given the opportunity to refuse changes to their service with Albyn. At any time, FitHomes residents can decline to continue providing data to the network and request for their data to be removed from the system. Albyn is also required by law to provide a resident with their data if they make a Subject Access Request (European Parliament & Council of the European Union 2016). These mechanisms ensure residents retain full autonomy over the data produced in their FitHome.

Transporting and delivering data from FitHomes to residents, Albyn, and external bodies highlighted the need for two main resolutions to support ethical conduct:

First, the location of our data storage and processing servers. This impacts the rights afforded to residents and their data. The FitHomes network currently aggregates data on a central server at BrightSolid's Aberdeen Data center which is governed by the GDPR and DPA 2018 (Parliament 2018). If the network continues to grow there may be a requirement to move the system to a larger cloud services provider. In this case, only providers offering services located in the UK or Ireland will be considered as they meet our standards for data safeguarding and privacy.

Second, resident data is provided to RGU in a psuedonymised format. This ensures we receive data with no identifying information which can be attributed to a real person, however a shared identifier is used which Albyn can resolve to a physical house. This is necessary to support communication mechanisms where needed for research. For instance, the in-home study conducted in Chapter 7 required the identifier of the house used for the prototype deployment to access the data within the FitHomes network. We do not store any resident names in the FitHomes network databases, and all data administration procedures related to personal information are managed by Albyn.

1.6 Thesis Organisation

This work is divided into 6 chapters with original contributions. Chapter 2 contains a literature review. Common methodologies for health assessment are identified and investigated to determine how they can be replicated using in-home monitoring. Modern approaches to ambient sensing are also evaluated to establish our new sensing technology. In Chapter 3 the FITsense system is detailed. Our improvements over the initial implementation are highlighted in Section 3.3 and Section 3.5. Chapter 4 establishes the concepts behind CSI for wireless sensing applications. This section also introduces CSIKit. In Chapter 5, we demonstrate 2 CSI sensing applications developed using CSIKit. This highlight the capabilities of COTS CSI sensing hardware and how they outperform PIRs. Our sensing application experience enables the work in Chapter 6, as we compare COTS CSI collection hardware performance. This allows us to establish a baseline for viable COTS sensing hardware, enabling our in-home study in Chapter 7. In this study we first developed a hardware solution for deployable CSI sensors in a FitHome, which was used to perform investigatory data analysis for sensing applications. Finally, we performed an experiment to identify whether behavioural annotation data from FITsense could be used automate CSI dataset generation for sensing application research and development.

Chapter 2

Literature Review

The research presented in this thesis required a review of three fields: inferential measurement, behavioural monitoring, and ambient and radio-frequency sensing. Our previous work in the early development of the FitHomes network focused on inferential measurement, which involves sensors that do not directly provide information about the domain of interest but rather make use of online measurements to provide context that can be used to infer an unknown variable. For example, a PIR motion sensor detects movement in a room but a behavioural monitoring sensing apparatus must consider additional context via inference to determine whether the occupant has transitioned to a different location or has simply stood up. This section aims to address the relevant considerations when designing inferential sensors from their designations of sensitivity and accuracy, response to and mitigation of interference, and co-located deployment for fusion applications.

We recognise that we are not the first to employ behavioural monitoring techniques using inferential sensing. Thus in this review we investigated existing cases in the same field to understand potential applications, and how to bridge the gap between behavioural observations, and monitoring health and wellbeing through feature extraction.

Finally, we explored modern ambient sensing solutions for use in smart homes and such sensors capable of providing rich data. Among the various ambient sensing solutions, we selected CSI as the most cost-efficient and potentially capable technology. We performed a review of the potential applications observed in previous research to demonstrate the usefulness of this technology. Overall, this review of inferential measurement, behavioural monitoring, and ambient sensing solutions provided a foundation for the research presented in this thesis and demonstrated the potential of these technologies for use in our work.

2.1 Inferential Sensing

A sensor is a device which observes an input stimulus to produce an output signal. They employ a sensing mechanism to monitor a given input. For instance, a temperature sensor mechanism could function by measuring the resistance of a thermally-sensitive resistor with the input stimulus being the external temperature of the immediate environment. While typical sensors are valuable in many measurement scenarios for monitoring a single output stimulus, they may not be suitable for broader or more complex tasks due to their limited scope.

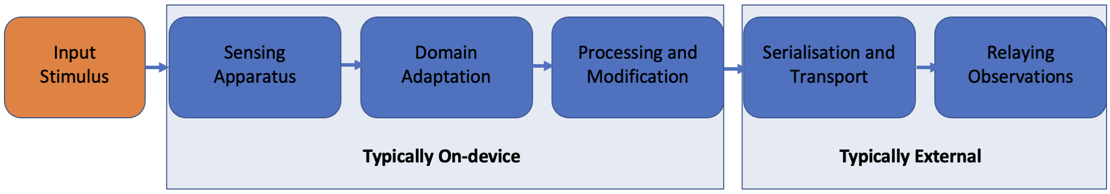


Figure 2.1: Diagram showing the common behaviours of measurement systems and the flow of output data.

In Figure 2.1, the basic flow of data in a generic measurement system is outlined. An input stimulus, as previously discussed, is monitored as a single variable which our device aims to measure. The sensing apparatus operates to sample this input stimulus. To produce an output signal, the sampled stimulus must be converted to a digital signal which is adaptation of the source to the output domain. Once this signal is produced, it can be processed. General examples of such processing stages include amplification and filtering. Each of these stages are usually performed on the measurement device itself, especially in scenarios where the device is used for any online applications. For instance, a radio will usually contain its entire amplification and filtering chain on the same PCB. Finally, output data can then be transferred from the device itself to external processing chains or storage.

Inferential sensors, sometimes referred to as a virtual or soft sensors, differ from traditional measurement tools. They do not aim to monitor a single variable, but rather estimate a variable which cannot be directly measured. A model or algorithm is employed to factor in multiple measurements either from a single sensor or an array of sensors. In ambient sensing environments, it is crucial to adopt this sensing approach because the sensing device is often physically disconnected from the stimulus that we attempt to measure and interpret.

An example of inferential sensing is the use of sensor belief networks (Dodier, Henze, Tiller & Guo 2006). This approach utilizes Bayesian probability theory to estimate the occupancy status of an observed environment. Dodier and his team used three PIRs and a phone hook sensor for their experiment. Typically, when dumb PIRs are deployed in sensing applications, a single PIR activation indicates that someone is present. However, Dodier and his team established that typical occupancy-based lighting monitoring systems fail to reduce energy usage due to misfiring sensors and rarely-used spaces.

The use of sensor belief networks allows for the establishment of a sensor hierarchy, where some sensors will only activate in specific scenarios, while others may always trigger. Sensors that rarely activate may have a narrow field of influence or be placed with obscure edge cases in mind. In Dodier's example, their Room 1 PIR 3 was less sensitive than PIRs 1 and 2. Once trained, the probability model heavily favored this sensor to indicate occupancy as it is more likely to activate in a true positive scenario. Conversely, Room 2 PIR 3 activated many times in unoccupied scenarios, and so its activations are much less likely to be considered in occupancy modeling. Using a mathematical model allows us to establish variable utility for a given sensing apparatus. This means that we can employ multiple sensors and utilize their measurements for a single sensing goal, by assigning different weights to each sensor based on the likelihood of their activation in the presence of a person. More modern approaches to belief networks build on this through using deep neural networking for Deep Belief Networks (Hinton 2009).

Inferential sensors can potentially have increased error compared to regular sensors due to the additional complexity in employing sensing mechanisms. To establish a sensor is fit for function it must be evaluated. We discuss some relevant performance factors and how such metrics impact inference.

2.1.1 Sensitivity and Resolution

Sensitivity refers to the rate at which a sensor's output signal changes in response to a given change in the input modality, by quantifying how responsive the sensor is to changes in the input. In addition to affecting the number of data points produced, sensitivity also affects the accuracy of a sensor. A highly sensitive sensor can detect even the slightest changes in the input, but is more likely to be affected by noise and other unwanted signals. On the other hand, a less sensitive sensor may miss some subtle changes but produce more consistent readings. Overly sensitive sensors can cause battery drain issues, as each sensor data report will incur battery usage. Larger data sizes, such as those from high resolution sensors, typically consume more power due to the increased bandwidth requirements for data transmission. This means it is important to balance the decision on how much data we actually need for efficient sensing.

The resolution of a sensor also affects its accuracy. Higher resolution sensors can detect smaller changes in the input, which can lead to more accurate readings. However, they also generate larger amounts of data, which can be challenging to process and store. In some cases, the largest possible resolution may not be necessary, and a lower resolution sensor can be sufficient. Balancing the sensitivity and resolution of a sensor is crucial to creating efficient and accurate sensing systems. This is especially important in battery-operated sensors, where power consumption is critical. By finding the optimal balance between sensitivity and resolution, it is possible to create sensors that produce accurate and reliable data without generating large quantities or consuming excessive power.

In order to balance sensitivity and relevancy of data points, it's important to choose appropriate sensors for a given application. For example, highly sensitive sensors may be necessary to capture small changes or fluctuations in specialised applications. In other cases, such as industrial monitoring or environmental sensing, less sensitive sensors may be required in order to avoid generating large amounts of irrelevant data (Touqeer, Zaman, Amin, Hussain, Al-Turjman & Bilal 2021). It's also important to consider the accuracy of the sensor readings in relation to the overall accuracy of the model or system being used. While highly sensitive sensors may be able to capture small changes in data, if the accuracy of the readings is low, the resulting model outputs may be inaccurate or unreliable. Therefore, steps should be taken to ensure sensor readings are accurate and reliable. Mitigations for measurement deviations and interference must be considered, as they represent inherent risks to sensor accuracy.

2.1.2 Deviations and Interference

Deviation refers to the difference between a sensor's measured output and the real stimulus being measured. The term refers to how measurement devices differ from the quantity they aim to measure, and can result from internal issues such as poor design or calibration. Deviation is typically consistent across multiple measurements as it is a systemic issue. Various measures can be used to quantify it such as the mean absolute error and the root mean square error may be used. Sensor manufacturers often deal with deviation by providing expected values and margins of error for their products, which they explain to customers in product specifications and other documentation. For instance, the RSSI accuracy for the standard operating range of the Raspberry Pi CM4's wireless chipset is listed as +/- 5dBm (Cyp 2019). This allows engineers to account for these expected deviations in their design and implementation of such equipment.

Interference is caused by external factors that can influence the sensor's output, such as electromagnetic signals or environmental conditions. Interference is also a systematic error but is often more difficult to control and mitigate than deviation, as it can be caused by a wide range of factors and may vary over time and location.

Finally, noise refers to random variations in the signal being measured that are not related to the parameter of interest. Noise is a stochastic error that can be caused by factors such as electrical interference, thermal fluctuations, and sensor imperfections. Unlike deviation and interference, noise is not consistent across multiple measurements and can vary over time and location.

While an individual sensing mechanism may be capable of making accurate measurements, deviation in inferential sensors is a dissonance between sensor outputs and the overall inference task. In a compound sensing apparatus these risks are manifold. The mathematical models used in inferential sensors crucial must account not only for the relationship between each sensor's data points, but also for the variations of said points from deviations, interference, and noise. In compound sensing, we must also take into account that each sensor may be affected by different levels of interference. Researchers have dealt with this issue by carefully modelling such deviation and accounting for it in their analysis (Zhang & Dong 2004).

Ambient sensors are subject to various sources of interference that can affect their accuracy and reliability. Interference can arise from many factors, including changes in temperature, humidity, and other environmental factors. These factors can cause the sensor to detect false positives or negatives, resulting in inaccurate data. For example, PIRs can be affected by infrared heaters, which emit the same type of radiation as the sensors are designed to detect. This can cause false activations and inaccuracies in the data. Similarly, movements outside the window or in other areas can also trigger the sensor, leading to erroneous results. Interference can also arise from the sensor itself, where the sensor may be functioning properly but picking up data that we do not want to observe. This can occur due to the sensor's placement and positioning, which can be addressed by considering better sensor placement.

Moreover, interference can also affect the transmission mediums, such as the 2.4GHz frequency space. Since 2.4GHz is a crowded frequency space, it is expected to suffer from interference. In fact, one study considered the interference patterns inherent in popular sensor data transmission protocols, such as ZigBee, Bluetooth Low Energy (BLE), and WiFi (Marinčić, Kerner & Šimunić 2016). They found BLE to be the most robust method when operating in crowded network environments. In a similar study, Ghayvat (2015) investigated approaches to interference mitigation using intelligent channel hopping. They proposed an adaptive channel hopping algorithm that can dynamically select the best channel for transmission, reducing the effect of interference on the sensor data.

Some ambient in-home monitoring studies have been conducted with the assumption is no noise in the data produced by the sensors (Hyväri, Elo, Kukkohovi & Lotvonen 2022). While this may not significantly affect the study's objectives if they are not concerned with tracking every single action in the home environment, it can pose a challenge for more fine-grained activity tracking. If sensors are blindly trusted to produce valid data points without accounting for noisy or invalid data due to interference, we risk losing the ability to track logical sequences of events. In this context, sensor belief networks offer a promising way to maintain this sequence by incorporating probabilistic reasoning into the sensor data processing pipeline. By contrast, traditional rule-based methods or classifiers do not explicitly account for uncertainty in the sensor data and may produce less accurate results in the presence of noise or interference.

As deviation, interference, and noise are inherent in sensor data collection, we must consider the impact of their inclusion in datasets, which generally results in overfitting.

2.1.3 Overfitting

Overfitting occurs when a model or sensor is trained or calibrated on a dataset which contains data points which are not representative of the real-world application it was designed for. As a result, the inferential model being employed may produce accurate results for the data it was trained on, but perform poorly when exposed to new, unseen data. In areas of research where real-world data can be expensive and inaccessible, overfitting is a real concern. Simulated datasets are often employed for experimentation or initial training, especially in ambient monitoring research. When using simulated datasets, there is a risk that the model may adjust to the specific details of the simulation rather than the underlying patterns that are intended to be learned (Yeomans, Thwaites, Robertson, Booth, Ng & Thewlis 2019). In cases where the trained model becomes too complex, minor fluctuations may be learned as opposed to the general concepts being conveyed in the dataset.

There are various approaches to mitigate overfitting. The most common approach is to use machine learning regularisation approaches that are designed to handle noisy or incomplete data, such as by training with a dropout layer (Baldi & Sadowski 2013). Dropout layers thin out models by drop a random number of units in a given training stage. Essentially this averages out the model, retaining a more general image of the domain rather than getting caught up in fine details. Various boosting methods are also common in regularisation, such as Structured Gradient Tree Boosting (2019). These methods have been shown to greatly reduce overfitting across multiple simulated datasets.

When it comes to balancing the relevance of data versus the amount of data collected, there is a trade-off to be made. Increasing the level of detail captured by a sensor can provide valuable insights, but it also increases the risk of overfitting and the need for larger storage and processing resources. Therefore, it's important to carefully consider the specific requirements of the application and strike a balance between the level of detail and the overall accuracy of the sensor system.

Overfitting is also a concern in RF sensing where ambient signals can be influenced by a wide range of factors, such as interference from other devices, environmental conditions, and human activity. To address this, RF sensing systems may incorporate advanced signal processing techniques, such as wavelet decomposition, to eliminate consistent or high frequency noise from raw signals (Palipana, Rojas, Agrawal & Pesch 2018). Regularization is also common in many RF modelling applications, for instance L2 regularization employed in CSI sensing (Ma, Zhou & Wang 2019). L2, or Ridge, Regularization employs a penalty mechanism to constrain the weights applied by the cost function. This can reduce model complexity and the possibility of overfitting.

In general, it is vital to make considerations when designing and deploying ambient sensing systems to avoid overfitting and ensure that the collected data is relevant, accurate, and useful for the intended purpose. Alternatively, a more radical approach is to incorporate additional sensors or features that can provide complementary information and help improve the accuracy of the overall sensing system, through sensor fusion.

2.1.4 Sensor Fusion and Aggregation

Readers are likely familiar with the issue of false negatives in lighting systems, where lights may turn off after a prolonged period of no detected movement. This problem arises from the lack of sufficient data points being collected by the event-based PIR sensors, which only trigger after movement occurs. This means the issue cannot be resolved through the use of PIRs alone, as they are too myopic in this context. The solution involves employing the right tool for the task: using bespoke sensors designed for monitoring occupancy rather than just movement (Yang, Zou, Jiang & Xie 2018). Implementing such sensors not only enhances the accuracy of inference-based approaches to occupancy monitoring, but also enables the identification of stationary versus moving subjects. This example showcases sensor fusion through its improved and more robust functionality.

Sensor fusion refers to the process of combining data from multiple sensors to obtain a more complete and accurate understanding of a problem space. Many inferential sensors benefit from fusion of data from multiple sensors in the same space. A single sensor provides only a single perspective on the problem, which may not be sufficient to fully capture the complexity of the environment being monitored. By adding other sensors, the same space can be monitored from different angles or modalities, producing a more complete picture of the observable environment. For example, antenna arrays may be employed when one single antenna cannot provide a proper image of the observed space. Multiple observations of the same space can be combined to provide a clearer image.

There are different approaches to sensor fusion, including centralized, decentralized, and hierarchical. In centralized sensor fusion, all the sensor data is collected and processed by a central unit, which then produces a single output. In decentralized sensor fusion, the sensors communicate with each other to share information and make decisions collectively. In hierarchical sensor fusion, the sensor data is processed at different levels, with higher levels integrating information from lower levels to produce a more comprehensive output.

By combining information from multiple sensors, sensor fusion can reduce uncertainty and improve the accuracy of measurements. Different levels of sensor fusion can be used to achieve different levels of accuracy and complexity, ranging from simple data aggregation to complex data analysis and decision-making.

For instance, combining different sensing technologies for additional insight is a primary component in the process of medical diagnosis. Many health conditions present with multiple symptoms which cannot all be detected with the same apparatus. Diagnosticians will consider the results of different medical tests to confirm a diagnosis. The sensing modality of a medical apparatus predicates its capabilities, fidelity, and its applications. For instance, a Magnetic Resonance Imaging (MRI) scanner uses strong magnetic fields and radio waves to slowly generate high resolution 3-dimensional internal images of the human body. These can be used to locate and identify tumours, brain injuries, strokes, and a range of conditions which cannot be detected with other equipment. An ultrasound scanner, or sonograph, uses high frequency sound waves to generate real-time 2D images of internal organ structures. An MRI scan is more intrusive than an ultrasound. Typically more intrusive modalities generate more data. Ultrasound waves do not penetrate bones or gases as easily, and so tend to be used for soft tissue. As such, there are multiple selection criteria when choosing between these different sensors. Multiple sensing technologies can be used for provide greater insights when combined. If these tests were not as intrusive, the patient experience would improve.

Walk-in patients can be assessed both physically and verbally when they are initially considered. They may be asked a series of questions their physical capabilities, symptoms, or recent activities. Less-intrusive measurements such as temperature and blood pressure may also be taken. This assessment used to identify the severity of the patient's condition and whether additional tests will be needed. In some cases, an early warning score will also be calculated. Several measurements are combined to generate a score of the patient's overall risk. This can streamline the process and identify whether additional testing or treatment is needed. Getting patients to this stage is one of the main problems facing the accessibility of hospital care. Many patients have no major problems and have been fully addressed at this stage. If we could identify whether there is a heightened need for a hospital visit, this may eliminate the doubt in a patient's mind. Providing an in-home early warning system may be sufficient to ensure patients feel comfortable living in their home. However this approach requires careful selection of sensors and their deployment configuration.

Common off-the-shelf IoT sensors, such as PIRs and door switches, are relatively low fidelity. PIRs provide a coarse binary overview of activity in a room. Specialised switch or pressure sensors can be attached to appliances to monitor their usage and provide a separate data stream. These can be used for high level behavioural monitoring, such as activity recognition. This approach can be effective but additional sensors means higher cost and more maintenance. Intrusive sensing technologies capable of providing rich data insights do exist. For instance, cameras and wearables can provide us with much more detailed information about a resident's behaviour and physiology. However their intrusiveness predicates the potential uptake, which reduces their long-term viability. A viable solution requires us to find a middle ground sensing technology which maximises sensing capabilities while limiting intrusiveness. Channel State Information is a form of RF sensing which can be performed using off-the-shelf WiFi hardware. We investigated the use of RF-based sensing as an ambient sensing modality potentially capable of greater fidelity. This could us to target rich sensing applications with low-cost ambient sensors.

Ambient health monitoring smart home systems can capture sensor data from a resident's home to generate high-level insights into their behaviour and underlying health.

They cannot accurately take measurements in the same way as medical equipment and shouldn't be compared. Instead, they can tell us what the patient is doing, how much they do it, and when. To establish which data we wish to gather, we must consider how a medical professional assesses patients for a variety of conditions in a hospital. Once such assessments have been highlighted, we can then consider what sensing capabilities are offered by the available ambient sensor technologies we could use in a smart home environment. A reasonable goal would be to approximate the measurements physically or verbally gathered during physical assessment which could perform intervention to inform the resident they need to be tested.

2.2 Establishing Factors of Health Risk

Vital signs are a desirable metric when observing patient health. At a basic level the presence of vitals indicates the resident is alive. The resident's breathing rate may indicate they are struggling to breathe, they are stressed, or if they are even unconscious. Heart rate and blood pressure also offer rich insight into the resident's physical condition and wellness. Typically, residents living independently very rarely have their vitals measured. Consider a patient who is in the early stages of a long term health condition. They observe indications of their condition in daily life which they are unable to attribute to a specific condition. Eventually they consider it necessary to get checked out. They visit a doctor or general practitioner who assesses them. Through this process they may perform various structured health assessments including establishing an Early Warning Score (Smith, Redfern, Pimentel, Gerry, Collins, Malycha, Prytherch, Schmidt & Watkinson 2019). The following features are considered to generate an overall score:

- Respiration rate
- Oxygen saturation
- Systolic blood pressure
- Pulse rate
- Level of consciousness or new confusion

- Temperature

When taking these vital measurements, the medical professional will identify whether any values sit outside an expected range and score them based on their deviation. This process requires the patient to physically attend the hospital and spend time with a medical professional. While ambiently monitoring pulse rate and breathing rate in the home would be desirable the rise of wearables has largely achieved this, providing an easy way to accurately monitor vitals (Castaneda, Esparza, Ghamari, Soltanpur & Nazeran 2018). Instead, monitoring in the home should focus on behavioural observations. We consider medical assessments performed to identify behavioural aberrations which we could monitor ambiently. Through this process, we aim to establish a candidate risk score which could indicate whether the resident should seek medical assessment.

One such assessment, the One-Leg Standing Test (OLST), aims to assess the subject's balance by measuring the length of time a subject can support themselves on one foot (Michikawa, Nishiwaki, Takebayashi & Toyama 2009). Unlike other assessments previously mentioned, OLST has no link to mortality, however it can be linked to relative frailty and fall risk in elderly patients. Some researchers argue that current research relating fall risk to OLST times can be inconsistent. However Ayena (2016) designed a methodology for performing the OLST in a home environment which accurately evaluated fall risk in a range of patients, including healthy, elderly, and those with physical impairments and conditions. This assessment and many similar ones commonly aim to identify parameters of a patient's gait.

Parameters of gait can be monitored to identify patterns in movement as well as variances which may indicate deterioration or other conditions. Gait velocity is widely used to indicate fall risk in elderly patients (Verghese 2009, Montero-Odasso & et al 2005). Using machine learning techniques statistical links between gait and a range of conditions, spanning from fall risk to dementia and death, can be modelled with increasing accuracy (Yang, Zhang, Wang & Tomizuka 2013, Ismail, Radwan, Suominen & Waddington 2017). Fritz and Lusardi (2009) went as far as to label walking speed as the "sixth vital sign" and noted its potential in predicting the length of hospital stays, medical costs and general health status. Gait velocity is the most widely observed parameter of movement.

Balance, cadence, accelerometer-based motion patterns and postural sway can also be observed, however most models which successfully predict falls appear to give importance to gait velocity by design. Behaviour-based patterns in movement have also been recently explored, such as path tortuosity (Kearns, Fozard, Becker, Jasiewicz, Craighead, Holtsclaw & Dion 2012). As less intrusive sensor equipment for continual positional monitoring becomes available, it is expected that more research will be performed in this area (Adib & Katabi 2013).

Monitoring gait and other motion parameters provides excellent insight into the movement of a subject, their long term health risks, and even morbidity. The non-discrete sensors required for measuring these expressions can prove more financially and computationally expensive than simple sensor implementations. Many studies have linked behaviour and activity recognition with binary sensor outputs using a variety of machine learning techniques, such as Dynamic Bayesian Networks, Case Based Reasoning (CBR), and Hidden Markov Models (Tapia, Intille & Larson 2004, Zhou, Jiao, Chen & Zhang 2011, Suryadevara & Mukhopadhyay 2012). Observations from these studies have been used to make assessments on subject behaviour, stability of motion and observable conditions. There has been only limited research on the applications of behavioural modelling and activity recognition for predicting falls, and identifying risk of falling, using simple sensors. Parameters of gait, movement, and heart rate can strongly indicate fall risk, however these factors can currently only be measured accurately using active sensors in laboratory environments (Vestergaard, Patel, Bandinelli, Ferrucci & Guralnik 2009). A less intrusive approach would be monitoring residents in a home environment with passive sensors, allowing for authentic behaviour to be observed and modelled with accuracy. In these conditions, gait velocity and activity performance can be assumed natural and indicative of regular behaviour. During a 5 year study by Phillips (2017) where motion capture information was taken using an embedded Kinect device in a home environment, a model was derived which estimated that a 5.1cm/s decrease in gait velocity was associated with an increased risk of a fall within the next 3 weeks. However residents are still likely to reject deployment of vision-based sensors in their own home. An accessible solution must match the ambience of simple IoT sensors, especially if we wish to avoid performative patient behaviour.

The way in which a patient assessment is performed can affect its results. The standard practice for performing these tests in hospitals has a minor methodological flaw, as they can only take measurements based on how the patient presents. In the OLST a patient may not have recently intentionally tried to stand on one leg. They may be unstable on their feet, experience pain on one side, or face other issues which would hinder them in performing the test. Jefferis (2019) noted that when requested to do so in a medical setting they may perform the test regardless. This is performative behaviour in the sense that the resident is actively modifying their behaviour when observed. In extreme cases performativity can even mask behavioural symptoms (Lange, Lamanna, Watson & Maier 2019). Some patients want to present themselves at their best under observation. They psych themselves up and exceed their known capabilities in an attempt to satisfy the test and improve their result (Van Gestel, Clarenbach, Stöwhas, Rossi, Sievi, Camen, Russi & Kohler 2012). This causes issues for both the patient and their physician. A representative of the patient's day-to-day performance is more valuable to a physician than their best case. The patient could also injure themselves in the process, or become completely misdiagnosed. Repeated ambient measurements in the home could avoid this behaviour, as the resident is not physically observed. The resident would be unlikely to sustain performative behaviour after sensors were installed in their home. Next we consider the available sensor technologies, their data collection methodologies, and sensing capabilities for behavioural monitoring.

2.3 Smart Home Sensing Technologies

Various sensor technologies exist for use in Smart Homes and IoT environments. They differ in price, capabilities, and their sensing modality. They can also be grouped as either polled or event-based. This distinction can be based on their behaviour when reporting data to a centralised system. Polled sensors repeatedly send data points with a constant delay. Examples of polled sensors include cameras and accelerometers. Event-based sensors only generate a point when a perceptible change occurs. They tend to produce significantly less data as a result. The amount of data reported is a determination made on the sensor controller itself. The dimensionality of sensor data has a direct impact on the sensing applications which can be delivered.

Sensor data dimensionality reduction can occur at several stages throughout the collection process. The most fundamental limitation in collection is how the modality is monitored.

Consider a quartz-based clock. A piezoelectric charge is generated by passing current through a quartz crystal. This causes the crystal to oscillate at its resonant frequency. The output is fed into sequential binary clock circuit which overflows once every second. The crystal's oscillations are polled, however the clock's sensor controller outputs a data point once per second. The sensor controller reduces the dimensionality of its output data through inference.

We therefore group sensors by whether the data they output has scope for further inference. We define these groupings as low-fidelity and rich sensors. Establishing these categories is important to aid the sensor technology selection process. Given our goal is to monitor resident behaviour and activity, we used these categories to identify a candidate sensing technology which could provide improved fidelity over our existing sensors.

2.3.1 Low-fidelity Sensing

In this context, low-fidelity sensors are those which produce sparse data points, derived through inference on the sensor controller. The controller handles incoming data and determines which data points to generate or forward. For example, some industries make use of polled infrared sensors. However in IoT applications this would produce significant amounts of redundant data. Instead most IoT-focused PIRs generate points when the data changes, for instance when a motion event is detected. In this case polling would not produce any additional meaningful data or insight. The sensing application monitors changes on the sensor itself which reduces load on the IoT network.

Low-fidelity sensors often serve a single purpose or function. This can be within the scope of a larger sensing application utilising data from multiple devices, but without context the data from a low-fidelity sensor has limited use. Additional context can be provided with other low-fidelity sensors deployed in higher density. Tapia (2004) used dense deployments of simple switch-based sensors in an attempt to monitor low level behaviours. These sensors were deployed throughout the home, with specific sensors used for oft-used appliances and surfaces, such as the medicine cabinet or kitchen counter. This provided additional context for higher fidelity sensing applications which could not be delivered with a single sensor. The cost of such a solution has steadily dropped in price over the last decade, however the main limitation of this approach is the increased maintenance required for each sensor. This approach is not viable at scale. As such, we consider the behaviour monitoring capabilities which can be achieved using sparse sensor deployments.

PIRs and other low-fidelity sensors can be used to capture behaviour at a high level (Ogawa, Suzuki, Otake, Izutsu, Iwaya & Togawa 2002, Aipperspach, Cohen & Canny 2010). For instance, they may indicate there is activity in a room but cannot localise this within the environment. Alternatively they may only monitor a small part of the environment alone which is inherently localised. High level behaviours such as room transitions and the presence of activity can be observed. In most cases additional sensors are required to provide context for these observations, this limiting the sensing utility of an individual sensor. If we wish to achieve high fidelity sensing capabilities at scale, we must consider low-cost rich sensors.

2.3.2 Rich Sensing

Many polled sensors can produce large amounts of data. What distinguishes rich sensors the ability to use their output data for additional inference without external context. A polled temperature sensor can only report back the current temperature in its local environment, which is low-fidelity in this context. The video feed from a camera sensor can be used for multiple sensing applications, even simultaneously. For instance, cameras can be used for motion detection, object detection, and face recognition simultaneously (Yazdi & Bouwmans 2018). Wearable accelerometers can simultaneously be used for motion detection, activity recognition, and gait monitoring (Tao, Liu, Zheng & Feng 2012). A model can be produced to provide additional low-level insights into the data rich sensors generate, which cannot be performed with low-fidelity sensors. Rich sensors can usually be deployed as a single sensing apparatus with multiple capabilities. They tend to be more computationally expensive as a result of the significant increase in the amount of data generated, and cause more strain on the supporting communication networks (Baraniuk

2011). For instance, a 480p 30fps camera feed, standard in surveillance applications, typically generates 3100kbps of video data (Hikvision USA Inc. 2013). It is infeasible to have multiple rich sensors feeding raw data into a centralised network for storage if surveillance is not the primary application.

One solution to efficient sensing using rich sensors is to deploy applications on edge. Deployed sensing applications perform inference on rich data inputs to establish low level insights (Nachman, Huang, Shahabdeen, Adler & Kling 2008). These systems typically require more processing power than low-fidelity sensors, which may be deployed on the sensor PCB itself or a supporting computer. For instance, a popular approach is to use an edge-oriented compute unit such as Nvidia's Jetson Nano or the Raspberry Pi 4 (Süzen, Duman & Şen 2020). These systems support the real-time deployment of ML models using popular frameworks such as Tensorflow. These integrated sensing applications require significant amounts of research. They must be vigorously tested in expected environmental conditions and be demonstrated to perform well. The development of these applications typically involves significant amounts of research-oriented data collection, which involves gathering targeted data with meaningful annotations.

These rich sensor examples are commonly used in research for behavioural monitoring in smart home environments (Stone & Skubic 2013). Their sensing capabilites are demonstrated to provide insight into behaviour, physiology, and underlying health characteristics. However residents can find vision-based sensing to be too intrusive to their personal privacy (Townsend, Knoefel & Goubran 2011). Residents may also object to wearable sensors for similar reasons. This contributes to low uptake which drastically reduces the overall reach of smart home health monitoring systems. To facilitate large-scale deployment of these systems, we need a low-cost unintrusive sensor capable of rich insight in behaviour and wellbeing.

RF-based sensing modalities output data in the form of complex numbers through the measurement of electrical signals that arrive at an antenna (Shah & Fioranelli 2019). We consider RF-based sensing technologies as rich sensors due to the high dimensionality of their output data. Simultaneous sensing applications could be deployed using the same input radio signal. RF technologies typically use expensive equipment which limits their potential for large-scale deployment (Adib & Katabi 2013). However their fine-grained monitoring capabilities vastly outperform commercially-available IoT sensors. Most notably, this approach is typically ambient and unintrusive as the sensing modality relies on wireless signals which are not visible to the human eye. Channel State Information, an RF-adjacent sensing technology, can be employed using commercial off-the-shelf WiFi devices. We observed this potential and considered its sensing capabilities for use in smart housing and health monitoring.

2.3.3 Identifying Behaviour using Sensors

There are a variety of approaches to behavioural monitoring using sensor data. Low level behavioural observations typically relate to the actions of a person as they move through their environment or interact with objects. Activities are a higher level form of behavioural observation oriented at grouping actions into behavioural sequences. For automated healthcare, accurate activity recognition is paramount as forms the foundation for the identification of behaviour and decision-making ability in subjects. Activity recognition can be performed with basic binary sensors. Actions and events can be interpreted by monitoring sensor changes as a user moves through their home and uses certain areas and or appliances. These events can be assigned relative importance, and be used to construct a logical sequence of state changes. For instance, if doors at either side of a room were triggered in sequence, it can be assumed that a user has crossed from one side of the room to the other (Ogawa et al. 2002). These assumptions are valid when abstracting simple events, such as getting out of a chair, moving from one room to the other or turning on an appliance. More complex activities however become far more difficult to identify.

Some of the challenges in identifying complex activities are a result of dimensional limits in low-fidelity data capture such as:

- Recognising concurrent and interleaved activities, in which users can perform multiple actions simultaneously with some overlapping others at varying intervals. Some actions may also interrupt others.
- Ambiguity of interpretation, where single actions can have multiple meanings under relevant perception.
- Multiple residents, requiring parallel activities to be recognised for each user.

In activity recognition research a researcher will create a high-level conceptual model to identify features of an activity as they are expressed through sensor outputs (Kim, Helal & Cook 2010). The two primary approaches are rule-based sequential systems, and machine learning.

Rule-based approaches to classifying a resident's behaviours are well established and suited to basic sensor networks containing few binary sensors (Ogawa et al. 2002, Gaddam, Mukhopadhyay & Gupta 2010). Regular behaviours are easily observable in binary motion sensor networks, allowing them to be used for basic health monitoring. While rules are time consuming to generate manually, they work well in static configurations that rarely change and have only a limited number number of behaviours that need to be identified.

Specifically engineering a rule-based classifier for an environment can be costly and time-consuming (Witten, Frank, Hall & Pal 2016), and raises several challenges. Theoretically a rule-based system can cover each possible combination of events in a sensor network, however it is often infeasible to write accurate rules that achieve full coverage. Multiple rules may be activated and priorities need to be established. In addition maintenance can be difficult as small changes can have knock on effect to other rules. While these systems can technically be extensible, the time investment required to expand scales exponentially. Consideration of all stages of the network is required when adjusting the system as changes may ripple out through seemingly unrelated functionality.

Modern approaches to activity recognition aim to overcome the limitations of rule-based systems by focusing on the use of machine learning, with a more recent focus on deep learning. Sensors which produce complex multi-dimensional data, such as wearable accelerometers and gyroscopes, can capture patterns which are not easily perceptible to a human observer. Rules could not be easily written for activity recognition on such data, whereas a ML model can generally be trained to build effective models. For instance, a popular model used for activity recognition is a Recurrent Neural Network (RNN). RNNs are a form of temporally aware neural network, as they can retain and reuse knowledge from previous inputs in a sequence as they produce an output. While RNNs have a form of memory, Long Short Term Memory units (LSTMs) utilise an additional memory cell per unit, which allows them to retain knowledge for longer periods of time. This makes them perform well in tasks with temporally-relevant features, such as classification of data captured from an accelerometer (Chen, Zhong, Zhang, Sun & Zhao 2016).

The presence and importance of sequential events in the data from smart houses gives rise to opportunities to improve activity recognition by capturing the sequences as part of the representation. This may be explicitly by employing a feature engineering approach, however manually chosen features using heuristic processes may be overshadowed by modern deep learning research (Ordóñez & Roggen 2016). Ordóñez found that identifying the temporal dependencies that exist within human activity expressions was key to improving the performance of an activity classifier. The experiments performed in their work made use of wearable sensor data, though similar relationships with no derived heuristics may exist between binary sensor activations (Tapia et al. 2004).

Smart house sensor data presents an opportunity to identify the significance of temporal dependencies in activity recognition, independent of interfering factors such as performative behavioural differences in accelerometer data. Existing smart house sensor data research has identified the importance of temporal dependencies in activity recognition, with Tapia assigning additional labels to their representations to indicate the rough time of day at which an activity took place (Tapia et al. 2004). This manually engineered approach to temporal representation did not yield significantly improved performance when compared to a more basic representation, however the underlying assumption holds weight. Manually selected temporal relationships rely on a deeper understanding of the dataset and may not translate well across varying behavioural routines. For instance, some residents may prefer to make a hot beverage in the morning, while some may prefer to do so in the evening. However, all of them must turn on the kettle before doing so. Understanding temporal dependencies at the representation level is key to effectively learning routine behaviours.

A key challenge for ML approaches is the acquisition of labelled data; it is expensive to manually annotate sensor data with activity labels, and variations in the labelling decisions of ambiguous sensor sequences may affect the overall quality of the dataset (Rockinson 2008). Smart house data presents its own challenges, as the data is often in the form an irregular sequences of sensor activation events rather than a time sequence of polled sensor data, such as with wearables.

An ML-based classifier is an effective solution for a scalable system for smart house activity recognition. Rule-based approaches to smart house activity recognition tend to not scale efficiently for continued expansion in large housing networks, due to increasing house layout complexity. However, moving to an ML-based classifier presents alternative challenges, such as the acquisition of labelled activity examples from residents. Existing smart house sensor datasets contain relatively few labelled examples, which makes the use of modern deep learning more difficult. Data is represented in these datasets as sequence of sensor activations and associated activity labels, which highlights the existence of temporal relationships between sequential activities and activity steps. These could potentially be exploited to improve classification performance, however current approaches to encoding these relationships in data representations have yielded no significant performance improvement. Learning routine behaviours using underlying temporal relationships may be a more effective use of labelled sensor activation sequences.

2.3.4 Data Collection

Public datasets for activity recognition from smart house sensor data are typically rare, as collecting and labelling data is time-consuming. A few have been made available by Van Kasteren, CASAS and MIT (Van Kasteren, Noulas, Englebienne & Kröse 2008, Cook, Schmitter-Edgecombe, Crandall, Sanders & Thomas 2009, Tapia et al. 2004), with most public datasets containing at maximum 2 weeks of data. The CASAS dataset uses a combination of binary motion sensors and magnetic switches placed on key objects for activity tracking, such as a bowl and medicine cabinet, to track 5 basic activities (Phone Call, Wash Hands, Cook, Eat, Clean). The Van Kasteren dataset contains data from magnetic door sensors and switches to track 7 activities, with the notable inclusion of longer term activities such as Leave House and Go to Bed. This introduces the concept of activities with significantly variable lengths. The most complex of these datasets is from MIT, which contains 2 similar experiments using 77 and 84 magnetic switch sensors respectively to track 22 and 24 activities. Several of these activities are largely similar, such as Going out for Shopping, and Going out for Entertainment. However while this dataset has a large number of unique classes, it also contains relatively few instances with only 208 labelled examples. Sensor data is represented in each of these datasets as a time series of sensor activation sequences annotated with activity class names. Selecting a classification algorithm for smart house activity recognition requires an understanding of the relationships between sequential activity performances. Regardless of the approach, the annotation process has the most impact on the value of the underlying data.

While activities can feasibly be given labels, it is more difficult to produce examples and labels of routines. A routine performance can be simulated, but the strength of routine observation is being given the understanding of emergent behaviour. Yin's (Yin, Zhang & Karunanithi 2015) unsupervised approach to routine discovery is influential as the identification of routines as clusters is a more scalable approach than conventional labelled examples. Useful concepts are also identified, such as that of "hub" rooms, which naturally serve as a transitional area between two rooms where an activity would occur. A motivation of this work was to monitor the decline of capability for elderly residents, since disruptions in routine are typically observed in dementia patients. This work can feasibly be performed using motion sensors alone as it relies on locomotion patterns, and natural extension would be tracking finer resident behaviours in an effort to better understand what is happening, rather than just observing movements between rooms. By identifying finer resident behaviours in the context of an activity sequence, there is additional scope for identifying variations in behaviour which may be linked to a decrease in mobility or cognitive function.

2.4 Channel State Information

Radio frequency (RF) transmissions can travel through the air and walls, with solid objects leaving an imprint on the resultant signal as collected at receiving antennas. This can be employed as a form of ambient sensing, as human beings are a significant obstacle to RF signal paths in the GHz bands. RF-based sensing technologies have experienced a recent upsurge, after a significant challenge with through-wall transmissions was overcome (Adib & Katabi 2013). Since this breakthrough, RF has seen wider applications in ambient sensing, such as gait recognition, imaging and human localisation using no additional equipment (Seyfioğlu, Gürbüz, Özbayoğlu & Yüksel 2017, Adib & Katabi 2013, Adib, Kabelac & Katabi 2015). While RF-based sensors could have a potentially massive impact in ambient health monitoring, the equipment required is prohibitively expensive. Systems using RF typically use multiple high-end enterprise software-defined radios, such as the USRP N210 and B210. These configurations are prohibitively expensive which makes their deployment unlikely in large scale smart housing networks. As a more cost-efficient option, common off-the-shelf (COTS) WiFi devices communicate using radio waves, usually in the 2.4GHz or 5GHz range. Internally, many WiFi devices measure the quality of their connection using a metric called Channel State Information (CSI). For each channel in the given spectrum, the device collects the measured phase and gain of the signal at each antenna, which allows for the identification of signal disturbances. Typically, this used at an engineering level to optimise WiFi network links and identify problem areas for connections. Outwith its intended use however, CSI has also been used to monitor respiration, heart rate, and for gesture recognition (Liu, Wang, Chen, Yang, Chen & Cheng 2015, Khamis, Chou, Kusy & Hu 2018, Zou, Liu, Wu & Ni 2017) in contact-free scenarios. CSI collection from common off-the-shelf WiFi devices offers an inexpensive method for ambient RF-based sensing, which could also benefit from the existing WiFi infrastructure in most homes.

As a low-level feature of some wireless devices, CSI has been used to monitor wireless performance internally. Some research has been performed using easily retrievable metrics such as Received Signal Strength (Wen, Tian, Wang & Lu 2015), however CSI is more desirable as it is a rich measurement with significant dimensionality. CSI measurements cannot be retrieved without permission from the device driver. A modified firmware must be used with a supported WiFi device to provide access to CSI, such as the Linux 802.11n CSI Tool (Halperin, Hu, Sheth & Wetherall 2011a) which was released in 2011. This tool supports the Intel IWL5300 and was the first to enable CSI collection on a 802.11n network card, allowing for gain and phase to be monitored across 30 subcarrier channels. This fostered interest in the use of CSI for alternative uses beyond channel performance analysis, with diverse applications such as human identification (Zeng, Pathak & Mohapatra 2016). However while this tool allowed for development with fairly modern wireless technologies at the time, many researchers are still currently using the Linux 802.11n CSI Tool despite its reliance on older hardware. One reason for this is that the development libraries produced for the tool are written in MATLAB, and contain specific calibration parameters for the IWL5300. Another is the lack of modified wireless drivers for most modern wireless hardware. There are currently no published studies identifying the difference in performance in device-free wireless sensing applications, such as respiration monitoring, between data captured from the Intel IWL5300 and more modern wireless hardware. An effective performance comparison is necessary to inform the selection of CSI hardware for deployed sensing applications.

To understand the possibilities in CSI-based sensing, we consider modern health and behavioural monitoring applications. As the significant majority of research to date is still performed using the IWL5300, we cover applications developed using this hardware solution. We observed a variety of CSI-based health monitoring capabilities in literature demonstrated using the Intel IWL5300. The most prominent sensing applications exploited fine-grained data generated in controlled environments. This approach is most common in state of the art CSI research, as RF noise from external sources is minimised and subject movement is limited to that the researchers wish to monitor. CSI filtering and signal processing are the primary methods to extracting discernible features for sensing applications. In this section we investigated the efficacy of these approaches in respiration and heart rate monitoring, and gesture and activity recognition tasks.

2.4.1 Vitals Monitoring

Respiration monitoring systems aim to track the depth and period of a subject's breathing pattern. This monitoring task is usually performed using wearable electrodes or a sensor mat placed on the bed. CSI-based methods rely on tracking the movement of a subject's chest through the observation of periodic changes in the amplitude or phase of a specific subcarrier channel. Liu (2015) processed CSI amplitude using high frequency noise filters to produce a sinusoidal breathing pattern in the presence of a subject. This method requires filter parameters to be tuned for specific environments, with performance varying based on the subject's average breathing depth. A peak counting algorithm can then be used to estimate a respiration rate based on the extracted signal's period. While a sinusoidal pattern is visible in the presence of one subject, two subjects will produce interference. In this case, a fourier transform can be applied to the signal in order to identify underlying breathing frequency of each subject. This approach achieved comparable vitals monitoring accuracy to commercial sleep trackers such as SleepIQ (Grifantini 2020).

80% of breathing rate estimation errors were less than .5 breaths per minute, even in cases where the resident was up to 10m away from the sensing apparatus. As sleeping was the original focus of the work, Liu also highlighted the system's potential capability to highlight sleep apnea in a subject, observable as interruptions in a sinusoidal breathing pattern. Similar signal processing-based approaches for CSI data analysis are common in controlled scenarios.

Heart rate estimation is another area where signal-processing can be used. Currently, wearable sensor technologies are almost uniformly adopted for heart rate estimation, using either chest-mounted electrodes or light-based wrist mounted sensors. CSI-based heart rate estimation aims to measure diastole and systole, however current research which successfully performs this task remains nebulous on the actual measurement their systems capture. It is speculated that skin vibrations produce the regular patterns being observed, however this remains unclear (Liu, Chen, Wang, Chen, Cheng & Yang 2018). Liu's approach to heart rate estimation is largely similar to that of respiration, however a butterworth bandpass filter is applied to the data after the running mean filter, which isolates signals within a given frequency range. This approach yields consistent heart rate measurements, with 90% of estimation errors falling within 4bpm.

Later works in vitals monitoring centred around two primary novel approaches. Khamis (2018) proposed a method for isolating relevant frequency components from CSI data through an automated calibration method. This greatly improved the robustness of otherwise similar approaches to vitals monitored as used by Liu. In general, modern approaches have benefit from the use of CSI data from multiple antennas. Zeng developed a method for data fusion using CSI from multiple colocated antennas to reduce phase noise which they denoted as CSI Ratio (Zeng, Wu, Xiong, Yi, Gao & Zhang 2019). While the performance gains in controlled scenarios are minor, the system showed increased accuracy at greater subject distance. In respiration monitoring, a mean absolute error rate of 3.6bpm was observed while the subject was 7m away from the sensing apparatus. These approaches have proved key in low-level monitoring tasks where fine-grained detail is required.

2.4.2 Behaviour Modelling

CSI-based sensing also shows demonstrable propensity in high level behavioural monitoring tasks (Wang, Jiang, Hou, Dou, Zhang, Huang & Guo 2019). Vitals monitoring applications work well with signal processing methodologies as they can rely on repetitive low-level common instinctual motions like respiration. More significant body movements are far less consistent between different subjects. Personalised approaches to CSI-based sensing rely on modelling techniques to adapt to varying subjects. While standard ML algorithms such as support vector machine (SVM) have been used for some CSI research (Wang, Liu & Shahzad 2016a), these solutions can suffer from performance reductions as additional unique participants and environments are introduced. To produce CSI representations for basic ML algorithms, features are manually extracted such as the minimum/maximum/average values measured for each channel. However, this can produce models which are optimised specifically for small numbers of training examples, rather than to achieve the high level task. Signal processing-based methods alone can be sufficient for some basic physiological estimations, however deep learning is of increasing relevance in CSI due to its high dimensionality. Deep learning algorithms, while requiring significantly more training examples, can extract features from the dataset rather than relying on manually engineered features. Coarse body movements, crowd counting, and even rudimentary gait analysis can be performed by training deep learning models on CSI data (Zou, Zhou, Yang, Gu, Xie & Spanos 2017, Wang, Liu & Shahzad 2016b). These systems exhibit models which can scale effectively across an increasing number of participants. The capability to train a common model for behaviour monitoring using complex CSI data is important, as generating personalised models for individual subjects is infeasible.

