

FORBES, G., MASSIE, S. and CRAW, S. 2020. Wifi-based human activity recognition using raspberry pi. To be presented at *32nd International conference tools with artificial intelligence (ICTAI 2020)*, 9-11 November 2020, [virtual conference].

Wifi-based human activity recognition using raspberry pi.

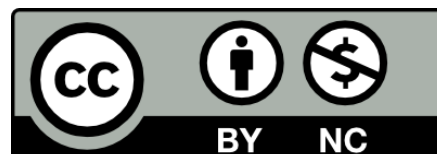
FORBES, G., MASSIE, S. and CRAW, S.

2020

© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

 OpenAIR
@RGU

This document was downloaded from
<https://openair.rgu.ac.uk>



WiFi-based Human Activity Recognition using Raspberry Pi

Glenn Forbes, Stewart Massie, Susan Craw

School of Computing Science & Digital Media
Robert Gordon University, Aberdeen, UK

{*g.r.forbes, s.massie, s.craw*}@rgu.ac.uk

www.rgu.ac.uk/dmstaff/lastname-firstname

Abstract—Ambient, non-intrusive approaches to smart home health monitoring, while limited in capability, are preferred by residents. More intrusive methods of sensing, such as video and wearables, can offer richer data but at the cost of lower resident uptake, in part due to privacy concerns. A radio frequency-based approach to sensing, Channel State Information (CSI), can make use of low cost off-the-shelf WiFi hardware. We have implemented an activity recognition system on the Raspberry Pi 4, one of the world’s most popular embedded boards. We have implemented a classification system using the Pi to demonstrate its capability for activity recognition. This involves performing data collection, interpretation and windowing, before supplying the data to a classification model.

In this paper, the capabilities of the Raspberry Pi 4 at performing activity recognition on CSI data are investigated. We have developed and publicly released a data interaction framework, capable of interpreting, processing and visualising data from a range of CSI-capable hardware. Furthermore, CSI data captured for these experiments during various activity performances have also been made publically available. We then train a Deep Convolutional LSTM model to classify the activities. Our experiments, performed in a small apartment, achieve 92% average accuracy on 11 activity classes.

Index Terms—Activity Recognition, Smart Home, IoT, RF

I. INTRODUCTION

The available choice for sensing technologies in smart home health monitoring environments has increased in variety over the last decade. Typically basic technologies such as passive infrared sensors (PIRs) for motion detection, and magnetic switches for tracking doors have been deployed in dense configurations, with commercial offerings largely matching the state of the art [1]. These sensors produce simple activation events which can be used to indicate presence and activity levels in the home [2]. Many basic sensors are also considered to be ambient in that residents do not perceive them to be invasive. While their ambient nature is a strength, these sensors produce comparatively very little data. Sensor technologies capable of producing richer data representations, such as cameras and wearables, are becoming popular but can be considered too invasive for some tasks. As a result, this reduces their overall uptake and residents may modify their behaviour if they are continuously aware that they are being monitored. Ambient sensor technologies which can produce large amounts

of rich data could overcome this limitation, especially in health monitoring.

Channel State Information (CSI) captured from off-the-shelf WiFi devices is a developing area of research, with significant progress being made in Gesture [3] and Activity Recognition [4], [5]. This technology leverages the ambient nature of RF transmissions, while also potentially allowing for a system which benefits from the existing WiFi infrastructure in smart homes. While commercial applications are only now becoming practical [6], many publications in this area address the issues facing practical deployment. One such issue is the specific hardware/software combination which is necessary to facilitate the extraction of CSI data.

Since the Linux 802.11n CSI Tool was released in 2011 [7], CSI research has primarily been performed using the Intel IWL5300 wireless card. This hardware has been the cheapest and most consistent way to extract CSI data from off-the-shelf hardware. Over time, this hardware configuration has become less relevant as embedded hardware boards are used more in sensor configurations. Following the release of Nexmon CSI [8], it has now become possible to extract CSI from a Raspberry Pi 3B+/4, which makes it easier to conduct research in this area. As this is a recent release, the CSI capabilities of the Raspberry Pi 4 have not yet been fully established.

The Raspberry Pi 4’s price and potential capabilities position it as a powerful tool in smart home health monitoring. If it can offer 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. This is a good challenge for the 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).

While CSI research has been developing over the past decade, it can be difficult for researchers to start working with this technology. Many of the tools and scripts written for interacting with this technology are written for specialised

languages or tools such as MATLAB. There are few public CSI datasets, which have a variety of activity classes that may not be relevant to specific scenarios. Hence, it is difficult to perform research without assembling hardware configurations and collecting your own data. To address this, alongside the code used for our experiments, we are making a CSI interaction framework available for use, called CSIKit¹. This framework is written in Python, and uses Numpy and Scipy which should improve accessibility for data science-focussed research. The data used for these experiments, and some others, will also be made available alongside CSIKit. This is aimed at allowing researchers to get started with CSI research, without spending significant amounts of time assembling their own configuration.