2.4.3 Activity Recognition

We investigated literature in activity recognition as it presented both a method of preserving individual privacy by categorising coarse movements, and achieving our wider goal of supporting and enhancing the capabilities of behavioural monitoring systems currently powered by low-fidelity sensor solutions. There is a common conflation between activity recognition, and its lower-level counterpart, gesture recognition. While gesture recognition covers small movements such as hand waving, arm and leg lifts, activity recognition refers to a sequence of movements which form an activity, such as washing dishes or preparing food. Gesture recognition relies on a much smaller feature space since these movements are short, whereas activities require a understanding sequential behaviours relate. However an overlap exists in the underlying techniques used for these systems.

Activity recognition can be performed by monitoring changes in the CSI spectrum. There are a variety of approaches to this task including dimensionality reduction techniques such as principle component analysis (PCA), or automated feature extraction with deep learning models such as a convolutional neural network (CNN). Prominent methodologies can be categorised as using or combining statistical feature extraction and digital signal processing.

Systems using statistical features extracted from CSI rely heavily on calibration techniques designed to reduce the influence of RF noise. Arshad (2017) filtered incoming CSI packets based on their Modulation Coding Scheme (MCS). CSI frames are sent using a given wireless channel denoted by their WiFi configuration. The MCS determines the data rate outgoing frames are sent with. Frames sent using different MCS are transmitted on incompatible channels, and so ensuring the MCS is consistent for a CSI stream is important. A second order butterworth low pass signal filter was then used to remove high frequency noise components, and preserve low frequency signals generated by human movement. Statistical features of the resultant CSI data could then be expected to remain consistent between different activity performances. The mean, standard deviation, 25th and 75th percentiles, median absolute deviation, and max of the filtered CSI were extracted for use with kNN and SVM classifiers. Average accuracies of 94.2% and 94.0% were achieved with kNN and SVM respectively. A worst case accuracy of 78% was observed in classifying walking activities, and 94% on "hands moving" performances. This approach showed CSI activity recognition could be achieved with relatively similar approaches to those used in accelerometer-based activity recognition (Ordóñez & Roggen 2016).

Despite its successful use in published works, some researchers have indicated that passband filtering techniques could introduce distortion to the resultant signal. Palipana (2018) outlines that passband filtering is designed for narrow bands. Human signals produced during movement typically occupy larger bands which cannot be effectively filtered this way. Instead, FallDeFi utilised a Discrete Wavelet Transform (DWT) for threshold-based filtering. The CSI signal is decomposed into its constituent wavelets, which can be independently thresholded before the signal is reconstituted. This resulted in a filtered signal which retains more high frequency components, and does not have the overly-smooth appearance characteristic of passband filtering. This signal is used to produce legible spectrograms for activity performances using a Short Term Fourier Transform (STFT). An SVM classifier was then trained using vectors of geometric features extracted from the spectrogram images. FallDeFi's activity recognition performance was tested in 5 different environments with 12 activity classes. Their novel approach reduced the dependence on environment-specific training, with an observed 80% accuracy when trained with data from a different environment to the target.

More advanced signal processing techniques can also be performed with CSI data gathered from multiple antennas simultaneously. State-of-the-art approaches utilising doppler shift modelling have been employed for CSI activity recognition to improve fidelity by tracking individual limbs during movement (Yan, Zhang, Wang & Xu 2019). These approaches rely exclusively on signals from multiple antennas and cannot be accomplished using single antenna radio solutions.

2.5 Summary

In this review we established our desired methodology and technical framework for employing multi-modal sensors for smart home health monitoring.

We detailed the current solutions for diagnosis of long-term health conditions and the issues faced in getting patients to hospitals at the right time. Additionally, we identified challenges which could impede both the physician and patient in the process of assessment due to performative behaviour. Various assessments for patient health were outlined, and the National Early Warning Score (NEWS) was noted as a concise overview of any cause for concern. Many of these assessments feature behavioural components which are better-suited to observation with non-medical devices. The importance of patient independence and their ability to remain living in their home was also highlighted, and ambient smart home health monitoring solutions were presented as an option to support independent living. By monitoring natural behaviour of the resident over a long-term in their home environment, an opportunity to highlight cause for concern presents itself through approaches similar to the NEWS method. Factors related to risk of long-term health conditions could be extracted using sensor data to establish an overview of patient risk, which could then be used to provide alerts when residents should get themselves checked.

The state of smart home health monitoring solutions using commercial IoT sensor equipment was detailed. Various sensor technologies were grouped into categories based on the way data was gathered from its sensing modality, being polled or event-based. Thus inference is performed on the sensor controller or the sensor network. If the dimensionality of the data is reduced such that inference is performed on the sensor controller itself, a sensor is deemed to be low-fidelity. If additional inference can be performed after the fact with the data provided to the sensing network, then the sensor is rich. Low-fidelity sensors are common in smart home IoT networks, and are demonstrated to have some behavioural monitoring capabilities given domain knowledge and context for the home environment. However their insights are limited as they provide coarse sensing data and require dense, economically unviable deployments to expand upon their basic capabilities. To improve upon coarse behavioural monitoring applications, rich sensors are needed. However many rich sensing technologies are intrusive and infringe upon the privacy of residents. RF-based sensing technologies are typically expensive, however they are both rich and ambient which makes them an attractive option for smart housing. Smart home monitoring using WiFi Channel State Information (CSI) was then considered as a low-cost ambient rich sensing solution which could supplement low-fidelity sensors.

Commerical off-the-shelf CSI sensing hardware is largely limited to the Intel IWL5300 WiFi chipset which is no longer in production. As such it is a not a viable solution for wide-scale deployment as a smart home sensing solution. We highlighted prominent literature demonstrating CSI sensing applications for health monitoring capabilities, including vitals monitoring and activity recognition. Common techniques for signal processing and feature extraction were discussed, and notable performance in desirable applications was observed. These methods are considered when we perform our own CSI sensing research in this thesis.

Chapter 3

FitHomes and FITsense: Enriching Sensor Data Representations for Smart Home Health Research

This chapter will cover the sensor data interpretation and processing methodology developed for the FITsense system. FITsense is a project commissioned by the Albyn Housing Society with the goal of generating resident fall risk profiles using data from the FitHomes network. By observing residents as they move throughout their home, their daily activities and routines can be tracked using a rule-based annotation system. Relevant features are then extracted from this behavioural data to score their fall risk through various factors. These scores can be used to generate a profile of the resident's overall fall risk, which can be tracked over a long-term. Core behaviour and location monitoring systems were developed in the initial stages of this project. These underlying systems now form the primary functionality of FITsense. Initial manual evaluation through labelled data collection with a small subset of residents showed coarse behaviour and location could be accurately tracked. Section 3.2 describes the design of the FITsense system. We then focus on the new work performed to improve the Data Processing and Annotation stage. Section 3.3 outlines the steps taken to sanitise the sensor data ingress.

The FitHomes network has continued to expand beyond the initial deployment of homes. As new FitHome developments are completed, challenges arise in adapting the FITsense system to new floor plan layouts. Homes in the original Dalmore site all follow the same reference layout which allows for the same rules to be used in each house, with some minor refinements to meet personal circumstances. Our newer housing developments, such as those in Nairn and Dingwall, use different layouts. In addition, we also plan to retrofit existing homes with varied layouts into the FitHomes network. In order to expand the FITsense system for use across all house layouts, the current rule-based annotation solution must be changed. It is impractical to develop and maintain an effective rule-base for all layouts, as such a more flexible approach is required. This variation in structural house layout represents a form of personalisation needed to adapt behaviour and activity classifications to a resident's home. To attempt to move beyond the rule-based implementation, we developed an enhanced representation for sensor data for use with personalised deep learning models. These representations are built by encoding the temporal dependencies between sequential sensor events, alongside the sequence itself. The process of gathering labels for FitHomes data for dataset generation is shown in Section 3.4. Section 3.5 investigates the underlying reasoning temporal representations are built on, and details an experiment performed to determine their impact on performance compared to raw sensor event sequence classification.

In this chapter the following contribution is presented:

- **A temporally enhanced data representation for activity recognition models using smart home sensor data.**

3.1 Disambiguation

FitHomes and FITsense are distinct terms which are not used interchangeably. Figure 3.2 illustrates the divide between the FitHomes network and the FITsense system. FitHomes refers to a network of smart homes that are equipped with sensors to monitor residents' health and wellbeing, while FITsense is a behavioural monitoring system that uses sensor data from a given FitHome to determine health risks associated with falls and other conditions. The original contribution in this work relates to FITsense and its implementation with regards to data representation and preprocessing. Houses from the FitHomes network are also used extensively throughout this work, and so we provide context for better understanding of both terms.

FitHomes is a network of smart homes run by Albyn Housing Society Ltd. using sensors to provide health and wellbeing informatics to residents, and provide a testbed for health research. This covers both the architectural and engineering challenges of delivering new housing developments and technological components. These homes are equipped with ambient IoT sensors to collect data about resident behaviours and activity performances. This data is then analysed to identify patterns and trends that can be used to inform care plans, detect early warning signs of health problems, and improve overall health outcomes. FitHomes provide residents with a data dashboard which they can share with family, carers, and healthcare professionals to monitor health and wellness remotely and provide personalised care based on real-time data.

FITsense is a behavioural monitoring system, developed by RGU, to determine health risks associated with falls and other similar conditions. This system uses sensor data from a given FitHome to monitor resident behavioural patterns and factors which may contribute to falls and other health problems. These factors are extracted weekly to produce a profile of resident risk which is used to identify trends which may indicate a specific area of concern.

This chapter outlines our design and implementation of the FITsense system, and our primary contribution which aims to improve the data representation used for FITsense's Activity Classification stage.

3.2 System Design

The FitHomes network and FITsense are complementary systems. The aim of the FITsense project was to produce meaningful insight into a resident's health and wellbeing using sensor data from FitHomes. The system diagram in Figure 3.1 outlines the approach used by FITsense to make use of FitHomes sensor data. Each FitHome contains sensors which relay data to a IoT hub. This hub forwards the raw sensor data to our database running on a remote central server. Thus the FitHomes network provides ingest for the FITsense system. This data is then handled in daily batches by the FITsense sensor data processing system for annotation, activity classification, and profile generation. Each stage of the system generates additional comprehension of the life of the resident, culminating in the visualisation stage. Their FITsense data is provided to the resident with daily updates. The resident can also choose to share their data with their family, carers, and healthcare professionals.

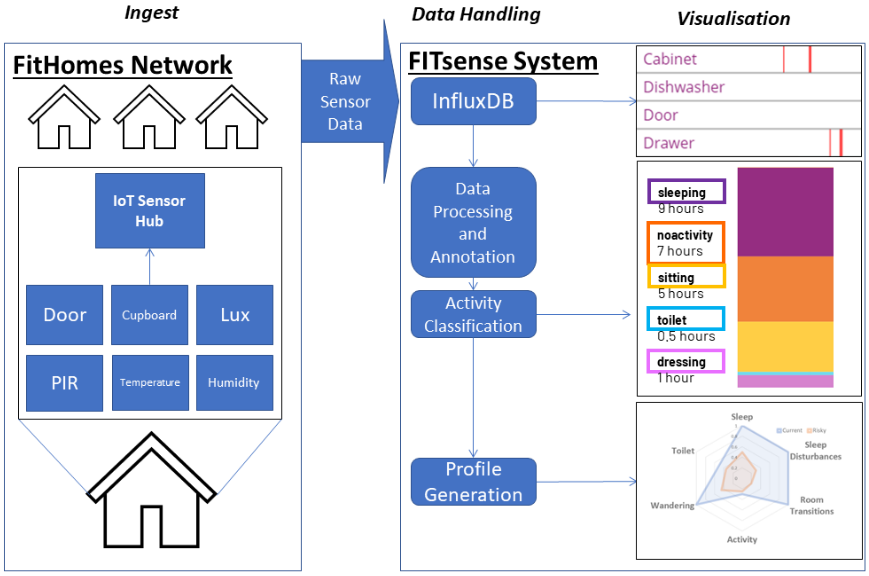


Figure 3.1: System diagram for FITsense and the FitHomes sensor network.

Ingest

Off-the-shelf IoT sensors are used in the FitHomes network: primarily the Aeotec Multisensor 6 and Fibaro Door/Window Sensor 2. They were chosen for their reliability, simple design, and supported integration with the IoT hub used in FitHomes. The simple design for each sensor is key to ensuring their ambient nature when in a resident's home. Each sensor was chosen to blend in with the home design itself, such that the resident is not constantly reminded of their presence. This ambient nature accomplishes two goals, in both increasing the attractiveness of the FitHomes design and concept, and ensuring residents do not modify their behaviour in the presence of the sensors. Initial feedback to the FitHomes proposals showed the older residents, to whom the system was targeted, responded positively to sensors they did not have to perceive, nor interact with. Wearables, cameras (including IR-based ones), and other more intrusive sensing modalities were fully rejected by these residents. As the system has continued to develop, we also observed this positive response with younger residents. Uptake is important at this stage as the project continues to expand, ensuring more potential recipients are happy with the sensors in their home. Additionally, residents may modify their behaviour in the presence of more conventional sensors, such as cameras, as discussed in Section 2.4.2. With our chosen sensors blending into the background, the residents simply trigger the sensors through normal activity around the home.

To capture the resident's movements throughout the home environment, a multi-sensor was placed in each room. The primary functionality of the multi-sensor is its PIR, for motion detection. It also contains sensors for monitoring temperature, humidity, lux, and CO2. The Dalmore FitHome layout has an open plan Living Room/Kitchen space, as can be seen in Figure 3.2. Additionally, the doors between rooms use a sliding design which the residents do not always diligently close. This precipitated careful planning for the placement of each sensor. We optimised placement to allow for power supply to each sensor, while also covering the entire space such that one side of the room does not favour detection over the other. These placement decisions have lead to a generally coarse level of fidelity, allowing for the tracking of one resident in the home. As movement within a room containing two residents cannot be discerned from one-another, it was determined that the sensing applications within FITsense be limited to a single resident at this stage.

Data Handling

FITsense has two stages which receive sensor data input from the FitHomes network: data processing and annotation, by which resident location and behaviour are identified from raw sensor data, and activity classification, performed using the annotated sensor data.

FITsense was originally designed for fall risk determination, and as such, the activities considered in this initial implementation were the core Activities of Daily Living (ADLs). These ADLs are identified through segmentation of the annotated sensor events using room transitions. Each of the ADLs we detect are largely room-specific: sleeping, foodprep, awayfromhome, grooming, dressing, livingroomactivity, and toileting. This assumption-

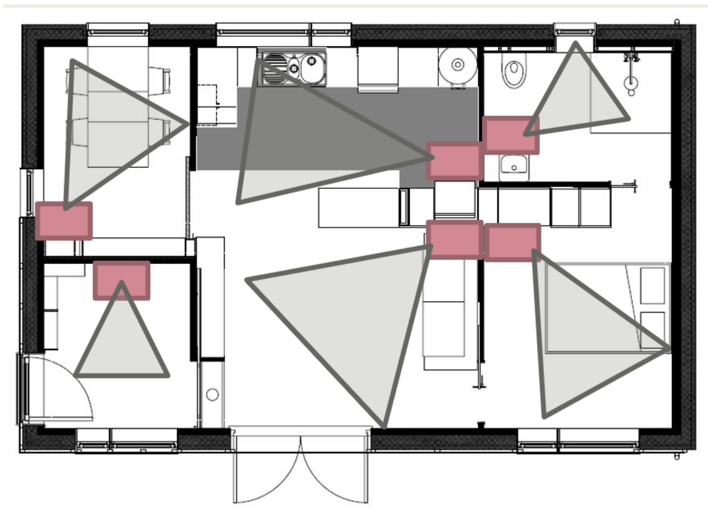


Figure 3.2: Floorplan of the Dalmore FitHome layout annotated with multi-sensor (PIR) locations.

based system is limited in its fidelity, however we consider this to be the best use of the available sensing hardware. As such, we limit the system to the activities we are confident in the system's capability to detect.

From these annotated behavioural sequences, relevant features can then be extracted to generate profiles for various health monitoring applications. The initial prototype of FITsense was implemented to perform identification and monitoring of fall risk factors. However, the implementation remains generic at this profile generation stage, such that other health monitoring applications can also be considered.

Visualisation

The FITsense dashboard produces an array of visualisations for the various data received, processed, and generated through the FITsense system. Most of these dashboard views are time-series-based. These are tailored for different applications and consumers, with some remaining as internal diagnostic tools, while others are polished such that they can be interpreted by front-end consumers. On the right side of Figure 3.1, 3 different visualisations can be seen. The top graph shows direct output from each of the sensors around the home, with a dark red mark showing the sensor was active during that period of time. The bottom graph shows the output from the Profile Generation system, showing compounding scores for fall risk factors. The orange segment in the middle shows the score threshold for risky behaviour which indicates imminent fall risk, while the blue section shows the resident's current score. This particular resident seems to be exhibiting largely normal healthy behaviour, though their overall activity score is low. Family, carers, and health professionals to whom the resident has granted permission to view these graphs can discuss this issue with the resident themselves, and identify the underlying cause for their low activity levels. At this stage, the FITsense system produces meaningful insight into the resident's sensor data which can be used by a viewer to identify when intervention is needed to support health and well-being.

3.2.1 Limitations

The FITsense system is designed to consider input data from a single occupancy home with one permanent resident. However, some homes on the network have more than one permanent occupant or regular carers. In these cases the system cannot track multiple residents as they make divergent movements throughout the home. This also means it is unable to generate more than one profile for a home. It is crucial to note that PIRs produce a single data point regardless of the amount of movement observed, which means that the system is unable to track more than one resident. This will likely cause invalid inferences and impact data quality regardless of any preprocessing stages.

The current FitHomes installations mainly rely on binary infrared motion sensors. However, it is desirable to expand the sensor network with more modern sensor modalities that can deliver higher fidelity sensing applications. During the development of the initial FITsense prototype, a focus on passive sensor technologies was encouraged as residents requested this. This limitation means that the system cannot use cameras, wearables, or other state-of-the-art tools for measuring physiological health data such as gait, heart rate, and other fine-grained activities or behaviors. Despite this limitation, our long-term approach to employing rich sensors is to attempt to achieve similar measurements using ambient sensors.

This issue with sensor fidelity also affects the system's ability to classify activities and fine-grained behaviors. As such, expanding the sensor network to include more modern sensor modalities may improve the system's ability to track multiple residents and classify their activities and behaviors.

3.3 Data Processing and Annotation

Motion detection events from the PIRs used in FitHomes are foundational in FITsense's location and behaviour sensing. An activation from a PIR indicates the presence and motion of the resident in the room. Sustained motion from the resident will cause the sensor to remain activated for a longer duration. These characteristics can be observed to identify both the location and coarse behavioural state of the resident. Resident location can be identified using the room labels assigned to each PIR, whereas coarse behaviour can be classified as either inactive, active, or moving from one room to another. The latter process we refer to as a "room transition". Additional room-specific sensors, such as those on the fridge and cupboard doors, then provide additional insight into a room-specific behaviour. Location and behaviour labels can then be assigned to each sequential sensor activation, establishing a logical sequence of events.

However these annotations rely on sensors only producing activation events when desired. Undesired sensor activation events are considered noise, which detracts from the performance of a sensing application. These events are denoted as undesired, as the sensor may be operating as designed even if this is not as intended. Overlapping sensor influence is the main factor incurring undesired activation events. More than one sensor may share the same input surface, such as PIRs with intersecting areas of influence. This can be observed in Figure 3.2, between the Living Room and Kitchen rooms in the centre of the diagrams. When reaching the threshold between the two rooms, the resident activates both sensors which generates interleaved activation events. These events are inconsistently paced. The last sensor to be activated is often not the PIR in the destination room, thus the interpreted sequence of location transitions is incorrect. This behaviour can also occur when the sliding doors in a FitHome are left open, which is a variable behaviour depending on the resident. If the doors are left open the PIRs will observe spaces beyond their selected room. Undesired activations caused by overlapping sensors typically occur in adjacent rooms. We denote this occurrence as "flickering", referring to the location information from these events implying the resident is quickly moving back-and-forth. This is a minor issue, but can over-represent the amount the resident is moving. More significant undesired activations occur due to sensor interference. This is a common wide-ranging issue which is not unique to the FitHomes sensor network. Interference manifests as sensor events specifically not caused by a resident or their behaviour. These undesired activations can incur a room transition between two incompatible rooms, denoted as those with no direct path between each other. These undesired activations break the logical sequence of events this stage attempts to establish.

Behavioural monitoring is also affected by invalid transitional data. The interpreted sequence of events should follow a logical pattern, just as with location. If the resident is motionless in a room, FITsense should determine them to be located in that room, exhibiting an inactive behavioural state. When they stand up and leave the room, they should be detected to change behaviour from inactive to active, before transitioning from the origin room to their destination.

Undesired sensor activations are the core issue with tracking resident location and behaviour states, and so we implemented two prototype approaches to attempt to eliminate them. The first, called the "Event State" system uses predefined links between adjacent rooms to establish a filter for invalid location transitions. The second, "Managed State", is a naïve approach aimed at mapping the links between rooms. Room transitions are validated with secondary sensor activations after the transition event, using retrospective passes over the data. These approaches are detailed, and then their invalid transition rate is compared using a batch of data retrieved from multiple FitHomes.

3.3.1 Event State Approach

We define an event state as a compound key, representing both the location and behavioural state of the resident. For instance, when the resident is moving in the living room, their event state would be "Livingroom_Active". This representation allows us to assign the event state to the sensor event in the database as additional metadata. The system can then track these states across the sequential sensor events to follow the logical sequence. The goal of this approach was to eliminate the effects of undesired sensor activations.

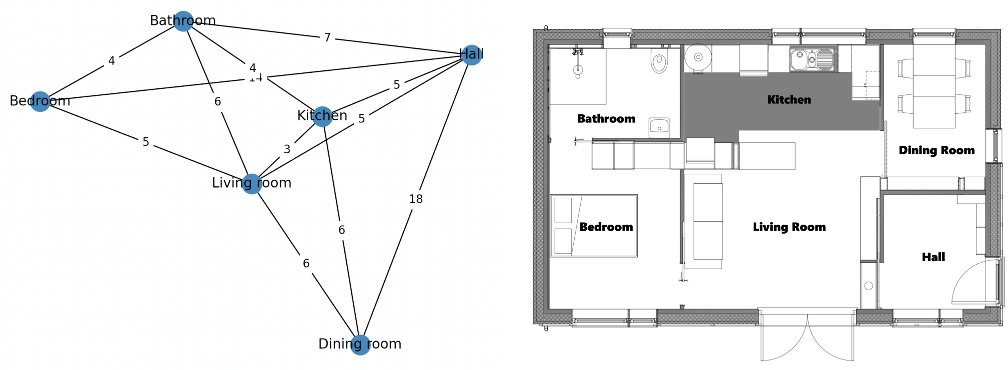
For each daily batch of data, the sequence of sensor events is considered. An initial state is assumed from the first observed room transition. The system then retains an internal event state, used to establish the next logical event state in the sequence. The next sensor event is evaluated to identify whether the location or behaviour component of the event state will change. If the location has changed, the system will check the pre-defined list of room links within the home to establish whether this sensor event is desired or undesired. If a link does not exist between the new room and the previous room, then the new sensor event is discarded. Behaviour is also tracked as either inactive, active, or a room-specific behaviour. Room-specific behaviours are defined by a specific sensor, such as the door sensor on the fridge, or flush sensor on the toilet. In cases where the internal state has reached an event state inconsistent with reality, exceeding a threshold of 10 attempted invalid transitions in a row will reset the event state.

The event state system was designed for the first FITsense prototype, and acted as a reasonably performant data pre-processing solution. In a large portion of cases, the logical sequence of the resident's location and behaviour transitions could be observed. Its main downsides are its reliance on predefined room locations and links, and its dependence on a fully functional sensor complement. These predefined room links may be easily gleaned in cases where the layout is consistent for a set of FitHomes such as the Dalmore installation. However as new and retrofit homes are added to the network, it may prove a less scalable solution. Sensor failures are also common with real-world installations for a variety of reasons. While the event state system functions well in simulated environments, it can suffer from degraded performance with real-world data produced by the FitHomes network.

3.3.2 Managed State Approach

The idea for the "Managed State" approach follows on from feedback and evaluation of the first FITsense prototype. We identified a key difference in the characteristics of resident behaviour and undesired sensor activations: the latter is typically faster and not sustained. As a resident moves through the environment, they usually produce multiple sensor activations from the same sensor as they move within a space. To exploit this, we proposed using secondary sensor activations to validate room transitions.

When a resident transitions from one room to another, the first sensor activation in the destination room does not initially constitute a room transition. Instead, it becomes the candidate room. Once a second event occurs, either from the same PIR or a room-specific sensor such as the fridge, the candidate is validated and the room transition is retroactively applied for the first event. This ensures sustained activity is used to indicate the transition, as opposed to treating each individual sensor event as if it were desired. Invalid room transitions can be filtered through their lack of validation. This approach not only allows us to almost fully nullify the flickering effect, but also eliminate the reliance on predefined room links.



(a) Estimated (b) Actual

Figure 3.3: A network graph generated with the Managed State system, used to estimate the Dalmore FitHome layout from resident data. Each edge is annotated with the average transition duration between rooms.

The home layout itself can also be mapped using this approach. Figure 3.3(a) shows a personalised network graph generated using data gathered from a resident's home, processed with the managed state system. Once a batch of sensor events has been processed, the valid links which exist within the home have been identified. The transitions for each link in either direction can then be collated. This allows us to generate the graph using each potential room as a node, and weight the edges between nodes using their transition durations as the most common transition time. This allows us to roughly estimate the distance between each room with reasonable accuracy. The approach is also inherently personalised, as residents exhibit different average transition times. Using Figure 3.3(b) as a reference, we can see the majority of the links between rooms in the home are accurately represented. We observe the estimated distance between the Bedroom, Bathroom, Kitchen, and Living Room are all similar. The most commonly-used link, Living Room to Kitchen, is 3 seconds, being the shortest path in the home. This is consistent with the floor plan, owing to the open plan layout reducing the average transition time.

Despite this success, the system is not without flaws. While the invalid transition rate decreased significantly, they do still occur. In Figure 3.3(a), an invalid link is present between Bedroom and Hall. The duration marked for this period is estimated as 14 seconds. As this does not align with the summative duration of Bedroom to Living Room to Hall, this link could be discarded. However, it has highlighted a recurring behaviour. The Hall PIR occasionally activates overnight while the resident is in their bed, likely due to the placement which can incur undesired activations caused by outdoor movement. These "invalid" links represent the flexibility of this system to react to recurring behaviours which can be used to diagnose real-world issues with the usage of the sensors.

3.3.3 Performance Comparison

A batch of sensor events was gathered from each of the homes which was then processed separately by the Event State and Managed State systems. [The rate of transitions between invalid room pairs was used to identify both systems' performance in filtering undesired sensor events]. As such, the evaluation only considers the location aspect of annotation. Additionally, the Event State system will be assumed to perform better as it relies on a list of predefined links between rooms. The Managed State system is likely to perform worse due to undesired sensor events negatively affecting link inference, though comparable performance is expected.

Table 3.1: Room transition counts and invalid transition rates generated through the Event State and Managed State systems for one month of FitHomes network data from 8 houses.

FitHome Resident	**Room Transitions**	**Event State (Invalid Rate %)**	**Managed State (Invalid Rate %)**
A	12924	1.3	6.4
B	1355	1.9	4.6
C	4218	1.8	4.8
D	7685	2.1	7.5
E	3532	2.4	6.6
F	4152	3.3	4.8
G	3833	2.0	3.8
H	9904	1.6	7.8

Both systems were compared using data gathered from 8 FitHomes between December 2020 and January 2021. Table 3.1 shows the number of room transitions and invalid transition rates for both systems, generated for this period. On average, the Event State and Managed State systems exhibit invalid transition rates of 2.1% and 5.8% respectively. These rates differ per home, but show no observable trend with respect to transition count. Clearly, reduced performance can be observed using the Managed State system due to the lack of prior knoweldge of room links. As the system requires consistent and sustained behaviour to introduce a new potentially physically invalid room link, this recurring behaviour may be of interest.

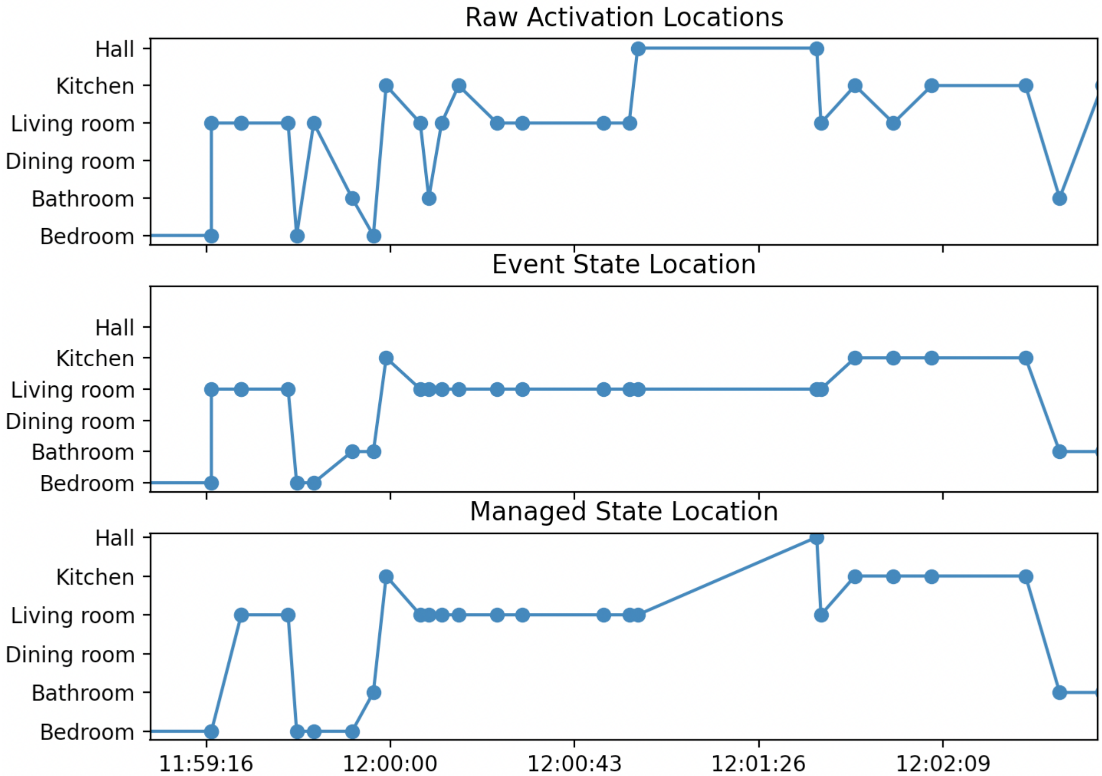


Figure 3.4: Comparison of interpreted locations from raw sensor events, and annotations from the Event State and Managed State systems.

To better understand how these systems compare we provide a demonstration. The example in Figure 3.4 shows a real morning routine observed from just 3 minutes of a resident's FitHome data. Each point on the line represents a sensor event, and each shift of the y-axis represents a transition from one room to another. The lines produced by each system show a different interpretation of the room transitions the resident performed in the 3 minute period. Significantly less flickering can be observed in the lower two graphs. Additionally, many invalid transitions are observed from the raw activation locations. A transition from the Living Room to Bathroom can be observed between 12:00:00 and

12:00:43, despite no direct path existing between those rooms. This transition is filtered by both the Event State and Managed State systems. This is shown as an additional point in the Living Room during this period on the two lower graphs. A key difference between the Event State and Managed State interpretations can be seen in how it handles the Hall sensor event just after 12:00:43. Two activations occur which are filtered by the Event State system, with the latter being retained by the Managed State system. As no other sensor activations occurred during this period, it can be assumed the resident likely entered the hall. The Managed State system uses both Hall sensor events to validate the transition, before returning to the Living Room. While some issues still remain with both systems, they offer a clear benefit over the location interpretations from raw sensor events.

To preserve the desirable performance characteristics from both systems, we developed the enhanced Event State system. This system retains the link inference from the Managed State system while still using a list of predefined links as a backup. This ensures new and retrofit home environments can still be mapped using the inside-out method from the Managed State system. The invalid transition rate can optionally be minimised by providing a list of physical room links within the home. The personalised home layout mapping generated through inference alone is retained for both diagnostic modelling, and for future research considering resident behaviour over a long term.

3.4 Labelled Data Acquisition

In order to obtain labelled data for evaluation and dataset generation, a timeline document was circulated to predisposed residents. This timeline broke the day down into 15 minute intervals, with an option to select which room(s) were occupied during the period and which of the tracked ADLs the resident may have been engaged in. Additional space was also left to allow residents to add information they felt may be useful to activity and pattern recognition. The given timeline spanned 7am to 1am with the assumption that residents would be asleep for the remainder of the 24 hour period. The form was designed to ensure that the data collected would not be entirely curated around several rule-based assumptions that had been formed during development of the FITsense system. For instance, residents can select that they performed an ADL spanning across two rooms which the current system does not consider.

Of the evaluation timelines completed by residents, two were selected due to the high quality of data outputted by those houses at the time of evaluation. The key observation was that the Event State system performed well. Resident locations and ADL expressions matched up well with the established event-states in FITsense data. In most cases where a mismatch occurred a sensor functionality issue existed, or the resident documented unexpected behaviour.

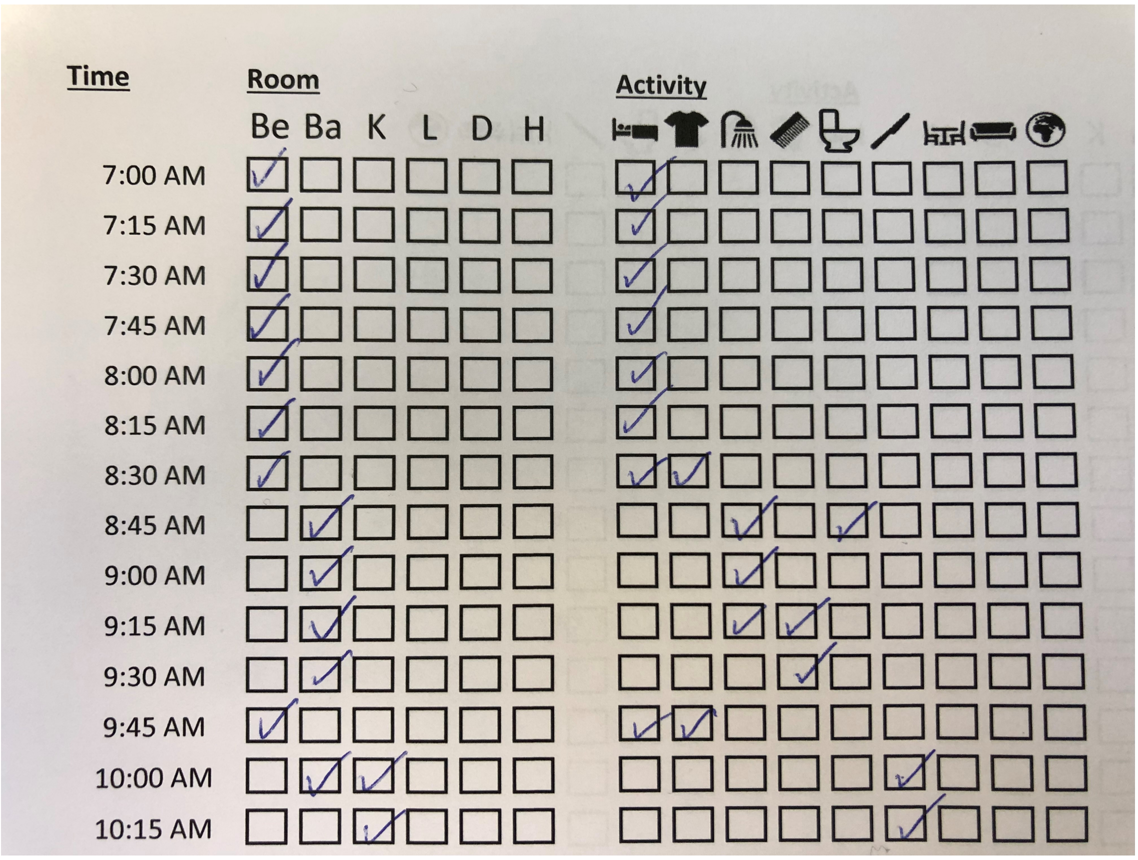


Figure 3.5: A completed evaluation timeline from a FitHome resident.

While the labelled data provided in the evaluation timelines has merit, there are too few examples for extensive use with deep learning. Instead of collecting additional data, an additional 13 surrounding days of unlabelled data was retrieved from the homes. This data was then split into windows, using room transitions to mark each end of a window. By viewing the sensor activation sequence associated with each split, the ADL could be identified considering the residents' documented behaviour as seen in their returned timeline. For example, in Figure 3.5, it can be observed that the resident moves from the bedroom into the bathroom after waking up. During this visible period, the resident has transitioned from sleeping to dressing, then to toilet, shower, and grooming. In unlabelled sensor data, this can be viewed as activations of the motion sensors in the bedroom (with sustained movement for a short period) and the bathroom. This sequence then becomes useful for manual annotation, as it allows for long term prediction of regular behaviours. The recurring activity patterns observed in Figure 3.5 can be used to assist manually annotating the data with ADL labels, producing the fitsense 1 and fitsense 2 datasets.

The FITsense system performs well in real scenarios where houses contained a full complement of functional sensors. Cogent sequences of ADL expressions can be observed, with few noticeable issues in the event-state transitions. When observing differences between the rule-based labels on the same period of FITsense data, it is notable that the "grooming" ADL was rarely identified. To a person performing manual labelling, "grooming" ADLs can be easily observed after a "toileting" instance. However it is notable that the temporal dependence exhibited here is the key identifier. Through sensor activations alone, it could not be expected that a classifier would be able to correctly identify the dependent ADL. In order to improve the performance of FITsense ADL classification, additional sensors would need to be added to the configuration in order to allow more ADLs to be correctly telegraphed. Alternatively, the temporal dependencies which exist in the data could be encoded as part of an activity representation for a ML classifier.

3.5 Representing Temporal Dependencies

Sequences are important in activity recognition. An issue in making use of sequential sensor activation data is to develop effective approaches for harnessing temporal dependencies. The aim of this work is to identify and investigate the importance of different types of temporal dependencies and propose specific representations and algorithms which can take advantage of these relationships.

In this section, we explore the use of different forms of temporal relationships within the sensor data sequence representation. Several alternative approaches to classifying activities from low level, raw data inputs are investigated and effective solutions which leverage temporal relationships are identified. The main contribution of this work is the development of a novel temporal dependency-aware ML approach for activity recognition from event sequence sensor data.

3.5.1 Identifying Activities from Sensor Data

The activity recognition task is typically broken down into two separate parts: windowing and ADL classification. Windowing involves separating the continuous sequence of sensor activations into smaller sequences likely to contain a single ADL expressions, while ADL classification is the process of assigning activity labels to each windowed sequences of sensor activations. We currently employ a rule-based approach for windowing and classification. As a first step the plan is to keep the windowing approach but change to a more flexible ML approach for classifying.

A popular approach to split data into windows is to use a sliding window (Dahmen, Thomas, Cook & Wang 2017). This has been shown to be effective in accelerometer datasets, however there is the potential for windows to overlap and not correctly represent realistic activity behaviours. Our approach to the windowing of long sequences of sensor activations stems from our enhanced Event State system, which tracks the resident's movements and activity through the home. This is an additional filter layer between binary sensor activations and the classification task. Location annotations from Event States can be used to split sequences of sensor activations as the resident transitions from one room to another. Due to the Event State system requiring states to depend on previous locations and behaviour, this rule-based classifier effectively makes use of the implicit temporal dependencies between ADLs which occur sequentially. While this provides the benefit of reducing window overlap, the specificity of ADLs which can be captured is reduced. A clean Event State sequence can be used to split data into smaller sequences for ADL classification.

3.5.2 Temporal Dependencies

In Activity Recognition tasks, McKeever observed that some activities can act as natural precursors to others (2010). An example of this is the understanding that after using the restroom it is more likely that the subject will wash their hands. This sequential behaviour is widely present in the FITsense data and may support activity recognition tasks based on a relationship to behavioural routines which occur in daily life. Some classifiers already take advantage of these sequences to a basic degree. We propose that these regular behaviours can be learned more effectively through the representation of temporal dependencies to a ML classifier.

In data from the FitHomes network, we have observed two significant forms of temporal dependencies likely to be of use in activity classification:

Implicit Dependencies

- the order and sequence in which ADLs take place; and
- the order and sequence of sensor activations within an ADL.

Explicit Dependencies

- the duration of an ADL;
- the duration of sensor activations within an ADL; and
- the time of day at which events occur.

Implicit dependencies refers to the sequential information incidentally encoded in the order and timing of sensor sequences in the training data. In addition, the classification of previous learned examples can influence the learning and prediction of future sequences. In activity recognition data this allows the sequence of ADLs, as well as the sequence of sensor activations, to be encoded as additional knowledge. This may potentially be negatively affected by dataset gathering methods, such as non-contiguous collection. Implicit relationships in this scenario refer to the importance of the ADLs that take place before and after the current instance being classified. Some classifiers, e.g. bi-directional LSTMs, can inherently take advantage of these sequential dependencies. Significance in this model is placed on the order of ADLs; the resultant class of the classification task.

Explicit dependencies in this context refers to the extension of the feature set to include additional temporal features in the representation. The additional features could potentially capture any additional temporal information, such as the total length of a sensor sequence, the time of day at which it occurred, etc. For instance, by splitting the day into quadrants, a coarse timestamp identifying the quadrant of the day at which a sensor activation occurred could be an explicit representation. In initial experimentation, this specific feature was found to have little impact on the performance of our classifiers. Similar manually engineered features have had varying impact across multiple sensor sequence representations. In order to encourage the discovery of temporal dependencies, we implemented cumulative and relative timestamps into the representation. Manually engineering the representation by selecting specific features relies on observations and assumptions of temporal importance in timestamp data, whereas fine timestamps allow for the algorithmic identification of dependencies.

Basic motion sensor data representations typically comprise a set of encoded sensor activations which have occurred within the time window, and fail to capture the implicit or explicit forms of temporal dependency e.g. event sequences and timings. As implicit temporal dependencies incidentally exist in this basic form of sensor data representation, it is the choice of classification algorithm that will determine how effectively these dependencies are represented; Whereas explicit temporal dependencies are represented by manually engineered features their influence will be dependent on how well the manual representations present the additional temporal information for learning.

We investigate a variety of representations and classification algorithms to understand how different data representations affect the influence of explicit temporal dependencies and different classifiers learn implicit temporal dependencies. It is expected that a temporally aware LSTM classifier supplemented with a richer representation containing explicit temporal information may provide better overall performance over the baseline.

3.5.3 Rule-Based Classification

ADL classification in the initial FITsense system lacks granularity but can be effective in real scenarios. The system could theoretically be expanded to capture more complex ADLs but the work required to write and maintain robust rules which consider the full state of the system is substantial and unmanageable as the system scales. Each room in the home is assigned a set of ADLs which could potentially occur. Within these ADLs are specifications which qualify a sequence as an expression of this ADL. These specifications usually confirm the existence of specific event-state transitions and the total length of an ADL. After larger sequences have been windowed using event-state room transitions, each sequence is checked against the available specifications. The key assumptions behind this ADL implementation are: ADLs are limited to one per window and ADLs can only be performed in one room at a time. The master list of ADLs is ordered by importance, to ensure important ADLs are misclassified less often than unimportant ones. In the event more than one ADL specification is matched to a sequence, the specification with the highest importance rating is assigned. The system functions well on the single site with identical home layouts. However, the activity recognition rules rely on manual observation of the specific environment.

The approach has limitations and in particular will not transfer easily to future developments with different sensor configurations or home layouts. This highlights the main issue with rule-based classifiers: the rules must be designed specifically for the environments they operate in. The rules must be tested to ensure edge cases and unexpected sensor behaviours are correctly handled, which is a time consuming process. Rules can also become dependent on 100% sensor uptime. While ambient binary sensors can be cheap, they require upkeep to ensure the general quality of the data being produced is maintained. Additionally, the produced rules are not flexible to sensor configuration changes and other unforeseen events. If a reference house design is not available, manual rule design is required for each unique house layout. This is infeasible for future expansion as these issues will scale exponentially.

3.5.4 ML

ML is the standard approach employed for HAR. Advances in deep learning, recurrent and convolutional neural networks have directed the field for classification of complex sensor data, such as accelerometers and gyroscopes (Zeng, Nguyen, Yu, Mengshoel, Zhu, Wu & Zhang 2014). However, more basic sensor configurations, such as the binary sensor networks seen in smart home environments, have been well-served by Naïve Bayes, Decision Trees and other established classification algorithms (Tapia et al. 2004, Bourobou & Yoo 2015).

While these traditionally established ML methods can perform well in ADL classification tasks, they do not make use of the implicit temporal dependencies found in sensor data. The temporal relationships which exist in sensor data have relevance in ADL classification, as they offer an opportunity to extract additional useful knowledge from sensor activation data. Recurrent Neural Networking methods, specifically Long Short Term Memory (LSTMs), can make use of the temporal knowledge encoded in the sequences of sensor activations and ADLs which occur in training data.

We propose using a hybrid method to improve the temporal awareness of an ML-based ADL classifier. First, we plan to enrich sequential sensor data representations by adding relative timestamps between subsequent activations to representations for training data. Then by using LSTMs to learn intrabatch relationships between instances of ADLs.

Some low-level temporal dependencies have been acknowledged by other researchers, however implementing these relationships efficiently within representations for these classification algorithms has historically had little effect on the overall performance. An impactful approach to representing temporal dependencies represents an opportunity to make use of additional existing data to improve performance in a ML activity classifier.

Rule-based systems, for instance, may make use of the presence or order of specific sensor activations, time of day or delay between activations. This is limited by the degree of domain knowledge for specific environments and relies on rules remaining intercompatible with the existing complement. ML systems are typically less verbose than rule-based systems, however they can infer better internal rules during training than those manually designed. Each sample in the training data is considered when the ML model is synthesised. This ensures that a ripple effect, where incompatibilities with new rules cause issues with existing functional rules, is unlikely to occur. While their black box design makes ML classifiers less explainable as AI systems, the performance of many such implementations is highly desirable.

3.5.5 Experiment

The aim of this experiment is to evaluate the performance of baseline classifiers on binary sensor datasets and compare their performance with LSTM implementations which have implicit and explicit temporal knowledge made available. A selection of popular classifiers were used to establish the baseline performance of traditional classifiers on this problem. These do not make use of the implicit temporal knowledge provided through the sequences in the data.

LSTMs can use previous sequential learned examples to inform their decisions. We hypothesise that the performance of an ADL classifier can be improved by forming long term knowledge based on existing temporal dependencies. LSTMs were selected due to their demonstrated strength in time-series classification. Additionally, LSTMs can make use of both the implicit and explicit temporal information from our data representations. Four LSTM configurations were compared to evaluate the performance impact of implicit and explicit temporal knowledge in ADL classification.

Iterations on our LSTM models were used to identify how performance could be improved by supporting implicit temporal dependencies and adding explicit temporal information. The previous learned examples can influence the learning and prediction of future sequences in a traditional (or unidirectional) LSTM. Stateful LSTMs can make use of additional implicit knowledge, specifically the order in which ADLs are performed. Stateful LSTMs retain the hidden states of neurons between batches during training, allowing intrabatch dependencies to be inferred. These implementations are marked in the results with the prefix “State”.

Relative timestamps were selected as the preferred form of explicit temporal dependency representation. They were found to be most effective through initial experiments. “Exp” marks experiments performed using representations containing explicit relative timestamps.

Table 3.2: Overview of the datasets used.

Dataset	**Classes**	**Attributes**	**Instances**
adlnormal	5	39	120
kasteren	7	14	242
tapia1	22	76	295
tapia2	24	70	208
fitsense1	7	13	744
fitsense2	7	13	990

Data Collection

These experiments make use of three publicly available datasets, alongside two datasets from the FITsense project. The datasets used in these experiments document instances of ADLs and their accompanying binary sensor activations which have been captured using a variety of windowing methods. Details of the six datasets are shown in Table 3.2.

CASAS[1] (adlnormal) This dataset contains the fewest classes, with 5 total activities observed by 39 independent sensors. While there is are small number of instances at 120, the activities tracked in the dataset are diverse enough to present little challenge for most baseline classification methods. As only 5 ADLs are tracked, large timegaps between activities can exist which may impact the performance of stateful LSTMs. This small dataset is non-contiguous.

Van Kasteren[2] (kasteren) The kasteren dataset follows a structure most similar to that of fitsense1/2, with similar tracked activities and sensors. The ADLs expressed in this dataset have relevance to health monitoring and add a layer of complexity which may present a challenge in classification. Prepare_Breakfast and Prepare_Dinner are observably similar as activation sequences, however the time at which they are performed is important. This small dataset is non-contiguous.

MIT[3] (tapia1/2) The most complex of the datasets used in these experiments due to the large number of sensors and classes. Several ADLs could be considered beyond the scope of capability for the sensor network (e.g. Going out shopping vs Going out for entertainment), however the complexity presents a useful challenge for classification. This large dataset is non-contiguous.

FITsense/FitHomes[4] (fitsense1/2) These datasets contain the largest number of instances while using the least sensors of the datasets used in these experiments. The tracked activities were selected for health monitoring applications such as “sleeping”, “grooming”, and “foodprep”. These large datasets contain contiguous streams of ADLs.

[1] `http://casas.wsu.edu/datasets/adlnormal.zip`
[2] `https://sites.google.com/site/tim0306/kasterenDataset.zip`
[3] `https://courses.media.mit.edu/2004fall/mas622j`
[4] `https://www.rgu.ac.uk/fitsense`

3.5.6 Experimental Design

WEKA was used to run a set of baseline classifiers on the datasets. Each dataset was converted from its original format to a zero-padded sequence of discrete values representing binary sensor activations in the ARFF format. Baseline algorithms were selected to evaluate the performance of a representative variety of classification techniques. LibSVM, J48, Bayes and k-NN were selected due to their established significance in the field. Each of these classifiers were run with default configurations as supplied by WEKA.

The following (default) WEKA configurations were used:

- SMO - batchSize=100, c=1.0, toleranceParameter=0.001, epsilon=1.0E-12
- J48 - batchSize=100, confidenceFactor=0.25, minNumObj=2
- BayesNet - batchSize=100, searchAlgorithm=K2
- IBk - KNN=1, batchSize=100, nearestNeighbourSearchAlgorithm=LinearNNSearch

Temporally aware LSTM implementations were configured using Keras, using the Tensorflow backend. The LSTMs are trained using the "categorical_crossentropy" loss function and "adam" optimizer. Different batch size, units and epochs values were used for each dataset due to variation in sequence length, sensor makeup and activity composition. Standard LSTMs were used with a batch size of 256, 128 units and run for 100 epochs. Stateful LSTMs were implemented using the "stateful" option in Keras with a batch size of 256, 128 units and run for 100 epochs. ExpLSTMs were implemented with a batch size of 256, 512 units and run for 50 epochs. While LSTMs may offer better classification performance in some scenarios, they require a long training time in comparison to baseline classifiers such as k-NN. Deep learning backends such as Tensorflow offer distributions which may use GPUs for training instead of CPUs, which can significantly improve the length of time taken to train.

Both WEKA and Keras implementations were run using Leave One Out cross validation, with each dataset being split by day. This ensures fold contain contiguous sequences ensuring realistic meta-sequences are represented, with each fold starting and ending with "sleeping" ADLs. However in the data for baseline classifiers metasequences of sensor activations could be broken up as the implicit temporal dependency between samples is not considered. Each session of training was also repeated three times with fixed seeds to ensure repeatability.

3.5.7 Results

Table 3.3: Baseline Classifier results (macro f1 scores).

Dataset	**LibSVM**	**J48**	**BayesNet**	**k-NN**
adlnormal	0.898	0.934	**0.983**	0.910
kasteren	**0.901**	0.891	0.871	0.892
tapia1	0.162	**0.303**	0.246	0.248
tapia2	0.129	**0.314**	0.070	0.219
fitsense1	0.281	0.613	0.600	**0.667**
fitsense2	0.464	**0.620**	0.530	0.560

The results for the baseline classifiers are shown in Table 3.3. Each algorithm delivers the highest result on at least one dataset, however the overall winner is narrowly J48. This is due to its performance on more complex datasets such as tapia1/2 and fitsense2. Bayes and LibSVM demonstrate impressive performance on the adlnormal and kasteren datasets respectively. Performance of the baseline classifiers on the tapia datasets is relatively poor in comparison to the strong performance on others. Performance on the fitsense datasets is relatively good in comparison to tapia, however it still falls short of that seen on adlnormal and kasteren.

The LSTMs display a more balanced performance across all datasets. Slightly poorer performance can be observed between the top performing baseline classifiers and ExpLSTM implementations on adlnormal and kasteren datasets. On the more complex tapia datasets, improved performance over J48 can be observed in the ExpLSTM implementations. This improved performance on more complex datasets can also be seen in fitsense datasets, with ExpStateLSTM exceeding the baseline classifiers. The overall winner in

Table 3.4: LSTM results (macro f1 scores).

Dataset	LSTM	StateLSTM	ExpLSTM	ExpStateLSTM
adlnormal	0.932	0.951	**0.975**	0.918
kasteren	**0.874**	0.831	0.867	0.856
tapia1	0.212	0.202	**0.331**	0.287
tapia2	0.133	0.240	**0.359**	0.256
fitsense1	0.853	0.833	0.740	**0.864**
fitsense2	0.676	0.728	0.586	**0.752**

the LSTM implementations is ExpLSTM, with most results showing a leaning towards implementations utilising additional temporal knowledge.

On the fitsense datasets, ExpStateLSTM achieves impressive F1 scores (0.864 and 0.752 on fitsense1 and 2 respectively) and outperforms all other baseline and LSTM approaches. fitsense1/2 are the only contiguous datasets in the experiments, and so are expected to benefit most from the implicit temporal dependencies captured by maintaining the sequence of ADLs. The performance of ExpStateLSTM supports this intuition.

3.5.8 Discussion

Of the baseline classifiers, J48 shows the most balanced performance across all datasets. Decision trees display strong performance in general activity recognition tasks. A potential future area of interest may be to investigate the methods by which temporal knowledge could be effectively represented in the training of decision trees for ADL classification.

The baseline classifiers did not have access to any temporal relationships, however on the simpler datasets (adlnormal and kasteren) they all achieved impressive results that outperformed LSTMs with some algorithms. This suggests that for simple classification tasks temporal relationship information is not required. However, on more complex tasks, including the FITsense data, the baseline algorithms' performance was poor highlighting the need to harness temporal relationships.

All LSTM implementations displayed more balanced performance than the baseline classifiers, with variants making use of additional temporal information giving better performance on the more complex classification tasks. Stateful LSTMs performed better than temporally unaware LSTMs across several datasets, however they did not perform best overall. While our temporally unaware LSTM narrowly won on the simpler kasteren dataset, ExpLSTM gave the best overall performance on the 4 publicly available datasets. This highlights the importance of capturing specific event timings as part of the representation for more complex tasks.

The key motivation for this section was to achieve good performance on FITsense data. On fitsense 1/2, ExpStateLSTM was a clear winner, highlighting the importance of the activity sequences for this data. This is because the FITsense datasets are different to the others in that they are a formed from a continuous time-stream and have contiguous windows. The inclusion of null or "none" ADL states ensures the complete sequence of activities is retained. As a result, the implicit meta-sequences of ADLs which occur in the data can be effectively used as an additional source of temporal knowledge.

The inclusion of relative time-stamps as explicit temporal information improved performance in most scenarios. This approach to temporal knowledge representation appears to have been successful in encouraging the discovery of temporal relations. The combination of implicit and explicit temporal representation performs best on the fitsense datasets, which are entirely contiguous.

3.6 Summary

In this chapter, we detailed the FITsense system, its integration with the FitHomes network, its outputs, and the limitations faced in scaling the system to additional homes. We presented novel contributions both in our Data Processing and Annotation methodology, and our temporally-enhanced representation for sensor data.

Our approach to Data Processing and Annotation using the Event State and Managed State systems allows the FITsense system to establish a logical sequence of location and behaviour transitions from smart home sensor data. This approach demonstrates a clear improvement over raw event interpretation. Using predefined room links with the Event State system, we observe an average invalid transition rate of 2.1%. The Managed State approach allows us to use inferred links, effectively mapping the resident's home layout with an personalised inside-out approach. Inferring the home layout, without predefined links, we observe an average invalid transition rate of 5.8%. Our current system now makes use inferred links to generate personalised mappings based on how the residents use their homes, with the option to provide predefined links to reduce the invalid transition rate.

The enhanced sensor data representation defined in Section 3.5.2 showed an improvement in performance over more basic representations. Our stateful LSTM implementation using our explicit temporal dependency representation improved performance by 20% over baseline classifiers. This improvement in performance provides additional value from the scant labelled data we can currently collect through residents in the FitHomes network.

These contributions provide further refinement to the FITsense system which improve its ability to scale as the FitHomes network expands. To proceed further with FITsense, we must continue to improve the system in two key areas. We need to continue developing our scalability, by investigating additional ways we can exploit the soft rule-based labelling system for dataset generation. We also look forward to progress beyond the room-specific, generic ADLs we detect. The primary limiting factor affecting the fidelity of the behaviours we can detect is our sensing hardware. Moving beyond simple PIRs, we require sensor technologies capable of providing us with rich data while remaining completely ambient and unintrusive.

Chapter 4

Utilising Channel State Information using Off-the-Shelf Hardware

This chapter introduces Channel State Information and the work performed to address challenges faced in using this technology with commercial off-the-shelf hardware (COTS). Channel State Information measurements identify the wireless conditions for a given transmission, typically measured at the receiver. Variations in the wireless conditions caused by human presence can be used for various sensing applications. The key concepts of CSI-based wireless sensing are introduced in Section 4.1. These are fundamental to our understanding of CSI data, and how features can be extracted or inferred for wireless sensing applications.

Typically, sensing applications are developed in controlled environments with specific hardware. This requirement is a major obstacle in developing CSI sensing solutions to be used in real-world environments (whose wireless conditions vary) and deployed at scale, with low-cost, readily-available hardware. As such, few real-world commercial applications of CSI sensing exist. Hardware-specificity is another key issue at play in researching commercially viable deployment. A wide array of CSI-capable hardware solutions exist, however few sensing works have been published using more recent offerings. Newer CSI-capable hardware differ significantly. Direct comparisons are difficult to perform due to the bespoke software implementations for each hardware platform. This is our base motivation in developing CSIKit: a CSI parsing, processing, and visualisation toolkit written in Python, which supports all COTS hardware.

CSIKit is aimed at three key demographics: researchers performing work with multiple CSI hardware solutions, data scientists, and researchers new to CSI research. We aim to enable a new generation of CSI researchers by homogenising approaches to CSI data exploration and development. CSI-based sensing research is typically performed by and targeted to wireless engineers, due to a relatively high barrier of entry. The documentation and software resources supplied with COTS CSI toolkits are difficult to interpret and inconveniently designed. They're also largely written in MATLAB, experience with which is lacking among data scientists. For instance, Nexmon CSI's parsing software requires each configuration parameter to be specified for an input file. Identifying that this issue exists with most COTS CSI toolkits, we developed CSIKit to work with all COTS devices and auto-detect all formats and configurations. This enables identification of arbitrary CSI files without manual configuration. After data is parsed in CSIKit, it is returned in a homogeneous format regardless of the COTS device used. This enables common sensing implementations which can be applied to data from any COTS device, reducing the barrier to entry for new and existing researchers. CSIKit is detailed in Section 4.2.

Once we finished initial development on CSIKit, we set out to observe differences in between COTS CSI hardware. We identified the variable impact of Automatic Gain Control (AGC) on COTS devices. This affects CSI sensing performance due to an arbitrary scaling factor which is applied as a result of AGC. The most popular CSI toolkit includes a solution for mitigating AGC, though it does not apply to other COTS devices. Gao et. al suggested a hardware-invariant solution for AGC mitigation (2019). However their demonstration of this approach is domain-specific and does not prove its efficacy. In Section 4.3, we use CSIKit to demonstrate the link on which this solution is based and perform an experiment to compare its efficacy across COTS devices. We have also made an open source implementation of this solution available through CSIKit.

In this chapter the following contributions are presented:

- **CSIKit**: a CSI parsing, processing, and visualisation toolkit written in Python, which supports all COTS hardware

- A **demonstration of the link between RSSI and pre-AGC CSI**
- An **open source implementation** of hardware-agnostic AGC mitigation using RSSI
- An experiment **comparing the performance of hardware-agnostic AGC mitigation** using CSI from several COTS CSI devices

Our goal is to identify a viable WiFi CSI sensor solution for dense deployments in FitHomes, to enhance the behavioural monitoring capabilities of the FITsense system. We first need to investigate the capabilities of COTS hardware. To do this, we developed a software toolkit for use with data from all COTS CSI hardware: CSIKit, covered in 4.2. This was then used to process data for the WiFi CSI sensing applications we developed in 5.1 and 5.2. We can compare COTS hardware performance using these sensing applications with the homogeneous data representations provided using CSIKit. In 6 we compare RF and sensing performance of the available COTS CSI hardware solutions. This allows us to identify the lowest-cost hardware required for CSI-based sensing. We can then deploy these sensors in a real-world FitHome, to evaluate how behavioural monitoring fidelity can be improved using both the CSI sensors and the existing ones. We detail this in our real-world in-home study in 7. Through this approach we demonstrate a viable cost-effective approach to multi-modal sensing for personalised smart home health monitoring.

4.1 Channel State Information (CSI)

Several forms of wireless communication employ some form of channel measurement. This refers to the measurement of effect of propagation on a transmission, which is impacted by several factors such as scattering and attenuation. The attributes of these measurements are referred to as Channel State Information (CSI). This work focusses on the use of WiFi-based channel measurements, however this technology is not limited to WiFi. Many OFDM wireless communications standards employ channel measurements, including LTE, 5G, and LiFi. We opt to use WiFi due to the low cost of the underlying hardware, its ubiquity both in usage and documentation, and lack of licensing costs.

In order to understand how WiFi signals propagate, and the challenges we face in generating viable data for wireless sensing applications, we must understand how WiFi devices communicate. We then detail the contents of CSI data and methodology for constructing a timestream. Finally we outline the hardware restrictions we face in using COTS devices for CSI sensing, and our approach to investigating long-term ubiquitous deployment. This section provides necessary context to understand the usage of CSI for wireless sensing before we detail our contributions to the field.

4.1.1 WiFi

WiFi is an incrementally stacked set of protocols designed to support networking and communication between devices. Original Equipment Manufacturers (OEMs) implement various configurations of the IEEE 802.11 standards to support wireless transmissions for many applications. These devices communicate with one another using fundamental Protocol Data Units (PDUs) known as Frames. Frames can travel from a transmitting device to a receiving device through multi-path propagation. More than one path is traced by the transmit radio waves, which makes them vulnerable to interference sources from any direction. This is core to most wireless sensing approaches: the subject of the sensing application generates a disturbance in the wireless transmission.

CSI allows us to monitor these disturbances. A single channel measurement can be provided with a WiFi Frame. Continuous CSI can be collected by appending multiple instantaneous channel measurements over time. Variations and disturbances in the channel can then be measured within a given time period. However we must ensure each measurement is from a coherent channel, such that sequential measurements are comparable. Transmit frames are sent through a fixed-bandwidth transmission on multiple carrier frequencies which are modulated to encode data. The IEEE 802.11 standards specify different methods by which frames can be modulated. Through the top level IEEE 802.11 standards, being Legacy/B/A/G/N/AC/AX/BE, and then through the standard-specific Modulation Coding Schemes (MCS) and other functionality. These distinctions in hardware capability ensure the widest range of compatibility while allowing newer features between supported devices. The work in this thesis focuses on the use of 802.11n, unless otherwise specified. The given configuration of these standards and features constitute a specific channel when used. This channel is not identical to that which would be measured with a different configuration. This means channel parity is necessary for a coherent continuous CSI measurement.

Next, we consider the contents of channel measurements and how the representation is constructed.

4.1.2 CSI Representation

A single frame of CSI data is represented by a matrix of complex numbers. This matrix has 3 dimensions, defined by the bandwidth of the WiFi frame, the number of transmit streams, and the number of receive streams. The latter will be referred to as the number of antennas. The bandwidth used for a WiFi frame is separated into component carrier frequencies through Orthogonal Frequency-Division Multiplexing (OFDM), by which S subcarriers are used to encode and transmit data independently. As each subcarrier can serve separate data streams, CSI will be different as captured from each subcarrier. In a hardware configuration using t transmit antennas and r receiving antennas, this can be represented in a matrix as CSI for a given packet transmission i.

$$CSI_i = \begin{vmatrix} H_{1,1} & \dots & H_{1,r} \\ H_{2,1} & \dots & H_{2,r} \\ \vdots & & \vdots \\ H_{t,1} & \dots & H_{t,r} \end{vmatrix} \tag{4.1}$$

$H_{t,r}$ for a given transmit and receive pair represents a vector containing complex pairs captured for each subcarrier. The number of subcarriers available depends on the hardware configuration used for both the transmitting and receiving device and the channel bandwidth they operate on. In general, a 20/40/80MHz channel uses 64/128/256 subcarriers respectively. Notably, some subcarriers function as guard carriers (or guard bands) to reduce interference, and so these remain empty or invariate by design. The remaining subcarriers are for data, which we use in wireless sensing applications. If the number of available subcarriers is S, a given $H_{t,r}$ pair can be expressed as:

$$H_{t,r} = [h_{t,r,1}, h_{t,r,2}, \dots, h_{t,r,S}] \tag{4.2}$$

The complex number h_s generated for each subcarrier contains the effect of transmission on the signal from a subcarrier, from which phase θ and gain $|h_s|$ can be derived.

$$h_s = |h_s| \exp(j \cdot \theta) \tag{4.3}$$

$H_{t,r}$ contains the summative impact of the effects of propagation on the transmit signal. The combination of these external factors make up the effect of multi-path propagation:

- Reflection - Change in phase as the signal rebounds;
- Scattering - Variations in path affecting signal shape;
- Attenuation - Reduction in observed amplitude.

Static objects in the environment affect multi-path propagation in a consistent manner, whereas dynamic objects such as human bodies have a variable effect. This variation in the observed CSI data can be interpreted as impacts on the multi-paths caused by human activity. This may be through passive motion such as respiration, or active movement such as walking.

To interpret multi-path disturbances in a temporal context, a coherent channel must be measured through repeated sampling. Digital radio signals are typically sampled at a fixed rate. Many Digital Signal Processing (DSP) techniques rely on uniformly-sampled signals. The channel for a given CSI measurement can be usually sampled at any given moment, however a uniformly-sampled continuous channel measurement is not possible with WiFi. The key requirement for a WiFi CSI measurement is the transmission of a Frame, which is used to generate instantaneous CSI.

Due to many factors, 802.11n WiFi frames cannot be repeatedly sent with a guaranteed interval. This is both a limitation on both the transmitter, as a result of Carrier Sense Multiple Access (CSMA) collision avoidance and detection, and the receiver, through traditional full-stack packet loss. WiFi CSI should be assumed to be inconsistently sampled (high jitter), regardless the method of traffic generation. This can occur for a variety of reasons, with the primary one being interference from other devices.

4.1.3 Interference and Noise

Our approach to CSI sensing requires a transmitter and receiver to be located separately, which means that both devices must be operated independently with some degree of synchronization. Frames are scheduled to be transmitted with a given interval, for which the receiver can sample CSI to generate a timestream. However, achieving a specific timing goal with CSI measurements is not always possible. This is especially challenging in home environments, where WiFi is a shared space with other devices.

As other wireless devices operate normally they can impact the timing and consistency of our CSI measurements in several ways. For example, capturing CSI from wireless traffic of another device accidentally can result in not measuring the same wireless channel or PHY. Another problem arises when a device attempts to transmit data at the same time as other devices. This can lead to collisions and packet loss, resulting in a decrease in the accuracy and reliability of the CSI measurements. There is no guaranteed airtime for a device in 802.11 transmissions, and so the more devices operating in a given 802.11 channel, the less airtime available for consistent intervals.

To mitigate these issues, our main strategy is to use a rarely-used WiFi channel exclusively for our CSI measurements. Transmission and forwarding are done on separate channels, which limits traffic from other devices to our collection device. Even with this mitigation, it is unlikely that perfect measurement timings between devices can be achieved. This is due to the variations in signal propagation through the environment caused by factors such as temperature and humidity. Thus, we must account for inconsistent CSI timings.

To achieve a periodic measurement of a coherent channel, interpolation is used to account for the noise caused by inconsistent CSI timings. Through a time-series representation of the wireless channel gathered through CSI, we can observe disturbances in the multi-path over time. These disturbances can be observed for device-free human sensing applications, where we can detect the presence and movement of humans in the environment without the need for any additional hardware.

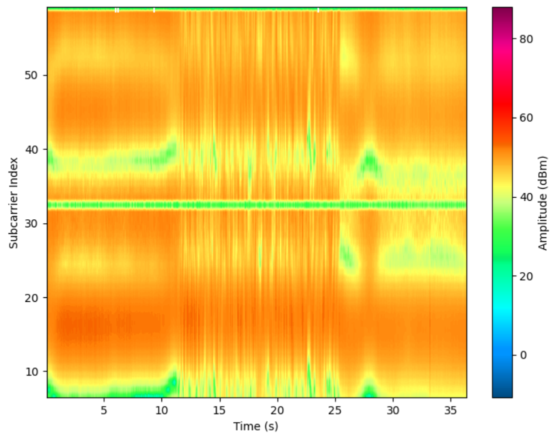


Figure 4.1: CSI heatmap example showing a period of movement between 11s and 25s.

4.1.4 Sensing Applications

Human movement typically causes large, rapid changes in CSI. To illustrate this, we captured CSI in a small environment for 40 seconds. Figure 4.1 contains a heatmap showing CSI amplitude on the Z axis. Each subcarrier is stacked on the Y axis, while sequential frames of CSI form the time-series shown on the X axis. Between 5-10s the human subject remains stationary, and as such the channel is relatively consistent. Between 11 and 25s, the subject stands up, walks around the room, and then sits back down. During this period, the CSI amplitude changes rapidly across all subcarriers. This increased variation consistently occurs when human motion disturbs the wireless channel, which forms the basis for our approach to device-free wireless human sensing.

This example shows a basic approach to using CSI amplitude for wireless sensing applications. Most CSI sensing applications make use of some form of feature engineering by using statistical features of CSI data. These features can be easily integrated in statistical modelling approaches such as Support Vector Machine (SVM). Alternatively, features of CSI can be inferred given large amounts of labelled data. Deep learning approaches to CSI sensing are increasingly common, and could likely result in more personalised models for fine-grained sensing applications. In this thesis, we focus on statistical and deep learning approaches to CSI sensing.

Tight control over hardware operation is needed to maintain a coherent channel for wireless sensing. The deployment and RF front-end configurations for available COTS hardware differ significantly, especially in the level of user access provided in their settings. We consider the challenges faced in using COTS hardware for WiFi CSI sensing.

4.1.5 Hardware Challenges

While all devices which follow the 802.11 specification could theoretically be capable of CSI collection, support is rare on consumer hardware. Many first and third party solutions for WiFi CSI support exist, but each is limited to a specific chipset or manufacturer subset.

The majority of COTS CSI research and development is performed using the Intel IWL5300. The chipset itself was released in 2008 and it is no longer in production. Despite this, research using this chipset is still regularly published. This is largely due to its low secondhand cost, MIMO antenna support, and the accurate CSI collection solution released by Daniel Halperin in 2011 (Halperin et al. 2011a). State-of-the-art functional applications have been achieved using this hardware, ranging from movement detection and activity recognition, to environment mapping and water pressure monitoring (Arshad, Feng, Liu, Hu, Yu, Zhou & Li 2017). The primary limitation of this hardware is that it cannot be deployed in any meaningful scale. The hardware deployment options are inflexible, and increase base cost and physical size. The current off-the-shelf options require power-hungry x86-based compute units. This necessitates the requirement for power and cooling, which in many cases requires a fan which can be loud. Finally, many of the applications delivered using this hardware are developed in a hardware-specific manner. To transition from CSI being solely for research applications rather than an off-the-shelf sensing solution, these challenges must be addressed.

In acknowledging the limitations of the popular hardware approaches to CSI, the larger question becomes: how can CSI sensors be deployed to maximise sensing coverage? An attractive approach is to utilise dense deployments with redundant purview. This ensures human movement can be monitored throughout the environment, mirroring PIR sensor deployment strategies. This approach isn't feasible to apply with the IWL5300 or other similar solutions, due to their size, cost, and power consumption. To even deploy one of these solutions per room would be prohibitively expensive. Conversely, ubiquitous deployment can be achieved with the ESP32, a low-cost, low-power micro-controller with CSI collection support. While there is a strong value proposition for ESP32, the quality of the CSI data collected likely differs to other solutions. A comparison of common and modern approaches to CSI collection is needed, to ensure the trade-offs between hardware options are clear.

4.2 CSIKit

In CSI research, many developers reinvent the wheel. Published CSI sensing applications regularly contain re-implementations of basic parsing and processing functions provided with standard CSI tools. Researchers may use any of a wide variety of tech stacks and programming languages, but they are limited by the support provided by their chosen CSI tool. Most public CSI tools use a packed binary format and provide MATLAB parsing code. The extent to which toolkit-provided code supports various stages of parsing and processing varies. For example, the Linux 802.11n CSI Tool is provided with extensive MATLAB code for both parsing and processing CSI data from the IWL5300. Conversely, Nexmon CSI is provided with a single MATLAB script for parsing. As such, many researchers rely on third-party support for parsing and processing CSI data from their devices. The research tasks involved are largely analogous to modern data science, however experience in both RF engineering and digital signal processing is effectively required. Despite the expertise data scientists can offer the CSI sensing field, there has been a significant lack of CSI software resources tailored to them.

To remedy this we developed CSIKit: a Python parsing, processing, and visualisation library. It allows researchers to easily use CSI data from COTS devices with popular Python libraries such as Numpy, Scipy, and Tensorflow. Originally, CSIKit was developed with the goal of providing up-to-date Python parsers for the Linux 802.11n CSI Tool and Nexmon CSI formats. As of writing, CSIKit now supports CSI data formats from all popular COTS CSI tools, including the Linux 802.11n CSI Tool, Atheros CSI Tool, Nexmon CSI, ESP32 CSI Tool, and PicoScenes. Tests are included alongside these implementations to ensure parsed data is accurate within 64-bit precision. As the project progressed, CSIKit has become a research toolkit for the development of CSI sensing applications supporting all popular COTS CSI hardware. More complex processing functionality has also been implemented, such as denoising with Discrete Wavelet Transform-based thresholding.

4.2.1 Data Parsing

CSI data is most commonly stored in binary formats. This is because the Network Interface Cards (NICs) themselves generate the CSI data through firmware, and the lowest level representation available is a packed C structure. As such, the typical approach to interpreting this data is through a C program. The layout of the structure can be replicated in C syntax and the program can iterate through chunks from the binary file to parse each frame of CSI data. CSI toolkit providers include MATLAB code (with compiled C extensions) which allow users to parse the data within a MATLAB workspace. This is useful for researchers familiar with MATLAB, however many data science students do not have relevant experience. MATLAB is also not supported to run directly on systems like the Raspberry Pi, which limits the deployability of potential sensing applications.

A small number of third-party open source solutions exist for parsing CSI data from toolkits like the Linux 802.11n CSI Tool and the Atheros CSI tool. However these solutions are toolkit-specific, and typically require manual parameter configuration. The latter is a significant issue. This means files must be labelled with the configuration used with the CSI toolkit. When parsing an arbitrary CSI file, the size and formats of the underlying C structures can differ depending on the chosen bandwidth, number of antennas, use of optional features, and more. CSIKit automatically identifies the configuration used, which allows it to parse any given file. An example of this automated detection can be observed the top half of Figure 4.3. The example CSI file provided with the Linux 802.11n CSI Tool is provided to the CSIKit command line interface, which generates a readout of information on the file. No other CSI parsing utility has offered this functionality before CSIKit.

In developing CSIKit, we published open source parsing implementations for all of the formats used by publicly available COTS CSI collection toolkits. Each parser is Pythonically implemented with no external C-based solution, ensuring the code is clean and readable. The formats currently supported are:

- **Linux 802.11n CSI Tool** (IWL5300)
- **Atheros CSI Tool** (AR9XXX series)

- **Nexmon CSI** (Raspberry Pi 3B+/4, Nexus 5/6P, ASUS RT-AC86U)
- **ESP32 CSI Tool** (ESP32-series microcontrollers)
- **PicoScenes** (IWL5300, Intel AX2XX, AR9XXX, USRP SDRs, HackRF One)
- **Custom CSV formats** (Open, used with our prototypes)

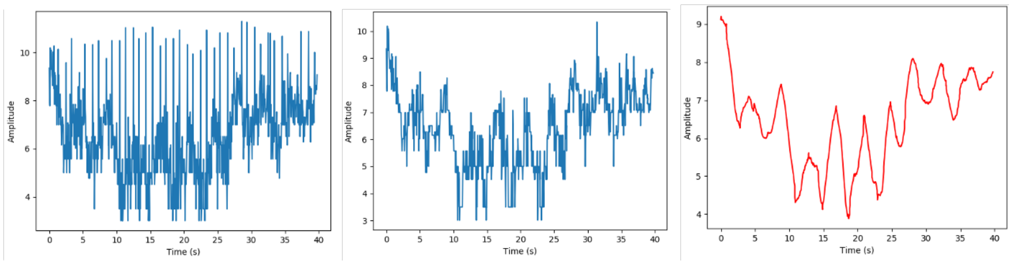
When parsed with CSIKit, any of these formats can be read into a centralised `CSIData` Python class instance. This class structure is uniform, containing metadata and the list of CSI frames. Each frame contains both its metadata and CSI matrix. Numpy matrices are used for CSI, which ensures both compatibility with Python frameworks like Tensorflow, Keras, and PyTorch, and support for exporting to other serialised formats.

The method for parsing any supported file with CSIKit is fully homogeneous, ensuring no stage of the CSI pipeline is hardware-specific. Following the example provided in the documentation, users can prepare a matrix from any CSI file with the same 3 lines.

4.2.2 Processing

Raw CSI matrices of complex pairs are almost never used on their own. Each complex pair represents gain and phase measured for a given subcarrier. Sensing applications make use of gain (amplitude), phase, or both for their CSI input. The gain is calculated as the magnitude of the complex vector, while phase is calculated as the angle of the vector. CSIKit provides a `get_CSI` function which can generate a coherent CSI matrix from a `CSIData` instance. This returns a numpy matrix for the entire CSI time-stream.

This centralised structure allows for optimised processing pipelines to be developed for CSI data from all supported COTS devices. CSIKit offers a focused range of signal filters for preprocessing CSI data observed in literature. This includes statistical methods such as running mean and hampel filters, and finite impulse response (FIR) passband filters. These methods are common in established literature and form the basis by which most CSI data is preprocessed for sensing applications and more. Examples for using CSIKit's built in filters for preprocessing are included in the README, ensuring new and experienced researchers can use and expand upon these techniques.



(a) Raw (b) Hampel filter (c) Hampel + Running mean filter

Figure 4.2: Example of CSI subcarrier plot using CSIKit, showing the impact of hampel and running mean filtering.

4.2.3 Visualisation

CSIKit offers matplotlib-based plotting functionality for `CSIData` structures. The most basic plotting option provided is a 1-dimensional line plot for subcarrier streams. A single subcarrier plot is useful for comparing the impact of filtering techniques. In Figure 4.2, a 40 second CSI subcarrier stream can be observed at various stages of the filtering process described in the CSIKit readme. The raw amplitude in 4.2(a) exhibits both significant high frequency noise and digital noise. In 4.2(b) a hampel filter has been applied to remove the high frequency noise. As a result, the underlying signal can be resolved however digital noise still remains. In 4.2(c), the hampel filter is combined with a running mean filter to smooth out the digital noise. A signal with a clear trend can be observed, which is useful in various sensing applications.

Visualising CSI data can be difficult, as it is formed of multiple subcarrier signals. While a 1-dimensional line plot of a single subcarrier shows the overall trends in the signal, it does not scale well with the many subcarriers provided in even the most basic CSI data. A common method to visualise all subcarriers is a heatmap, with amplitude on the Z axis. This allows us to observe correlated variations across the entire wireless channel measurement. In Figure 4.3, the command line interface for CSIKit is used to identify a file produced with the Linux 802.11n CSI Tool. A summary readout is then produced to establish its CSI data content and various relevant metadata stored alongside. The graph option is then used to render a heatmap of the amplitude of the CSI data from the first spatial stream. This example is from a file provided with the Linux 802.11n CSI

Tool which does not contain valid timestamps. As such, the X axis is rendered using the CSI frame index as opposed to a time.

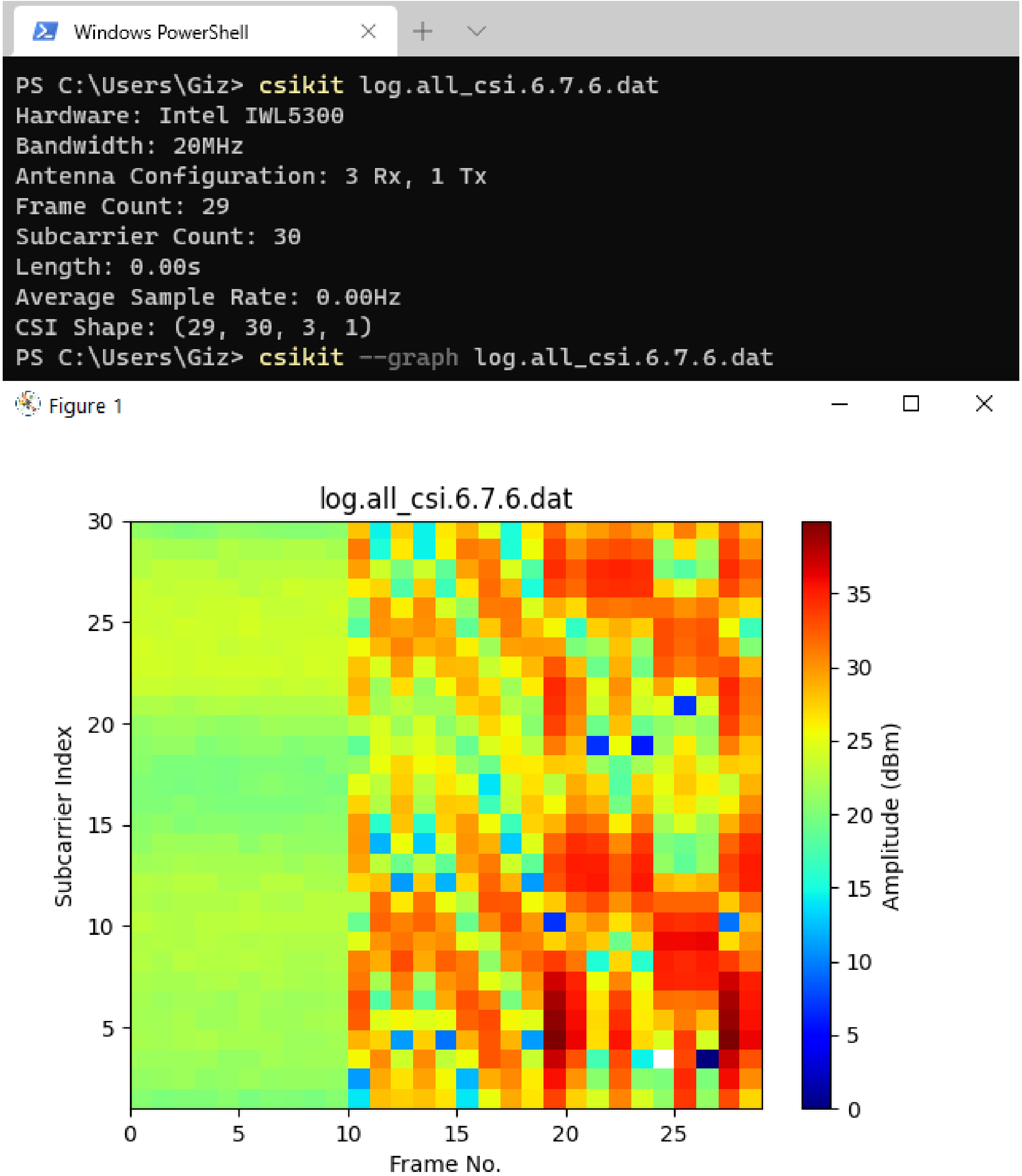


Figure 4.3: Demonstration of CSIKit's command line interface.

Previously, these visualisation operations would require a user to use the MATLAB parser within their own script to generate this overview and plot a heatmap figure. Alternatively, the user could implement their own parser and the relevant tools to verify its accuracy against the MATLAB implementation. The CSI hardware and software configurations used to generate each file must also be known for the parsing process. These steps would also have to be repeated for each unique hardware platform the researcher planned on using. With CSIKit, regardless of the COTS hardware platform, this can be achieved with either no code through the command line interface, or fewer than 10 lines of Python.

4.2.4 Impact

To date, CSIKit has been downloaded over 40,000 times since its first release in September 2020. The repository has garnered over 100 stars and 20 forks on GitHub. Throughout its development, other developers have contributed various quality-of-life features such as variations of the built-in parsers, and formats for data re-encoding. International researchers have also reached out through email and the GitHub issue tracker to report bugs and request assistance and inspiration for their CSI research. Our publication through which CSIKit was first introduced demonstrates CSI activity recognition using the Raspberry Pi 4 and has 8 citations. A recent publication from TU Chemnitz refers to CSIKit as a useful alternative to MATLAB-based offline processing (Kindt, Turetta, Demrozi, Masrur, Pravadelli & Chakraborty 2022).

CSIKit provides a unique opportunity to consider device-agnostic CSI sensing applications. The majority of CSI data formats are built of frames, with a CSI matrix and corresponding metadata. Within CSIKit, this data is stored in a CSIData object which shares common attributes and functions regardless of the data source. As such, CSIKit can be used with valid files containing CSI data with no reference to a specific hardware platform. This aids new researchers, who may be unaware of the specific contents of a file they are working with, and researchers working with diverse datasets spanning multiple device platforms.

The work in Section 6 demonstrates the first hardware-agnostic CSI sensing application, designed and developed with CSIKit. The remaining works in this thesis were all delivered using CSIKit.

4.3 Estimating pre-AGC CSI with RSSI

Several COTS WiFi NICs can be used to collect CSI for sensing applications. As solutions are developed for collecting CSI on more hardware platforms, CSI continues to become a more approachable field of research. This also raises the question of which hardware performs best in device-free wireless sensing applications. The most popular device in current COTS CSI research are the Intel IWL5300, closely followed by the Atheros QCA9300 series of cards. Developers have made modified device drivers for these NICs available, which facilitate the collection of CSI and other useful metadata for WiFi frames. As Peripheral Component Interconnect Express (PCI-E) cards, these NICs must be connected to an x86 system such as a laptop or Next unit of Computing (NUC) which require considerable physical space, power, and cooling. Some low-power hardware such as the Raspberry Pi 3B+/4 and ESP32 have recently gained CSI collection capabilities through either third party modification or support directly from the manufacturer. These devices are considerably smaller, lower-cost, and have a much smaller power footprint. These factors can have a strong effect on the viability of deploying these sensors at scale. However the metadata supplied with their CSI measurements is less complete than that of the PCIe based NICs. A key missing variable is the amount of gain applied to incoming signals through the Automatic Gain Control (AGC) circuit. This process also rescales the CSI data measured for the frame, which effectively destroys fine-grained information encoded in the CSI data stream. As many sensing applications benefit from this information, estimating or recovering the original signal is an important stage of data processing.

Both the Intel and Atheros-based solutions have a defined process for reversing the gain applied to the incoming signal, owing to support from the manufacturer. As more devices gain support for CSI collection, there is no guarantee this manufacturer support will be available for a given device, nor that a third party could reverse engineer the process. Instead a hardware-agnostic method for estimating the original CSI values is required. In their 2020 work, Gao (2019) proposed an approach using separately-acquired RSS to rescale frames and estimate pre-AGC CSI. Their approach relied on the statement "Given that the sum of CSI squared over all the subcarriers should be consistent with RSS...". However, their comparison of the CSI before and after their rescaling is domain-specific which masks the extent to which their approach estimated pre-AGC CSI. It is not clearly presented that this relationship exists, and we wished to further investigate the performance of this rescaling solution. By showing their approach estimates pre-AGC CSI with sufficient accuracy, we can demonstrate the viability of CSI from newer COTS devices.

The works described in this section are as follows:

- First, **we demonstrate the relationship between pre-AGC CSI and RSS**, which underpins the RSS-rescaling process. This relationship can then be exploited to rescale CSI from generic COTS devices to estimate the pre-AGC CSI.
- Second, **we compare rescaling accuracy across popular COTS CSI devices**, by measuring the improvement in correlation against a control with no AGC.
- Finally, **we provide an open source implementation of pre-AGC CSI estimation using RSS**, along with the code and data used in the experiments to produce our figures.

4.3.1 Automatic Gain Control

Automatic Gain Control (AGC) circuitry is used in RF amplification chains to ensure output amplitude stays within a suitable range, despite variability in input amplitude. In wireless transmissions, this process is important as the RSS at the receiver will vary with distance and orientation. When RSS is low, the AGC circuit will increase the antenna gain to improve the Signal-to-Noise Ratio (SNR). If the RSS is high, such as when the transmitter and receiver are colocated, the AGC circuit will attenuate the incoming signal to avoid signal degradation or clipping. This transformation occurs early in the RF chain, as gain variability must be addressed before the Analogue to Digital Converter (ADC). As a result, gain is adjusted for the entire transmission. For 802.11 OFDM, this includes the preamble containing CSI.

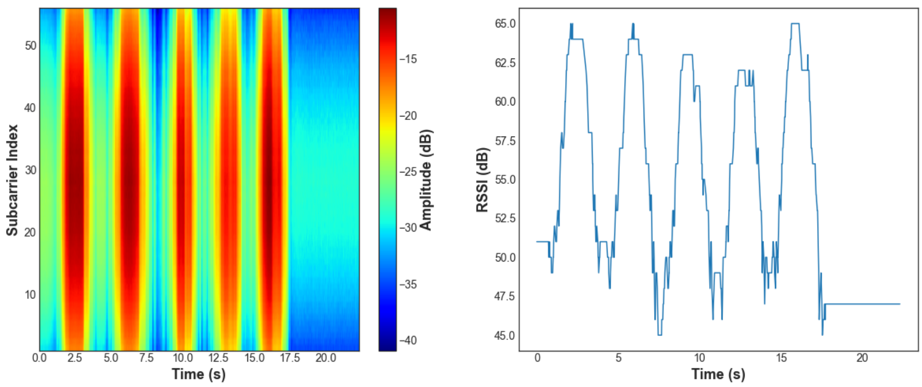


Figure 4.4: Received CSI amplitude and RSSI as the transmitter moves towards and away from the USRP B210

To understand the impact of AGC on CSI we can observe a transparent RF chain. By using an 802.11 transceiver implementation on a USRP B210 we ensure no AGC is employed, and generate an unmodified CSI stream. To demonstrate the behaviour of pre-AGC CSI, a transmitter producing 802.11 traffic is moved towards and then away from the receiver in a repeating pattern. Figure 4.4 shows the CSI amplitude during this motion, showing a sharp increase in amplitude as the transmitter is moved close to the receiver. There is also significant range of observed amplitude, peaking at -10dB. This pattern can also be observed in the RSS, similarly peaking when the transmitter is closest to the receiver.

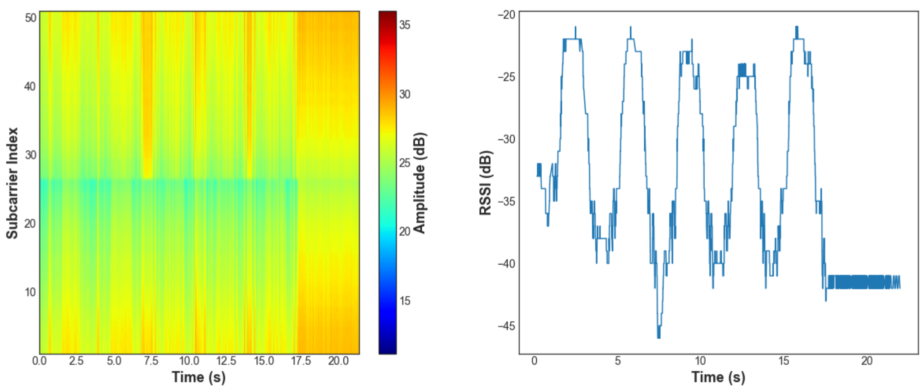


Figure 4.5: Received CSI amplitude and RSSI as the transmitter moves towards and away from the ESP32

However the same pattern is not present in CSI as-collected from COTS devices, due to the gain applied to incoming signals through AGC. Figure 4.5 shows CSI from the same antenna and time period captured alongside 4.4. A similar RSS pattern can be observed, however the range in observed CSI amplitude is very small. These gain modifications effectively mask data encoded in the CSI. Some CSI-based sensing applications rely on this behaviour in CSI amplitude, which may be linked to poor performance when applied to COTS CSI data (Gao, Gao, Wang, Li, Xu & Jiang 2019). While the CSI does not follow the pattern observed in Figure 4.4, the RSS curve in Figure 4.5 does. If RSS does represent the pre-AGC gain curve, it could be used to rescale CSI data to pre-AGC values. To confirm this, we must demonstrate there is a linear relationship between CSI and RSS.

4.3.2 Relationship between RSS and CSI

As reported by the NIC, both CSI and RSSI are measured for the preamble of a given OFDM frame j. In this section, RSS and RSSI will be used interchangeably. Both measurements use a logarithmic scale and represent the received signal strength. RSS can be mapped directly to a specific power level in mW. RSSI has no standard and is an internally scaled to a manufacturer's implementation. While a given manufacturer's implementation may result in an RSSI measurement which is also valid RSS, a generic approach to interpreting RSSI must assume this is not the case. Most NICs will provide an RSSI reading from which RSS will have to be derived. The transformation between RSSI and RSS is not always available, however this hypothesis does rely on the assumption RSSI is collected either without AGC employed, or before AGC is applied. For this work, we use RSSI to demonstrate the relationship between signal strength and CSI.

CSI is represented by a vector of complex values, containing gain and phase measured for each OFDM subcarrier. RSSI is an collective measurement of the total gain observed across all subcarriers. To compare, we must reduce the CSI vector to a single value by summing the squares. This approach ensures an invariant measurement of summative CSI magnitude, which can be compared to establish a relationship between CSI and RSSI. To scale this value to a comparable domain for RSSI, the square root of the result is divided by the number of subcarriers. If the number of available subcarriers is S, the average subcarrier magnitude for CSI vector H is defined as:

$$\overline{H} = \frac{\sqrt{\sum_{n=1}^{S} |h_n|^2}}{S} \tag{4.4}$$

We can then observe $\overline{H}$ for an active range of RSSI values R. For the aforementioned reasons in 4.3.1, data captured from the USRP B210 is used.

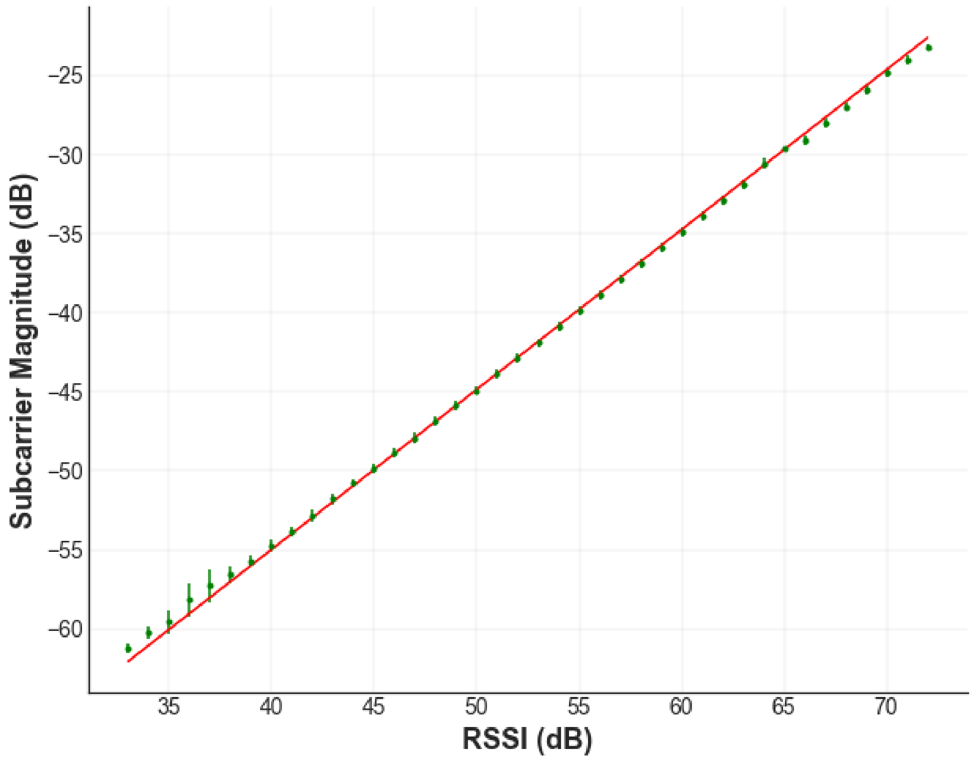


Figure 4.6: Relationship between pre-AGC CSI (amplitude per subcarrier) and RSSI, measured with a USRP B210

Figure 4.6 shows $\overline{H_j}$ plotted against R_j for k frames. The error bars represent +/- a standard deviation. R increases alongside $\overline{H}$, indicating a linear relationship between both variables. At the low and high ends of the RSSI scale, some inconsistency can be observed. However 94% of the data falls within a standard deviation. A line of best fit was chosen to minimise the squared error, defined as E. Numpy's polyfit function was used.

$$E = \sum_{j=0}^{k} |p(R_j) - \overline{H}_j|^2 \tag{4.5}$$

This approach yields line 4.6, whose y-intercept c remains generally consistent for a given hardware platform.

$$\overline{H}_j = mR_j + c \tag{4.6}$$

As seen in Figures 4.4 and 4.5, the scales for the CSI and RSSI magnitudes differ across unique hardware platforms. While the relationship between RSSI and CSI is not present in the ESP32 data as-collected, a similar RSSI curve can be observed for both the USRP and ESP32 samples. This indicates that the RSSI collected by the ESP32 is measured pre-AGC, and may represent the relative scale of expected CSI magnitude. Next, we consider the standard approach for AGC mitigation on the most common COTS NIC used for CSI research. This will allow us to establish whether the relationship between RSSI and pre-AGC CSI can be used to rescale post-AGC CSI.