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;
- and an implementation of an established activity classification model structure on the above dataset.

In this paper, 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 they will affect our future work.

II. RELATED WORK

The Internet of Things offers improved inter-connectivity between devices which has allowed for basic off-the-shelf sensors which can form cohesive networks throughout an entire home. These networks have assisted researchers in performing more long-term smart home monitoring studies, which previously could be limited in scope and size due to communication overhead [9]. Early studies looking at residential healthcare informatics highlighted subject behaviour using simple sensor equipment [10]. In doing so, they were able to identify regular behavioural routines. As the underlying technologies used in these studies improved, such as video, the richness of the data produced by sensor equipment improved. This allows for more insights to be extracted from the same behaviours demonstrated by a resident. However activity tracking still lacked the richness necessary for accurate health monitoring.

Assessments of deteriorating health condition, such as mobility, are traditionally performed under laboratory conditions

[11], [12]. For example, Vestergaard found a link between performance in the 400-Meter Walking Test and remaining life span. However, these methods are slightly artificial in that the assessments are outwith the normal environment and can be treated as a one-off performance rather than a measure of day-to-day capabilities. If an approximation of these tests can be achieved in a home environment, not only would performative behaviour be greatly reduced, but deteriorating health conditions could be caught sooner. While the measurements taken for these tests are not easily replicated using basic sensor equipment, newer technologies can be used to extract similar metrics.

Sensor technologies which are capable of producing richer data have typically more intrusive modalities, such as video or wearable sensors. The rich data streams they produce can be used to extract similar measurements to those taken during hospital testing, such as gait speed in video [13], and heart rate from wearables [14], [15]. This data is valuable, however residents tend to resist these sensor modalities as they are intrusive or inconvenient. Residents have privacy concerns with long term video capture, and wearable sensors require upkeep and maintenance.

Conversely, many unintrusive sensor modalities are limited to more basic technologies, such as infrared motion detectors, or magnetic switches. Researchers have had varying success in performing measurements using these sensors [16], however these technologies appear to be better suited to behavioural tracking rather than physiological estimations [9], [17]. The basic technology used in these sensors is often cheap, however IoT-connected versions of these sensors can be much more expensive due to the additional wireless hardware and branding.

Another form of ambient sensing technology, 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. Research in RF-based sensing technologies has experienced a recent in activity after a significant challenge with interference in through-wall transmissions was overcome [18]. Since this breakthrough, RF has seen wider applications in ambient sensing, such as gait recognition, imaging and human localisation [19], [20]. Software defined radios have demonstrated capabilities for gait tracking, respiration and heart rate monitoring, and even accurate localisation. However from both a research and deployment perspective, the radio equipment used can be prohibitively expensive. The potential applications for RF-based technologies in smart home health monitoring are numerous but a cheaper hardware solution is needed to improve the accessibility of this technology.

As a cheaper 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, CSI is used at an engineering level

¹<https://github.com/Gi-z/CSIKit>

to optimise WiFi network links and identify more efficient connection solutions. CSI has also been used to monitor respiration, heart rate, and for gesture recognition [21], [22] in contact-free scenarios. CSI collection from COTS WiFi devices offers an inexpensive method for ambient RF-based sensing, which can benefit from the existing WiFi infrastructure available in most houses.

III. CHANNEL STATE INFORMATION

CSI has been shown to be effective in performing a range of activity monitoring tasks, such as smoking recognition [23] and crowd counting [24]. Medically-focussed studies have also shown it to be effective in monitoring respiration and heart rate [21]. We plan to use CSI for activity recognition in an ambient smart home environment to assist health monitoring.

While the key objective of this paper is to establish the capabilities of the Raspberry Pi 4 for activity recognition, a brief outline of the fundamentals of CSI assists the discussion.

A. Preliminaries

The available frequency space for 802.11 WiFi is separated into component carrier frequencies through Orthogonal Frequency-Division Multiplexing (OFDM), by which each subcarrier is 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 .

$$CSI_i = \begin{bmatrix} H_{1,1} & \dots & H_{1,r} \\ H_{2,1} & \dots & H_{2,r} \\ \vdots & & \vdots \\ H_{t,1} & \dots & H_{t,r} \end{bmatrix} \quad (1)$$

$H_{t,r}$ 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. Popular CSI hardware such as the Intel IWL5300 can access 30 subcarriers when paired with a device over 2.4GHz, whereas newer devices which can access 802.11ac and 5GHz channels can access around 256 subcarriers. Notably, some subcarriers function as guard carriers (or guard bands) to reduce interference, and so these remain empty by design. If the number of available subcarriers is S , a given $H_{t,r}$ pair can be expressed as:

$$H_{t,r} = [h_{t,r,1}, h_{t,r,2}, \dots, h_{t,r,S}] \quad (2)$$

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.

$$h_s = |h_s| \exp(j \cdot \theta) \quad (3)$$

Multi-path propagation is an effect inherent to wireless transmission systems, as transmitted signals do not travel directly to the receiving antenna. 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 will affect multi-path propagation in a consistent manner, whereas dynamic objects such as human bodies will 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. While both phase and gain are modulated by this activity this paper will focus on the use of gain.