4.3.3 COTS CSI Limitations

The Intel IWL5300 wireless card supports CSI extraction via a modified firmware (Halperin, Hu, Sheth & Wetherall 2011b). As this firmware was built with access to the proprietary source code, all transformations to the captured data can be accounted for. In the IWL5300 the RSSI represents the signal strength after AGC has been applied. The original RSS is obtained by removing both a magic constant and the `agc` value provided with IWL5300 CSI measurements. In summary, Halperin highlighted the following factors which needed to be considered when interpreting IWL5300 CSI to account for various undesired transformations:

- **CSI/RSSI factor**: Calculated as the ratio of RSSI power to subcarrier magnitude, to account for AGC
- **Thermal noise**: Noise floor incurred by background entropy
- **Quantization error**: Caused by the disparity between the 8bit complex resolution for CSI and the 6bit ADC in the NIC

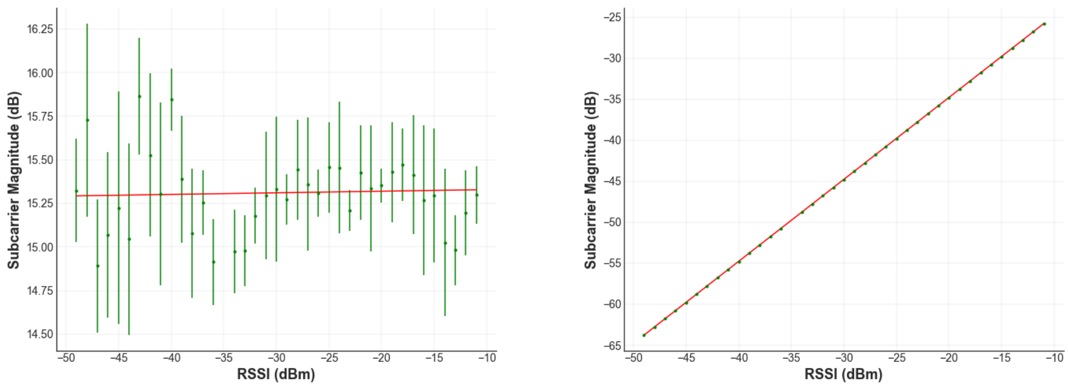


Figure 4.7: IWL5300 CSI/RSSI relationship plotted before and after applying Halperin's rescaling method

Figure 4.7 shows Halperin's rescaling method accurately reintroduces the expected ratio of pre-AGC CSI to RSSI. All points fall along the line of best fit. This seems to indicate this approach to rescaling CSI calculates an estimated pre-AGC value. As the IWL5300 offers relatively accurate channel measurements with MIMO, it stands as the most popular NIC for COTS CSI collection and research.

In an ideal scenario, this accurate rescaling process would be possible with all COTS devices capable of exporting CSI data. Unfortunately, full transparency from COTS NICs cannot be expected. For instance, the manufacturer may add support for extracting CSI but not provide details of the RF chain. This is usually to protect intellectual property or sensitive info. In other cases, COTS CSI tends to be gathered through third-party methods such as modified device drivers (Xie, Li & Li 2015), or firmware flashpatching (Gringoli, Schulz, Link & Hollick 2019). Reverse-engineered methods require far more effort to retrieve details of the RF chain. For instance, despite "nexmon_csi" releasing in 2019, a preliminary solution for isolating elements of the RF mixing pieline wasn't released until late 2021[1]. It is unreasonable to expect that every first or third-party solution will be able to provide this information. As more options become available to use COTS NICs for CSI collection, applications for sensing and otherwise will benefit from a method for estimating the pre-AGC CSI with common metadata. As the CSI/RSSI ratio underpins Halperin's approach, it is clear this relationship is present in pre-AGC CSI and manually

[1] `https://github.com/T3r0/nexmon_csi_gain`

reintroducing it through rescaling appears to recover the pre-AGC CSI curve.

4.3.4 Estimating H with RSSI

Gathering low level details on implementation for generic NICs can be difficult, and in some cases impossible. In (Bardwell 2002), practical examples for RSSI conversions to RSS for a few manufacturers are provided. For instance, Atheros chipsets use a formula for the conversion, by which the percentage is converted into RSSI and a magic constant of 95 is subtracted. These transformations depend entirely on manufacturer implementation. For instance, in some cases RSSI may be represented with a percentage. In a common scenario for generic NICs, RSSI values are obtained pre-AGC and the manufacturer's magic constant is unknown. While obtaining the magic constant would be preferred, not accounting for it renders subcarrier magnitude to an arbitrary scale. This is not an issue as the scale is constant per the hardware configuration, and CSI sensing applications (using amplitude) rely on underlying trends as opposed to absolute values. RSSI can only be used to rescale CSI if it can be accessed while remaining a logarithmic representation of signal strength.

Gao et. al discusses this area as CSI Calibration in their work focussing on the use of CSI fingerprinting for localization (2019). By relying on the relationship between CSI and RSS, they proposed equation 4.7 for calculating a scaling factor for rescaling CSI to pre-AGC values.

$$s = \sqrt{\frac{10^{RSS/10}}{\sum CSI_i^2}} \tag{4.7}$$

Where s is the resultant scaling factor, RSS is the RSS measured in dBm, CSI is the vector of CSI subcarrier amplitudes for a given frame i. At the time of CRISLoc's publication, nexmon did not natively capture RSSI measurements for incoming CSI frames and so RSS measurements were obtained separately. This approach hinges on the assumption that "the sum of CSI squared over all the subcarriers should be consistent with RSS". Their application functioned well and represented the state of the art at the time, however their demonstration of the efficacy of this method is domain-specific. The benefits to this approach for fingerprinting tasks are obvious, however it is unclear how this might apply to other applications. To demonstrate the generic efficacy of this approach, the estimated pre-AGC CSI should be compared to that of a control employing no AGC.

4.3.5 Experiment

To effectively demonstrate Gao's proposed approach to RSSI rescaling for pre-AGC CSI estimation is a viable generic approach, the resultant estimated CSI should be compared to that of a device which does not employ CSI. For this task, a USRP B210 Software-defined Radio (SDR) was used with the PicoScenes SDR backend (Jiang, Luan, Ren, Lv, Hao, Wang, Zhao, Xi, Xu & Li 2022). This ensures a control with a transparent RF chain, whose RSSI and CSI were consistent as demonstrated in Figure 4.6. Through this work, the goal is to ascertain whether the pre-AGC CSI estimations provided through RSSI rescaling represent the original CSI curve, and to measure the accuracy of this approach.

These experiments aim to both visualise and quantify the improvement in representing the pre-AGC curve in COTS CSI data as a result of rescaling with RSSI. To do this, a range of COTS devices are configured alongside the B210 control. An ESP32 with an onboard antenna was used to generate 802.11n frames via injection, ensuring the device could be mobile during the captures. This allows for the strength of the incoming signal to be easily modulated by the moving the ESP32 injector closer to and further away from the receiving antenna. By connecting all devices to a homogeneous antenna configuration, the same input signal can be guaranteed ensuring the observed differences in the CSI data will be entirely due to difference in the RF chains.

Experimental Setup

To ensure AGC is employed by the COTS devices throughout the experiment, the following process was used to generate samples for capture. Each capture ran for 25 seconds. Before the capture begins, the injector is placed on the desk 50cm away from the receiving antenna. At the start of the capture, the injector is moved against the receiving antenna, and then back to its starting place. This process is performed 5 times, before the injector comes to rest at the 50cm marker. This was designed to produce 5 distinct peaks in RSSI, which ensures AGC (if present) will be employed. 10 captures were gathered from each device.

Hardware Configuration

To provide a homogeneous antenna configuration, an LPD410 4-way signal divider was used. This ensures the same input signal can be shared across each device, with each device simultaneously connected to the divider via SMA or u.fl. By doing so, the observable differences in the collected CSI will be incurred by RF chain variations rather than the antenna configuration.

4 CSI collection devices are used in this experiment. The captures were orchestrated using SSH and the specific toolkits listed for each device.

- **USRP B210**: Control. Operated using PicoScenes (Jiang et al. 2022).
- **ESP32 (WROOM U.FL)**: Lowest-cost CSI collection solution. Operated using ESP32 CSI Toolkit. (Hernandez & Bulut 2020).
- **Raspberry Pi Compute Module 4**: Modern and accessible hardware. Operated with Mzakharo's fork[2] of nexmon_csi supporting RSSI collection (Gringoli et al. 2019).
- **Intel IWL5300**: Most common COTS CSI solution. Operated using Linux 802.11n CSI Tool (Halperin et al. 2011b).

An ESP32-WROOM-32D was used for frame injection to generate traffic for CSI collection. 802.11n beacon frames were generated at 50Hz. This device was moved by hand during the experiments in a smooth repeatable motion along a 50cm track.

Evaluation Metric

The main challenge in measuring similarity between neighbouring streams of CSI is synchronisation. As each NIC is operated by independent system, direct synchronisation to a suitable standard is not possible. While the ESP32 does run a realtime operating system (FreeRTOS), the onboard crystal oscillator cannot be easily linked with the other NICs used for these experiments. On average, any one of the devices could be up to 1s delayed from another.

[2] `https://github.com/seemoo-lab/nexmon_csi/pull/256`

To combat this, the periodogram of the mean across all subcarriers is used as our input. By converting the resultant signal to the frequency domain, the impact of desynchronisation in the time domain can be minimised. The improvement in correlation after applying rescaling is then used as the evaluation metric. This ensures that devices with weak synchronisation with the control are not unfairly penalised for their performance, and the result of rescaling is the primary point of measurement.

Open Implementation

An open source implementation of RSSI-based pre-AGC CSI estimation is provided as a keyword parameter (scaled=True) for parsing files with CSIKit (Forbes, Massie & Craw 2020a). Additionally the code to produce all figures in this work has been released [3].

[3] `https://github.com/Gi-z/AGC-Reversal-Demo`

4.3.6 Results

Comparison after Rescaling

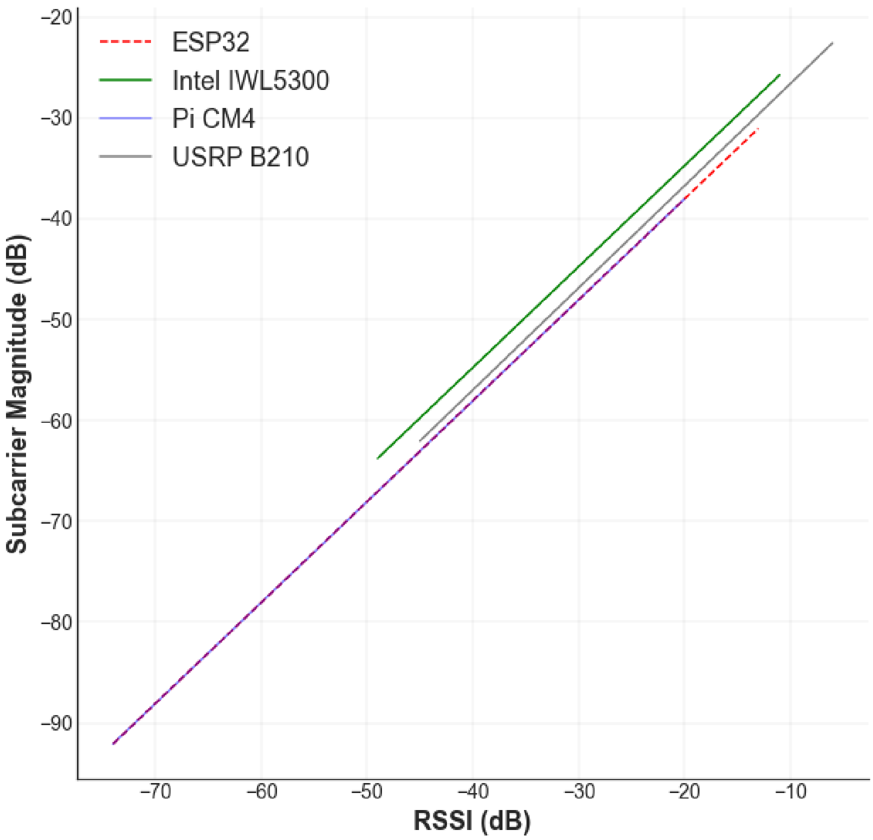


Figure 4.8: Comparison of the Subcarrier Magnitude/RSSI relationship from each device.

Figure 4.8 shows the relationship between RSSI and the sum of the squares of CSI after the available rescaling method was applied, for each device used in the experiments. Each line of best fit shows this relationship can be generally reintroduced to the CSI data through RSSI-based rescaling. It can also be noted that the y-intercept is different for each device.

As previously observed in Figure 4.6 the USRP B210 demonstrates a largely consistent relationship between both variables. The USRP data shows an imperfect but mostly linear relationship, most likely due to non-linear aspects in the RF chain. Inconsistency can be observed in USRP subcarrier magnitude between 30-40dB, however to a much less significant extent. At the lower end, it appears these inconsistencies appear to be more pronounced. RSSI-based rescaling is inherently limited by the accuracy and consistency

Table 4.1: CSI amplitude features before and after applying rescaling (in dB).

	Before			**After**		
	Max	**Min**	**Range**	**Max**	**Min**	**Range**
USRP B210	-13.11	-36.35	23.25	-13.11	-36.35	23.25
ESP32	33.5	15.99	17.51	-22.95	-58.15	35.2
Pi CM4	57.78	53.38	4.39	-42.94	-76.82	33.88
Intel	30.49	26.14	4.36	-15.82	-52.76	36.94

of RSSI measurements. Section 4.3.6 inspects features of the RSSI measurements from each device.

In the heatmaps in Figure 4.9, we can observe the CSI curve across all subcarriers for 4 devices. The control, USRP B210, shows no change after RSSI-based rescaling since no AGC was applied in its RF chain. A clear banding pattern is present in the control data, aligning with the movements of the transmitter towards and away from the receiver. As the transmitter was moved closer to the receiver, both RSSI and the CSI amplitude increased. This banding pattern is not present on the Unscaled data for the COTS devices. Traces of the banding can be observed when viewing the data very closely, but the pattern has largely been smoothed out in the CSI as-collected from the COTS devices.

When rescaled with its reference implementation, the Intel system demonstrates the same banding pattern as the control. Notably, its rescaled amplitude is closest to that of the control, than the other COTS devices. Table 4.1 shows features of CSI amplitude for each of the devices. The maximum values observed after rescaling for the Intel and USRP are within 3dB of each other, closer than the other COTS devices. The minimum values observed after rescaling for each of the COTS devices is lower than that of the control, likely owing to spikes in RSSI which are not observed on the USRP B210 (see Figure 4.10). This demonstrates the accuracy of the reference rescaling implementation provided by Halperin. When the RSSI-based rescaling implementation is applied, the banding pattern can be observed to be reintroduced in both the ESP32 and Pi CM4. Section 4.3.6 aims to measure its performance.

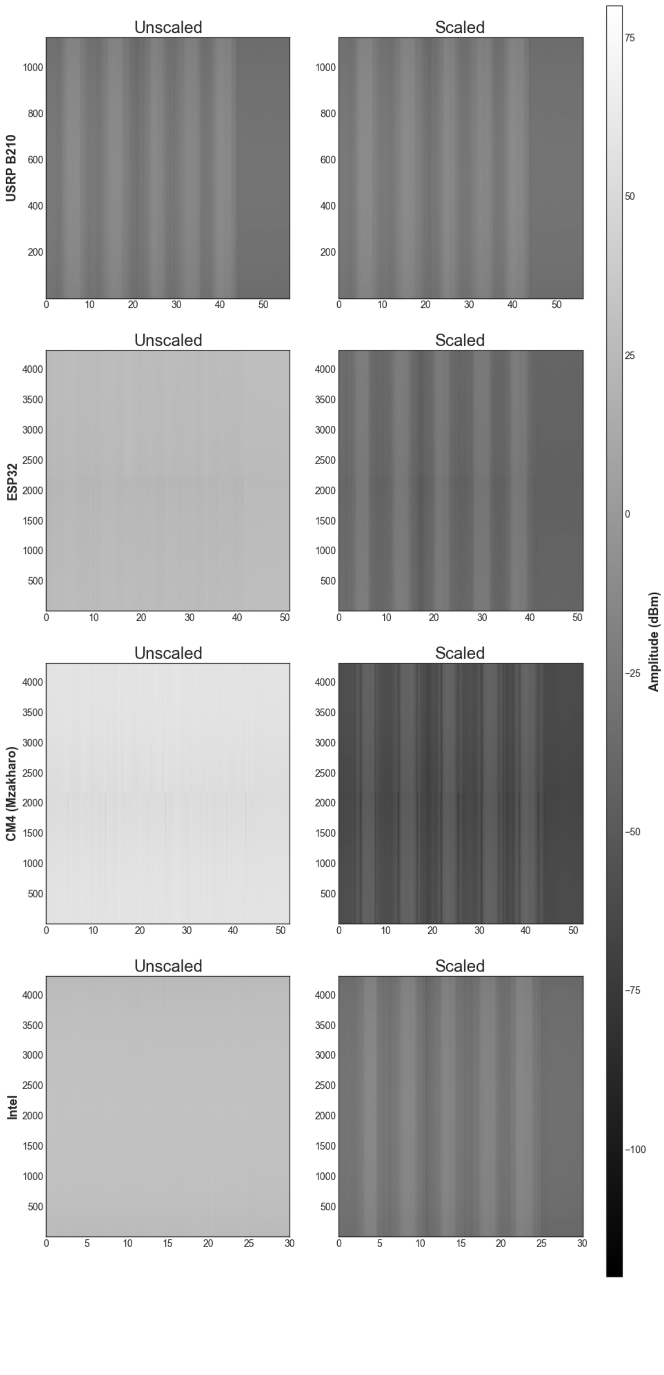


Figure 4.9: Comparison of Heatmaps from all devices.

RSSI Accuracy

Figure 4.10 shows RSSI measurements across each device for an experimental capture. The data shows the RSSI captured with the USRP B210 is the most stable.

Each of the COTS devices exhibits different behaviours which deviate from that of the USRP. The COTS devices show additional noise during the stationary period at the end where stable RSSI is observed on the USRP. The most significant noise at rest can be observed for the Intel system, where this instability fluctuates between -38dB to -32dB. Halperin's method appears to account for this by also considering the noise floor and quantisation error, which cannot not be relied upon for generic COTS devices.

During the period where the transmitter was mobile, the CM4 produced several readings close to 0dB which appear to be invalid. This behaviour was found to be present in each of the captures on the CM4. RSS, and so RSSI, should follow a gaussian distribution (Yadav 2018). Gaussian filtering methods should be investigated to account for the instability observed for the COTS RSSI measurements.

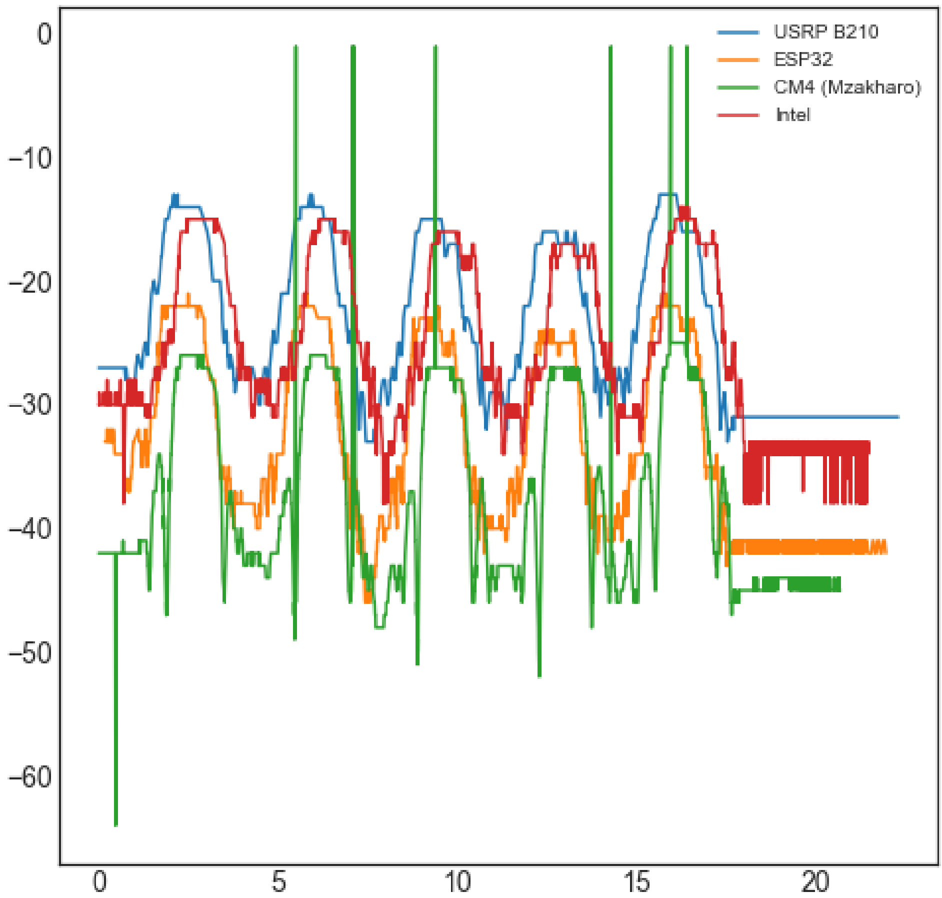


Figure 4.10: Comparison of RSSI measurements from all devices.

Figure 4.8 also seems to indicate the RSSI retrieved from the CM4 behaves differently to that of the other devices. It may be the case that the RSSI retrieved using this version of nexmon is retrieved after the first stage of amplification. The RSSI accuracy for the standard operating range of the Pi CM4's wireless chipset is listed as +/- 5dBm (Cyp 2019). Page 5 of this document details the amplification chain, which is currently being reverse engineered[4]. As this cannot be expected for generic COTS CSI collection solutions, a heuristic is needed for determining whether RSSI is representative of the pre-AGC RSS. A possible approach may be to use an FFT to identify high frequency noise in the RSSI.

[4]`https://github.com/T3r0/nexmon_csi_gain`

Measuring Improvement

Figure 4.11 shows the improvement in correlation of the average CSI after applying the available rescaling method, measured for a given device against the control. For the ESP32 and Pi CM4, this is the RSSI-based rescaling method, while the Intel system uses Halperin's method. An improvement in correlation was observed for each experimental capture, as a result of rescaling.

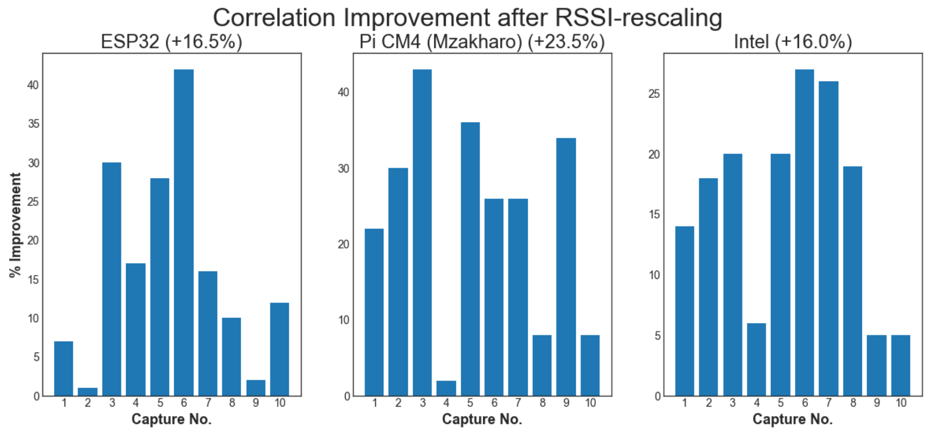


Figure 4.11: Improvement in Correlation after RSSI-rescaling.

Table 4.2: Average correlation through RSSI-rescaling, by device.

	Before	**After**
ESP32	.46	.69
Pi CM4	.37	.54
Intel	.69	.78

With the ESP32, an average improvement in correlation of 16.5% was observed. Initial correlation against the control for the ESP32 is higher than the CM4. In some cases, correlation improvements were insignificant (less than 5%). Greater overall correlation is observed for the ESP32 than Pi CM4 after rescaling.

As observed in Figure 4.9, the Pi CM4 shows the largest difference in overall scale both before and after applying this rescaling method. As such, a smaller change to the CM4's correlation against the control can yield a stronger improvement than other the other devices. We observe a larger average improvement in correlation for the Pi CM4.

Halperin's method on the Intel hardware shows the most consistent improvement in correlation. This is to be expected, as this method is hardware-specific and takes several proprietary measures into consideration. Table 4.2 shows the initial correlation is already equal to or higher than can be achieved with the other devices, indicating the IWL5300 generates CSI measurements which are closest to that of the control regardless of AGC.

4.3.7 Conclusions

The experiments performed in this work demonstrate RSSI can be used to rescale CSI to estimate the pre-AGC curve. While the performance of this method varies, an improvement in correlation against the control was achieved for every experimental capture. These tests also demonstrated the importance of accurate RSSI measurements, which must be taken pre-AGC to re-encode this curve into the unscaled CSI signal. Despite its smaller power envelope, physical size, and hardware cost, the ESP32 has been demonstrated to perform comparably to other COTS CSI collection devices and show an improvement in performance with RSSI-based rescaling. Future work to further improve the viability of CSI data for WiFi sensing may investigate a heuristic for identifying whether RSSI is representative of the pre-AGC received signal strength, through identification of noise in the high frequency components of RSSI.

4.4 Summary

In this chapter we introduced CSI and its use in device-free wireless sensing. To identify the most viable low-cost hardware entry-point for WiFi CSI sensing, we developed CSIKit: a software solution for processing CSI data from all COTS CSI solutions. Through this process we produced a representation for CSI data which supports homegeneous implementations for application using CSI from all COTS CSI solutions. We then extended CSIKit's generic COTS sensing software support by implementing an approach to pre-AGC CSI estimation using RSSI. The approach was then evaluated and we determined the pre-AGC CSI curve could successfully be estimated using RSSI. CSIKit was our base software solution for CSI processing for the remaining work. We now consider our sensing applications developed using CSIKit, which demonstrate its functionality for processing generic CSI data into homogeneous data structures. This will then be used to compare sensing performance across COTS CSI hardware.

Chapter 5

Sensing Applications with Off-the-Shelf CSI Hardware

This chapter details the proposal, implementation, and performance testing process for two applications using COTS CSI hardware for wireless sensing. In the project we investigated many sensing capabilities using CSI data which were observed in literature. We investigated these sensing approaches using modern COTS hardware to validate their efficacy. We then developed successful novel approaches for movement detection and activity recognition which effectively scales with the newer features offered on modern COTS hardware. These build upon the coarse behavioural monitoring capabilities currently offered in the FitHomes, while retaining the ambient nature of our existing sensor complement at low-cost.

Movement detection is the foundational sensing application in FITsense. Our existing PIR-based solution delivers room-level movement detection which feeds into our coarse behavioural monitoring systems. We require sensors with greater fidelity to improve upon the fundamental limitations of the PIR modality. CSI-based sensing allows us to monitor the disturbances in a wireless channel caused by human movement in the environment. By using COTS WiFi hardware for device-free wireless sensing, we provide a improvement in temporal fidelity over the existing motion detection solution in the FitHomes.

One of the key advantages of using CSI for wireless sensing is the option to deploy multiple sensing applications using the same input CSI data from a single sensor. Multiple interpretations are possible with the same CSI matrix due to the complex channel representation contained therein. This approach can allow us to incrementally improve upon the sensing capabilities the CSI sensors can provide in the FitHomes, without replacing the sensor hardware. As we investigated sensing capabilities for high-level behavioural monitoring, we observed a gap in the literature as no activity recognition application had been implemented using CSI data from modern COTS hardware like the Raspberry Pi 4. While activity and gesture recognition works are common in CSI research, these had previously only been delivered using either the Intel 5300 or Atheors AR9XXX chipset series. The newer chipset used on the Raspberry Pi 4 offers increased capabilities, including support for higher bandwidth channels up to 80MHz. This increased bandwidth increases the number of subcarriers in the CSI measurements from 30 to 256. We prepared a deep learning approach to CSI-based activity recognition using the Raspberry Pi 4, to attempt to infer features from the larger feature space provided with the increased bandwidth. This represents a significant shift in the fine-grained behavioural monitoring capabilities of our sensing platform.

In this chapter the following contributions are presented:

- A statistical approach to **motion detection** using CSI data which **adapts to changes in subject distance and movement intensity**.
- The first **activity recognition** application demonstrated using CSI from the **Raspberry Pi 4**, using a deep learning approach with DeepConvLSTM.

5.1 Movement Detection

Various sensor technologies can be employed to monitor movement, which is useful in behavioural monitoring. The choice of technology has a direct impact on the quality of behavioural monitoring, through the richness of the data they collect, their cost, and their perceived intrusiveness to residents. Many homes already contain Passive Infrared motion sensors (PIRs), however these are event-focused and so produce small amounts of data which is insufficient for high quality behavioural insights. Alternative modalities such as wearables and cameras capture richer data with diverse applications, however residents may reject them as they are considered too intrusive for a home environment (Sucerquia, López & Vargas-Bonilla 2017). Radio Frequency (RF)-based solutions, such as (Adib & Katabi 2013), are unintrusive but as presented are too expensive to deploy at a large scale. Selecting a sensor for smart housing currently requires balancing the inherent characteristics of the modality. The ideal solution would be an ambient low cost sensor like a PIR, which could also collect rich data for high quality behavioural insights. Recent developments have allowed for CSI to be collected using the Raspberry Pi 4, one of the world's most popular single board computers (SBC) (Gringoli et al. 2019). This provides a scalable deployment solution for CSI-based applications on a SBC with on-device processing capabilities.

While PIRs are effective at detecting if motion has occurred in a given environment, they do not accurately measure the duration and intensity of the motion taking place. Once activated, PIRs have a reset time of typically a few seconds, which makes accurate measurement of the extent of movement taking place difficult. Most PIRs require line-of-sight which means they are visible to residents. As some can find PIRs relatively intrusive, this can result in residents consciously or unconsciously modifying their behaviour when near PIRs (Moretti, Marsland, Basu, Sen Gupta, Group et al. 2013). A common example of this is residents waving at the sensors, generating additional sensor activation data. CSI-based sensors can potentially eradicate performative behaviour, as they do not require line-of-sight in order to function. The device can easily be hidden behind a surface or wall.

In this work, we establish a scalable and unintrusive solution for motion detection in smart home environments, which can replace existing PIR sensors. Through this we introduce a novel statistical approach for coarse movement detection using CSI, demonstrated with the Raspberry Pi 4.

5.1.1 Implementation

Our system aims to perform movement detection using variation in the correlation of sequential CSI frames which is impacted by human movement. Periods of increased variation can be identified as being caused by the impact of the subject's body movements in the RF environment. By measuring this and applying a statistical classification approach, we can effectively perform both movement and occupancy detection.

This system is designed to work both as a replacement for a PIR, but could also be used to enhance their functionality by working alongside them. A functioning implementation of this system can be built at comparable cost to an off-the-shelf Z-wave connected PIR sensor, but with increased temporal precision and the ability to potentially offer additional health monitoring functionality as the technology progresses.

While a raw threshold-based approach has been employed in other work, it is a static approach and does not proactively adapt to changing environmental conditions. The system proposed in this work adapts to changes in the environment through an automated calibration process. Threshold-based approaches are limited in sensitivity, which can cause movements performed further away from the sensor to produce signals of lower magnitude. We aim to mitigate this by using a proportional threshold which offers fine control over sensitivity. Furthermore, we attempted to ensure that data windows beginning with movement could also be correctly classified, as a typical statistical approach may require initial data to establish a norm.

The system can be split into 6 sections as seen in Figure 5.1. Data is captured using CSI hardware, which can either be passed to the system as a batch, or buffered on a realtime system. The data is then preprocessed to remove noise and inconsistencies, after which the Pearson Correlation Coefficient (PCC) is calculated. Statistical analysis of variance is then performed on the resultant PCC, through which movement can be identified from the input signal. The movement output can then be relayed to the viewer as a binary value with relatively high temporal precision, as opposed to a PIR.

5.1.2 Preprocessing

Using the hardware configuration defined in Section 5.3, CSI amplitude data is captured at approximately 100Hz. This inconsistency is due to the ambient packet loss which can occur in a managed network solution in typical environmental conditions. The data is then linearly resampled to 100Hz. Pilot, guard, and null subcarriers are removed from the signal, from which the lower 20MHz band is extracted. A low pass filter is then applied with a lower frequency bound of 10Hz, to reduce noise from unwanted frequency components in the signal. This output is then downsampled to 10Hz to reduce processing overhead. Negligible accuracy improvements were observed using 100Hz data, and it was

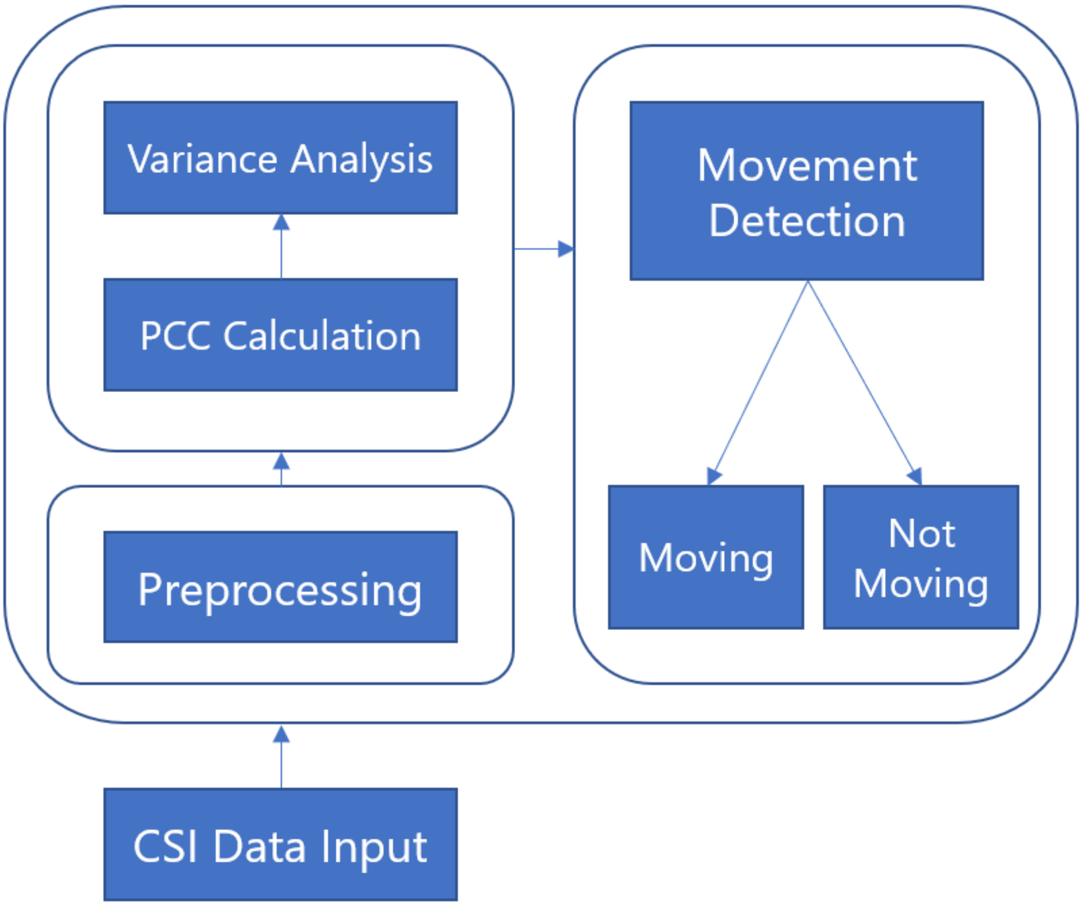


Figure 5.1: System overview diagram.

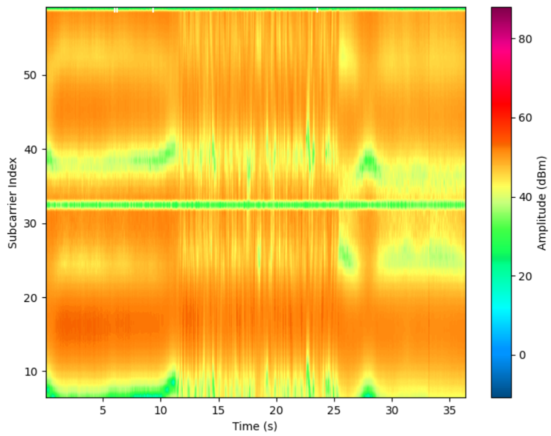
not found to significantly improve precision.

5.1.3 Pearson Correlation Coefficient

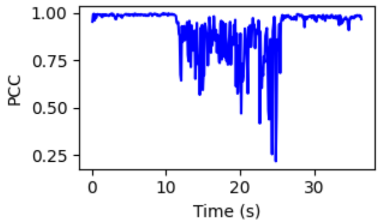
Human movement causes large, rapid changes in CSI, shown in Figure 5.2. As a result, the correlation between adjacent frames of CSI fluctuates. These fluctuations may not always be immediately visible in a heatmap plot, however the underlying pattern in PCC variation tends to be present. We quantify the correlation between the CSI at frames t and $t+1$ through the Pearson Correlation Coefficient.

$$PCC_{t,t+1} = \frac{cov(CSI_t, CSI_{t+1})}{\sigma(CSI_t) \cdot \sigma(CSI_{t+1})} \tag{5.1}$$

Where σ is the square root of the frame's variance and cov is the covariance of the



(a) CSI Amplitude



(b) PCC

Figure 5.2: CSI heatmap example showing a period of movement between 11s and 25s.

two frames. As the PCC is invariant under linear transformations to the data, it allows us to compare the shape of the graphs numerically. Other works have used the Signal Tendency Index (STI) to quantify the changes to the CSI over time and detect movement using a threshold (Yang et al. 2018). STI is a transformation of the PCC between adjacent frames, and our testing found no benefit to using this over the PCC.

$$STI = \sqrt{2n(1 - PCC)} \tag{5.2}$$

As we are using the variance of the PCC for movement detection we do not rely on its robustness, but in future work we could consider investigating Fisher's Z-Transform. As the extent to which PCC varies can differ significantly in real-life motion detection scenarios, we measure the variance of the PCC within two sequential 0.5 second windows, rather than identifying when the PCC has dropped below a fixed threshold. This approach ensures relative reactivity to changes in the location of the movement source, which may be closer or further away from the receiver. For example, given the PCC curve in Figure 5.2(b), one could reasonably assume a threshold could be established for 0.50. However, the same intensity of movement performed at the other side of the room would yield a smaller change in PCC and so not be detected. By measuring the variance between two sequential 0.5 second windows, we can monitor for a consistent relative intensity of movement, as opposed to a fixed value.

Algorithm 1 Sliding Variance Analysis

Input: Running Variance of PCC, thresholds for movement and non-movement
Parameter: Window size
Output: Array of detected movement

```
while i <length(input) do
    X = input[i : i+windowsize]
    Y = input[i+windowsize : i+(windowsize*2)]
    diff = abs(mean(X)−mean(Y))
    if moving then
        output[i] = bool(diff > mov_thresh)
    else
        output[i] = bool(diff < nomov_thresh)
    end if
    moving = output[i]
end while
```

5.1.4 Variance Analysis

This sliding variance analysis algorithm is designed to handle automated signal shape detection on the PCC output. It aims to overcome issues faced when using a threshold-only approach, such as inflexibility to magnitude changes. By using the running variance of the PCC as our input, spikes and short-term changes in the floor can be more robustly classified. Once calibrated, this algorithm can be used to identify periods of movement in varying environments and deployment configurations.

Calibration is needed to ensure the system is capable of identifying whether an input stream contains any movement activity whatsoever. Five data captures are performed for ten seconds each in order to establish normal CSI trends in an empty room. PCC values for this data are then calculated, from which the mean and mean range are extracted. A "containsmovement_threshold" can then be established as the mean minus twice the mean range. The "mov_threshold" is set to 0.15, and the "nomov_threshold" is set to 0.05. The "windowsize" is set to 5.

The "containsmovement_threshold", established to indicate the presence of movement within a data capture, is used to indicate whether variance analysis should be performed. In which case a running variance filter with a window size of 10 samples is applied to the PCC and passed to Algorithm 1. Iterating over each sample in the data, two sequential half second windows are established. The absolute difference between the means of both windows is then calculated. Movement can be identified as periods where the difference between the means of both windows exceeds 15% (based on "mov_threshold") of the max observed value in a given data capture. This proportional approach helps ensure that differences in magnitude of the CSI signal do not have a direct impact on the detection accuracy. Additionally, the use of a separate "nomov_threshold" ensures that consistently lower variance will indicate movement has terminated, as the beginning of movement produces more erratic signals. A binary stream of movement status for each frame is then returned.

5.1.5 Experiment

The objective of this experiment is to demonstrate that the system described in Section 4 can offer movement detection, at least analogous to that of a PIR. CSI and PIR sensors are placed in two environments and data is collected while the resident performs a set of 5 movement-related activities. The output from both sensors is compared to the ground truth timings and a ratio of correctly-classified to incorrectly-classified time is established to score the accuracy of each sensor. Each movement type is repeated 5 times. While the CSI system may be capable of performing movement detection in through-the-wall scenarios, this experiment will establish the capabilities for movement detection within one room.

5.1.6 Environments

Data was captured from two different environments, in order to establish the system can be calibrated to environments with different external factors. These factors may include but are not limited to: interference from other WiFi devices, thickness of walls, and building density. While the impact of the environment and external factors has been identified, it has not yet been independently quantified (Lee, Ahn, Choi, Kim & Lee 2019). This experiment will begin to identify how performance compares between different environments, however additional experiments will be needed to more definitively demonstrate that the system is robust to environmental changes.

Environment 1 is a small, open plan apartment with CSI collection performed in a 4.2 x 3m living room. Environment 2 is a two storey house with CSI collection performed in a home office, approximately 3 x 2.2m. Object clutter is representative in both environments. The "containsmovement_threshold" is calculated through calibration as 0.87 and 0.92 in environments 1 and 2, respectively.

5.1.7 Movement Design

Five movement patterns were chosen for data collection, as listed below. These were designed to test limitations in the movement detection capabilities of both the PIR and CSI systems, while also acting as representative samples of behaviour which could be observed in smart home health monitoring scenarios. Notably, movements 4 and 5 were chosen to test the ability of the solution to respond to changes in distance between the subject and sensor while retaining similar sensitivity to both movements.

1. 10 seconds of standing still, followed by 15 seconds of waving arms, and 10 seconds of standing still.
2. 7 seconds of waving arms, followed by 7 seconds of standing still, and 7 seconds of waving arms.
3. 10 seconds of standing still, following 10 seconds of walking, and 10 seconds of standing still.
4. With the subject close to the sensor, 5 seconds of sitting, followed by stand up activity over 5 seconds.
5. With the subject further from the sensor, 5 seconds of sitting, followed by stand up activity over 5 seconds.

The subject initiated and concluded captures using a smartphone. The ground truth, against which sensor data was compared, was generated from these timelines. At 10 samples per second, accuracy is measured as the ratio of correct to incorrectly classified samples.

5.1.8 System Configuration

The collection devices are configured as shown in Figure 5.3:

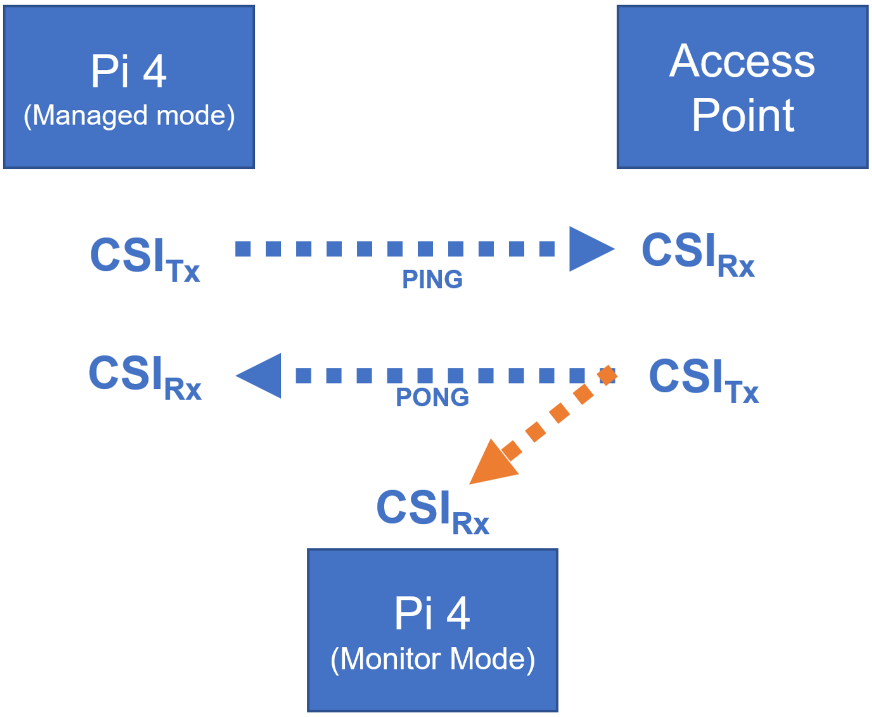


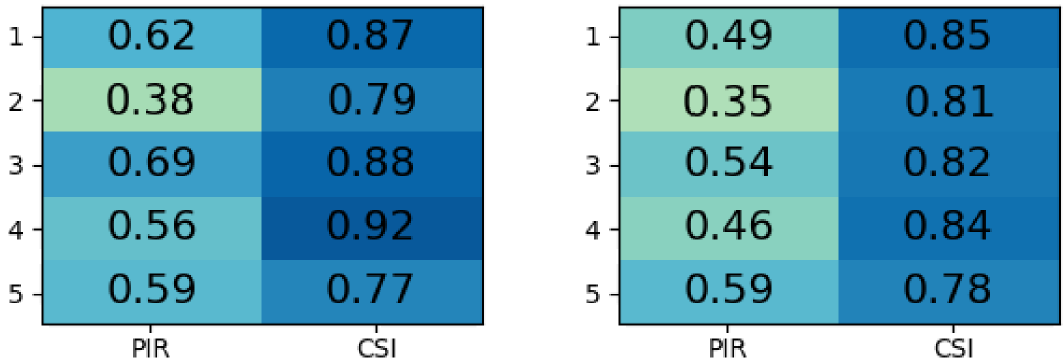
Figure 5.3: CSI capture device configuration.

- **Access Point**: Sky ER110 Router running a 5GHz wireless network on channel 36 at 80MHz bandwidth;
- **Traffic Generator**: Raspberry Pi 4 running in Managed mode, sending 100Hz ping packets to the AP;
- **CSI Collector**: Raspberry Pi 4 running in Monitor mode, listening to the Traffic Generator. In addition, a BIS0001-based PIR was connected to the collector device for comparison.

This system produces CSI data at a rate of roughly 100Hz, which can then be resampled to meet the desired 100Hz source rate.

5.1.9 Results

The performance observed shows the PIR system detects the extent of movement with an average accuracy of 56.7% and 48.6% in environments 1 and 2 respectively, while the CSI system does so with an average accuracy of 84.6% and 82%. In all movements across both



(a) Environment 1

(b) Environment 2

Figure 5.4: Accuracy of the statistical system for movement detection using CSI data from the Raspberry Pi 4.

environments the CSI system performs better than PIR by an average of 32%, with a best case improvement of 46%.

5.1.10 Discussion

The goal of this experiment was to identify whether our CSI movement sensor can offer similar or improved performance to a PIR. The strong performance observed indicates that our CSI sensor can outperform PIRs in monitoring the instance and duration of movement events in a smart home environment. While the activity performances captured for each movement pattern may not perfectly match up with the fixed-time ground truth, it is clear the CSI sensor is performing well with respect to both the PIR and the ground truth.

Performance is largely similar between the two environments, showing the system can be accurately calibrated for different environments as described in Section 4.3. Slightly lower performance can be observed in environment 2. This may be caused by external factors affecting the RF conditions in the environment. The subject's performances for each data capture may also display poorer adherence to the ground truth timelines. Due to the increased time precision of the CSI sensor, it would be advisable to consider a more sophisticated ground truth for movement capture such as video.

Consistency can be observed among most of the movement styles, however there is a notable drop in performance on movement 2 samples for the PIR. Movement 2 begins and ends with movement activity, a design decision aimed to test whether the variance analysis algorithm could correctly classify samples beginning with the subject already moving. Our results showed our systems could take up to 1s to identify movement from the start of capture in these examples. As the subject controlled data capture, movement may have also in fact concluded for a short period before the data capture ended. Both the PIR and CSI sensors demonstrate this behaviour as shown in Figure 5.5. However the CSI system more accurately captures both sustained movement and the termination of movement as the subject ends the capture.

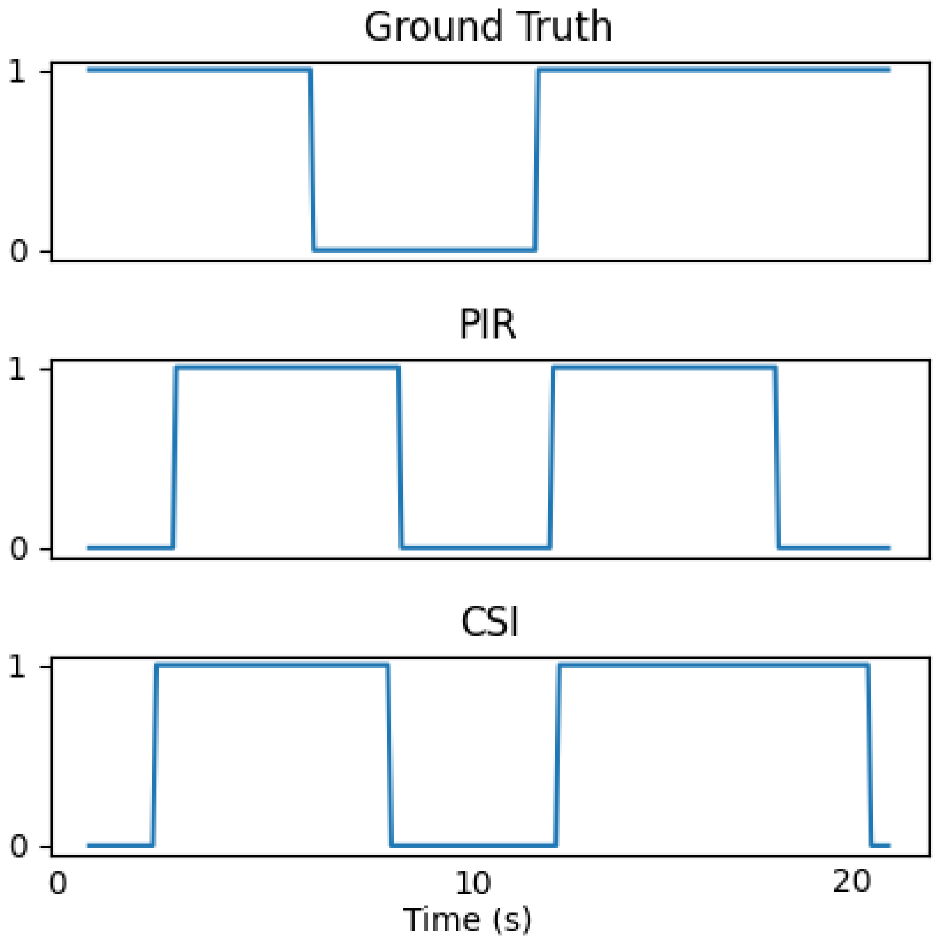


Figure 5.5: Binary movement output for a movement 2 capture in environment 2.

5.2 Activity Recognition

The Raspberry Pi 4's price and potential capabilities position it as a powerful tool for smart home health monitoring using CSI. If it can be provide health monitoring data of similar value to more standard technologies such as vision and wearables, then its completely ambient nature would allow for more ubiquitous deployment in smart housing. Establishing the capabilities of a technology can be a slow, iterative process. To address whether further research will be necessary, it would be beneficial to identify the Pi's performance in activity recognition as this is a well-rounded challenge for CSI-capable hardware. In our experiments, we aim to establish the Pi 4's activity recognition capabilities in a home environment, in both Line of Sight and Non-Line of Sight scenarios (NLOS).

The key contribution of this work is to establish the Raspberry Pi 4 as a capable device for ambiently monitoring activities in the home. Further contributions from this work are:

- **CSIKit**, a Python framework for interacting with data from a range of CSI hardware
- **A public dataset** of generated CSI and activity annotations from Raspberry Pi 4 hardware
- **An implementation** of an established activity classification model structure on the above dataset

In this work, we examine the placement of the Raspberry Pi 4 in CSI hardware deployment and the differences in performance we could expect to see in targeted scenarios. The activity recognition task is then outlined, including how it is distinguished from gesture recognition. We then outline the selected activities given the available environment and hardware, and how they were varied to represent realistic activity performances. The configuration used for the deep learning model structure is detailed. Finally, the results from our experiments are discussed, followed by a discussion on how this could affect future implementations.

5.2.1 Methodology

The aim is to identify whether the Pi 4 can effectively be used to perform ambient smart home activity recognition in a representative environment, as has been demonstrated is possible with IWL5300. Traditionally, standard machine learning classification algorithms such as SVM have been used on CSI data for activity recognition (Lee et al. 2019). However recent research has shown that CSI data is well suited for use with convolutional LSTMs (Wang, Jiang, Hou, Huang, Dou, Zhang & Guo 2019). We use a deep variant of this model, denoted as DeepConvLSTM, which requires more training and can potentially learn higher level concepts. As part of setup, the structure of our model will be established experimentally based on the available data for the task. It is expected that this configuration using data from the Pi 4 will benefit more from a convolutional structure, due to the increased number of available subcarriers over the more common IWL5300.

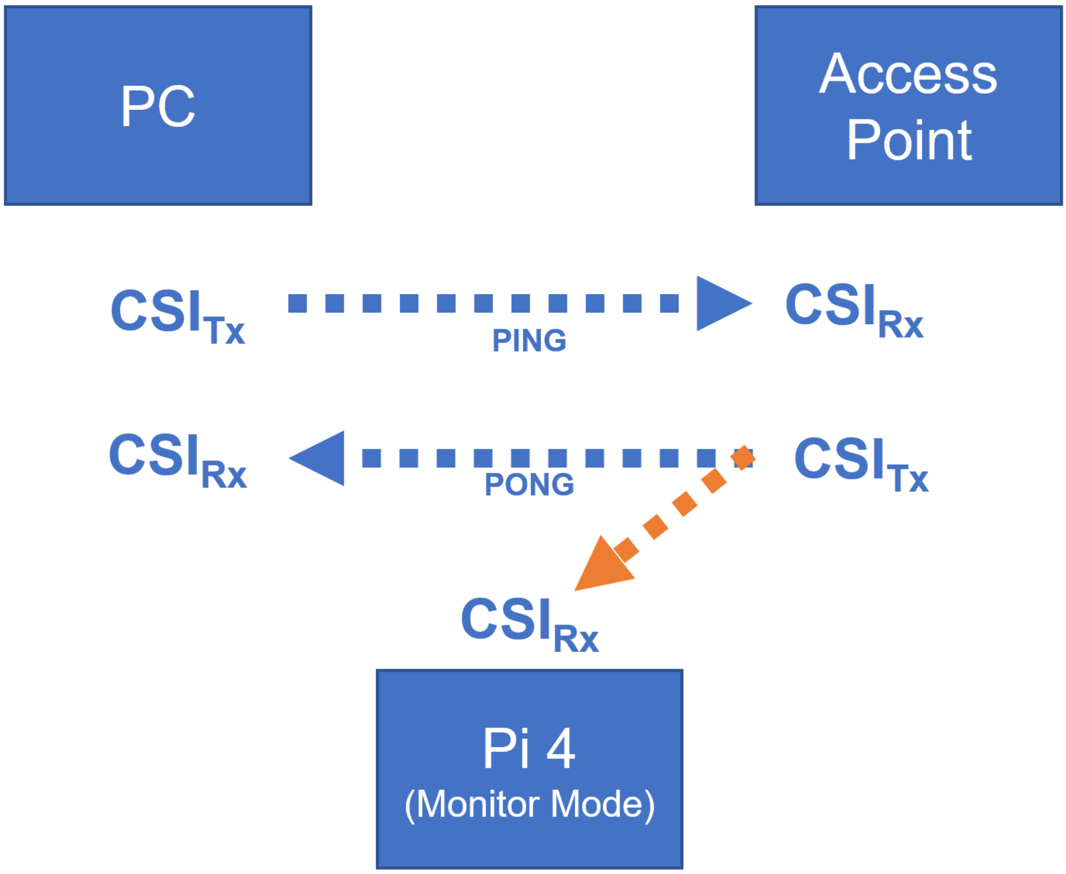


Figure 5.6: Activity capture layout, showing CSI as captured at each device.

A device configuration will be assembled to collect CSI data from the Pi 4. CSI data can then be captured as activities are performed in the environment. Once this data has been collected, a classification model can be trained on the labelled examples. The efficacy of this model can then be identified using labelled examples from the test set. The device configuration for data collection can have a massive impact on the quality of the activity performance captures. The chosen configuration aligns with what would be expected in a realistic in-home monitoring scenario and is similar to those used in other CSI activity recognition studies. Figure 5.6 details how each device in the configuration communicates. The role of the PC is to generate traffic, from which CSI can be captured. As the PC sends ping packets to the Access Point (AP), the AP will reply with pong packets. The Pi will then capture CSI for each pong packet as they are sent out, which will contain information on the RF and physical disturbances between the AP and Pi.

No data preprocessing was performed on the collected CSI amplitude values. This is because any preprocessing or filtering could affect the performance of a real-time system in a way that may reduce overhead for other simultaneous applications. If the results of this experiment are found to be disappointing, then basic signal preprocessing such as a low pass filter could be applied to reduce high frequency noise components. The raw CSI amplitude values are compiled into a 256 x 1 vector and passed to the model for training.

These CSI vectors can then be packed into windows, which are ready for training and classification with a model. Training this model can be performed on a dedicated system. This allows for far more complex models to be produced than would be possible on the Pi hardware alone. While training a deep learning model can be very intensive, the completed models can be easily deployed on the Pi hardware at runtime with fast classification performance.

5.2.2 Experimental Setup

In this experiment, we aim to measure the Pi's ability at classifying a set of performed activities. A range of activities were each performed 100 times in a home environment. CSI data was captured at 100Hz while these activities were being performed. This data is then read into overlapping windows and passed to a model for training.

Equipment

The Raspberry Pi 4 is configured with Debian 10 (Buster/Linux 4.19.97) with the main branch of Nexmon CSI[1] installed. Nexmon was configured to ensure the device operated

[1] `https://github.com/seemoo-lab/nexmon_csi/`

on 5GHz Channel 36, observing an 80MHz bandwidth, and collected CSI with a 30us Delay. A MAC address filter was set to ensure only frames generated by the AP were collected. Data collection was controlled from a smartphone connected to the Pi over SSH, communicating over a separate 2.4GHz network to reduce interference.

The AP used is a Sky ER110 wireless router operating a 5GHz WiFi network on channel 36 at 80MHz. Finally, a separate wireless device is paired with the AP to generate traffic for which the Pi can capture CSI. This is accomplished by sending flood pings at a consistent 0.01s interval to the AP. While this does not guarantee a consistent sampling rate, the resultant timestream can be linearly interpolated to approximately 100Hz.

Environment

This experiment was performed in a small apartment within a terraced block. Due to COVID-19 lockdown, these experiments were limited to a single home environment and a single subject. The building has granite outer walls, with a drywall interior. These factors may have an effect on the overall performance of this system, however the extent of this has not been fully explored (Lee et al. 2019). The apartment has an open plan layout and doors were kept open for the duration of the experiments. Five other 5GHz wireless networks were operating at the time of the experiment, however this was deemed to be indicative of realistic interference which might be observed in an occupied smart home environment. Similarly, the apartment is also shared with a small house cat which operated autonomously throughout and moved between rooms during activity performances. This was also deemed to be representative interference.

Figure 5.7 shows the placement of the Pi and AP in relation to where activities were performed in the apartment. The Living Room is the largest room in the space. so the Pi and AP are placed at opposing corners. Both devices were placed at the same height 1m above the ground to ensure there was an unobstructed signal path.

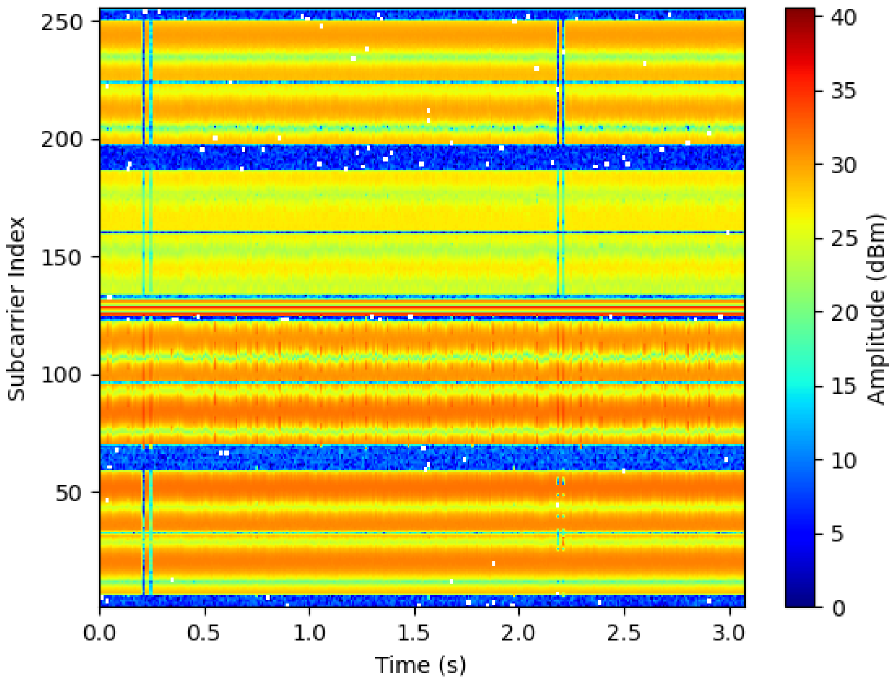
Activity Performances

Each activity being performed in the dataset has been designed to both be easily repeatable and representative of realistic in-home behaviour. The selected activities were also chosen to provide a wide range of both similar and distinct activities to effectively assess the

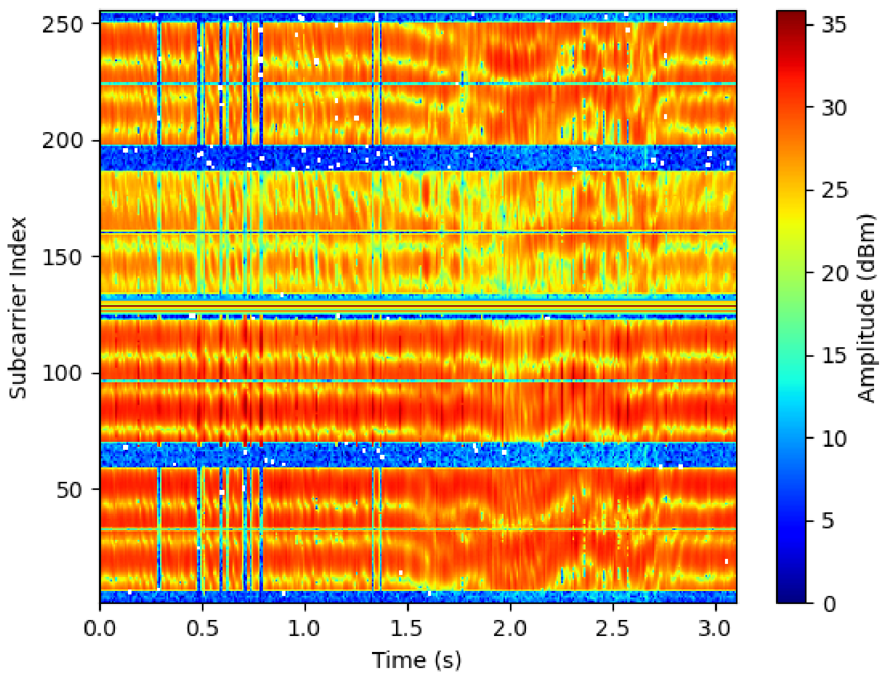
Figure 5.7: Layout of the environment and activity locations.

performance of the classifier, and the quality of the data the Pi produced. These activities are as listed: nothing, standup, sitdown, getintobed, cook, washingdishes, brushteeth, drink, petcat, sleeping, walk. Data was produced by commencing capture as the activity was about to be performed and concluded once the activity was completed. This can observed in Figure 5.8(b), with data remaining mostly stable at the start of the capture before significantly changing, and then returning to a stable state. The capture procedure was controlled by the subject, and so there may be slight variations in the length of time taken before the activity fully begins and after it concludes. The overlapping windows being used for the model should mitigate this in some fashion.

The "nothing" activity was designed to allow the system to classify instances where there is no clear activity being performed, however this method may not be fully effective at providing a null space representation. These captures were performed with the subject sitting on the floor in the living room with no significant movements.



(a) still



(b) walking

Figure 5.8: Comparison of still and walking CSI readings.

Figure 5.7 details the location at which each activity was performed. This shows that activities which took place in the Bedroom, Bathroom and Kitchen would be considered NLOS in that neither the Pi or AP have a direct line of sight with the activity space. It is expected that performance will be reduced on these activity classes due to this.

CSI data is captured using nexmon and rendered with tcpdump, which produces a pcap file. This file is then interpreted using our CSIKit which generates 256 x 1 numpy matrices, which can then be used in Tensorflow. From these, the CSI amplitude is derived. The raw amplitude values are then windowed using a sliding window of 1 second at 100Hz, with a .5 second overlap.

Model

The DeepConvLSTM model was implemented in Keras, using the Tensorflow backend running on a system using an Nvidia GTX 1080 GPU. Our implementation of the model is defined as 2 x Conv1D, 1 x MaxPooling1D, 4 x BidirectionalLSTM. The Conv1D layers were configured using the "relu" activation function, 128 filters and a kernel size of 5. The BiLSTM layers used 200 units. The model was then trained to 200 epochs, with a batch size of 128. Multiclass macro f1 scores were calculated using 10-fold cross validation.

5.2.3 Results

Overall, strong multiclass performance can be observed. Several classes show clear certainty with complete precision, recall and F1. The largest overlap can be observed in Figure 5.9 as "standup" and "sitdown" show a clear overlap in confusion. Activities performed in NLOS scenarios appear to show no clear performance trend, with both "washingdishes" and "brushteeth" working well, while "cook" shows a slight drop. It is noted that classes with the largest numbers of instances show strong performance. The average accuracy across all classes is 92% which indicates this system functions well.

5.2.4 Discussion

The key objective of this experiment was to identify the Raspberry Pi 4 can effectively be used for smart home activity recognition. The strong performance observed in these results indicate that the CSI data produced by the Pi 4 does appear to be nuanced enough

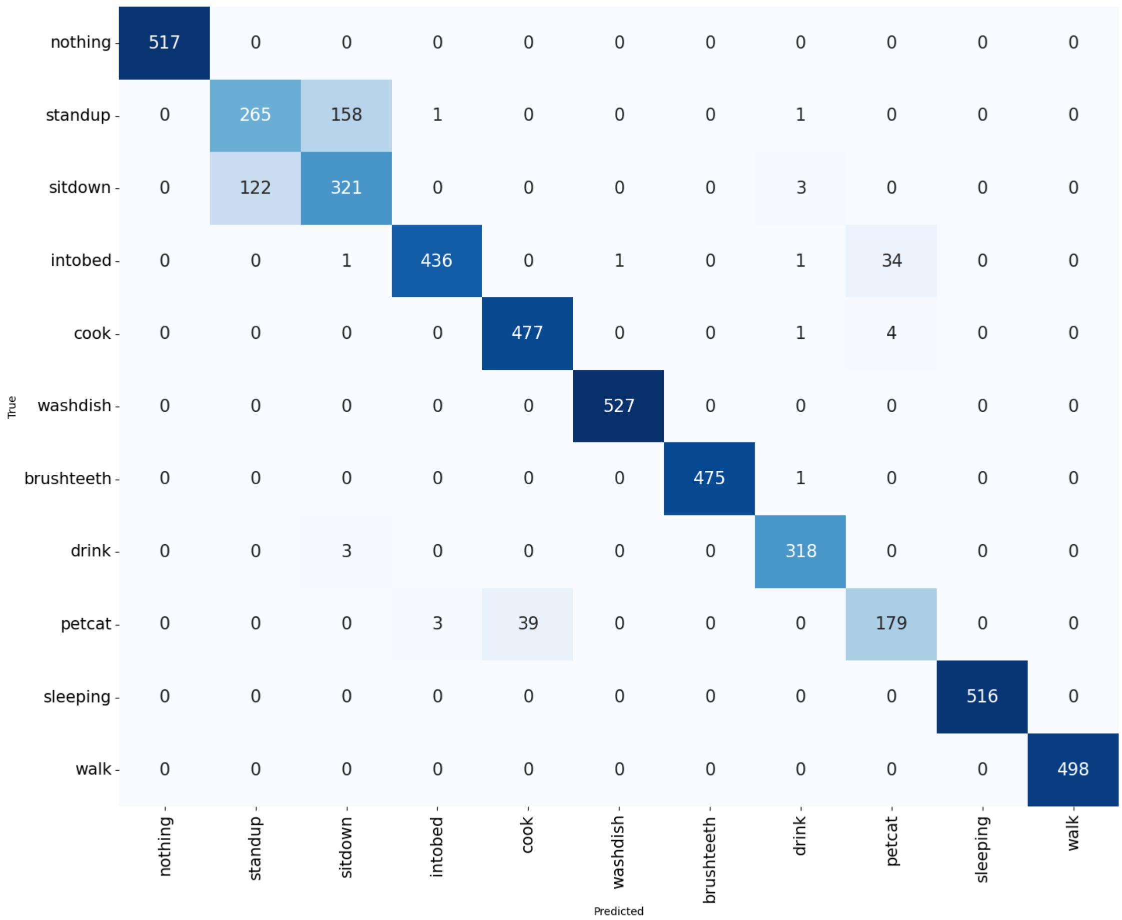


Figure 5.9: Confusion Matrix for 100Hz results.

Table 5.1: Classification report.

Activity	Precision	Recall	F1	Support
nothing	1.00	1.00	1.00	517
standup	0.68	0.62	0.65	425
sitdown	0.66	0.72	0.69	446
getintobed	0.99	0.92	0.96	473
cook	0.92	0.99	0.96	482
washingdishes	1.00	1.00	1.00	527
brushteeth	1.00	1.00	1.00	476
drink	0.98	0.99	0.98	321
petcat	0.82	0.81	0.82	221
sleeping	1.00	1.00	1.00	516
walk	1.00	1.00	1.00	498
Accuracy			0.92	4902
Macro Avg.	0.92	0.91	0.91	4902
Weighted Avg.	0.92	0.92	0.92	4902

to allow our DeepConvLSTM model to classify activity instances well. Even considering some of the classes are quite similar we achieved 92% accuracy which demonstrates effective performance.

The clearest overlap in confusion, as seen in Figure 5.9, is between "standup" and "sitdown" classes. These activities do appear to be very similar. They take place on the same chair in the environment. One aspect which may affect this confusion is the style of windowing being used here. As there is a very short, but variable amount of time taken both before the action in the activity capture occurs and after, there may be additional windows being passed to the classifier that actually do not contain the act of standing up or sitting down. In these instances, the windows will still be labelled which may serve to further confuse the classifier at training. By more tightly controlling the data collection procedure and ensuring windows do contain activity behaviours, this may mitigate some of the observed confusion here.

Another area of confusion concerns two somewhat dissimilar activities, "petcat" and "cook". Each activity takes place in a different room and at different heights, but it appears the repetitive arm movements may have some impact on this. As a NLOS activity, "cook" activity performances may produce less distinct data patterns for the model.

Real-world Performance

While this system performed well in this experiment, this may not be fully representative of real world performance. We acknowledged many factors in this experiment that may have an impact on performance which has not been quantified, such as the impact of interior wall materials. Furthermore, this system would be deployed across many smart home environments and so it cannot be expected that training examples can be provided specifically for each resident/environment combination. Training a general model which can be deployed across each home will be necessary. A recent study has addressed this issue by performing domain adaptation in order to learn different environments (Narui, Shu, Gonzalez-Navarro & Ermon 2020). Potentially transfer learning may be of interest here.

In a real-world scenario, it is expected that classification models will be deployed and run on the Pi hardware itself. As the Pi has limited processing power compared to the systems on which our models are typically run, we will need to consider the sampling rate and window sizes being used for classification. Several studies have investigated the effect of sampling rate on CSI system performance, and this seems to indicate anything up to a 20% drop in performance when dropping from 100Hz to 10Hz (Liu et al. 2015, Lee et al. 2019). Downsampling our activity capture data to 10Hz, we repeated our experiments with a similar model structure. In Figure 5.10, we compare performance observed for each activity class for both our 100Hz and 10Hz configurations. Overall, we can see a slight reduction in performance in some tougher classes like "standup" and "sitdown". However, some classes such as a "cook" show no reduction in performance despite the significant reduction in sampling rate. Additionally, operating at this sampling rate could allow for the removal of the separate PC for traffic generation in the system if the Pi could capture CSI for frames ambiently generated by the AP. This would allow for deployed Pi systems to utilise the existing WiFi infrastructure in smart homes.

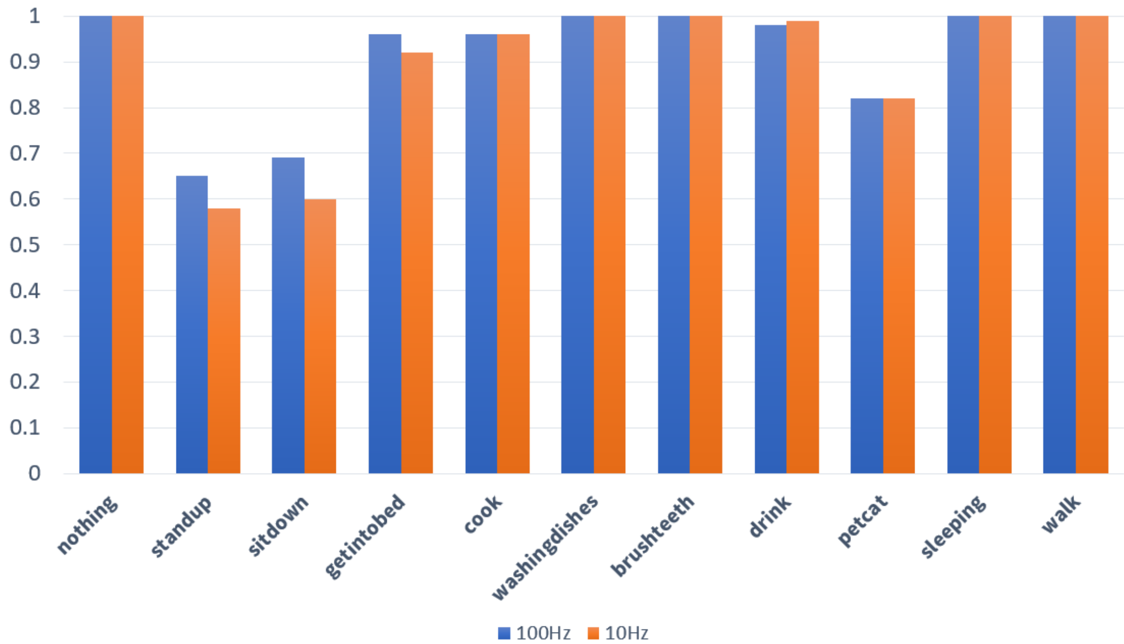


Figure 5.10: Comparison of per-class performance on data captured at 100 and 10Hz (macro F1 scores).

5.2.5 Conclusion

Our results confirm the Rasberry Pi 4 has capabilities for use in ambient activity recognition in smart homes, and can be deployed in similar environments to those used in studies using the IWL5300. It appears the DeepConvLSTM model is well-suited to the CSI data produced by the Pi 4. Potentially, other models may be worth investigating, such as autoencoder recurrent networks. An exciting aspect of these results is the performance observed when using the model with data captured at 10Hz. This potentially further reduces the cost of a Pi-based system for real-world deployment, allowing it to benefit from the existing WiFi infrastructure in most smart homes.

Many non-standard applications have also been explored for CSI. Targeted research implementations may in fact have value in residential health informatics where it may not be immediately obvious, such as smoking recognition (Zheng, Wang, Shangguan, Zhou & Liu 2016) and crowd counting (Liu, Zhao, Xue, Chen & Chen 2019). These applications have demonstrable value in ambient health monitoring and the strength of a deployed in-home solution would be in merging these capabilities given they make use of the same input data stream. Combination CSI extraction and analysis systems making use of several health monitoring solutions would represent a significant step forward in this field.

5.3 Summary

In this chapter, we detailed our process in establishing the capabilities of a CSI-based sensor. Initial evaluations have highlighted the sensor's aptitude for measuring human movement and distinguishing activity performances. The results observed demonstrate CSI from devices using a single omni-directional antenna can be used for sensing applications tracking human movement in a coarse manner. As it outperforms PIR-based sensors in our initial experiments, CSI sensing has been demonstrated to offer increased accuracy in measuring the duration of movement events.

Our experiments also show that while a modern single-antenna solution reduces the scope of potential applications over common multi-antenna options, the coarse sensing capabilities demonstrate a clear benefit over PIR-based coarse sensing. By demonstrating and testing the implementation of two sensing applications using CSIKit's homogeneous data structures, we establish its functionality as a data processing platform for generic CSI data. CSIKit provided us with a platform which we used to support a comparison of COTS hardware for CSI-based sensing in 6. It was expected that differences in hardware capabilities and sensing performance could be linked to hardware costs. We used this comparison to identify the viable hardware entry-point for low-cost device-free wireless sensing.

As additional sensing applications are likely possible with the same data collected for these initial applications, we then aimed to develop a platform to collect this data over a long term to produce a large dataset for a variety of sensing applications using deep learning. Section 7.3 addresses the challenges faced in continuously collecting CSI data in the FitHomes network. Our hardware configuration can also be further optimised to reduce the power profile and hardware costs, which aligns with the goals of our funding partners.

Chapter 6

Comparing COTS Hardware for CSI-based Sensing Applications

If CSI sensing solutions could be easily deployed in homes worldwide, it would have already been done. Compelling sensing applications with CSI have been presented since 2013 (Xiao, Wu, Yi, Wang & Ni 2013), yet very few commercial deployments have been seen (Linksys 2019, Origin Wireless 2019). The reasons for this are two-fold: the COTS hardware used for the majority of CSI research and development are no longer in production, and regardless, they cannot be efficiently deployed at scale for ambient sensing. High-end CSI hardware is common in research, as is older mid-range hardware. Low-cost CSI collection solutions are now available, though it is unclear how their performance may differ in sensing applications. To identify a viable entry-level solution for COTS CSI sensing research, the performance of COTS devices must be compared.

Many hardware options exist for COTS CSI collection. Previously, CSI research was limited to ray-traced simulations and enterprise radio equipment. Once Halperin's Linux 802.11n CSI Tool was released for the Intel IWL5300 in 2012, it became the dominant COTS CSI collection platform. This is largely due to its low secondhand cost, MIMO antenna support, and the accuracy of Halperin's solution. Following this, the Atheros CSI Tool was released for AR9XXX chipsets. In more recent years (2018-present) the available device selection has rapidly expanded. The Raspberry Pi 4, ESP32, and Intel AC9260/AX200/AX210 have broadened the hardware options for modern COTS CSI research. Many researchers still primarily use the Intel IWL5300 to date, due to its reliability and wide use in published works. With the increased device choice, the selection criteria for a given hardware platform is still unclear as little comparative research exists.

The most significant factors affecting hardware selection are accessibility, cost, and potential for deployment. These factors are usually linked. For instance, more accessible hardware offering fine-grained control over data tends to be more expensive. Cheaper hardware for ubiquitous deployment can require more specific hardware placement and signal processing. Consumers for CSI collection hardware have varying requirements for their hardware. A comparison of the available hardware options is required to support selection decisions. The criteria for this comparison will focus on identifying how the available hardware options differ in functionality, modern 802.11 features, wireless performance, cost, and their potential to scale for large-scale deployment. We developed novel approaches to perform this comparison, by utilising a homogeneous antenna configuration and software implementation for generic COTS CSI hardware.

The contributions in this chapter are as follows:

- First, **we outline the selection criteria** for available CSI-capable COTS hardware.
- Second, **we establish a novel methodology for CSI hardware comparison**, by utilising a homogeneous antenna and software implementation for CSI collection and processing.
- Third, **we directly compare wireless performance between COTS hardware**, by comparing signal quality metrics captured from each device on a homogeneous antenna configuration.
- Finally, **we compare CSI sensing performance with COTS hardware**, through evaluation of a novel generic sensing application for human motion detection. This is the first sensing application deployed using generic COTS CSI hardware which we have observed.

6.1 Hardware Options

The CSI collection hardware selection compared in this work is not fully exhaustive. Rather, this comparison aims to cover a range of hardware deployment paradigms, 802.11 capabilities, and price points, while also supporting direct comparison for traffic streams from the same injecting device. Through comparison of these COTS devices, the relevant hardware for various use cases will be considered. For instance, research-focused users may care less about size and power consumption, in exchange for more hardware capabilities or accuracy. Applied researchers may prefer low power hardware for ubiquitous deployments, which may offer less accurate data. The following evaluation will identify whether a trade-off is being made in both wireless and sensing performance, and whether this presents an issue for research accessibility.

A notable exclusion from the listed hardware is the Atheros AR9XXX series. It does not support monitor mode-style collection for packets following the 802.11 format criteria needed to use the Intel IWL5300. As such, it is not included in these tests. The most analogous solution to it would be the IWL5300 itself, as it supports 802.11n with 3x3 MIMO.

6.1.1 USRP B210

Ettus Research produces state-of-the-art SDR equipment, operating under the National Instruments umbrella. Their USRP range of hardware is one of the leading standards in enterprise SDR offerings, and their UHD API is widely supported. For most USRP hardware, the FPGA source code and board schematics are freely available. The open nature of the hardware ensures accountability for specifications and encourages developers to integrate UHD with support from Ettus. This positions USRP hardware as a strong reference for a performant CSI collection device. A USRP B210 was used for the experiments in this work, which exceeds the specifications necessary for an accurate comparison with COTS devices. This unit is 155x120mm.

The USRP B210 is a dual-channel full-duplex SDR, supporting frequency ranges from 70MHz to 6GHz at a maximum bandwidth of 56MHz. It features a 12-bit ADC/DAC, and has a listed frequency correction error of +/- 2ppm.

The software stack used for CSI collection is entirely configurable, with various options for assembling the necessary RF chain. A common approach is to transmit and receive OFDM symbols and extract the preamble containing CSI with GNURadio. The raw complex values can then be saved to a file for processing, or visualised in realtime with a QT sink. Another approach is to use a proprietary processing chain such as PicoScenes (Jiang et al. 2022), which can capture CSI into their container format. The main benefit of using either stack is the ability to finely tune most parameters of the transmission and the receiver, including gain and automatic gain control (AGC). Additionally, raw baseband signals can also be captured alongside CSI. These can be used to measure Error Vector Magnitude (EVM) to establish low level 802.11 performance and interference characteristics.

Primarily, SDR-based CSI collection is a research pursuit. To support real-time 802.11 transmit and receive, an FPGA-based 802.11 modem implementation must be used. Few of these are available, and most are hardware-specific. Due to these modem implementations running in-silicon, observed packet rates tend to be lower than that of dedicated NIC hardware(Jiang 2022). The high cost of most SDR hardware, coupled with the experience necessary to efficiently operate it are significant barriers to entry. While the hardware itself is typically small, many SDRs do not run applications themselves and instead rely on external compute. This includes the collection and storage of the data itself, regardless of its destination. As such, SDR-based deployment involves consideration of the physical space requirements of the SDR, and a compute unit for processing. This method of CSI data capture suits the purposes of research and development.

6.1.2 Intel IWL5300

The Intel IWL5300 was released in 2008 with support for the 2006 draft-N 802.11 specification. The common mPCI-E half-height SKU is sized at 27x30mm. Marketed as the Intel Ultimate N WiFi Link 5300, it was a well-equipped NIC for the time. Today, this chipset remains the most common COTS CSI collection device.

The IWL5300 is a wireless network adapter operating on the 2.4GHz and 5GHz frequency bands. It supports 3x3 MIMO, supporting a maxmimum theoretical data rate of 450mbps. Both 20MHz and 40MHz bandwidths are supported in standard NIC operation, however CSI collection is limited to 20MHz. As standard with COTS NICs, more information beyond these top level specifications can be hard to gather. Through Halperin's work, the onboard ADC/DAC has been revealed to use 6-bit precision.

The primary method of CSI collection with the IWL5300 is the Linux 802.11n CSI Tool (Halperin et al. 2011b). This modified firmware and kernel driver, produced with support from Intel directly, spawned a massive body of CSI research currently growing to date. The extensive implementation details most elements of the RF chain, with scripts provided for mitigating most manufacturer-specific transformations such as RSSI, AGC, and spatial mapping. Many newer CSI collections such as PicoScenes build upon the strong base provided by Halperin. This level of detail is unmatched in other COTS CSI collection solutions.

Despite the evolving support for the IWL5300, basic limitations of the chipset cannot be transcended. The modified driver and kernel support for the chipset are not supported on ARM platforms. This means an x86 compute unit is necessary to run the NIC for CSI collection. Additionally, the IWL5300 is no longer in commercial production. For research purposes, the hardware is still accessible and low cost, however it cannot be purchased or deployed at meaningful scale. The IWL5300 is the ideal device for researchers aiming to recreate and build upon published works which use it.

6.1.3 Raspberry Pi 4

The Raspberry Pi 4 released in 2019, maintaining the Raspberry Pi Foundation's dominant position as producer of the world's most popular single board computer. The board itself is sized at 86x56mm. Inheriting the Broadcom BCM43455(c0) wireless chipset from its predecessor (Pi 3B+), the Pi 4 supports the relatively modern 802.11ac specification from 2013 and features increased CPU clock speeds. The Raspberry Pi series is positioned as a powerful, accessible board for both entry level and experienced users. As such, the Pi 4 is one of the most accessible CSI collection solutions available.

The BCM43455 is a single-chip WiFi and Bluetooth transceiver which operates on the 2.4GHz and 5GHz frequency bands. As installed on the Raspberry Pi 4, it features an onboard dual-band omni-directional antenna. While this antenna is not configurable on the Pi 4, the Raspberry Pi Compute Module 4 features a u.fl socket for an external antenna connection. This single antenna solution operates with a maximum theoretical data rate of 433mbps, following the 802.11ac standard. 20/40/80MHz bandwidths are supported in 5GHz mode, however 2.4GHz is limited to 20/40MHz. ADC/DAC bit-depth information is not available.

Researchers from TU Darmstadt's Secure Mobile Networking Lab released Nexmon CSI in 2019. Initially, the Raspberry Pi 3B+/4 and Nexus 5/6P were supported, with the Asus RT-AC86U being added later. This reverse-engineered approach to CSI extraction is unprecedented, with additional functionality being added over time. As is the case with COTS solutions developed without support from the manufacturer, some manufacturer-specific transformations such as AGC are still unclear. Additional software solutions for parsing and processing CSI data from Nexmon CSI have been released (Forbes 2020, Kindt et al. 2022).

The relatively small physical size of the Raspberry Pi 4, and its single board usage present it as an accessible, low net cost approach to CSI research. Its ARM SoC consumes far less power than standard research CSI collection solutions, while operating silently. While the body of work using the Raspberry Pi 4 for CSI sensing research is small in comparison to the IWL5300, its modern and accessible wireless chipset and onboard SoC with edge processing capabilities will likely encourage significant future research.

6.1.4 ESP32

Espressif Systems released the ESP32 in 2016, a low-power dual-core 32-bit microcontroller with extensive peripheral device support. The SoC features 802.11n wireless capabilities, marketed for a wide range of commercial applications from IoT to edge AI solutions. The unit size varies, with the average model being 18x17mm. It is priced around £3GBP, making it the lowest-cost CSI collection solution.

In wireless performance, most ESP32 variations do not differ significantly in specification or performance. One of the most common ESP32 chips is the ESP32-WROOM-32, which will be used for this comparison. The ESP32 is a single-chip WiFi and Bluetooth transceiver operating on the 2.4GHz frequency band. ESP32-WROOM-32 features an onboard omnidirectional PCB antenna, though subvariations offer u.fl connectivity for an external antenna. It supports a theoretical max data rate of 150mbps, supporting

20/40MHz bandwidths. The ESP32 has 2 onboard 12-bit ADCs, with one reserved for the WLAN radio and other internal functionality.

Support for CSI collection through the ESP32 WiFi driver was added by Espressif in 2019. Documentation, example code, and best use practices were released alongside. As with most COTS hardware, the source code and block diagrams for the WiFi driver and hardware are not available which means some manufacturer-specific transformations such as AGC are unclear. Shortly afterwards, Hernandez released ESP32-CSI-Tool for collecting captured CSI data in CSV files (Hernandez & Bulut 2020). The low-cost and small size of the ESP32 positions it as a strong option for large-scale deployment for CSI sensing. Few works have been published using the ESP32 for CSI research, though initial results are promising (Hernandez & Bulut 2022).

6.1.5 Intel AX200

The most modern wireless chipset in this comparison is 2019's Intel AX200. Of all the COTS devices compared, the AX200 features the most recent complete WiFi specification: 802.11ax. Similar to the IWL5300, the main SKU is a peripheral PCI-E card sized at 22x30mm. Primarily sold as an OEM product, many modern Intel-powered laptop and desktop systems are preinstalled with the AX200.

The AX200 is a wireless network adapter with WiFi and Bluetooth capabilities. 2x2 MIMO is supported, with support for 2.4GHz and 5GHz frequency bands at 20/40/80/160MHz bandwidth. Through 802.11ax, the chipset is capable of a maximum theoretical data rate of 2.4Gbps. ADC/DAC bit-depth information is not available. By far, the AX200 is the most capable NIC in this comparison.

Intel added CSI reporting to the `iwlwifi` driver in Linux kernel version 5.17. Through this support, the AX-series driver can dump frames and CSI measurements through the kernel. Various applications can then register to receive this data and collect CSI measurements for storage, parsing, and processing. To date, no open source applications of this nature exist, with the main public solution being PicoScenes. PicoScenes is currently only supported on x86 hardware, which incurs similar hardware requirements to the IWL5300 in terms of compute and power consumption. At this stage, the status of AGC and other manufacturer-specific transformations to the CSI data are unknown.

Given its modern feature set, coupled with manufacturer support, and the growing body of research making use of the AX200, it could potentially replace the IWL5300 in the long term. While it is unlikely to be used in a deployed capacity due to its reliance on an x86 compute unit, it appears to be an ideal device for modern CSI research.

6.2 Methodology

In CSI research, various factors affect performance in both collection and application. By definition, wireless systems can experience inconsistencies in the RF chain. As a stream of CSI data should be formed of evenly spaced measurements, packet loss and inconsistent frame pacing can have a negative impact on data quality. Similarly, variations in the strength of received signals can introduce significant noise into the data. This evaluation aims to identify device performance in relation to these factors

To fairly evaluate device performance, the same RF input must be provided to each device. For this purpose, we developed a novel approach using a homogeneous antenna configuration for device comparison. We first used this approach in Section 4.3. Through the use of a signal divider, the signal from a single antenna can be split to 4 SMA outputs. Each device can then be connected to an SMA output, replacing the original antenna onboard. This approach ensures the same incoming signal is provided to each device with no significant modification to gain or phase. While some attenuation may occur as a result of the signal division, this operation is consistent across each of the connected devices. As such, the observed wireless performance is indicative of that of the COTS device being measured, rather than external aspects of the RF chain.

This experiment establishes whether the lowest-cost COTS platform is capable of measuring high quality CSI for ubiquitous sensing applications. This ensures the trade-off between cost and value is economical. As the common feature-set across devices is 2.4GHz/20MHz/802.11n with a single omni-directional antenna, this configuration was used across all devices.

6.2.1 Link Quality

To establish a reference of the signal quality of received frames, a USRP B210 is used to capture raw baseband signals. This allows the RF conditions for the environment and antenna configuration used for the experiment. From those, we derive EVM, SNR, and constellation diagrams. This allows us to establish the overall Transmit/Receive (Tx/Rx) link quality used for the experiment.

6.2.2 Packet Loss

In WLAN transmissions, packet loss is defined as transmitted frames which do not reach their destination. There is no singular cause for this, rather a series of compounding factors at each layer of the chain from hardware to software. This includes but is not limited to: weak SNR resulting from poor antenna performance, inefficiencies in the NIC firmware, and issues with routing within the operating system's networking stack.

Packet loss incurs 2 significant challenges in CSI data collection: unpredictable loss of frames, and inconsistent frame pacing. The vast majority of signal processing methods used in CSI research rely on data being sampled at a consistent rate. Dropped frames manifest as missing samples, disrupting the sequence within the CSI stream. Similarly, delays at both the transmitter and receiver can cause inconsistent inter-frame spacing. The primary cause of this, which typically cannot be easily mitigated, is CSMA CA/CD. CSMA determines the cooperative procedure for reducing wireless interference which cannot be avoided in most cases on COTS hardware. As such, inconsistency will likely always be present.

Acknowledging the inconsistency of the CSI stream in non-simulated environments is a necessity, though the extent to which it impacts data can be managed with performant hardware and a strong RF chain. Typically this factor is mitigated through the use of interpolation, estimating the missing values in a sequence using a smoothing filter.

In this evaluation, packet delay will be monitored across each of the devices, as the inter-frame spacing measured by the devices' onboard clocks. The distribution of interval between frames will be estimated using Kernel Density Estimation (KDE) with a gaussian kernel. As each device can observe variable performance in packet rate, the difference from the max observed packet rate for a given device/capture will be used to evaluate performance.

6.2.3 CSI Stability

Unlike RSSI for which a single value is recorded for each frame, CSI is a vector of complex values for each subcarrier. As such, it cannot be easily compared across devices. While direct comparisons of the CSI itself can be useful, sensing performance is most significantly affected by the consistent behaviour of the device's CSI measurements. This can be monitored by measuring the inter-frame correlation of CSI, using the Pearson Correlation Coefficient. PCC behaviour and its use in sensing applications is detailed in Section 5.1.3.

This evaluation aims to measure the consistency of CSI measurements across the COTS devices, and identify which devices may over or under-observe environmental variations through PCC.

6.2.4 Human Motion Sensing Performance

To evaluate the performance of COTS devices for sensing applications we use an LSTM-based approach trained on data from each of the compared devices. The goal in this comparison is to identify differences in device performance to determine the utility of their CSI data for generic sensing applications.

An off-the-shelf webcam is used with a threshold-based motion detection implementation in OpenCV to establish a ground truth for data labelling and evaluation. This method is an industry standard and largely accepted as an accurate method of motion detection, ensuring a high quality ground truth.

6.3 Experiment

This comparison is established by collecting CSI data from each of the devices while a set of repeated human motions are performed between the transmitter and receiver. Both the general wireless and motion sensing performance characteristics of the devices are measured and compared. Using the collected data, the performance metrics defined above are measured, and a motion sensing application is used to further evaluate COTS device performance for general CSI sensing.

6.3.1 Hardware Configuration

An ESP32-WROOM32 with PCB antenna was used to inject 802.11n frames at a target rate of 100Hz to generate CSI data. In practice, the observed frame count is lower due to varying RF conditions and CSMA CA/CD on the ESP32 OSI Layer 2 implementation. As such, the injection rate will not be considered for packet loss.

The LPD410 signal divider setup defined in Section 4.3.5 was used to provide the same RF input to each device in the comparison. Each device was connected using a shielded SMA-to-SMA cable with the appropriate u.fl/RP-SMA adapters where needed. As this is a 4-way signal divider, all 5 devices cannot be operated simultaneously. Instead, 3 separate sets of captures were performed with the USRP B210 acting as reference present for all sets. The device sets are as follows:

1. **Intel IWL5300**, **ESP32**, **Intel AX200**
2. **ESP32**, **Intel AX200**, **Raspberry Pi 4**
3. **Intel IWL5300**, **ESP32**, **Raspberry Pi 4**

The device set was coordinated using SSH and timed bash scripts to initiate capture on each device, maintain general synchronisation, and collate the resultant data. CSI was collected from each device using the following configurations:

- **USRP B210**: A desktop system with an AMD 3700X CPU with 32GB of RAM was used with PicoScenes running on Linux Mint 20.04. The B210 was connected over USB 3.0.
- **Intel IWL5300**: A laptop system with an Intel i5-3320M CPU and 8GB of RAM was used with Linux 802.11n CSI Tool running on Ubuntu 13.04. The mPCI-E half-height model of the IWL5300 was used.
- **Raspberry Pi 4**: A Raspberry Pi Compute Module 4 with 32GB eMMC and 2GB of RAM was used with stock Nexmon CSI.

- **ESP32**: A common ESP32-WROOM-32D-based NodeMCU board was used with ESP32-CSI-Tool.

- **Intel AX200**: A desktop system with an Intel i5-7600k CPU with 16GB of RAM was used with PicoScenes running on Ubuntu 20.04. A generic PCI-E adapter for the M.2 E-key based AX200 model was used.

The February 14th 2022 build of PicoScenes was used for each of the tests.

6.3.2 Experimental Setup

Five movement types were chosen for data collection, as listed below. These were designed to test limitations of statistical approaches to movement detection with CSI, while also acting as representative samples of behaviour which could be observed in smart home health monitoring scenarios. Notably, movements 4 and 5 were chosen to test the capability of a sensing application to respond to changes in distance between the subject and sensor while retaining similar sensitivity to both movements.

1. 10 seconds of standing still, followed by 10 seconds of waving arms, and 10 seconds of standing still.

2. 10 seconds of waving arms, followed by 10 seconds of standing still, and 10 seconds of waving arms.

3. 10 seconds of standing still, following 10 seconds of walking, and 10 seconds of standing still.

4. With the subject close to the sensor, 5 seconds of sitting, followed by stand up activity over 5 seconds.

5. With the subject further from the sensor, 5 seconds of sitting, followed by stand up activity over 5 seconds.

Each movement type is a performance which contains both periods of movement and no movement. These are then split into .5s windows of CSI data annotated with the motion state.

A webcam was used to capture 800x600px footage of the subject as they moved through the environment. The positioning of the shot was chosen to fully include the subject in frame during each movement. A binary threshold-based movement detection implementation was used with OpenCV, establishing motion as a dilated contour exceeding 400px in size. This configuration aimed to ensure motion sensitivity was not significantly affected by the subject's distance from the camera, though it cannot be fully mitigated.

6.3.3 Motion Detection Models

Our **LSTM-based** model for movement detection was initially developed in pursuit of a deployable real-time system for generic human motion detection with COTS devices. This approach uses 0.5s of 10Hz CSI data, for which PCC and normalised subcarrier magnitude is extracted. In Figure 6.1 it can be observed that features are then passed to a bidirectional LSTM. A dropout layer was used to drop half of the LSTM's units during training to reduce overfitting. Overfitting is of primary concern when training this model, due to the variations in the relationship between PCC and absolute gain for CSI signals for different COTS hardware.

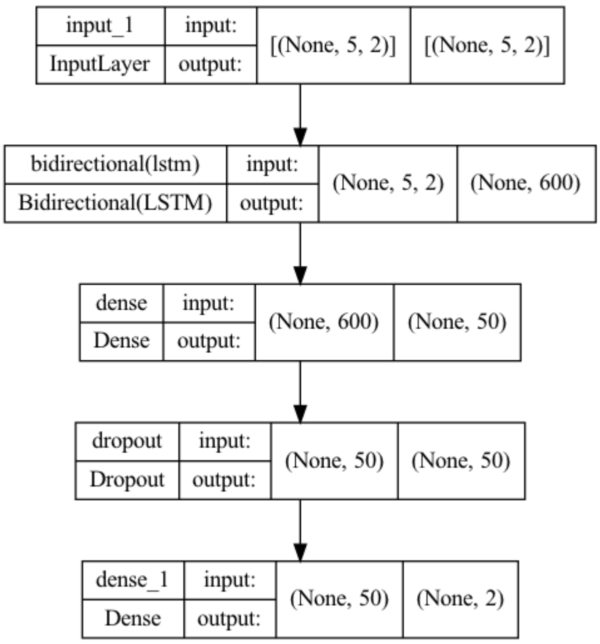


Figure 6.1: Model structure for our LSTM-based movement detection implementation.

The output of this model is a binary classification, determining whether the 0.5s time slice contained human movement. An LSTM was chosen due to its prevalence in time-series classification tasks and its demonstrated comprehension for sequential data. Initial performance during design and testing showed promise.

Our goal in designing and implementing this model was to demonstrate a fully generic approach to CSI-based movement detection. The same model could then be used with data from any of the available devices. This approach has the potential to expand the accessibility of CSI sensing research and development by abstracting hardware-specificity in sensing applications.

A balanced dataset containing 56,208 .5s windows was built using data from all of the devices collected during all of the runs. Data from each device was sparsely sampled to match the number of instances used.

Experimental models were trained using data from each of the devices used in this comparison, with one device being isolated at a time. This is because each device was used to measure the wireless channel for the exact same input signal from the divider. By training one model with data from all of the devices bar one, each device can then be evaluated by isolating its data in a separate test set fold. We then collate data from each of the other devices to build a training set which contains an average estimation of the wireless channel. Individual device performance will then be highlighted in its test stage. Using this model, we compare COTS devices to identify their aptitude for sensing applications.

6.4 Results

The results of the comparison experiments are split into:

- An initial evaluation of the general Link Quality using EVM measured using the USRP B210;
- The Packet Loss observed by each of the COTS devices for the same stream of WiFi frames;
- Overall stability of the CSI data collected by all of the devices used in the comparison;

- and finally a comparison of Human Motion Sensing application performance achieved using data from each of the devices.

6.4.1 Link Quality

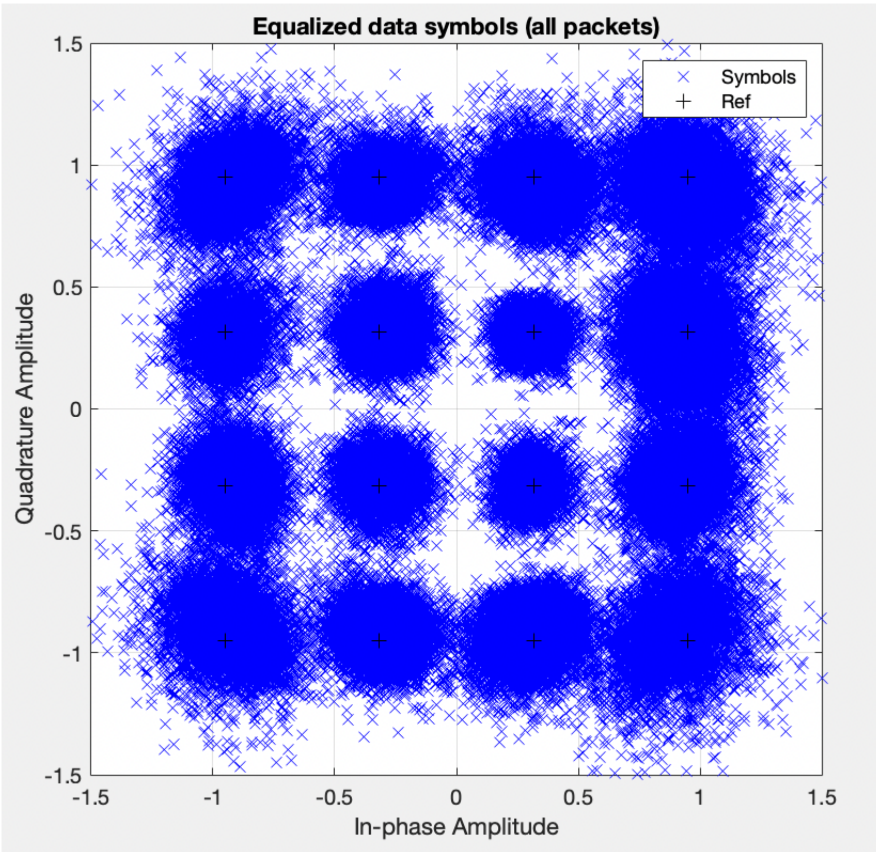


Figure 6.2: Constellation plot for packets received during reference configuration EVM measurements with the USRP B210.

General wireless performance and Link Quality was measured using the baseline USRP B210. COTS devices do not support these measurements with raw baseband signals, and so their wireless signal strength is later compared using RSS. Injected frames were transmitted with the High-throughput Mixed Format (HT-MF) packet format using Binary Convolutional Coding (BCC) with MCS4, 16-QAM at 3/4 coding rate confirmed at the receiver. An average Signal to Noise Ratio (SNR) of 16.77dB was observed through the baseband signals captured using the USRP B210. A low SNR may be attributed to the signal divider used for these experiments, among other aspects of the configuration. The average constellation plot for each reference capture can be seen in Figure 6.2. A general

Table 6.1: Signal quality features for injected frames measured on the USRP B210.

	Min	**Max**	**Mean**
Power (dBm)	-.74	-.47	-.62
CFO (kHz)	-22.50	-16.72	-19.04
Offset (samples)	3024	4.71e+6	2.32e+6
RMS EVM (dB)	-18.51	-14.77	-16.86
Max EVM (dB)	-10.78	-1.41	-8.10

tilt is noted towards the positive ends of each axis, indicating minor phase differences between the I and Q paths. Significant noise can be observed.

Both Max and RMS EVM values are provided alongside other signal features in Table 6.1. At MCS4, the maximum EVM threshold is -19dB, with the both the mean and min RMS EVM falling above this value. As the mean RMS EVM is within 2.2dB of the threshold, the overall link quality is poor.

The IEEE 802.11 WLAN specification mandates devices use an oscillator with a 20ppm tolerance. As such, the Carrier Frequency Offset is well within the required EVM for 2.4GHz (96KHz at 40ppm) and should have little effect on the signal quality and CSI.

Overall this means a relatively low quality wireless link was measured by the reference USRP B210. Poor quality links can be common in WiFi transmissions due to improper device placement which is usually due to necessity of the environmental layout. By performing this comparison with an established low quality link, we can identify how each COTS device performs in network conditions representative of that commonly achieved with home layouts.

6.4.2 Packet Loss

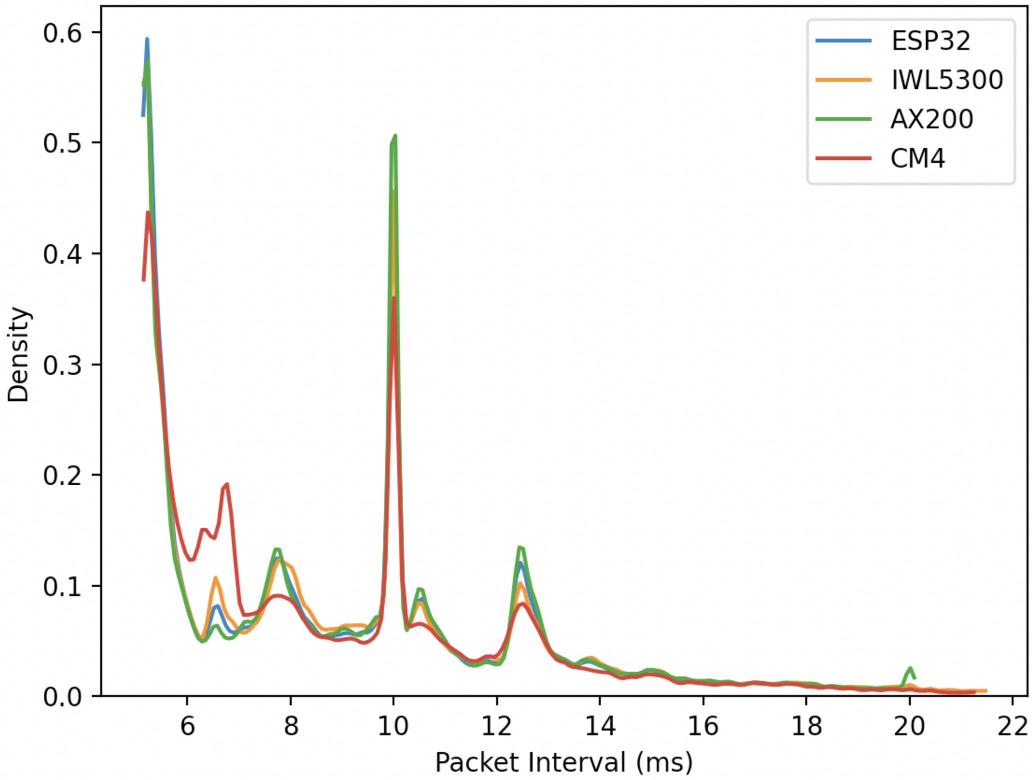


Figure 6.3: Distribution of the 95th percentile of measured delay for CSI frames across each device.

Packet delay is measured as the interval between successive packets for a stream of frames received by a device. In Figure 6.3, the distribution curve of packet delays observed for each device is plotted. The 95th percentile was selected from this data to observe the most significant range. Due to issues in configuration, the USRP B210 could not used as a reference for these tests.

The most significant proportion of intervals were around 5ms. The largest peak at 5.22ms is observed by the ESP32, with 36% of measurements falling between 5-7ms. A similar peak can be observed for each of the devices, with the others at 5.24ms. Another large peak on each device is observed at 10ms, indicating the injecting device is likely sending inconsistently spaced packets. The target injection rate is 100Hz which accounts for the peak at 10ms.

As 10ms is not the largest proportion for any of the devices, potential re-transmissions

Table 6.2: Statistical features of deviation from the highest observed packet rate (Hz).

	Min	**Mean**	**Variance**
AX200	-23	-2.82	12.02
CM4	-9	-0.47	1.91
ESP32	-23	-2.47	15.20
IWL5300	-7	-0.81	1.71

as a result of Tx buffering on the injecting ESP32 can be observed in neighbouring peaks. Complementary peaks can be observed between 7.75ms and 12.46ms, showing multiple repeated comparable intervals on each device. This demonstrates that as a low-cost form of injection, the ESP32's *esp_wifi_80211_tx* function offers baseline performance at 100Hz which may not be suitable for fine-timing applications.

The CM4 observed the highest average packet rate in 53.3% of captures, followed by the ESP32 in 25.6%, IWL5300 in 20%, and the AX200 in 1.1% of captures. Table 6.2 shows features of deviation from the observed max for each device. From this, we observe greater inconsistency in the rates achieved by the ESP32 and AX200 than that of the IWL5300 and CM4. On the IWL5300, the mean rate was slightly lower than that of the CM4. However the variance on the CM4 was higher, indicating that the IWL5300 is the most consistent COTS CSI collection setup used in this experiment. IWL5300 and CM4 show similarly small differences from the max, with the CM4 lagging behind. The ESP32 has the highest variance, showing more instability in frame detection than other COTS devices. From these results it can be determined that the highest instability in packet rate is observed on the ESP32 and AX200 in this experiment. In signal processing tasks using CSI data from these devices, more aggressive interpolation will be necessary to achieve a coherent CSI timestream.

6.4.3 CSI Stability

The PCC between sequential frames of CSI from each device was measured in each run. This was then compared to the PCC observed for each frame with the USRP B210 as a reference. Figure 6.4 shows the mean deviation observed by each device expressed as a percentage of the reference value. This metric indicates how much the PCC values differ from that of the reference, but not necessarily they are incorrect. Measuring PCC allows us to observe the overall variation in CSI for a given time step, and this can be negatively affected by inconsistent packet delay. In the figure, it can be observed that the ESP32 and AX200 show the greatest deviation at 26% and 33% respectively. The IWL5300 shows the lowest deviation at 18%, which further indicates it produces CSI most similar to that of the USRP B210.

This is similar to our findings in Table 6.2, with the ESP32 and AX200 observing significantly more variable packet rates than the IWL5300 and CM4. However while the AX200 has the greatest deviation, the ESP32 observed higher variance in packet rates. This indicates that packet delay is not the only factor causing PCC values to deviate from that of the reference. We then considered the average features of the PCC to further understand each device's unique behaviour.

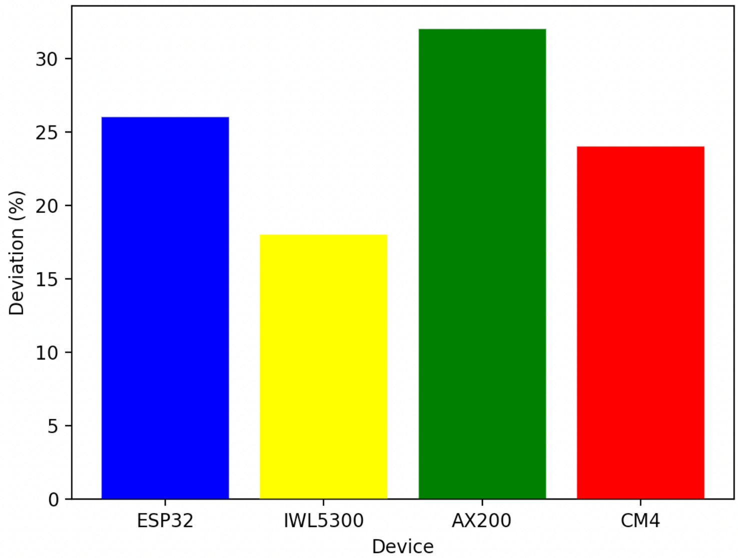


Figure 6.4: Mean percentage deviation from the USRP B210 (reference) PCC.

Table 6.3: Mean PCC features in Run 1.

	Min	Max	Kurtosis	Variance	Skewness
AX200	0.412	0.999	501.05	7.5e-4	-19.031
ESP32	0.716	0.991	28.694	5.8e-4	-2.885
IWL5300	0.596	0.994	21.971	18.8e-4	-19.825
USRP B210	0.413	0.999	18.733	22.1e-4	-20.840

Table 6.4: Mean PCC features in Run 2.

	Min	Max	Kurtosis	Variance	Skewness
AX200	0.510	0.999	380.196	5.5e-4	-17.080
CM4	0.313	0.994	58.396	66e-4	-5.980
ESP32	0.375	0.996	118.671	3.2e-4	-8.622
USRP B210	0.655	0.999	287.390	38.7e-4	-3.546

Table 6.5: Mean PCC features in Run 3.

	Min	Max	Kurtosis	Variance	Skewness
CM4	0.545	0.996	148.813	20.6e-4	-9.847
ESP32	0.779	0.997	151.789	1.9e-4	-8.739
IWL5300	0.460	0.995	42.085	27.2e-4	-4.574
USRP B210	0.412	0.999	36.034	36.1e-4	-4.860

Tables 6.3, 6.4, and 6.5 show the mean of PCC features gathered from each device. The max and min reflect the average link quality observed for each of the devices. The max value indicates the ceiling and shows the minimum channel variation observed in a time step. Both the USRP and AX200 observe max values of .999 in all runs, which indicates they are capable of near identical sequential CSI measurements.

Skewness and kurtosis indicate the shape of the distribution of PCC values. These values show no observable pattern between runs, indicating the wireless conditions have changed in some way between runs. This does not indicate any variation in device performance, but rather a constant factor of fading in the wireless channel over time. These fading changes occur over many hours and so historical data captured in an environment may not remain relevant for longer than a few days.

The most important metric we have observed in PCC is variance. This is based both on the values in the above tables, and the work performed in Section 5.1.3. Higher variance in PCC values is preferable as it indicates increased fidelity in measuring differences in the channel. The USRP B210 shows higher variance than the other devices on average. Similar, though generally lower, values are observed on the IWL5300. The CM4 also shows relatively high variance, though this behaviour was less consistent than that of the other devices. The lowest variance was observed on the ESP32

6.4.4 Human Motion Sensing Performance

Table 6.6: LSTM classification results.

	Precision	**Recall**	**F1 Score**	**Support**
No Movement	.81	.86	.83	6881
Movement	.85	.81	.83	6968

The LSTM model achieved 83% accuracy in classifying .5s periods of CSI data as containing movement or no movement. Table 6.6 shows the derived confusion metrics from the LSTM results. The recall for the "No Movement" class is higher than that of the "Movement" class indicating the model tends to favour False Negative predictions over False Positives. As seen in Figure 6.5, training and validation loss synchronised and relatively low. This indicates the model does not overfit in a significant way.

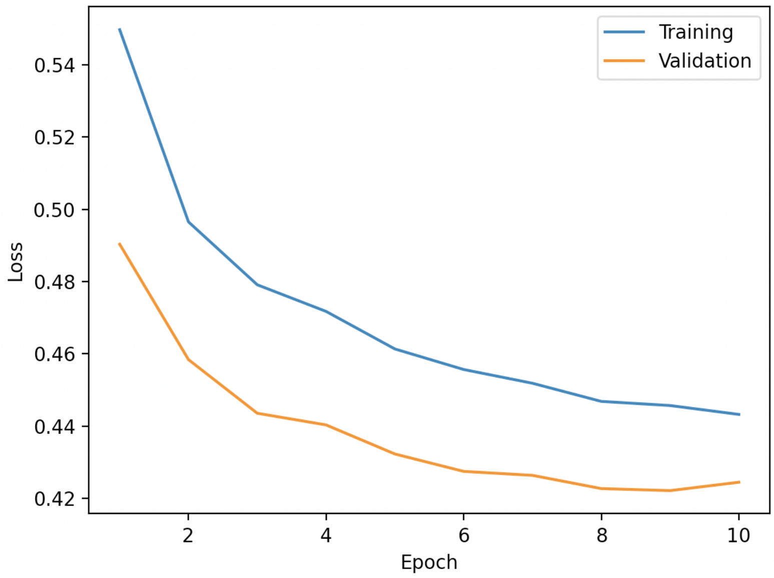
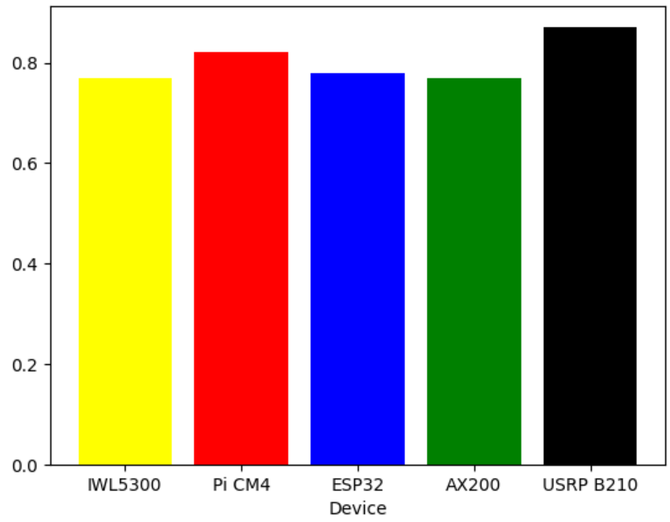


Figure 6.5: Loss measured at each epoch for each dataset while training the LSTM.

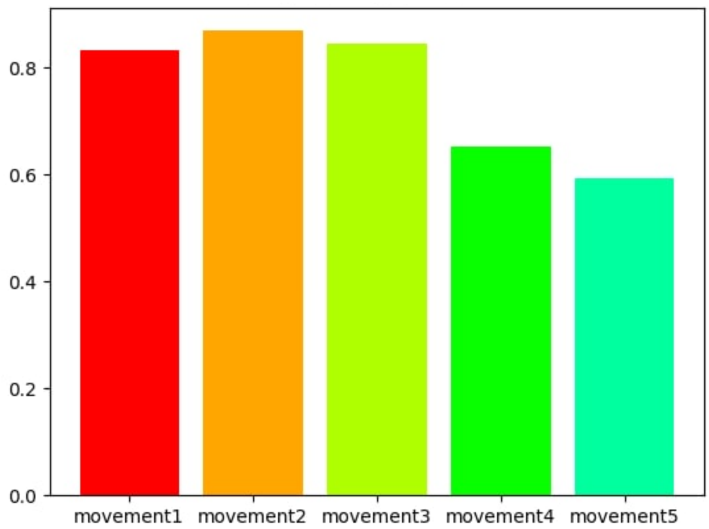
This model's performance is relatively weak for a real-time motion detection application. Its accuracy is low given the size of the windows used, and the false positive rate is far higher than the false negative rate. The chosen features have been shown to contain a discernible pattern underpinning the current classification performance. A stateful LSTM approach could be considered and may improve performance, given the statistical features observed and exploited in our work in Section 5.1. Despite its shortcomings, the model is fit for purpose to highlight the differences in sensing performance between the devices in a comparison using alternating train/test data subsets.

Performance Comparison via Feature-based Train/Test Subsets

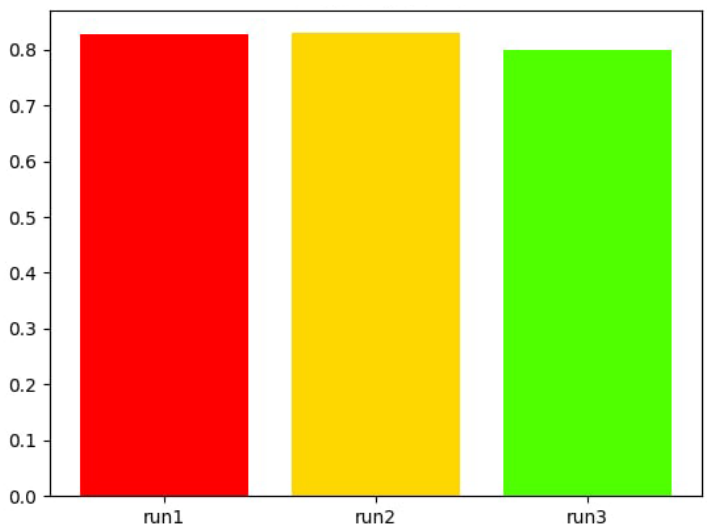
A feature-based performance comparison was performed to isolate specific subsets of the data to compare performance. The data was split into subsets, with one subset used as the test set and the remaining data acting as the training set. This stage is important to confirm no one subset provides critical training data without which the model would be crippled. It also highlights the individual performance of an isolated feature component, which is key in identifying device performance. We split the data based on the 3 most significant subsets: device, movement type (6.3.2), and run. When folded by device, this ensures the model is compatible with data from different devices. This is used to demonstrate the model can perform predictions using data for unseen devices, and to establish the performance differential for each device. When folded by movement type, this ensures we can identify the impact of the different types and lengths of source motion data. Finally, when folded by run, we compare the data from different time periods. This is important, as it isolates changes in the environment and wireless channel.



(a) Per Device



(b) Per Movement Type



(c) Per Run

Figure 6.6: LSTM Motion Classification performance by fold (F1 macro).

Figure 6.6(a) Device: The largest observed performance delta was 10% from the USRP B210 to the IWL5300. This indicates the factors in which we previously considered the IWL5300 to excel do not necessarily contribute in our coarse motion detection application. The USRP B210 performs best which is expected given its complete lack of AGC. The ESP32 came in second, indicating it produces CSI data which can be used for coarse sensing applications. The AX200, CM4 and IWL5300 are not far behind. This demonstrates device-nonspecific sensing applications are viable. Personalisation for device type may further improve performance, however a strong base can be achieved with a generic approach.

Figure 6.6(b) Movement Type: Small performance differences were observed between movement types 1 to 3. This is likely because they are similar in design and length. Types 4 and 5 show a large performance dropoff of 20% and 24% respectively, compared to type 2. 4 and 5 are both shorter, providing less data input windows. They are also performed at different locations to the other movements, including significantly variable distances relative to the sensor equipment. Our LSTM-based approach may be more susceptible to the distance and human disturbance falloff than our statistical approach in Section 5.1, as performance fell when tested using folds based on movement types performed at variable distance from the sensor equipment.

Figure 6.6(c) Run: Largely similar performance was observed across all 3 runs.

The comparison per device shows how the data produced by each CSI hardware aligns with the average wireless channel measured by the other devices. The LSTM model trained with these wireless channel measurements observes variations to determine whether motion can be detected. By measuring the channel with 4 devices simultaneously in each run, the model is trained on the common behaviour of the channel in which PCC decreases as human motion occurs. The best sensing performance was observed with the USRP B210 which supports its position as a baseline hardware producing high quality CSI. The COTS devices each fell within a 10% range of the USRP B210's performance. This performance deficit is notable, however each of the COTS devices achieve an F1 macro score within 5% of each other. This shows that viable CSI sensing application performance could be achieved using hardware at both the low and high end device price points.

6.5 Conclusion

In this work, we compared the wireless performance and CSI-based sensing performance of multiple CSI-capable hardware. We used a novel methodology with a homogeneous antenna and software implementation to compare 1-to-1 sensing performance in identical configurations for popular COTS hardware.

In wireless performance, the key observations were:

- The **USRP B210** and **IWL5300** produce similar CSI data, in both stability and variance. The IWL5300 performs well as a control and shares many characteristics of the CSI data from the USRP B210. The significant body of research making use of the IWL5300 is warranted given its MIMO capabilities and strong wireless performance.
- The highest COTS packet rate was observed on the **Raspberry Pi CM4**.
- The **ESP32** displayed relatively weak wireless performance.
- The **AX200** appears to perform relatively well, though its utility is limited by a lack of open source CSI tools.

All hardware showed a baseline level of performance sufficient for general usability in wireless applications. Weaker-performing devices are more likely to drop packets. Applications aiming to make use of these devices for wireless sensing should consider effective forms of interpolation to account for variable packet delays.

We also presented a novel LSTM-based approach for motion detection using .5s windows of CSI data from generic devices. The F1 macro score was consistent with the model accuracy, at 83%. This model showed relatively weak performance for the desired application, given the short window size. This approach should be reconsidered for real-time motion detection applications, likely considering the changing nature of wireless links. Additional feature engineering using statistical methods like those used in Section 5.1 should be considered. Despite this, the model was suitable for performance comparison across each of the devices.

Relatively consistent performance is achievable with each of the devices used in this comparison. No significant performance dropoff was observed between any of the COTS devices. This demonstrates that hardware-agnostic CSI motion detection is possible, and indicates the possibility for this behaviour to apply to other CSI sensing applications. In coarse device-free wireless sensing applications, the ESP32 shows comparable performance to the available COTS CSI hardware, and the USRP B210. This massively lowers the hardware entrypoint for CSI sensing development, to the price of a single ESP32 microcontroller. By increasing the accessibility of CSI sensing research, we provide a path towards ubiquitous, low-cost device-free wireless sensing.

Chapter 7

Real-World In-Home Study

In the previous chapter, we determined the ESP32 is a viable, cost-efficient hardware solution for CSI sensing. The comparison process indicated that the ESP32 displayed more inconsistent sample rates for the same packet streams than other devices, however this did not adversely affect its performance in sensing applications. When used to collect CSI over a serial bridge an ESP32 cannot function as a standalone IoT sensor. To achieve wireless sensing as proposed in the previous sections we cannot simply rely on the ambient CSI provided by nearby WiFi links. Instead, we must prepare a specialised deployment configuration to generate a continuous stream of CSI for each sensor. With the low cost of ESP32 modules in mind, several SoCs can be employed in a single sensing apparatus while maintaining a comparatively low unit cost. This approach affords the ability to put one CSI sensor in each room we aim to monitor in much the same way PIRs are deployed. Our new ESP32-based sensors are aimed at supplanting the sensing functionality offered by PIRs.

These new sensors are aimed at offering several improvements in data fidelity over the existing binary sensors deployed in the FitHomes. They provide fine-grained data, informing us not just of the subject's movements but also an analogue measure of movement relative to the sensor. We can also consider the style and actions within the movement, for both gesture and activity recognition, and potentially additional applications moving forward. To collect and label the data to investigate personalised sensing applications, we developed a prototype of the FitHomes CSI Platform. This systems facilitates the collection of a continuous CSI dataset of real-world in-home behaviour, soft-labelled using the FITsense platform and existing FitHomes sensors.

This chapter delivers the following contributions:

- **A data collection, processing, visualisation and storage platform for long-term CSI monitoring**: FitHomes CSI Platform provides a scalable pipeline for CSI data. This facilitates interaction with real-time data from the entire FitHomes network, after installing our deployable CSI sensors.
- A **functional prototype** of a **stand-alone networked ESP32-based CSI sensor**: A novel, fully implemented prototype sensor demonstrating the ability to collect CSI without a physical connection to the sensor unit.

7.1 Initial Sensor Prototype

This new CSI sensor is designed around the ESP32 microcontroller. As demonstrated in Section 6, the ESP32 is a competent hardware platform for CSI collection for wireless sensing applications. However, running code to collect and output CSI data over serial is just a single element of a fully integrated CSI sensor, much less one which could be deployed in the home. As no off-the-shelf integrated CSI sensor solutions exist, we designed and implemented a system using 3 ESP32 modules per sensing apparatus with a unit cost of £10 (at time of writing).

In most CSI collection setups there are 2 main components. A CSI collector, and a traffic generator. In some cases, the traffic generator may be an AP which the collector bounces frames off, while in others the generator may inject frames for the collector to passively monitor. Individual AP instances for each sensor would be excessive for dense deployments like ours. We use 3 ESP32-powered components in each sensor: a collector/forwarder pair, and a generator. A central Linux-based hub unit is also used to receive data from each sensor's forwarder.

The Generator naïvely injects 802.11n frames for the collector to receive. To improve efficiency and reduce wireless disruption, incomplete WiFi frames are sent. These frames initially follow the 802.11 beacon format before abruptly terminating. The ESP32 can still measure CSI for these incomplete frames, while most other WiFi devices do not parse them, thus reducing disruption. RAM usage on both the Generator and Collector is reduced, and overall stability is improved.

The Collector and Forwarder pair are soldered together to facilitate transmissions via a Serial Peripheral Interface (SPI) bus. Through this, the Collector can transmit the collected CSI to the Forwarder using Google's protobuf[1]. Each individual protobuf is roughly 160 bytes. These protobufs can then be verified for integrity on the Forwarder, before being sent to the hub for storage and processing. This method ensures the Collector does not drop incoming frames in the process of passing them along the chain, and allows the CSI generation/gathering to run on a different 802.11n channel to the forwarding operation. By default, it is impossible to use the 2.4G radio on a single ESP32 to both receive and forward the CSI data, hence the use of a secondary forwarder device.

It became clear during development that this design could also be implemented in a portable battery-powered capacity. Each component of this CSI collection system can be powered over USB (5V). This is only feasible for temporary operation and was used to support early tests to establish basic capabilities. To collect data and inform decisions we'd take in the deployment process for the FitHomes, we performed an initial evaluation with a portable prototype of this sensor.

7.1.1 Performance Evaluation

The purposes of this early evaluation were two-fold. Our first aim was to determine if the new sensor could be used with a CSI-based sensing application. Coarse human motion detection is a basic sensing application which we previously demonstrated could be performed with a variety of approaches, such as observing the PCC of sequential CSI frames (5.1, 6). By establishing the sensor could be used with a typical CSI sensing approach we demonstrate basic functionality and establish its fitness for purpose as a readily-deployable IoT sensor.

Second, we aimed to establish the consistent behavioural characteristics of CSI data which persist throughout multiple environments. In CSI sensing research there are common behaviours which can be observed across multiple works. While many aspects of

[1]**Protocol Buffers — Google Developers**: `https://developers.google.com/protocol-buffers`

the underlying data differ based on the environment and human subjects, the common behaviours tend to characterise the multipath propagation of the underlying WiFi signals. Tying these behaviours to specific sensing application performances can be difficult as the equipment in CSI research tends to be used with a static placement. A portable CSI sensor prototype was used as this allowed us to quickly alternate through varying sensor placements, configurations, and environments. To attempt this with other popular COTS CSI hardware like the IWL5300 would have been impractical as several of the components used in such setups are cumbersome and not easily battery-powered.

Experiment Design: The CSI sensor was operated in varying environments to establish motion sensing application performance in a variety of spaces. Two human subjects operated the portable sensor prototype in a large warehouse building. The warehouse contained uninterrupted open spaces and many enclosed rooms of varying construction. One person (the mover) placed the Generator and Collector/Forwarder sensor components in varying locations within each space. The mover then provided coarse motion stimulus by walking through the environment around the sensor components. The other person (the monitor) used a laptop to remotely monitor the sensor data output to determine when the subject had a measurable impact on the CSI, and establish cause and effect relationships between mover behaviours and CSI behaviour.

The CSI sensor produced data at a sample rate of 10Hz. The monitor viewed sensor data output through a basic real-time movement detection system. This was implemented by monitoring the PCC of CSI frames and applying a static threshold to classify each sample as containing coarse motion or no motion. The initial PCC threshold for the motion detection application was set to 0.70. The threshold was manually adjusted if the average PCC in the environment was higher or lower than the default such that the motion classifications were completely inaccurate. The monitor viewed the PCC values over time as the subject moved around the environment. Generally the PCC of CSI decreases as the subject moves throughout the environment. The monitor tracked when these changes were in response to the subject's motion. If these changes did not occur then the monitor identified and documented conditions of the environment and relative positioning of the Generator/Collector and mover.

These tests were relatively unstructured. They were repeated over 3 days using a variety of sensor configurations and environments. This approach was taken in an attempt to observe persistent high level sensing behaviours and highlight characteristics inherent to general CSI sensing usage.

Hardware Setup: We assembled a prototype focused on optimising the size, weight and power (SWaP) of the sensor (Figure 7.1). This approach uses two ESP32-powered TinyPicos as a Collector/Forwarder pair, powered with a single 1000mAh Lithium-Ion Polymer (LiPo) battery. These off-the-shelf boards were selected as they are smallest ESP32 boards which feature a fully featured ESP32 SoC, thus remaining representative of our planned in-home implementation. The boards were stacked atop each other which allowed for direct connection between the same GPIOs on each board. Pins 5, 18, 19, 23, 32, GND, and 5V were used. The connections facilitate communication over an SPI bus for CSI data transfer, as well as shared power and ground.

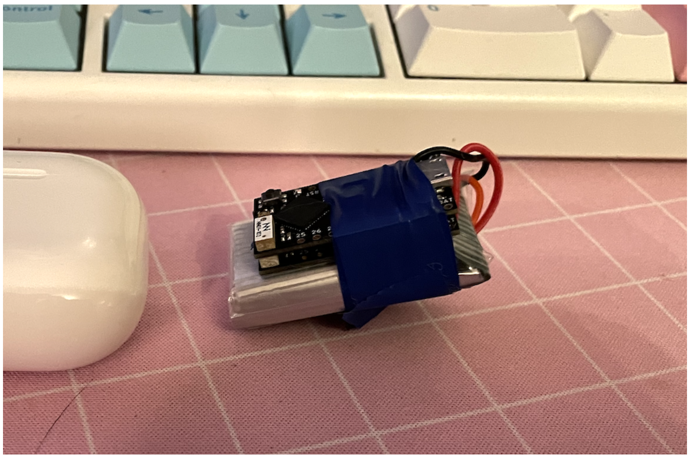


Figure 7.1: Assembled prototype of a portable CSI collector (sized at 4 x 2 x 2cm).

A single NodeMCU-style ESP32 dev board was used for the Generator. A Raspberry Pi 4 running a WiFi AP with hostapd was used as a central hub to receive data from the Forwarder. Both of these devices were powered over 5V USB from a 18650-based battery.

7.1.2 Results

The tests highlighted consistent behaviours in CSI data with relation to nearby human movement across varying environments and sensor placements. The key observations from our tests with the portable prototype are as follows:

The Generator and Collector have their own respective radii, in which human movement has a clearly observable impact on the PCC between frames. In a test performed in an open space within the warehouse, we noted an effective 1.8m radius around the Generator and Collector could consistently invoke a drop in PCC from 0.99 to 0.70. When stepping beyond 1.8m from either the Generator or Collector, the impact of human movement was reduced and a similar drop in PCC could not be invoked. A draft report from the FCC for an ESP32-WROOM-32E, whose PCB antenna is identical to that used in other ESP32 modules, includes a radiation plot showing an omnidirectional pattern with a slight tendency towards 30-150 degrees. In our tests, the sensors were for all intents and purposes fully omnidirectional. As such, when planning to deploy the sensors, we take note that the placement of the Generator and Collector does not necessarily set the boundaries for the sensing area, but rather where each radius originates.

Increasing distance between the devices also increased their local sensitivity. When the Generator and Collector were placed 50m apart in an open space with LOS, human movement in the middle of the two components did not significantly affect the PCC between frames. While the 1.8m radius could also be measured in this state, these movements also incurred negative PCC. This is likely due to signal strength in the link between the two devices dropping significantly with distance. As such, this indicates the need to maintain a relatively short distance between the devices, to maintain decent signal strength and to ensure the devices act as one sensor rather than two independent sensors.

Reflections from large surfaces are common. In several tests, a small room with concrete block walls was used. The Collector was placed in the room along with a subject, while the Generator was placed on the outside of the room against the door. It was expected that the movement of the subject in the room would be observable, however instead, consistently low and erratic PCC was observed. In this case, reflections from the door off the back wall behind the observers and their control station were occurring, capturing the movement of the observers instead. This functionality is undesired. The unintended signal paths as a result of the device placement should always be considered in sensor deployment, not just a hypothetical direct path.

7.2 FitHomes CSI Sensor Prototype

From the tests performed with the portable prototype, it was clear the ESP32-based sensor configuration generated good quality CSI for wireless sensing applications. It was determined we would continue development and testing with a similar design which could be deployed within the FitHomes installation in Alness.

7.2.1 In-home Prototype Design

For this prototype, a NodeMCU-style ESP32 devkit board was used. TinyPico boards are prohibitively expensive and their antennas underperform compared to the stock ESP32-WROOM-style PCB antenna. Figure 7.2 shows each component of the in-home prototype after assembly.

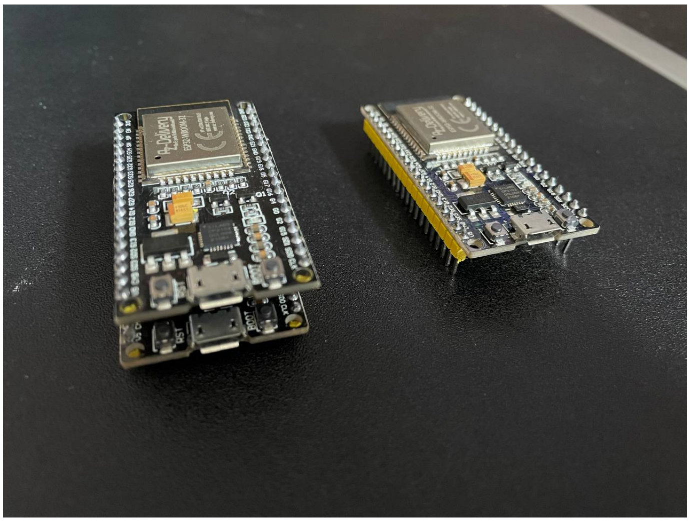


Figure 7.2: Collector/Forwarder (left) and Generator (right) used in the In-home Sensor Prototype (each sized at 5 x 2 x 3cm)

The power consumption of the prototype informed the physical deployment. Throughout development it was expected that continuous external power would be required. The Collector/Forwarder consumes a relatively consistent 117mA/575mW over microUSB. The Injector is much more erratic, alternating between consuming 109-167mA and 525-725mW. Battery power is not suitable at this stage as the device is planned for long-term continuous use. An off-the-shelf USB power adapter with a 1A output was used to provide 5V over microUSB to both the Collector/Forwarder and Generator.

This design was completed and implemented back in October 2021, however due to COVID-19 and other complicating factors, a single home deployment of these sensors was not completed until May 2022. During this period, the FitHomes CSI Platform was further planned and a prototype of the software implementation was developed.

7.2.2 Functional Assumptions

With experience working with CSI sensors both in an experimental and applied capacity, specific behaviours were expected in the CSI data. These assumptions informed decisions which affected the configuration for our initial deployment. The following assumptions were in place when this system was planned:

- **FITsense and the CSI sensors will complement each other**, providing additional perspective on the resident's movement and behaviour throughout the home. We should be able to align the time-series data from both the existing FitHomes sensors, the FITsense annotations for resident location and activity, and the CSI data for each time period.
- **One CSI sensor will be required for each room**. Initial experiments with the Movement Detection system explored in Section 5.1 indicated motion in adjacent rooms would not be detected. The difference in magnitude of the disturbances caused by human movement in the same and an adjacent room are observably different, and the system cannot consistently detect the latter. As the device cost for each sensor is low, it was determined that a single sensor would be used for each room.
- **Crosstalk between each sensor should be minimal**. Each device utilises a separate stream of 802.11n traffic on the same channel, separated by source MAC address for frames. Each Collector is also oriented such that the strongest incoming signals should be from their associated Generator. Typical forms of crosstalk where one device overhears from another should not be possible given the use of separate traffic streams.

We considered each of these assumptions in preparing our initial deployment. The FITsense and CSI sensor data is linked, as both systems generate time-series data in real-time. We aimed to identify whether these time-series could be compared for simultaneous observations which validated one another. Each room will be outfitted with a sensor. The data from sensors in neighbouring rooms will be compared to determine whether their observations overlap. Finally, the initial deployment was the largest we performed, with more sensors operated simultaneously than any of our previous tests. Each CSI data stream should be consistent and only contain channel measurements for that sensor. Despite this, we expected to observe a reduction in the overall packet rate due to the increased local traffic.

7.2.3 Performance Evaluation

Experiment Design: We performed the first test of our new sensor platform in one of the Dalmore FitHomes. 6 CSI sensors were deployed in a FitHome alongside the existing FITsense solution, with one in each room of the home. These sensors reported data to a Raspberry Pi 4 located in the Living Room. To deploy these prototype sensors we needed to supply power, produce an enclosure, and determine placements for each sensor. These factors are codependent and are the main issues which must be optimised to ease general deployment.

An off-the-shelf electrical cable enclosure was used as a case for both the Collector/Forwarder and Generator, ensuring a relatively homogeneous fascia with a similar aesthetic to the other FitHomes sensors. This assembly can be seen in Figure 7.3.

In ideal circumstances, the components would be installed in opposing corners of the room. This maximises the overlap between the sensing radii of the Collector and Generator. However availability of mains power limits where sensors can be placed. We used 3-meter micro USB cables to maximise our reach while minimising our physical impact on the resident's environment. Despite this, we had to make compromises on placement. Figure 7.4 shows the locations for each deployed CSI sensing apparatus. The Bedroom and Kitchen sensors could not be placed optimally. The desired placements were in opposing corners of these rooms, however there was no way to cleanly route cables for them. Each Tx/Rx pair is placed on opposing sides of the room, where opposing corners was infeasible.

(a) Collector/Forwarder (b) Generator

Figure 7.3: CSI sensor prototypes installed in one of the FitHomes in Alness.

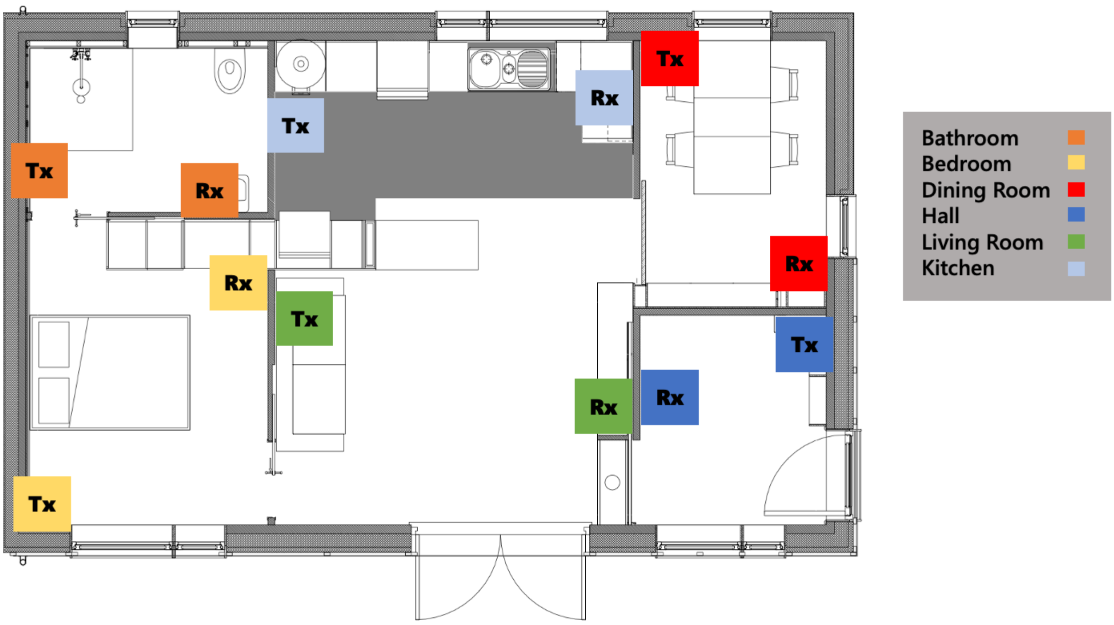


Figure 7.4: Annotated floorplan with CSI sensor Generator (Tx) and Collector (Rx) pairings.

The initial prototype hardware was tested internally for stability over several weeks.

Throughout the duration of the experiment all of the sensors remained functional and produced CSI data continuously.

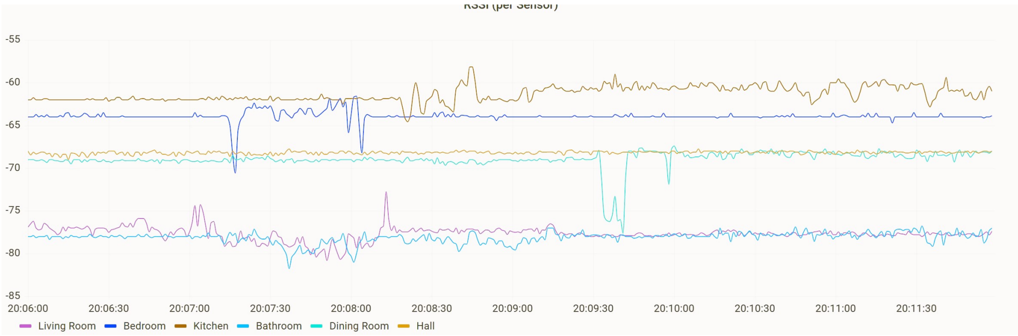
Experiment Setup: A Raspberry Pi 4 was used as a central hub, running the 64-bit April 2022 build of Raspberry Pi OS. Docker was used to manage the software stack, containing Redis server, the CSI server, and Grafana. The RedisTimeSeries module was used with Redis, to allow for timestamped values to be assigned to each sensor's key. The Python-based CSI server receives incoming CSI data from the sensors over UDP, where it is processed before being input into Redis. A local Redis instance is run with data batching for 30 minutes, before being sent to the Redis instance running on our central server alongside FITsense.

7.2.4 Results

CSI sensor data was forwarded to the central hub for storage. Features of the stored CSI data could then be viewed on a Grafana dashboard. While the raw CSI data is stored by this system, the CSI itself is too large to simultaneously display data from each of the sensors in a figure. For this work we use RSS and PCC as extracted features, to identify the capabilities of the CSI platform for sensing application development. In Figure 7.5(a), we see RSS (signal strength) data over time from each CSI sensor, plotted with a negative logarithmic scale on the Y axis. As the resident moves through the environment, they disturb the link between the Collector and Generator. This causes significant variation in the RSS which we can observe. Figure 7.5(b) shows PIR sensor activations captured during the same time period. These activations also occur as the resident moves through the environment, validating inference from the CSI sensors.

The behaviour of our CSI sensors can differ based on how the resident interacts with the environment. For instance, between 20:07:15 and 20:08:00 we observe a sustained PIR activation in the Bedroom. During this same period we can see increased variation in the RSS observed by the Bedroom CSI sensor. The RSS then returns to its previous average. This is a simple consistent behaviour where the resident's presence and movement cause a momentary increase in variation.

Granular changes in variation can also occur when the resident remains in a room and changes their position relative to the Collector or Generator. At 2:08:20, the resident



(a) Signal strength from CSI sensors (measured in dB)

(b) Sensor activations from FITsense PIRs

Figure 7.5: Data from FitHomes CSI Platform and FITsense gathered for the same time frame, showing the resident moving between the Living Room and Kitchen.

moves from the Living Room into the Kitchen, and a significant increase in variation occurs at the Kitchen CSI sensor. The PIR data confirms the resident is in the Kitchen for the remaining period. At 2:08:45, the variation on the Kitchen CSI sensor decreases, but still remains higher than observed before the resident entered the room. The increased variation the resident incurs is the result of changes in the multipath taken by the signals. If these changes persist, a new floor or ceiling in the extracted features is observed. This specific behaviour is consistent and has now been observed in both experimental data (Section 5.1.3) and real-world data from a FitHome.

Disturbances can also be observed in neighbouring rooms while the resident is mobile. Throughout the aforementioned period of movement in the Bedroom, the Bathroom sensor also shows increased variation. This could be misconstrued if the amount of variation which typically occurs during presence within the bathroom is not known. Most RF-based sensor systems use some form of environmental calibration. Our statistical approach to movement detection, defined in Section 5.1, uses a semi-automated calibration method which aims to combat this and has been shown to work in test environments.

These findings, though consistent throughout all of the data we collected from this deployment, could also be gleaned from just 5 minutes of data. We will use these to inform future areas of investigation as we aim to improve our sensor infrastructure. The key observation is the resident's influence on the RF environment and their ability to create observable disturbances in sensor data.

7.2.5 Discussion

The high level impressions from the initial deployment are as follows:

- **RSS is an underutilised metric** in device-free wireless sensing, which provides meaningful information on coarse signal disturbances around each sensor.
- Each sensor's data stream is separate, though **multiple sensors running simultaneously causes inconsistent timing** between CSI frames.
- Our **traffic generation method does not scale well** in environments with thin wall construction.

RSS is generally overlooked in device-free wireless sensing. It is clear RSS alone can be useful in occupancy monitoring, given the validated inference we performed using Figure 7.5. In this figure, we also observed the effect the resident's movement had on adjacent sensors. Due to the construction of the Fithomes, this effect is most prevalent in our deployed example and did not show up in any significant way during testing in controlled environments. During the period of movement in the Kitchen occurring from 20:08:20, we can see a small disturbance in the RSS of the Bathroom sensor, and then a large disturbance in the RSS of the Dining Room sensor. If the resident's location can be clearly identified, either through the CSI data itself or through a PIR, we could use these adjacent disturbances as a rudimentary form of localisation. For instance, we can assume the resident first enters the kitchen, moves towards the side closest to the Bathroom, and then the side closest to the Dining Room. This shows that the wireless signals do travel through the walls in the environment, and that this will have an effect on sensing applications. We see this as a potential use for the calibration steps we explored in our statistical approach in Section 5.1. The significance of the resident's impact on the RSS is much lower through walls, and so this must be quantified for sensing applications.

RSS is also an important metric for establishing the link quality of the Generator/Collector pair. PCC measurements are most useful for a link within an RSS "goldilocks" zone around -70dB. If the signal strength is too strong, such as the Kitchen sensor's -62dB, the ambient variation in PCC will be too low for meaningful insight into movement intensity. Similarly, if the RSS is too weak, such as the -78dB observed with the Bathroom sensor, the average PCC will be significantly lower than that of the other sensors. The average variation will also be too high to identify any disturbances incurred by the resident. This means we must additionally consider link quality when optimising sensor placement. The Generator and Collector should be placed such that the wireless disturbances caused by the resident will be observable within the RSS and PCC.

Figure 7.6 shows the RSS and PCC curves from the Kitchen sensor while the resident is not present. The RSS is consistent and confirms the resident is not in the room. However the PCC curve is very noisy and shows no real pattern. This behaviour is unlike our other PCC curves observed in the home or in our controlled environments (Section 5.1.3). This may be caused by the placement of the Kitchen sensor or poor wireless conditions in the Kitchen. For instance, the Kitchen has many large metal surfaces which can cause increased reflections over other rooms in the home. In its current state, the Kitchen sensor's PCC curve is not usable for sensing applications.

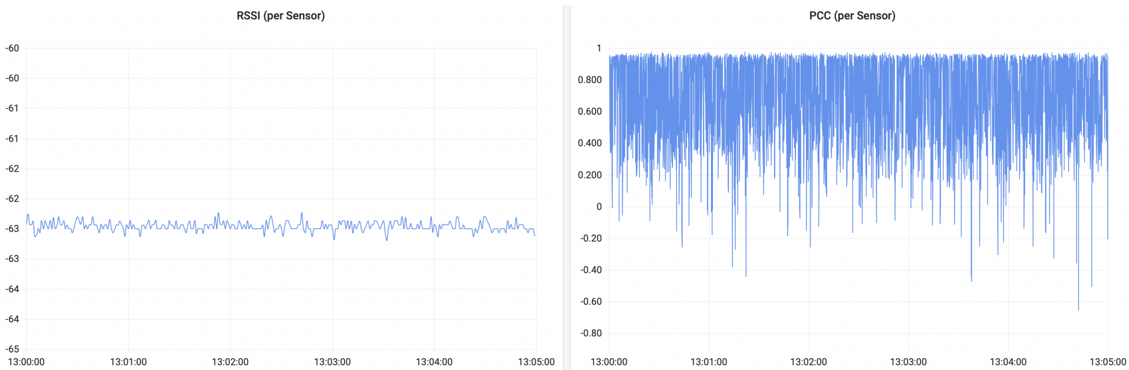


Figure 7.6: RSS and PCC curve from the Kitchen sensor while unoccupied.

Overall wireless performance was sufficient, but generally weak. The Generator devices aim to inject traffic at 100Hz when used individually. However, when multiple Generators are operated simultaneously we observe a drop in packet rate as measured at the Collector. Figure 7.7 shows this behaviour in action. The Hall sensor consistently achieved the highest packet rate, and the Living Room and Bedroom sensors generally performed worst. This behaviour manifested as a ranked system with the Generators producing more traffic the earlier they were turned on in sequence, due to CSMA CA/CD as previously discussed.

This is an obvious downside of operating all Generator/Collector pairs on the same wireless channel. To fully investigate this issue, we need to collect additional telemetry from the Generator and Collector, including Real Time Clock (RTC) synchronisation. This would allow us to measure the injected packet rate, the received packet rate, and the round trip time for each. If this confirmed our hypothesis then multiple simultaneous Generators would need to investigate alternative methods of traffic generation. A meshing implementation would be used as this would ensure all Generators could work together to prioritise traffic stability. Channel optimisation should also be considered.

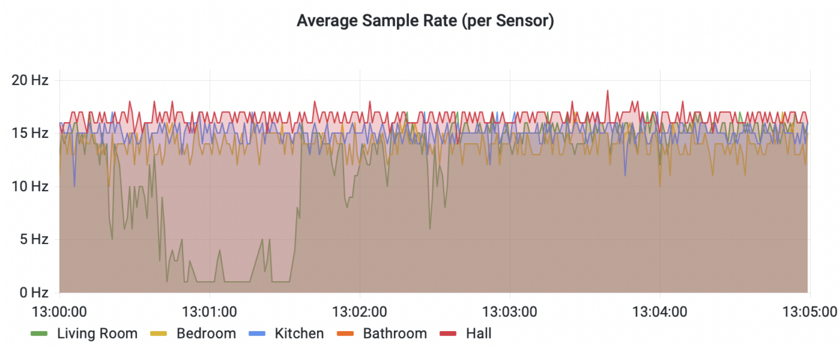


Figure 7.7: Sample rate observed from each CSI sensor.

Our initial data exploration confirmed we could match timestreams from both sensor solutions. We then performed an experiment to determine the effectiveness of our automated labelling approach, using FITsense data to label CSI for sensing application development.

7.3 FitHomes CSI Platform

While the data generated by the current complement of FitHomes sensors provides key insights for the FITsense system, the modality of the binary PIR sensors is an inherent limitation. With only the incidence and coarse duration of movement events to follow, it is unlikely the system can progress beyond assumption-based activity recognition. To expand on this system, additional sensing capabilities, and so modalities, are needed. For instance, we understand that the low level motions and patterns exhibited in the resident's movements can be be used to classify actions and gestures with CSI (Forbes, Massie & Craw 2020b). Established supervised model structures and implementations for CSI activity recognition exist, however their performance can typically be affected by two key factors: a lack of personalisation for the resident, and changes in the RF and physical environments (Palipana et al. 2018). Furthermore, they often struggle in the periods between actions because they are trained with CSI data from directed performances. Natural activity performances often blend into each another with a lack of distinction. These problems can be tackled by combining data from both the existing PIR sensors, and the new CSI sensors. In this work, we investigate the use of low level binary sensor data to label CSI data for generating targeted datasets for personalised sensing applications. Through FitHomes CSI Platform, we aim to perform long-term CSI data collection in FitHomes with our sensors, alongside the existing PIR sensors. To date, this is the first project deploying multiple CSI sensors in a smart home containing a long-term resident for continuous data capture.

7.3.1 System Design

The FitHomes CSI Platform expands upon the data pipeline originally developed for FITsense. Figure 7.8 outlines the system. The new CSI sensors developed in Section 7.1 will be deployed providing a CSI data source for the system. Data collected from the existing FitHomes sensors will continue to be sent to FITsense for resident activity and location annotation. This annotated sensor data can then be used to label periods of CSI sensor data, in the Data Fusion stage. This labelled CSI data can then be used in the development and realtime implementation of focused CSI sensing applications. It is expected that refined CSI applications will be developed to allow CSI-based sensors to replace the PIRs in future FitHomes installations.

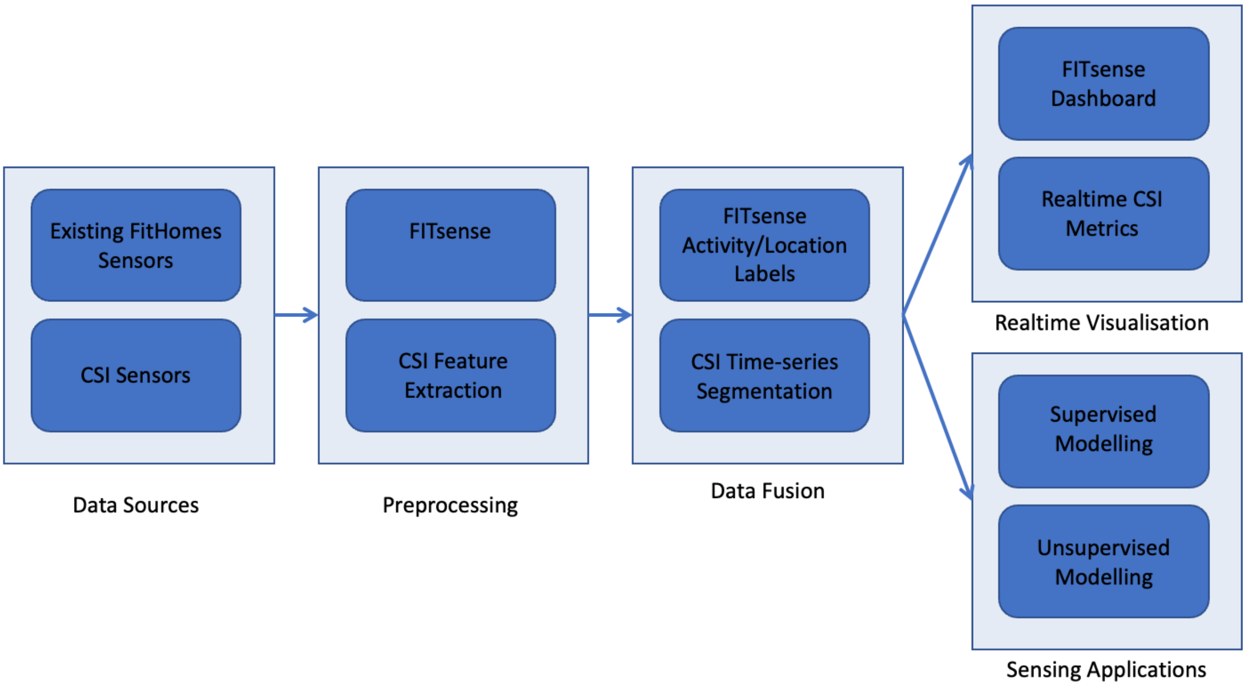


Figure 7.8: System Diagram for FitHomes CSI Platform

The platform was designed to produce a scalable data collection and processing solution for deploying many CSI sensors in a home. The coarse behavioural and location data generated through the FITsense system is a valuable source of supervision for CSI data modelling. This means we can generate large datasets over a long-term with automated soft labelling for CSI application development.

The initial proof-of-concept was prepared for the first CSI sensor installation. This was aimed at identifying whether the FITsense PIR data could be used to label CSI data to train a motion sensing model specific to the FitHomes environment.

7.4 Platform Evaluation

To evaluate the concept behind the FitHomes CSI Platform, we aimed to show our new CSI sensors could be used to achieve similar sensing performance to that of our previous comparison work. This will show the CSI sensing performance from our experiments in a controlled environment can be achieved in a real-world smart home using our new sensors. To accomplish this we make use of the new CSI sensors, the existing FitHomes sensors, and a modified version of our sensing application approach in Section 6. The main modification to this approach is to the CSI data annotation process, which is a barrier to research with real-world CSI data.

A key challenge in developing CSI sensing applications is in gathering labelled data, a process composed of segmentation and annotation. Our previous works used a manual form of segmentation: manually starting and stopping capture with a remote control. This approach was intrusive as it required the full attention of the subject, while also being expensive and relatively slow. This process was also not viable at scale. An automatic system for CSI segmentation and annotation was needed to operate at scale as sensors are deployed in many FitHomes. While a vision-based ground truth was previously used for annotation, they were considered too intrusive for use in the FitHomes project thus restricting their potential uptake. Instead the existing FitHomes PIR sensors were employed to both annotate and segment periods of movement in CSI data. This approach to gathering labelled data is both automated and ambient. This could support automatic dataset generation for targeted CSI sensing application development when employed over a long term.

In summary there are four key components we aim to demonstrate in this experiment:

- Our **new sensors produce viable CSI data** for use with device free wireless sensing applications;
- The **PIR data** generated by FitHomes and the FITsense system can be **used to annotate CSI data**;
- We can achieve **comparable sensing application performance** to that of **our previous work** in Section 6;
- A **proof of concept** for the **FitHomes CSI Platform**, which can aid **automated dataset generation** for CSI **sensing application development**.

7.4.1 Method

A motion detection system is used to evaluate the performance of the new CSI sensors. We collect CSI data from the new sensors in a real home to provide the input training data for the sensing application. We also collect data from co-located PIR sensors to segment and annotate periods of the resident's motion in CSI data. The motion detection system uses a supervised LSTM model trained using CSI data. This model's structure is based on our previous motion sensing work in Section 6, with alterations for the reduced sample rate of the new CSI sensors. If the model achieves similar classification performance to that of the previous experiments, then we can determine the new sensors produce data of similar utility to that collected in controlled experiments. This would show the new sensors were fit for purpose and could produce viable data for device-free wireless sensing applications. In this section, first the sensor installation and placement is detailed. Then the data collection and processing methodology for both sensor technologies is explained. Finally, the necessary modifications to the established motion detection LSTM model structure are covered.

The original plan for this evaluation was to test the new CSI sensors in multiple FitHomes, however due to the impact of the COVID-19 pandemic our installation capacity was limited. Our new CSI sensors were installed alongside the 6 PIRs in a resident's FitHome (3.2). These PIRs were used to establish a ground truth indicating the resident's motion state within a room. Each of the sensors are placed such that they cover the entire room.

The new CSI sensors measure the wireless link between their Generator and Collector sensing components, placed at opposing sides of the room. CSI data is polled at 4Hz to form the primary input for our motion detection model. PIR sensors are event-based and so only generate data points when motion is detected. They activate for a few seconds before reporting an inactive state once movement has concluded. These motion events are extracted from the shared timestream of new sensor data to generate input windows of CSI data which contain human movement. Windows of non movement were extracted by observing periods of time in which there was a constant period of no motion sensor events in each room. This data was then refined through our processing methodology.

Separate techniques were used to process data from both sensing technologies. Processing was performed with the goal of reducing the impact of interference and extract relevant features for our supervised model.

Undesirable variations can be observed in the PIR sensor performance which are inconsistent with the resident's actual behaviour. The undesirable behaviours observed in relation to this experiment are two-fold: latency, which is a by-product of the sensor's design, and invalid behavioural sequences, which are a result of flickering behaviour (3.3). Latency causes the sensor to remain active for short periods after the resident's movement has ended. Invalid behavioural sequences arise when the resident's movement through the home cannot be accurately tracked. The FITsense system significantly reduces the impact of these issues, and generates behavioural annotations which accurately inform us about movement in the home and where it occurs.

Raw CSI data is collected from each of the new sensors from which features are extracted. In this work RSS and PCC are used. These are used to populate a feature vector which forms the input for our LSTM model. RSS and PCC both reflect coarse variations in the wireless channel; consistent semi-permanent changes within which indicate human movement. General CSI behaviour and interference observed with the new sensors were both explored in 7.2.4. It was expected that data from both the kitchen and bathroom sensors would be of less utility for sensing applications, as the link quality is generally poorer and more inconsistent than the other rooms.

Various model optimisations were considered for use with data from the new CSI sensors. The initial deployed performance of the FitHomes CSI Platform limited the CSI sample rate to 4Hz. This mandated that we modify the model structure and allowed us to consider optimisations. The window size was adjusted to 4 samples, representing 1 second of sensor data. This also reduced the compute power required to pass data to the model and process predictions. We aimed to achieve similar performance to the original model structure, though this may be weaker than desired for a deployed sensing system.

7.4.2 Dataset

Data was gathered from both FITsense PIR sensors (annotations) and FitHomes CSI sensors (input).

FITsense PIR data is stored on a remote server using an InfluxDB model which can be queried for a specific time period. We isolated a 2 week period of FITsense data for the resident (7.2.3). Through this, we collated 165k data points from event-based sensors. Events were collected from each room PIR sensor. We identified 12,431 Movement events,

Table 7.1: Distribution of the annotated CSI data windows generated using real-world FitHome data (by room).

	Movement	**No Movement**
Bathroom	1839	2154
Bedroom	2710	5161
Dining Room	1050	5221
Hall	2158	5169
Kitchen	4296	5108
Living Room	14816	5233
Total	26869	28046

and 18,312 No Movement events. Each variable-length event is a continuous period of sensor data. As such, a single FITsense motion event could be used to generate multiple CSI data windows. They represent either motion or lack of motion following a motion period from the resident.

We then collated the same 2 week period of data from the FitHomes CSI Platform, which was operated onsite from a Raspberry Pi 4. FitHomes CSI Platform data is stored in RedisTimeSeries keys which can also be queried for a given time period. We then segmented CSI data gathered during a PIR-identified motion or No Movement event. 15 hours of CSI data was extracted to establish 54,915 1s windows from which a stratified train/test split could be derived.

Table 7.1 counts the CSI data windows in the dataset split by room and whether they contain movement. As a dataset generated in a real FitHome environment, the distribution is unbalanced with respect to each of the rooms. This is primarily because of the resident's general usage of the rooms in their home. We determined this to be a challenge of the approach in identifying the utility of the data generated through our automated annotation method. For instance, the dining room has the lowest number of movement windows as it is the least used room in the home. The number of no movement windows is slightly higher than movement, which was considered representative as a lack of movement is more common than movement.

7.4.3 Experiment Setup

Experiment Parameters: A bi-directional LSTM model was trained for 2 separate runs. In the first, the dataset contained annotated sensor data from all of the rooms in the home.

In the second run the data from the Bathroom and Kitchen sensors was excluded due to their observed weak link quality. A stratified train/test split of 70/30 was used in both experimental runs with a batch size of 128. The model was trained for 200 epochs using the Adam optimiser (learning rate=.0001) with a categorical cross-entropy loss function. This number of epochs was selected as the loss curve plateaued.

Sensor Configuration: We used Aeotec Multisensor 5 units for our FITsense PIRs. These were deployed with the resident's FitHome with placement detailed in Figure 3.2. Our CSI sensor hardware, implementation, and physical placement were previously outlined in this chapter.

Model Design: The model structure from our previous work uses a bi-directional LSTM with feature vectors containing RSS and PCC derived from CSI data. Due to the lower sample rate used for this evaluation, we reduced the window size to 4 samples which forms 1 second of CSI data. This reflects in a loss of fidelity over our previous model structure and so we used this opportunity to reduce the model complexity. As such, we use simplified model structure (Figure 7.9). We reduced the number of LSTM units from 300 to 100, which reduces the model complexity and training time. We also removed the first dense layer and subsequent dropout layer. This may cause the model to be more prone to overfitting. This new model structure is shallower than its predecessor which reduced training time per epoch from 5.6s to 3.4s.

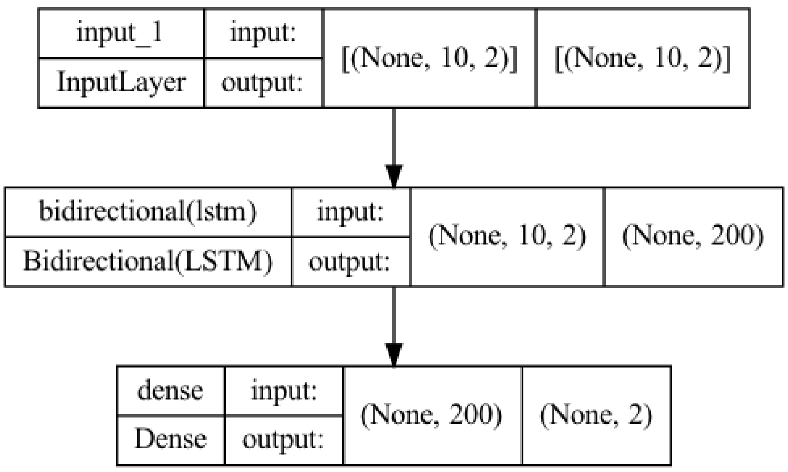


Figure 7.9: Model structure for our movement detection LSTM.

The model was composed using Keras with Tensorflow 2.8.0 on a 2021 Macbook Pro with M1 Pro, using Apple's conda channel.

7.4.4 Results

Table 7.2 shows the LSTM classification report from our first run, and Table 7.3 shows that of our second run. In our first run, the model achieved an accuracy of 74% in classifying 1s periods of real-world CSI sensor data as containing movement or no movement. In our second run, where data from the Bathroom and Kitchen sensors was dropped, the model accuracy increased to 84%. As our previous work achieved an overall accuracy of 83%, our first run shows notably weaker accuracy (-9%) while our second run shows a slight improvement (+1%). This highlights a negative performance impact when including the Bathroom and Kitchen sensor data.

In both runs the recall is higher for the No Movement class despite the varying class balance. This shows the model favours False Negative predictions over False Positives even when trained with more Movement instances. Precision is the same for both classes in the first run. In the second run, the precision of the movement class is much higher than no movement (+.14).

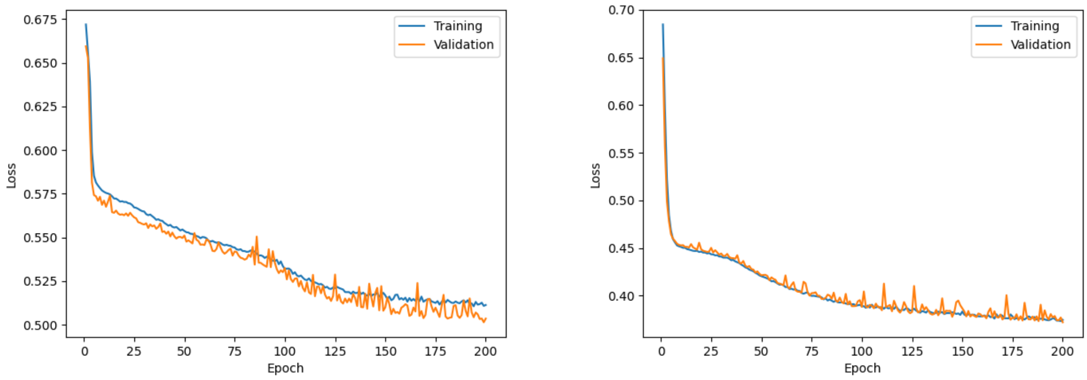
Table 7.2: LSTM classification results for first run, containing data from all rooms.

	Precision	**Recall**	**F1 Score**	**Support**
No Movement	.74	.76	.75	8414
Movement	.74	.72	.73	8061

Table 7.3: LSTM classification results for second run, excluding data from the Bathroom and Kitchen sensors.

	Precision	**Recall**	**F1 Score**	**Support**
No Movement	.78	.93	.85	6235
Movement	.92	.75	.82	6371

The model's loss curves are plotted in Figure 7.10(a) and (b). Both runs show higher variance in their loss curves than that of our previous model, with training and validation loss overlapping repeatedly. This could be attributed to the increased training time required to converge at 200 epochs. Performance gains were not observed on the previous model after exceeding 10 epochs, however this model took longer to converge. The general trend in loss still exists, with no significant variation between that of either set. The validation loss shows higher variation than training, indicating the learning rate should be reduced. A large drop in loss can be observed in the first 7 epochs which did not occur in the previous model. (b) shows a larger drop than (a) in these initial epochs. The curve is generally steeper in (a), however (b) reaches a lower final loss.



(a) All Rooms (b) Excluding Bathroom and Kitchen

Figure 7.10: Loss measured at each epoch for each dataset while training the LSTM.

7.4.5 Discussion

The changes in recall indicates the false negative rate of the model is significantly higher. Generally lower recall is observed on the movement class in both runs. This shows decreased sensitivity with the movement class, which is likely due to the use of the sensors in varying environments. While the previous work was performed in a single controlled environment, this experiment contends with the impact of multiple different room environments. Environmental changes (such as moving furniture or opening doors), resident behavioural variations, and ground truth noise are all contributing factors which can affect the overall quality of the data and balance of the model. Data was also gathered over a much longer period of time than our previous work; over 2 weeks as opposed to short experiments over 2 days.

Weak performance was observed when data from all of the sensors was used. This points to the poor link quality noted in the data analysis from the initial deployment of these sensors (7.2.4). The model trained on the dataset after excluding data from the Kitchen and Bathroom sensors performed similarly to our previous model implementation.

The features extracted for use with the LSTM model were shown to specifically behave differently on the Bathroom and Kitchen sensors. In Figure 7.6, the RSS is noted to be higher than that of the other sensors used in this experiment. However the PCC is much more variable, regularly dropping to below 0 despite the room being unoccupied. This indicates a wireless performance issue in this room causing unexpected behaviour in the CSI data. These deviations from expected behaviours negatively affect the model's performance as RSS and PCC form the sole input for the model.

The model's performance for the desired application is still relatively weak, despite the increase in window size and classification accuracy. However the model's tendency towards false negative classifications is preferred to the alternative in this application. Our automated segmentation and annotation approach relies on the highlighted windows actually containing human motion. False Positive classifications in this context would result in our model being given an incorrectly annotated data window. False Negative classifications result in less data windows in the resultant dataset, however they do not contribute to confusion within our model.

These results show our method of data collection and annotation for real-world CSI Data is viable and that FitHomes CSI Platform data can be of comparable utility to that collected in our controlled experiments.

7.4.6 Conclusion

Our movement detection model trained using our automated CSI annotation approach using PIR sensors achieved 74% and 84% classification accuracy in both experimental runs respectively. This shows the FitHomes CSI Platform can generate data of similar utility to that of our experiments in controlled environments, when the link quality for a given CSI is sufficient. This is a viable method to combine CSI data with the behavioural annotations generated by the FITsense system. We now aim to use the FitHomes CSI Platform in future work, for automated dataset generation for CSI sensing application development. Our first priority is to establish tailored metrics of wireless performance with deployed CSI sensors to assist in the diagnosis and optimisation of poor links. Our next goal is to improve our movement detection model, to increase detection accuracy such that it can be deployed as a real-time sensing application.

Automating the process of dataset generation could allow us to rapidly generate personalised models from new FitHome installations using both sensing modalities. This represents an important milestone given the environmental specificity of CSI data. The properties of the wireless channel change over a long term, both regardless of and in relation to changes in the environment and resident behaviour. Existing public datasets cannot easily capture or represent this behaviour. We consider using the platform to monitor trends in channel variations which may affect day-to-day sensing application performance. This work and our previous works acknowledge the clear semi-permanent changes in extracted CSI features which can be attributed to human motion and presence. In this work, we have demonstrated the FitHomes CSI Platform now enables us to identify changes to expected CSI behaviours over a long term, and establish how they vary with time and alterations in the wireless environment.

7.5 Summary

In this chapter, we establish the FitHomes CSI Platform to assist CSI sensing application research and development. This was achieved through novel contributions with our deployable fully-integrated CSI sensor solution and methodology for automated dataset generation.

Our novel hardware and software implementations with our new CSI sensors allow us to achieve a cohesive solution for deploying CSI sensing in real homes. We demonstrated our approach is functional, scalable, and cost-effective. No other solution for easily-deployable CSI sensing exists, and we are not aware of any case studies which achieve this goal. Our initial study has provided us with greater insights into the challenges faced in real-world deployment of CSI hardware than we have observed in our controlled experiments. Our ESP32-based sensing solution uses a single antenna for CSI collection, which could limit our approaches to coarse sensing applications such as movement detection and occupancy monitoring. To approach the sensing capabilities we observed in Section 2.4, we would benefit from using multiple antennas to view contrasting perspectives on the wireless environment. Despite this, we realise the potential for researching additional sensing applications with our single antenna CSI sensors, beyond those we investigated in previous chapters.

Our experiment in this chapter first demonstrated the efficacy of the CSI data produced using our new sensing hardware and software integration. We established the data produced by our sensors was usable, showing our success in CSI sensor deployment. This experiment also demonstrated the annotations provided by FITsense could be used to label CSI data for sensing application. Our results achieved using our modified movement detection LSTM with FitHomes CSI Platform data was capable of achieving a similar baseline performance to that of our previous experiments, which used data captured in controlled environments. We can use the behavioural annotations generated using FITsense and our original IoT sensors to automatically generate datasets from CSI sensor data for sensing application research and development. Using the wealth of data generated via the FitHomes and both sensing modalities, we can cheaply generate targeted personalised models or crowd source data for generalised models.

Chapter 8

Conclusions and Future Work

In this thesis and its constituent works novel solutions were developed to support the use of multi-modal sensor systems in smart homes for personalised health monitoring. Both hardware and software solutions for behaviour modelling were implemented to produce profiles of resident health risk factors and annotations for automated CSI dataset generation. In this final chapter we reflect on achievements made in relation to our research objectives, corresponding contributions, wider impact, and future research direction.

8.1 Achievement of Objectives

Each of the research objectives described in Section 1.3 were accompanied by deliverables. In this section we summarise how the objectives are accomplished through the contributions which emerged from the thesis.

Expand upon the FITsense system to improve scalability for new and retrofit home environments

The initial implementation of the FITsense system performs activity recognition and health risk profile generation using a rule-based system encoded with semantic rules. Environment-specific rules were used to filter out noise from the low-fidelity sensors. The links between adjacent rooms were used to validate candidate room transitions, as undesired sensor activations negatively affect localisation and behaviour monitoring performance. We developed an alternative method to reducing sensor noise which could be used without knowledge of environmental links. This approach uses a network graph model to estimate features of the environment and links between rooms based on how the resident moves throughout the home. Experimental results showed a representation of relative room relationships and relative distances could be estimated in FitHome environments.

Machine learning approaches for using smart home sensor data were investigated to identify methods of improving the activity recognition components of the FITsense system to support new environments. The primary challenge in targeting wide-scale deployment of these sensing applications is in gathering labelled data examples for training models. We established a methodology for collecting labelled smart home sensor data from residents, and worked with selected FitHomes residents to generate 2 datasets for activity recognition experiments. Our focus in this work was to optimise the utility of our available data, given the extensive cost in producing these labelled datasets. A novel representation for smart home sensor data sequences was used in experiments comparing baseline classifiers against various LSTM models for activity recognition. The results demonstrated LSTMs generally classified activity sequences more accurately with our representation using explicit timestamps to encode temporal dependencies. This work was presented at the IJCNN 2020 conference (Forbes, Massie, Craw, Fraser & Hamilton 2020).

Develop a homogeneous approach for handling data from all COTS CSI hardware platforms

The primary issue facing CSI research using COTS hardware is the use of bespoke data structures for each platform which require different implementations for parsing and processing. Most COTS CSI collection solutions provide software parsing implementations for use in MATLAB, which limits the accessibility of CSI sensing research. They also require data to be manually labelled with the device configuration for successful parsing. We designed and developed a software toolkit, CSIKit, as a single solution for each of these problems. CSIKit provides a Python toolkit for parsing, processing, and visualising data from all currently-available COTS CSI hardware platforms. CSI files from all popular formats can be parsed with automated detection for file type, hardware platorm and device configuration. Data is deserialised into a unified format which supports homogeneous implementations for processing, visualisation, and sensing application research. The framework adopts modern Python features and can be used with popular machine learning frameworks such as Tensorflow and PyTorch, making CSI approachable to data scientists and researchers without RF engineering experience.

Some CSI hardware platforms are provided with functionality to rescale generated CSI data to reverse the impact of Automatic Gain Control in the RF chain. We performed experiments to demonstrate the link between pre-AGC CSI and RSSI. A scaling factor can be generated to rescale CSI data, which reintroduces the pre-AGC curve. 3 device platforms were compared to a baseline hardware which did not employ Automatic Gain Control to establish the efficacy of this approach. Rescaled data from each COTS platform showed increased correlation against the baseline in all tests. This method applies to all COTS devices which provide both CSI and RSSI, and performs comparably to device-specific methods.

Show CSI is capable of higher fidelity sensing applications than basic IoT sensors

Two behavioural monitoring sensing applications were developed to demonstrate high-fidelity sensing applications could be delivered using CSI hardware. This shows CSI-based sensors can observe low-level behavioural insights with greater fidelity than ambient PIR sensors.

We developed a statistical approach to movement detection using CSI data. Our method uses relative thresholds based on the max values observed in a short window. An automated calibration system is also employed, which reduces the impact of common challenges faced with threshold-based methods. Experiments showed the system is capable of detecting movement of a subject within a room at distances between 0.5m and 5m. Our approach also performs well when a movement period is ongoing at the start of the data window, another limitation of standard methods. When compared against an off-the-shelf passive infrared motion sensor in 2 different environments, our CSI-based method showed an average of 32% greater accuracy at a precision of .1s. A preprint of this work has been published on arXiv (Forbes, Massie, Craw & Clare 2021).

A novel model structure was developed to perform activity recognition using CSI sensor data from the Raspberry Pi 4. This was the first CSI sensing application to be demonstrated on the Pi 4 hardware. We trained a model on data gathered from resident activity performances, and demonstrated an average classification accuracy of 92% for 11 unique activities. Only a marginal performance decrease was observed when the CSI data sample rate was reduced from 100Hz to 10Hz, showing the approach could be used with data gathered from ambient WiFi traffic. This work was presented as "WiFi-based Human Activity Recognition using Raspberry Pi" at the ICTAI 2020 conference (Forbes, Massie & Craw 2020b).

Identify differences in CSI collection performance across COTS hardware We performed experiments to compare COTS CSI collection hardware using CSIKit as a platform for homogeneous sensing application implementation. This allowed us to establish a performance comparison of the WiFi radios and CSI data quality for wireless sensing applications. The most consistent RF performance was observed with the Intel IWL5300, while the ESP32 and Raspberry Pi 4 were also shown to be viable hardware solutions for wireless sensing applications. Sensing performance was compared between devices using a single homogeneous motion detection application implementation. Our baseline hardware, the USRP B210, showed the highest average accuracy at 84% with the ESP32 coming in second at 81%. This demonstrates that the ESP32 is a viable low-cost hardware entrypoint for CSI research and wireless sensing application development.

Perform an in-home study using the selected best value CSI sensing hardware to demonstrate real-world sensing capabilities
The thesis was concluded with an in-home study performed using the ESP32 as a CSI sensing platform. To facilitate this study, we developed software and hardware solutions to support the use of multiple ESP32s as low-cost CSI sensors in an IoT network. A hardware design using 3 ESP32 microcontrollers was designed and prototyped for wireless CSI collection and delivery. A handheld variant of this sensor was developed and used to establish real-time performance characteristics which informed our hypothesis for real-world sensor deployment. Six ESP32-based sensors were deployed in an occupied FitHome alongside the existing sensor complement. Data analysis indicated the CSI sensors could be used for real-time observation of resident occupancy and movement intensity, with greater accuracy and precision over the existing approach using PIR sensors.

The FitHomes CSI Platform was proposed for the automated annotation of CSI data using behavioural labels provided by the FITsense system and existing low-fidelity sensors. This platform was tested in an experiment designed to identify whether data generated using our in-home ESP32-based CSI sensors could be used with FITsense behaviour labels to generate a dataset to train a movement detection LSTM. The movement detection application from our comparison work was repurposed for use with our new CSI sensor and trained using data gathered from a FitHome over a 2 week period. Experimental results showed comparable movement detection performance to that of our previous controlled experiments could be achieved using real-world data collected from a FitHome. The experimental models achieved accuracy values of 74% and 84% in classifying 1s periods of CSI sensor data as containing movement or no movement. This experiment demonstrates the initial capabilities of the platform, and shows our approach to automated CSI dataset generation using multi-modal sensors can work. This is the first real-world deployment of CSI sensors in long-term smart housing, and the only research test bed we are aware of performing entirely automated CSI data annotation.

8.2 Challenges

[these are challenges faced in the course of conducting the phd which altered our original objectives, and our approach to solving the problems we initially outlined] []

8.2.1 Sensor Data Quality

Maintaining sensor functionality is a critical aspect of the FITsense system, but it is not directly under our control. Instead, Albyn is responsible for sensor maintenance, which means that we can only report when sensors go offline, but they may not be addressed immediately. Unfortunately, there is no central fleet management system for the FIThomes sensors, so when a sensor goes offline, it simply stops reporting. To address this issue, we have developed a heuristic to determine if a commonly used sensor has not reported data within a reasonable timeframe, but it has not yet been implemented.

When sensors drop off the network, the logical sequence of actions that a resident takes in their home is disrupted, which prevents the FITsense system from functioning properly. This can occur for a variety of reasons, including battery or power issues, resident interference, and SmartThings hub changes. With all sensors functioning correctly, we can monitor resident behavior and activities to a certain extent, but missing sensors make it impossible to distinguish between missing activations and the resident performing a different sequence of actions.

To resolve this issue, Albyn needs to increase its commitment to maintaining data quality throughout its network. Installation technicians should maintain documentation that lists when sensors were installed, when batteries were last checked, and whenever any maintenance is performed. This documentation will also produce additional data that can support any long-term maintenance or sensor selection decisions.

8.2.2 FitHomes Hands-on

Prior to the impact of COVID-19, interaction with residents was already limited. However, we did have a hands-on day with residents near the start of the project, which allowed us to interact with them and receive feedback on the sensor installations and basic data dashboards.

Unfortunately, COVID-19 massively affected our plans for sensor prototyping and hands-on testing with residents. Originally, we had planned on installing basic prototype sensors in a small subset of homes and slowly cycling them out as we added new functionality. However, due to pandemic-related restrictions and safety concerns, we were unable to visit or prepare prototype sensors for the houses until late 2021, nearing the end of the project.

While we are grateful for the opportunity to perform the in-home study detailed in Section 7, this represented a stage of work we aimed to reach much earlier in the project. As a result, the thesis focused more heavily on CSI sensor data processing and application research than we had originally planned, given the limited ability to gather data from real-world environments. Despite these challenges, we were able to draw important insights and contribute to the field of CSI-based sensing, while recognizing the importance of engaging with end-users early and often in the design and development process.

8.2.3 CSI

While CSI has been proven in a research capacity, the focus of our work was on identifying approaches that could be employed for using CSI in commercial applications in real-world environments. We highlighted several issues with the approaches seen in research that would preclude their use in such settings. These issues fell under three main categories:

the devices used for CSI collection, the utilities for working with CSI collection devices and the data they produce, and the overall maturity of the field itself.

Firstly, the devices used for CSI collection in research settings are often bulky, expensive, and not designed for mass production. This makes them unsuitable for use in commercial applications that require low-cost, compact, and scalable devices. Secondly, the utilities for working with CSI collection devices and the data they produce are often complex, requiring specialized knowledge and skills. This limits their accessibility and usability in real-world settings. Finally, the overall maturity of the field itself is still relatively low, with many open research questions and a lack of established best practices. This limits the confidence that can be placed in the reliability and accuracy of CSI-based commercial applications.

8.2.4 Overfitting

Overfitting is a common problem in machine learning, and we aim to address this issue by leveraging the FitHomes CSI platform. Although general models can be built for the entire network to capture wave propagation characteristics inherent to the building style and shape, these models are not specific to individual residents' usage. To achieve specificity, we generate long-term datasets for each individual resident through their home. This approach allows us to train our models with a sufficiently high rate of dropout, which greatly reduces the risk of overfitting.

To monitor the rate of overfitting, we will continuously collect CSI data with soft-labels, as described in Section 7.3, and track standard model performance metrics over an extended period. If we observe a degradation in performance, we can build smaller datasets of sequential time periods to determine whether the issue is a creeping problem inherent to long-term data usage or a more recent occurrence.

8.2.5 Big Data in Sensing

In the context of sensing, Big Data refers to the large volumes of data generated by sensing devices, which can be challenging to store, process, and analyze. CSI data, in particular, presents unique storage challenges due to its high dimensionality and complex format.

One approach to addressing these challenges is to use databases that allow us to perform queries based on derived metrics, rather than storing the raw CSI data itself. By aggregating and processing the data to extract meaningful metrics, we can reduce the overall storage requirements and enable more efficient querying and analysis. This approach also allows us to easily compare data across multiple sensors and time periods, and to identify patterns and trends in the data.

In addition to database solutions, there are also emerging technologies and standards that aim to address the challenges of storing and analyzing Big Data in sensing applications. For example, the use of edge computing, where data is processed and analyzed locally on the sensing device, can reduce the amount of data that needs to be transmitted and stored. The adoption of standardized data formats and communication protocols can also improve interoperability and facilitate data sharing and analysis across different sensing applications.

8.3 Future Work

The FitHomes and FITsense projects are ongoing. In this section we state our future plans for expansion.

8.3.1 Sensor Deployability

In future work we seek to move entirely to a CSI-based sensing solution. Hardware designs are planned for focused implementations for new CSI sensors. The main plans are to merge the hardware apparatus for our sensing solution such that the collector and generator can be placed on the same board. A standardised enclosure will also be produced with mounting brackets to aid sensor placement. The Samsung SmartThings solution is currently being phased out and so we also consider the possibility of merging sensing modalities on the CSI sensors themselves to reduce maintenance and device cost.

As more robust CSI sensing applications are developed, we aim to deploy them on the edge to strengthen our approach to privacy-focused sensing. The links into the data wallet our industry partners are aiming for such that residents will be able to choose which data they wish to share with family, carers, and our research arm. The base functionality of our sensors should be provided regardless of this, and moving to the edge allows them to retain this functionality without our centralised cloud services.

8.3.2 FitHomes CSI Platform

In Section 7 we demonstrated our prototype of the FitHomes CSI Platform and performed an experiment to determine the utility of both its CSI data and annotations. Next, we aim to investigate sleep monitoring applications to establish new behavioural monitoring capabilities for our CSI sensing system. This will allow us to refine the system itself and our methodology to allow the platform to mature. By developing robust sensing applications using the platform we aim to attract research organisations to our test bed for long-term CSI sensing research. This will benefit FitHomes, FITsense, and ambient in-home health monitoring research as a whole.

8.3.3 Traffic Generation

Traffic generation is a crucial aspect of operating CSI sensing hardware. While WiFi sensing devices have the ability to both transmit and receive frames, their use of a single omnidirectional antenna makes the captured CSI for these frames unusable. Currently, our solution for traffic generation involves using a separate ESP32 to inject 802.11n frames. Although this solution worked well in a lab environment, when we attempted to deploy six injectors in a real-world environment, their performance was significantly reduced.

One of the challenges we face with injecting traffic on off-the-shelf hardware is that it is a black-box solution, and we have limited control over the injection functionality. There are options for injection using Nexmon-based tools such as the Nexus 5 jammer and the ASUS RT-AC86U scheduled frame injection tool, which we would like to investigate and evaluate their performance in congested scenarios. Additionally, we plan to evaluate the performance of frame injection using a software-defined radio, which would allow us to assess the effectiveness of this approach and explore various configuration options with simulated 802.11 frames.

By thoroughly investigating the available options for traffic generation, we aim to identify a solution that can be employed for commercial CSI sensing applications. This evaluation will enable us to select the most suitable traffic generation method for our specific requirements, considering factors such as performance, reliability, and scalability in real-world environments.

8.3.4 Generalisation

Most research on CSIapplications has traditionally been trained on data from a single environment. However, in other related fields such as accelerometer-based activity recognition, models are trained on data from a wide range of participants, allowing for good performance on unseen participants. This approach has not yet been widely used in CSI research due to the significant variability in WiFi propagation characteristics across different environments. However, with the use of the FitHomes CSI Platform, it is anticipated that large amounts of CSI data from diverse environments can be collected, providing an opportunity to leverage big data in CSI research. By training models on big data, which includes data from various activities, environments, and sensor inputs, inherent variability and patterns can be captured, leading to more generalizable models. Moreover, big data can facilitate robust model validation and evaluation, helping to identify potential issues and improve model generalization. Leveraging big data in CSI research has the potential to enhance accuracy and reliability in predicting activities in new, unseen data, making CSI applications more applicable for real-world scenarios.

8.3.5 Modern Hardware

The ESP32, although a relatively modern approach to CSI collection for sensing applications, still utilizes the 802.11n standard for WiFi communication. However, there have been recent developments in this space, as Espressif, the manufacturer of ESP32, has released engineering samples of their new ESP32 with 802.11ax, also known as WiFi 6.

The 802.11ax standard offers several advantages over 802.11n for traffic generation in CSI sensing applications. One notable improvement is better coexistence on the 2.4GHz frequency band, which is a commonly shared band across almost all legacy WiFi devices. With 802.11ax, there are mechanisms in place to mitigate interference from neighboring WiFi networks, resulting in more reliable and stable performance in congested environments.

Another advantage of 802.11ax is the ability to use wider bandwidths, including 20/40/80MHz on the 2.4GHz frequency band. This allows for higher data rates and increased capacity for traffic generation, which can be beneficial for CSI sensing applications that require high data throughput. Moreover, 802.11ax introduces more subcarriers and additional fidelity in the modulation and coding schemes, resulting in improved spectral efficiency and higher signal quality. This can lead to more accurate and reliable CSI measurements, which are crucial for precise sensing applications.

In addition to the improvements in the existing WiFi bands, 802.11ax also introduces a new WiFi band called WiFi 6E, which operates in the 6GHz frequency band. This additional band offers more available channels for traffic generation, which can help reduce interference from other devices operating in the overcrowded 2.4GHz and 5GHz bands. This can be particularly beneficial for CSI sensing applications that require interference-free environments for accurate measurements.

In conclusion, the new ESP32 with 802.11ax, or WiFi 6, offers several advantages for traffic generation in CSI sensing applications, including better coexistence on the 2.4GHz band, wider bandwidth options, increased subcarriers and fidelity, and the potential for reduced interference in the 6GHz band. These advancements in WiFi technology can potentially enhance the performance and reliability of traffic generation for CSI sensing applications, opening up new possibilities for commercial deployments in real-world environments.

8.4 Conclusions

In this work multi-modal sensors were employed to ambiently detect resident behaviour in smart homes for personalised health monitoring. The existing capabilities and limitations of simple IoT sensors for behaviour monitoring were detailed and informed hardware selection for our next generation sensing solution. CSI was investigated as an ambient rich sensor which expanded upon the fidelity of behavioural monitoring capabilities possible in real-world smart housing. A prototype solution for reducing cost and work behind CSI research was tested with real-world smart home residents. In future work, personalised models for residents could be explored using the FitHomes CSI platform to accelerate the development cycle. The development of robust CSI sensing systems rely on a strong understanding of the effects of the environment and resident on the wireless channel. The large-scale research test bed provided by FitHomes is a unique opportunity to observe CSI behaviour in both similar and varying environments. A combination of supervised and unsupervised techniques could also be employed to fully exploit the behavioural annotations provided through the FITsense system. This thesis serves as a demonstration of end-to-end smart home sensing research from sensor hardware development to long-term dataset generation and analytics.

Readers should take away the following key points from this work:

- Health monitoring systems employing basic sensors can only deliver sensing applications with basic capabilities. Binary IoT sensors can be used for inference-based sensing, but they cannot produce rich data for use with complex sensing applications. Sensors capable of producing rich data, such as cameras, wearables, or RF sensors, can allow for increased sensing capabilities and fidelity.

- An ambient approach to health monitoring in smart housing encourages uptake and reduces performative resident behaviour. To achieve large-scale deployment of smart home health monitoring systems, residents must not feel like they are constantly under surveillance. The primary issue affecting resident reception to these systems is not the concept of monitoring itself, but rather their perception of the apparatus. Residents are generally accepting of the technology when sensors which blend into the environment are used.

- While rich sensing technologies capable of advanced sensing applications do exist, they are often prohibitively expensive and too unwieldy for commercial deployment. Advances in sensor technology are only viable for real-world use at scale if the hardware is economical and usage is flexible. If we want rich sensing to be as widely-available as binary sensors, they must be just as low-cost and easily deployable.

While this work has not extensively focused on the implications of this sensing technology on modern privacy, we leave the reader with a final thought: As we continue to expand on the sensing capabilities achievable with cheaper and smaller hardware, we should also consider how this may influence modern ubiquitous surveillance. Cameras can be found everywhere in public spaces, however their reliance on visible light (or Infrared) preclude their omniscience. RF-based sensors also have limitations which inherently affect their sensing capabilities, however they can function without line of sight to their subjects. As the fidelity achievable with low-cost RF sensors increases, we must be wary and aim to preserve the level of privacy currently offered with our current approach to coarse to WiFi CSI sensing.

Bibliography

Adib, F., Kabelac, Z. & Katabi, D. (2015). Multi-person localization via RF body reflections, *Proc.s of the 12th USENIX Conference on Networked Systems Design and Implementation* pp. 279–292.

Adib, F. & Katabi, D. (2013). See through walls with WiFi!, *ACM SIGCOMM Computer Communication Review* **43**(4): 75–86.

Aipperspach, R., Cohen, E. & Canny, J. (2010). Modeling human behaviour from simple sensors in the home, *Proceedings Of The IEEE Conference On Pervasive Computing* **6030**(783): 62–79.

Arshad, S., Feng, C., Liu, Y., Hu, Y., Yu, R., Zhou, S. & Li, H. (2017). Wi-chase: A wifi based human activity recognition system for sensorless environments, *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, IEEE, pp. 1–6.

Ayena, J. C., Zaibi, H., Otis, M. J. & Menelas, B. A. J. (2016). Home-based risk of falling assessment test using a closed-loop balance model, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **24**(12): 1351–1362.

Baldi, P. & Sadowski, P. J. (2013). Understanding dropout, *Advances in neural information processing systems* **26**.

Baraniuk, R. G. (2011). More is less: Signal processing and the data deluge, *Science* **331**(6018): 717–719.

Bardwell, J. (2002). Converting signal strength percentage to dbm values, *WildPackets, Inc* pp. 1–12.

Bourobou, S. T. M. & Yoo, Y. (2015). User activity recognition in smart homes using pattern clustering applied to temporal ANN algorithm, *Sensors (Switzerland)* **15**(5): 11953–11971.

Castaneda, D., Esparza, A., Ghamari, M., Soltanpur, C. & Nazeran, H. (2018). A review on wearable photoplethysmography sensors and their potential future applications in health care, *International journal of biosensors & bioelectronics* **4**(4): 195.

Chen, Y., Zhong, K., Zhang, J., Sun, Q. & Zhao, X. (2016). Lstm networks for mobile human activity recognition, *2016 International conference on artificial intelligence: technologies and applications*, Atlantis Press, pp. 50–53.

Cook, D., Schmitter-Edgecombe, M., Crandall, A., Sanders, C. & Thomas, B. (2009). Collecting and disseminating smart home sensor data in the casas project, *Proceedings of the CHI workshop on developing shared home behavior datasets to advance HCI and ubiquitous computing research*, pp. 1–7.

Cyp (2019). *CYW43455: Single-Chip 5G WiFi IEEE 802.11n/ac MAC/ Baseband/ Radio with Integrated Bluetooth 5.0.* Rev. *O.

Dahmen, J., Thomas, B. L., Cook, D. J. & Wang, X. (2017). Activity learning as a foundation for security monitoring in smart homes, *Sensors (Switzerland)* **17**(4): 1–17.

Dodier, R. H., Henze, G. P., Tiller, D. K. & Guo, X. (2006). Building occupancy detection through sensor belief networks, *Energy and buildings* **38**(9): 1033–1043.

European Parliament & Council of the European Union (2016). Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation). Official Journal of the European Union, L 119/1.
URL: *https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679*

Forbes, G. (2020). CSIKit: Python CSI processing and visualisation tools for commercial off-the-shelf hardware.

URL: *https://github.com/Gi-z/CSIKit*

Forbes, G., Massie, S. & Craw, S. (2020a). Fall prediction using behavioural modelling from sensor data in smart homes, *Artificial Intelligence Review* **53**(2): 1071–1091.

Forbes, G., Massie, S. & Craw, S. (2020b). Wifi-based human activity recognition using raspberry pi, *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, pp. 722–730.

Forbes, G., Massie, S., Craw, S. & Clare, C. (2021). Motion detection using csi from raspberry pi 4, *arXiv preprint arXiv:2111.09091* .

Forbes, G., Massie, S., Craw, S., Fraser, L. & Hamilton, G. (2020). Representing temporal dependencies in smart home activity recognition for health monitoring, *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, pp. 1–8.

Fritz, S. & Lusardi, M. (2009). Walking speed: The sixth vital sign, *Journal of Geriatric Physical Therapy* **32**(2): 1–5.

Gaddam, A., Mukhopadhyay, S. & Gupta, G. S. (2010). Smart home for elderly using optimized number of wireless sensors, *Advances in Wireless Sensors and Sensor Networks* pp. 307–328.

Gao, Z., Gao, Y., Wang, S., Li, D., Xu, Y. & Jiang, H. (2019). CRISLoc: Reconstructable CSI Fingerprintingfor Indoor Smartphone Localization, pp. 1–18.

URL: *http://arxiv.org/abs/1910.06895*

Ghayvat, H., Mukhopadhyay, S. & Gui, X. (2015). Addressing interference issues in a wsn based smart home for ambient assisted living, *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, IEEE, pp. 1661–1666.

Grifantini, K. (2020). Tracking sleep to optimize health, *IEEE pulse* **11**(5): 12–16.

Gringoli, F., Schulz, M., Link, J. & Hollick, M. (2019). Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets, *Proc.s of the 13th Intl.*

Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, WiNTECH '19, p. 21–28.

Halperin, D., Hu, W., Sheth, A. & Wetherall, D. (2011a). Tool release: Gathering 802.11n traces with channel state information, *ACM SIGCOMM Computer Communication Review* **41**(1): 53.

Halperin, D., Hu, W., Sheth, A. & Wetherall, D. (2011b). Tool release: Gathering 802.11n traces with channel state information, *ACM SIGCOMM CCR* **41**(1): 53.

Hernandez, S. M. & Bulut, E. (2020). Lightweight and Standalone IoT Based WiFi Sensing for Active Repositioning and Mobility, *21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM) (WoWMoM 2020)*, Cork, Ireland.

Hernandez, S. M. & Bulut, E. (2022). Online Stream Sampling for Low-Memory On-Device Edge Training for WiFi Sensing, *Proceedings of the 4th ACM Workshop on Wireless Security and Machine Learning (WiseML 2022)*, Austin, Texas, United States.

Hikvision USA Inc. (2013). Recommended bitrates, *Technical report*, Hikvision, 908 Canada Court, Industry, CA 91748.
URL: *https://us.hikvision.com/sites/default/files/tb/recommended_bitrates_0.pdf*

Hinton, G. E. (2009). Deep belief networks, *Scholarpedia* **4**(5): 5947.

Hyväri, S., Elo, S., Kukkohovi, S. & Lotvonen, S. (2022). Utilizing activity sensors to identify the behavioural activity patterns of elderly home care clients, *Disability and Rehabilitation: Assistive Technology* pp. 1–10.

Ismail, H., Radwan, I., Suominen, H. & Waddington, G. (2017). Human postural sway estimation from noisy observations, *2017 IEEE 12th International Conference on Automatic Face & Gesture Recognition* pp. 454–461.

Jefferis, B. J., Parsons, T. J., Sartini, C., Ash, S., Lennon, L. T., Papacosta, O., Morris, R. W., Wannamethee, S. G., Lee, I. M. & Whincup, P. H. (2019). Objectively measured physical activity, sedentary behaviour and all-cause mortality in older men: Does volume of activity matter more than pattern of accumulation?, *British Journal of Sports Medicine* **53**: 1013–1020.

Jiang, Z. (2022). Supported Hardware - PicoScenes documentation.
URL: *https://web.archive.org/web/20220710193000/https://ps.zpj.io/hardware.html*

Jiang, Z., Luan, T. H., Ren, X., Lv, D., Hao, H., Wang, J., Zhao, K., Xi, W., Xu, Y. & Li, R. (2022). Eliminating the barriers: Demystifying wi-fi baseband design and introducing the picoscenes wi-fi sensing platform, *IEEE Internet of Things Journal* **9**(6): 4476–4496.

Juarez, J. M., Ochotorena, J. M., Campos, M. & Combi, C. (2015). Spatiotemporal data visualisation for homecare monitoring of elderly people, *Artificial Intelligence In Medicine* **65**(2): 97–111.

Kearns, W. D., Fozard, J. L., Becker, M., Jasiewicz, J. M., Craighead, J. D., Holtsclaw, L. & Dion, C. (2012). Path tortuosity in everyday movements of elderly persons increases fall prediction beyond knowledge of fall history, medication use, and standardized gait and balance assessments, *Journal of the American Medical Directors Association* **13**(7): 665–e7.

Khamis, A., Chou, C. T., Kusy, B. & Hu, W. (2018). Cardiofi: Enabling heart rate monitoring on unmodified COTS WiFi devices, *ACM Intl. Conference Proc. Series* pp. 97–106.

Kim, E., Helal, S. & Cook, D. (2010). Human activity recognition and pattern discovery, *IEEE Pervasive Computing* **9**(1): 1–14.

Kindt, P. H., Turetta, C., Demrozi, F., Masrur, A., Pravadelli, G. & Chakraborty, S. (2022). Wifieye – seeing over wifi made accessible.
URL: *https://arxiv.org/abs/2204.01830*

Lange, P. W., Lamanna, M., Watson, R. & Maier, A. B. (2019). Undiagnosed delirium is frequent and difficult to predict: Results from a prevalence survey of a tertiary hospital, *Journal of Clinical Nursing* **28**: 2537–2542.

Lee, H., Ahn, C. R., Choi, N., Kim, T. & Lee, H. (2019). The effects of housing environments on the performance of activity-recognition systems using wi-fi channel state information: An exploratory study, *Sensors (Switzerland)* **19**(5).

Linksys (2019). Linksys aware, Online.
URL: *https://www.linksys.com/us/linksys-aware/*

Liu, J., Chen, Y., Wang, Y., Chen, X., Cheng, J. & Yang, J. (2018). Monitoring Vital Signs and Postures during Sleep Using WiFi Signals, *IEEE Internet of Things Journal* **5**(3): 2071–2084.

Liu, J., Wang, Y., Chen, Y., Yang, J., Chen, X. & Cheng, J. (2015). Tracking vital signs during sleep leveraging off-the-shelf WiFi, *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing - MobiHoc '15*, pp. 267–276.

Liu, S., Zhao, Y., Xue, F., Chen, B. & Chen, X. (2019). Deepcount: Crowd counting with wifi via deep learning, *arXiv preprint arXiv:1903.05316* .

Ma, Y., Zhou, G. & Wang, S. (2019). WiFi sensing with channel state information: A survey, *ACM Computing Surveys* **52**(3).

Marinčić, A., Kerner, A. & Šimunić, D. (2016). Interoperability of iot wireless technologies in ambient assisted living environments, *2016 Wireless Telecommunications Symposium (WTS)*, IEEE, pp. 1–6.

McKeever, S., Ye, J., Coyle, L., Bleakley, C. & Dobson, S. (2010). Activity recognition using temporal evidence theory, *Journal of Ambient Intelligence and Smart Environments* **2**(3): 253–269.

Michikawa, T., Nishiwaki, Y., Takebayashi, T. & Toyama, Y. (2009). One-leg standing test for elderly populations, *Journal of Orthopaedic Science* **14**(5): 675–685.

Montero-Odasso, M. & et al (2005). Gait velocity as a single predictor of adverse events in healthy seniors aged 75 years and older, *Journal of Gerontology* **60A**: 1304–1309.

Moretti, G., Marsland, S., Basu, D., Sen Gupta, G., Group, M. et al. (2013). Towards a monitoring smart home for the elderly: One experience in retrofitting a sensor network into an existing home, *Journal of Ambient Intelligence and Smart Environments* **5**(6): 639–656.

Nachman, L., Huang, J., Shahabdeen, J., Adler, R. & Kling, R. (2008). Imote2: Serious computation at the edge, *2008 international wireless communications and mobile computing conference*, IEEE, pp. 1118–1123.

Narui, H., Shu, R., Gonzalez-Navarro, F. F. & Ermon, S. (2020). *Domain Adaptation for Human Fall Detection Using WiFi Channel State Information*, Springer International Publishing, Cham, pp. 177–181.

NHS (2017). The incidence and costs of inpatient falls in hospitals.
URL: *https://improvement.nhs.uk/uploads/documents/Falls_report_July2017.v2.pdf*

Ogawa, M., Suzuki, R., Otake, S., Izutsu, T., Iwaya, T. & Togawa, T. (2002). Long term remote behavioral monitoring of elderly by using sensors installed in ordinary houses, *2nd Annual International IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology - Proceedings* pp. 322–325.

Ordóñez, F. J. & Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition, *Sensors (Switzerland)* **16**(1).

Origin Wireless (2019). Official announcement and press release.
URL: *https://prtimes.jp/main/html/rd/p/000000003.000040517.html*

Palipana, S., Rojas, D., Agrawal, P. & Pesch, D. (2018). Falldefi: Ubiquitous fall detection using commodity wi-fi devices, *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* **1**(4): 1–25.

Park, Y. & Ho, J. C. (2019). Tackling overfitting in boosting for noisy healthcare data, *IEEE Transactions on Knowledge and Data Engineering* **33**(7): 2995–3006.

Parliament, U. (2018). Data protection act 2018. Chapter 12.
URL: *http://www.legislation.gov.uk/ukpga/2018/12/contents/enacted*

Phillips, L. J., DeRoche, C. B., Rantz, M., Alexander, G. L., Skubic, M., Despins, L., Abbott, C., Harris, B. H., Galambos, C. & Koopman, R. J. (2017). Using embedded sensors in independent living to predict gait changes and falls, *Western Journal of Nursing Research* **39**(1): 78–94.

Rockinson, R. J. (2008). *Activity Recognition with End-User Sensor Installation in the Home*, Master's thesis, Massachusetts Institute of Technology.

Seyfioğlu, M. S., Gürbüz, S. Z., Özbayoğlu, A. M. & Yüksel, M. (2017). Deep learning of micro-doppler features for aided and unaided gait recognition, *Radar Conference*, IEEE, pp. 1125–1130.

Shah, S. A. & Fioranelli, F. (2019). Rf sensing technologies for assisted daily living in healthcare: A comprehensive review, *IEEE Aerospace and Electronic Systems Magazine* **34**(11): 26–44.

Smith, G., Redfern, O., Pimentel, M., Gerry, S., Collins, G., Malycha, J., Prytherch, D., Schmidt, P. & Watkinson, P. (2019). The national early warning score 2 (news2), *Clinical Medicine* **19**(3).

Stone, E. E. & Skubic, M. (2013). Unobtrusive, continuous, in-home gait measurement using the microsoft kinect, *IEEE Transactions on Biomedical Engineering* **60**(10): 2925–2932.

Sucerquia, A., López, J. D. & Vargas-Bonilla, J. F. (2017). SisFall: A fall and movement dataset, *Sensors (Switzerland)* **17**(1).

Suryadevara, N. K. & Mukhopadhyay, S. C. (2012). Wireless sensor network based home monitoring system for wellness determination of elderly, *IEEE Sensors Journal* **12**(6): 1965–1972.

Süzen, A. A., Duman, B. & Şen, B. (2020). Benchmark analysis of jetson tx2, jetson nano and raspberry pi using deep-cnn, *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, IEEE, pp. 1–5.

Tao, W., Liu, T., Zheng, R. & Feng, H. (2012). Gait analysis using wearable sensors, *Sensors* **12**(2): 2255–2283.

Tapia, E. M., Intille, S. S. & Larson, K. (2004). Activity recognition in the home using simple and ubiquitous sensors, *in* A. Ferscha & F. Mattern (eds), *Pervasive Computing*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 158–175.

Tian, Y., Thompson, J., Buck, D. & Sonola, L. (2013). *Exploring the system-wide costs of falls in older people in Torbay*, King's Fund.

Touqeer, H., Zaman, S., Amin, R., Hussain, M., Al-Turjman, F. & Bilal, M. (2021). Smart home security: challenges, issues and solutions at different iot layers, *The Journal of Supercomputing* **77**(12): 14053–14089.

Townsend, D., Knoefel, F. & Goubran, R. (2011). Privacy versus autonomy: a tradeoff model for smart home monitoring technologies, *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, pp. 4749–4752.

Turner, S., Kisser, R. & Rogmans, W. (2015). Falls among older adults in the EU-28 : Key facts from the available statistics, pp. 1–5.
URL: *https://eupha.org/repository/sections/ipsp/Factsheet_falls_in_older_adults_in_EU.pdf*

Van Gestel, A. J., Clarenbach, C. F., Stöwhas, A. C., Rossi, V. A., Sievi, N. A., Camen, G., Russi, E. W. & Kohler, M. (2012). Predicting daily physical activity in patients with chronic obstructive pulmonary disease, *PloS one* **7**(11): e48081.

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

Verghese, J. (2009). Quantitative gait markers and incident fall risk in older adults, *The Journals of Gerontology: Series A* **64**(8): 896–901.

Vestergaard, S., Patel, K. V., Bandinelli, S., Ferrucci, L. & Guralnik, J. M. (2009). Characteristics of 400-Meter Walk Test Performance and Subsequent Mortality in Older Adults, *Rejuvenation Research* **12**(3): 177–184.

Wang, W., Liu, A. X. & Shahzad, M. (2016a). Gait recognition using WiFi signals, *UbiComp 2016 - Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 363–373.

Wang, W., Liu, A. X. & Shahzad, M. (2016b). Gait recognition using wifi signals, *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 363–373.

Wang, Z., Jiang, K., Hou, Y., Dou, W., Zhang, C., Huang, Z. & Guo, Y. (2019). A survey on human behavior recognition using channel state information, *Ieee Access* **7**: 155986–156024.

Wang, Z., Jiang, K., Hou, Y., Huang, Z., Dou, W., Zhang, C. & Guo, Y. (2019). A Survey on CSI-Based Human Behavior Recognition in Through-The-Wall Scenario.
URL: *https://ieeexplore.ieee.org/document/8735849/*

Wen, Y., Tian, X., Wang, X. & Lu, S. (2015). Fundamental limits of rss fingerprinting based indoor localization, *2015 IEEE Conference on Computer Communications (INFOCOM)*, IEEE, pp. 2479–2487.

Wilson, D. & Atkeson, C. (2005). Simultaneous tracking and activity recognition (STAR) using many anonymous, binary sensors, *Proceedings of the Third International Conference on Pervasive Computing* **3468**: 62–79.

Witten, I. H., Frank, E., Hall, M. A. & Pal, C. J. (2016). *Data Mining: Practical machine learning tools and techniques*, Morgan Kaufmann.

Xiao, J., Wu, K., Yi, Y., Wang, L. & Ni, L. M. (2013). Pilot: Passive device-free indoor localization using channel state information, *2013 IEEE 33rd International Conference on Distributed Computing Systems*, IEEE, pp. 236–245.

Xie, Y., Li, Z. & Li, M. (2015). Precise power delay profiling with commodity wifi, *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, ACM, New York, NY, USA, p. 53–64.
URL: *http://doi.acm.org/10.1145/2789168.2790124*

Yadav, R. (2018). *Indoor Positioning System using Pedestrian Dead Reckoning and Bluetooth Beacon*, PhD thesis, Chosun University.

Yan, H., Zhang, Y., Wang, Y. & Xu, K. (2019). Wiact: A passive wifi-based human activity recognition system, *IEEE Sensors Journal* **20**(1): 296–305.

Yang, J., Zou, H., Jiang, H. & Xie, L. (2018). Device-Free Occupant Activity Sensing Using WiFi-Enabled IoT Devices for Smart Homes, *IEEE Internet of Things Journal* **5**(5): 3991–4002.

Yang, S. H., Zhang, W., Wang, Y. & Tomizuka, M. (2013). Fall-prediction algorithm using a neural network for safety enhancement of elderly, *2013 CACS International Automatic Control Conference Digest* pp. 245–249.

Yazdi, M. & Bouwmans, T. (2018). New trends on moving object detection in video images captured by a moving camera: A survey, *Computer Science Review* **28**: 157–177.

Yeomans, J., Thwaites, S., Robertson, W. S., Booth, D., Ng, B. & Thewlis, D. (2019). Simulating time-series data for improved deep neural network performance, *IEEE Access* **7**: 131248–131255.

Yin, J., Zhang, Q. & Karunanithi, M. (2015). Unsupervised daily routine and activity discovery in smart homes, *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* **2015-November**: 5497–5500.

Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P. & Zhang, J. (2014). Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors, *Proceedings of the 6th International Conference on Mobile Computing, Applications and Services* **6**.
URL: *http://eudl.eu/doi/10.4108/icst.mobicase.2014.257786*

Zeng, Y., Pathak, P. H. & Mohapatra, P. (2016). WiWho: WiFi-Based Person Identification in Smart Spaces, *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN 2016 - Proceedings* pp. 1–12.

Zeng, Y., Wu, D., Xiong, J., Yi, E., Gao, R. & Zhang, D. (2019). Farsense: Pushing the range limit of wifi-based respiration sensing with csi ratio of two antennas, *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* **3**(3): 1–26.

Zhang, L.-H. & Dong, Y.-H. (2004). Quorum sensing and signal interference: diverse implications, *Molecular microbiology* **53**(6): 1563–1571.

Zheng, X., Wang, J., Shangguan, L., Zhou, Z. & Liu, Y. (2016). Smokey: Ubiquitous smoking detection with commercial WiFi infrastructures, *Proceedings - IEEE INFOCOM* **2016-July**(Cv).

Zhou, F., Jiao, J., Chen, S. & Zhang, D. (2011). A case-driven ambient intelligence system for elderly in-home assistance applications, *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* **41**(2): 179–189.

Zou, H., Zhou, Y., Yang, J., Gu, W., Xie, L. & Spanos, C. (2017). Freecount: Device-free crowd counting with commodity wifi, *GLOBECOM 2017-2017 IEEE Global Communications Conference*, IEEE, pp. 1–6.

Zou, Y., Liu, W., Wu, K. & Ni, L. M. (2017). Wi-Fi Radar: Recognizing Human Behavior with Commodity Wi-Fi, *IEEE Communications Magazine* **55**(10): 105–111.

Appendix A

Published Papers

Forbes, G., Massie, S., Craw, S. (2020). Fall prediction using behavioural modelling from sensor data in smart homes *Artificial Intelligence Review (2020) 53*, pp. 1071–1091.

Forbes, G., Massie, S., Craw, S., Fraser, L., & Hamilton, G. (2020). Representing Temporal Dependencies in Smart Home Activity Recognition for Health Monitoring, *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8.

Forbes, G., Massie, S., Craw, S. (2020). WiFi-based Human Activity Recognition using Raspberry Pi *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 722–730.

Forbes, G., Massie, S., Craw, S. (2021). Visualisation to Explain Personal Health Trends in Smart Homes *eXplainable Artificial Intelligence in Healthcare workshop (2021).*

Appendix B

Additional Documentation

RESEARCH ETHICS SELF-ASSESSMENT (RESA)

ROBERT GORDON UNIVERSITY ABERDEEN

The aim of the University's *Research Ethics Policy* is to establish and promote good ethical practice in the conduct of academic research. This *self assessment* is intended to enable researchers to undertake an initial self-assessment of ethical issues in their research.

Ethical conduct is not primarily a matter of following fixed rules; it depends on researchers developing a considered, flexible and thoughtful practice.

This *self assessment* aims to engage researchers discursively with the ethical dimensions of their work and potential ethical issues, and the main focus of any subsequent review is not to 'approve' or 'disapprove' of a project but to make sure that this process has taken place.

The *Research Ethics Policy* is available at www.rgu.ac.uk/research-ethics-policy

LAY SUMMARY
Please describe the project in plain English (i.e. non-scientific terms)

Data collected from sensors in specially designed `Fit Homes' will be used to learn and update actionable knowledge for an evidence-based falls prediction system. The project will develop a case-based system that analyses the sensor data *to* identify patterns of activity and behaviours identified as being related to an increased chance of fall. The system will be delivered as a prototype application that takes inputs from unobtrusive sensors installed in the homes and analyses the data received in order to identify the timing, frequency and duration of the residents' normal activities (getting up, preparing a meal, watching TV, etc). Changes in a `Fit Home' resident's own activity patterns over time can be used to detect deterioration in health; comparisons with other residents can highlight risk of falls.

PART 1: DESCRIPTIVE QUESTIONS

1.	Does the research involve, or does information in the research relate to: *[see Guidance Note 1]*	**Yes**	**No**
	(a) individual human subjects	✔	☐
	(b) groups (e.g. families, communities, crowds)	☐	✔
	(c) organisations	☐	✔
	(d) animals?	☐	✔
	(e) genetically-modified organisms www.rgu.ac.uk/about/health-and-safety/health-and-safety-index/hazardous-and-dangerous-substances-and-genetically-modified-organisms	☐	✔
	Please provide further details:		
	Albyn Housing will collect data from residents in the purpose built houses, and make an anonymised version of the data available for analysis on this project.		

2.	Will the research deal with information which is private or confidential? *[see Guidance Note 2]*	**Yes**	**No**
		✔	☐
	Please provide further details:		
	Albyn Housing will gain informed consent from any participants.		

PART 2: THE IMPACT OF THE RESEARCH			
3.	In the process of doing the research, is there any potential for harm to be done to, or costs to be imposed on: *[see Guidance Note 3(i)]*	**Yes**	**No**
	(a) research participants?	☐	☑
	(b) research subjects? *[see Guidance Note 3(ii)]*	☐	☑
	(c) you, as the researcher?	☐	☑
	(d) third parties? *[see Guidance Note 3(iii)]*	☐	☑
	Please state what you believe are the implications of the research:		
4.	When the research is complete, could negative consequences follow:	**Yes**	**No**
	(a) for research subjects	☐	☑
	(b) or elsewhere? *[see Guidance Note 4]*	☐	☑
	Please state what you believe are the consequences of the research:		

PART 3: ETHICAL PROCEDURES			
5.	Does the research require informed consent or approval from: *[see Guidance Note 5(i)]*	**Yes**	**No**
	(a) research participants?	☐	☑
	(b) research subjects? *[see Guidance Note 5(ii)]*	☐	☑
	(c) external bodies? *[see Guidance Note 5(iii)]*	☐	☑
	If you answered yes to any of the above, please explain your answer:		
	This project will be working with anonymised data supplied by Albyn Housing rather than with research participants directly. However, informed consent will be required and obtained by Albyn Housing as part of the data gathering process.		

		Yes	No
6.	Are there reasons why research subjects may need safeguards or protection? *[see Guidance Note 6]*	☐	☑
	If you answered yes to the above, please state the reasons and indicate the measures to be taken to address them:		

		Yes	No
7.	Does the research involve any "regulated work with children" and/or "regulated work with protected adults", therefore requiring membership of the *Protecting Vulnerable Groups (PVG) Scheme*? *[see Guidance Note 7]*	☐	☑
	[Please note: if the research potentially involves "regulated work", this MUST be raised with your HR Business Partner immediately. In this instance, the Human Resources Department will conduct a detailed assessment and will confirm whether or not PVG Membership is required.]		
	(a) PVG membership is not required.	☐	☐
	(b) PVG membership may be required for working with children.	☐	☐
	(c) PVG membership may be required for working with protected adults.	☐	☐
	(d) PVG membership may be required for working with both children and protected adults.	☐	☐
	If you answered yes to (b), (c) or (d) above, please give further information about the work you will be required to undertake and the nature of the contact with these groups. Please provide as much detail as possible:		
	Are you already a PVG member?	☐	☑
	If yes, please provide your PVG Scheme number:		

		Yes	No
8.	Are specified procedures or safeguards required for recording, management, or storage of data? *[see Guidance Note 8]*	☑	☐
	If you answered yes to the above, please give details:		
	A confidentiality agreement will be put in place.		

PART 4: THE RESEARCH RELATIONSHIP

		Yes	No
9.	Does the research require the researcher to give or make undertakings to research participants or subjects about the use of data? *[see Guidance Note 9]*	☐	☑
	If you answered yes to the above, please outline the likely undertakings:		

		Yes	No
10.	Is the research likely to be affected by the relationship with a sponsor, funder or employer? *[see Guidance Note 10]*	☐	☑
	If you answered yes to the above, please identify how the research may be affected:		

PART 5: SELF ASSESSMENT REGARDING THE RESEARCH PROJECT'S ETHICAL STATUS

		Yes	No
11.	Does the research have potentially negative implications for the University? *[see Guidance Note 11]*	☐	☑
	If you answered yes to the above, please explain your answer:		

		Yes	No
12.	Are any potential conflicts of interest likely to arise in the course of the research? *[see Guidance Note 12]*	☐	☑
	If you answered yes to the above, please identify the potential conflicts:		

		Yes	No
13.	Are there any other ethical issues not covered by this form which you believe you should raise?	☐	☑

14.	Please select one of the following:	
	i. The research project should proceed in its present form – no further action is required	☑
	ii. The research project requires ethical review by the University's Research Ethics Sub-Committee	☐
	iii. The research project requires ethical review by an external body (N.B. Question 5 above). If this applies, please give these details:	☐

Title of External Body providing ethical review	
Address of External Body	
Anticipated date when External Body may consider project	

If any ethical issues arise during the course of the research, a further *RESA* form should be completed.

	Yes	No
Does the study relate to a funded studentship? If yes, please provide Research and Enterprise Services (RES) with a copy of the *Research Ethics: Research Student and Supervisor Assessment (RESSA)* by emailing this to res-ops@rgu.ac.uk	☐	☑

Please confirm the name of the person who has peer reviewed this *RESA*	Nirmalie Wiratunga
RESA completed by	Stewart Massie

☐ Guidance Note 1

Ethical principles normally apply to information, data, and derivative substances in the same way as they apply to the subjects themselves. Consequently, work with individual financial data is governed by the principles of work with individual human subjects, and work with animal tissue is governed by the principles of work with animals. [return to Question 1]

☐ Guidance Note 2

The Australian National Health and Medical Research Council argues: "Individuals have a sphere of life from which they should be able to exclude any intrusion ... A major application of the concept of privacy is information privacy: the interest of a person in controlling access to and use of any information personal to that person." This principle applies to all information about a person, whether or not it is obtained directly from that person. The area that is private is conventional and culturally defined; in the UK it commonly includes income and family arrangements.

The information obtained in research is not, however, necessarily private. Some material is in the public sphere, which includes published and broadcast material, academic discourse, and the activities of government. Activities undertaken in a public place are public, rather than private, if they are openly displayed (e.g. artistic exhibition or attendance at a public event) or subject to public regulation (e.g. driving)."

[return to Question 2]

☐ Guidance Note 3

(i) "Harm" refers to negative consequences beyond those which would occur in the normal course of events. Costs may include putting subjects under stress, causing them anxiety, or even wasting their time. The question asks only about potential harm. Potential harm is not cancelled out by potential benefit. Broader consequences are considered in the following question.
Reviews of information are also subject to ethical consideration. It should never be assumed that no harm can be done to people simply by writing about them.

(ii) "Research subjects" includes not just participants and informants but those about whom data is collected. The term covers any research subject, including humans, animals, and inanimate subject matter.

(iii) The University has a responsibility to avoid putting you at risk, and potentially dangerous situations should always be drawn to the University's attention.

(iv) "Third parties" include any person, group or organisation who may be affected by the process of the research. [return to Question 3]

☐ Guidance Note 4

"Elsewhere" is an open category, intended to include consequences for third parties, sections of the community (e.g. "the voluntary sector"), the economy ("the catering industry") or the environment. ("the national park"), globally, and generalities which are harder to identify (e.g. "animal welfare"). Student researchers should never assume that their work is harmless only because they don't believe others will read it.

[return to Question 4]

☐ Guidance Note 5

(i) Research in the public sphere (question 2) may not require the consent or approval of research subjects. The advice of the Canadian Tri-Boards is that "REBs (research ethics boards) should recognize that certain types of research - particularly biographies, artistic criticism or public policy research - may legitimately have a negative effect on organizations or on public figures in, for example, politics, the arts or business. Such research does not require the consent of the subject ... Consent is not required from organizations such as corporations or governments for research about their institutions".

There is a general presumption that consent should be obtained from subjects whenever the information is private. The requirement to seek consent can, however, be waived in certain exceptional cases, for example where there is necessary deception, or where the consent of a subject may jeopardise the welfare of an informant. All such cases require explicit ethical review and an extended justification.

(ii) The consent of research *subjects* cannot be presumed because the consent of *informants* has been obtained. For example, one member of a family cannot necessarily be taken to speak for others, and an employer cannot always give consent on behalf of employees.

(iii) The consent of *external bodies* is required for several types of research, including e.g.

- ☐ research relating to the NHS
- ☐ research for work with dangerous substances, and
- ☐ research involving experimentation with animals.

The existence of external consent does not ethically exclude the project from consideration by the University, or vice-versa. Please provide a brief description of the project as submitted to the external body for ethical review.

[return to Question 5]

☐ Guidance Note 6

This may apply, for example, to human subjects who are regarded as vulnerable (e.g. children or prisoners) and to animals. Consent should not be taken as sufficient protection.

[return to Question 6]

☐ Guidance Note 7

(i) Regulated work normally involves caring for, supervising or working with individuals who participate in an organised activity. There are two types of regulated work: regulated work with ***children*** and regulated work with ***protected adults*****.**

(ii) ***Children*** are all people under the age of 18.

(iii) ***Protected adults*** are individuals aged 16 or over who are provided with (and thus receive) a type of care, support or welfare service. It is a service-based definition and avoids labelling adults on the basis of disability. A person will be a protected adult for the duration that they are receiving the service. Therefore some adults will be protected most of the time (e.g. residents within a care home) whereas others will only be protected for short periods (e.g. whilst receiving medical treatment at a hospital).

(iv) Further details can be found at www.rgu.ac.uk/about/governance/policies-and-legal/disclosure-scotland and www.disclosurescotland.co.uk/pvg/pvg_index.html.
Alternatively, you may want to discuss this with your HR Representative:
https://you.rgu.ac.uk/org/hr/SitePages/Meet%20the%20HR%20Team.aspx.

[return to Question 7]

☐ Guidance Note 8

Private data should be presumed to be under the control of the person or organisation to whom it relates. Anonymity is not a sufficient condition for confidentiality. Removing names from a report, or using aggregate data, may not be enough to ensure that respondents cannot be recognised or identified; and even where material is not identifiable except by the person who gave it, using it in ways that go beyond the terms on which it has been given may be a breach of trust.

[return to Question 8]

☐ Guidance Note 9

The integrity of the researcher, and the status of future research, requires that such undertakings should be respected. Promises should not be given in circumstances where they cannot be kept. For example, a researcher is not at liberty to conceal criminal activity and consequently cannot offer unconditional confidentiality in a study of such activity.

[return to Question 9]

☐ Guidance Note 10

Students who are undertaking research within the context of a work placement or employment should be aware that this is likely to have implications for the research and should identify what those implications are.

Sponsorship includes the grant of access to material by a responsible organisation.

[return to Question 10]

☐ Guidance Note 11

The University needs to know if the research may jeopardise its reputation through, for example, work for oppressive governments or other research relationships (e.g. work for tobacco firms) that might compromise or bias the research. Negative consequences in the form of criticism of the University or negative evaluations by students are legitimate potential outcomes.

[return to Question 11]

☐ Guidance Note 12

This includes, for example, conflicts between researchers, funders, stakeholders, employers and other research projects.

[return to Question 12]

RESEARCH ETHICS: RESEARCH STUDENT AND SUPERVISOR ASSESSMENT *(RESSA)* FORM

The aim of the University's *Research Ethics Policy* is to establish and promote good ethical practice in the conduct of academic research. This *self-assessment* is intended to enable researchers to undertake an initial self-assessment of ethical issues in their research.

Ethical conduct is not primarily a matter of following fixed rules; it depends on researchers developing a considered, flexible and thoughtful practice.

This *self-assessment* aims to engage researchers with ethical dimensions of their work and potential ethical issues. Please note the RESSA form and its subsequent review is not an ethical review and is not to 'approve' or 'disapprove' a project but to make sure that ethical approval will be sought if required.

The *Research Ethics Policy* is available at http://www.rgu.ac.uk/about/planning-and-policy/policies/policies

Research Student Name	Glenn Forbes
Principal Supervisor	Stewart Massie
School	CSDM
Research Project Title	Employing Multi-Modal Sensors for Personalised Smart Home Health Monitoring

LAY SUMMARY
Please describe the project in plain English (i.e. non-scientific terms) – 300 words maximum.

The FITsense project aimed to develop a solution using in-home sensors to predict an increased risk of falling in the 16 pilot FitHomes in Albyn.

During the project it became clear that additional health monitoring could be performed with existing sensor equipment, beyond fall risk prediction.

This project aims to:
• Identify and monitor general health and serious conditions such as dementia, heart disease, obesity, etc.
• Investigate developing new sensor equipment and modalities which may be more capable of capturing diverse data which may expand health monitoring capabilities.
• Find a solution for retrofitting existing variable house layouts with sensor equipment, and transferring data collection and learning between homes.

PART 1: DESCRIPTIVE QUESTIONS

		Yes	No
1.	Does the research involve, or does information in the research relate to: *[see Guidance Note 1]*	**Yes**	**No**
	(a) individual human subjects	✓	
	(b) groups (e.g. families, communities, crowds)		✓
	(c) organisations		✓
	(d) animals?		✓
	(e) genetically-modified organisms http://www.rgu.ac.uk/about/planning-and-policy/policies/policies		✓

	Please provide further details:		
	Albyn Housing will collect data from residents in purpose built houses, and make an anonymised verison of the data available for analysis on the project.		
2.	Will the research deal with information which is private or confidential? *[see Guidance Note 2]*	**Yes** ✓	**No**
	Please provide further details:		
	Albyn Housing will gain informed consent from any participants.		

PART 2: THE IMPACT OF THE RESEARCH

3.	In the process of doing the research, is there any potential for harm to be done to, or costs to be imposed on: *[see Guidance Note 3(i)]*	**Yes**	**No**
	(a) research participants?		✓
	(b) research subjects? *[see Guidance Note 3(ii)]*		✓
	(c) you, as the researcher?		✓
	(d) third parties? *[see Guidance Note 3(iii)]*		✓
	Please state what you believe are the implications of the research:		
4.	When the research is complete, could negative consequences follow:	**Yes**	**No**
	(a) for research subjects		✓
	(b) or elsewhere? *[see Guidance Note 4]*		✓
	Please state what you believe are the consequences of the research:		

PART 3: ETHICAL PROCEDURES			
5.	Does the research require informed consent or approval from: *[see Guidance Note 5(i)]*	**Yes**	**No**
	(a) research participants?		✓
	(b) research subjects? *[see Guidance Note 5(ii)]*		✓
	(c) external bodies? *[see Guidance Note 5(iii)]*		✓
	If you answered yes to any of the above, please explain your answer:		
	This project will be working with anonymised data supplied by Albyn Housing rather than with research participants directly. However, informed consent will be required and obtained by Albyn Housing as part of the data gathering process.		
6.	Are there reasons why research subjects may need safeguards or protection? *[see Guidance Note 6]*	**Yes**	**No** ✓
	If you answered yes to the above, please state the reasons and indicate the measures to be taken to address them:		
7.	Has PVG membership status been considered? *[see Guidance Note 7]*	**Yes**	**No**
	(a) PVG membership is not required.	✓	
	(b) PVG membership is required for working with children.		
	(c) PVG membership is required for working with protected adults.		
	(d) PVG membership is required for working with both children and protected adults.		
	If you answered yes to (b), (c) or (d) above, please give details:		
8.	Are specified procedures or safeguards required for recording, management, or storage of data? *[see Guidance Note 8]*	**Yes** ✓	**No**
	If you answered yes to the above, please give details:		
	A confidentiality agreement will be put in place.		

PART 4: THE RESEARCH RELATIONSHIP			
9.	Does the research require the researcher to give or make undertakings to research participants or subjects about the use of data? *[see Guidance Note 9]*	**Yes**	**No**
			✓
	If you answered yes to the above, please outline the likely undertakings:		
10.	Is the research likely to be affected by the relationship with a sponsor, funder or employer? *[see Guidance Note 10]*	**Yes**	**No**
			✓
	If you answered yes to the above, please identify how the research may be affected:		

PART 5: OTHER ISSUES			
11.	Are there any other ethical issues not covered by this form which you believe you should raise?	**Yes**	**No**
			✓

STATEMENT BY RESEARCH STUDENT			
I believe that the information I have given in this form is correct, and that I have addressed the ethical issues as fully as possible at this stage.			
Signed:	Glenn Forbes	**Date:**	04/03/2019

Ethical issues should be reviewed periodically through completion of the project, in particular at the transfer application stage by completing a further RESSA form.

The *Research Ethics Policy* is available at http://www.rgu.ac.uk/about/planning-and-policy/policies/policies

PART 6: TO BE COMPLETED BY THE PRINCIPAL SUPERVISOR

		Yes	No
12.	Does the research have potentially negative implications for the University? *[see Guidance Note 11]*		✓
	If you answered yes to the above, please explain your answer:		
13.	Are any potential conflicts of interest likely to arise in the course of the research? *[see Guidance Note 12]*		✓
	If you answered yes to the above, please identify the potential conflicts:		
14.	Are you satisfied that the student has engaged adequately with the ethical implications of the work? *[see Guidance Note 13]*	✓	
	If you answered no to the above, please identify the potential issues:		

15.	Please select one of the following:	
	i. The research project should proceed in its present form – no further ethical approval is required and no further action is necessary.	✓
	ii. The research project requires ethical review by the School's Ethics Review Panel (SERP)	
	iii. The research project requires ethical review by the University's Research Ethics Sub-Committee (RESC)	
	iv. The research project requires ethical review by an external body (. If this applies, please give these details:	

Title of External Body providing ethical review	
Address of External Body	
Anticipated date when External Body may consider project	

AFFIRMATION BY PRINCIPAL SUPERVISOR			
I have read the research student's responses and have discussed ethical issues arising with the research student. I can confirm that the information presented by the research student is correct and appropriate to allow an informed judgement on whether further ethical approval is required.			
Signed:	Stewart Massie	**Date:**	04/03/2019

Ethical issues should be reviewed periodically through completion of the project, in particular at the transfer application stage by completing a further RESSA form.

ⓘ Guidance Note 1

Ethical principles normally apply to information, data, and derivative substances in the same way as they apply to the subjects themselves. Consequently, work with individual financial data is governed by the principles of work with individual human subjects, and work with animal tissue is governed by the principles of work with animals. [return to Question 1]

ⓘ Guidance Note 2

The Australian National Health and Medical Research Council argues: "Individuals have a sphere of life from which they should be able to exclude any intrusion ... A major application of the concept of privacy is information privacy: the interest of a person in controlling access to and use of any information personal to that person." This principle applies to all information about a person, whether or not it is obtained directly from that person. The area that is private is conventional and culturally defined; in the UK it commonly includes income and family arrangements.

The information obtained in research is not, however, necessarily private. Some material is in the public sphere, which includes published and broadcast material, academic discourse, and the activities of government. Activities undertaken in a public place are public, rather than private, if they are openly displayed (e.g. artistic exhibition or attendance at a public event) or subject to public regulation (e.g. driving)."

[return to Question 2]

ⓘ Guidance Note 3

(i) "Harm" refers to negative consequences beyond those which would occur in the normal course of events. Costs may include putting subjects under stress, causing them anxiety, or even wasting their time. The question asks only about potential harm. Potential harm is not cancelled out by potential benefit. Broader consequences are considered in the following question. Reviews of information are also subject to ethical consideration. It should never be assumed that no harm can be done to people simply by writing about them.

(ii) "Research subjects" includes not just participants and informants but those about whom data is collected. The term covers any research subject, including humans, animals, and inanimate subject matter.

(iii) The University has a responsibility to avoid putting you at risk, and potentially dangerous situations should always be drawn to the University's attention.

(iv) "Third parties" include any person, group or organisation who may be affected by the process of the research. [return to Question 3]

ⓘ Guidance Note 4

"Elsewhere" is an open category, intended to include consequences for third parties, sections of the community (e.g. "the voluntary sector"), the economy ("the catering industry") or the environment. ("the national park"), globally, and generalities which are harder to identify (e.g. "animal welfare"). Student researchers should never assume that their work is harmless only because they don't believe others will read it.

[return to Question 4]

ⓘ Guidance Note 5

(i) Research in the public sphere (question 2) may not require the consent or approval of research subjects. The advice of the Canadian Tri-Boards is that "REBs (research ethics boards) should recognize that certain types of research - particularly biographies, artistic criticism or public policy research - may legitimately have a negative effect on organizations or on public figures in, for example, politics, the arts or business. Such research does not require the consent of the subject ... Consent is not required from organizations such as corporations or governments for research about their institutions".

There is a general presumption that consent should be obtained from subjects whenever the information is private. The requirement to seek consent can, however, be waived in certain exceptional cases, for example where there is necessary deception, or where the consent of a subject may jeopardise the welfare of an informant. All such cases require explicit ethical review and an extended justification.

(ii) The consent of research *subjects* cannot be presumed because the consent of *informants* has been obtained. For example, one member of a family cannot necessarily be taken to speak for others, and an employer cannot always give consent on behalf of employees.

(iii) The consent of *external bodies* is required for several types of research, including e.g.

- research relating to the NHS
- research for work with dangerous substances, and
- research involving experimentation with animals.

The existence of external consent does not ethically exclude the project from consideration by the University, or vice-versa. Please provide a brief description of the project as submitted to the external body for ethical review.

[return to Question 5]

ⓘ Guidance Note 6

This may apply, for example, to human subjects who are regarded as vulnerable (e.g. children or prisoners) and to animals. Consent should not be taken as sufficient protection.

[return to Question 6]

ⓘ Guidance Note 7

If your research will involve some form of work with children or protected adults or both, you may need to apply to join the Disclosure Scotland PVG Scheme. For further details and notes on applying please refer to www.rgu.ac.uk/about/governance/policies-and-legal/disclosure-scotland and www.disclosurescotland.co.uk/.

[return to Question 7]

ⓘ Guidance Note 8

Private data should be presumed to be under the control of the person or organisation to whom it relates. Anonymity is not a sufficient condition for confidentiality. Removing names from a report, or using aggregate data, may not be enough to ensure that respondents cannot be recognised or identified; and even where material is not identifiable except by the person who gave it, using it in ways that go beyond the terms on which it has been given may be a breach of trust.

[return to Question 8]

ⓘ Guidance Note 9

The integrity of the researcher, and the status of future research, requires that such undertakings should be respected. Promises should not be given in circumstances where they cannot be kept. For example, a researcher is not at liberty to conceal criminal activity and consequently cannot offer unconditional confidentiality in a study of such activity.

[return to Question 9]

ⓘ Guidance Note 10

Students who are undertaking research within the context of a work placement or employment should be aware that this is likely to have implications for the research and should identify what those implications are.

Sponsorship includes the grant of access to material by a responsible organisation.

[return to Question 10]

ⓘ Guidance Note 11

The University needs to know if the research may jeopardise its reputation through, for example, work for oppressive governments or other research relationships (e.g. work for tobacco firms) that might compromise or bias the research. Negative consequences in the form of criticism of the University or negative evaluations by students are legitimate potential outcomes.

[return to Question 12]

ⓘ Guidance Note 12

This includes, for example, conflicts between researchers, funders, stakeholders, employers and other research projects.

[return to Question 13]

ⓘ Guidance Note 13

In signifying agreement, principal supervisors are accepting part of the ethical responsibility for the project.

[return to Question 14]