B. Hardware Accessibility

While most WiFi hardware could potentially generate CSI data for third party use, in practice this is usually not the case. The IEEE 802.11n standard defines CSI as a method of communicating phase and gain information for each subcarrier across a transmit and receive antenna pairing [25], however few manufacturers make this data available to developers. This may, in part, be as the standardised implementation of CSI was designed to facilitate link quality monitoring. More non-standard applications for CSI became relevant as commercial CSI solutions became more accessible.

When the Linux 802.11n CSI Tool was released in 2011 [7], it became the most accessible way to extract CSI using a standard hardware configuration with an Intel IWL5300. This has fostered a field of research through which alternative applications for CSI, beyond the original purpose, have been explored. Researchers have experimented with CSI systems capable of monitoring respiration [26], heart rate [22], and sleep posture [21] using off-the-shelf hardware. While many approaches to CSI data interaction make use of digital signal processing techniques, recent research has been shifting towards deep learning approaches for use in more general classification tasks. This has led to significant performance improvements in variable domain application spaces, such as activity recognition, where signal processing and manual feature extraction had previously been the state of the art. Domain adaptation continues to present a pervasive challenge in the CSI space, as RF technologies are significantly affected by their surrounding environment, which can make it difficult to train general models which can be adapted for different environments.

One typical deployment scenario for IWL5300 hardware consists of a host and an access point (AP). The Linux 802.11n CSI Tool allows the IWL5300 to operate in managed mode, which is standard WiFi functionality by which a device connects directly to an access point. The host, configured with the IWL5300, which will generate traffic and capture CSI, and the access point, which provides a transmit surface. The host device sends a stream of packets to the AP at a consistent rate, which generates CSI frames containing

connection information for the link between the host and AP. For example, in activity recognition scenarios, the host and AP will be placed at opposing ends of the room to ensure a significant portion the signal will be altered by disturbances created by subject movement inside the room.

Linux 802.11n CSI Tool is limited to older linux kernels which can introduce significant inconvenience. Similarly, the IWL5300 was a well-equipped wireless chipset at the time of release, including support for the draft 802.11n specification, however the wireless field has moved on significantly over the last decade. The IWL5300 supports MIMO antenna inputs, with many standard configurations using a 2x3 antenna array. However, the IWL5300 does not support many widely available wireless technologies, including 802.11ac. In a study exploring activity recognition performance with the IWL5300, Wang simulated the additional bandwidth offered in 802.11ac by operating two 40MHz channels [27]. They concluded that the additional frequency space and subcarriers could improve recognition accuracy. This indicates that research currently being performed with IWL5300 could be held back by hardware limitations.

High end hardware solutions for generating CSI data do exist in the form of USRP software defined radios (SDRs). Ettus’s USRP N210 SDRs can be used to manually implement the 802.11 specification, which can be used to generate CSI. While a deployment configuration using this hardware is not economical, there is value in the research which can be performed using precise radio equipment due to the flexibility of implementation. This indicates that high end hardware can be used to perform useful research, and that there is a gap in the research space for research performed using modern mid-range wireless hardware.

The Nexmon project run by the Secure Mobile Networking Lab [8] aims to provide enhanced functionality for a wide range of wireless hardware, through chipset-specific firmware patches. These patches can be used to enable monitor mode (as opposed to managed mode), frame injection, and some basic SDR functionality. Notably, the Nexmon CSI offshoot allows for CSI extraction on a small subset of Broadcom chipsets. Originally Nexmon CSI supported extraction on Google Nexus 5 hardware [28], however this is similarly inaccessible as IWL5300. Recently, Nexmon CSI has been updated with patches for the BCM43455C0, which is the wireless chipset used in the Raspberry Pi 3B+/4. This version of Nexmon CSI is also supported on modern publicly available linux kernels. One of the most ubiquitous embedded boards in production, the Raspberry Pi 4 supports newer technologies which were released after the IWL5300 including 802.11ac. As an embedded solution, the Raspberry Pi makes use of a single transmit/receive antenna pairing. This may reduce potential quality of its CSI data output, as compared to the IWL5300. The Raspberry Pi 4 is a cheaper and significantly more accessible CSI capture solution than other available options, which positions it as a potentially useful device for deployment in a smart home health monitoring scenario.

IV. 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. To do this, a device configuration will be assembled to allow data to be collected. 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. This method is similar to how many deep learning activity recognition studies were performed using the IWL5300, however the Pi benefits from a deep learning approach as it has access to significantly more subcarriers.

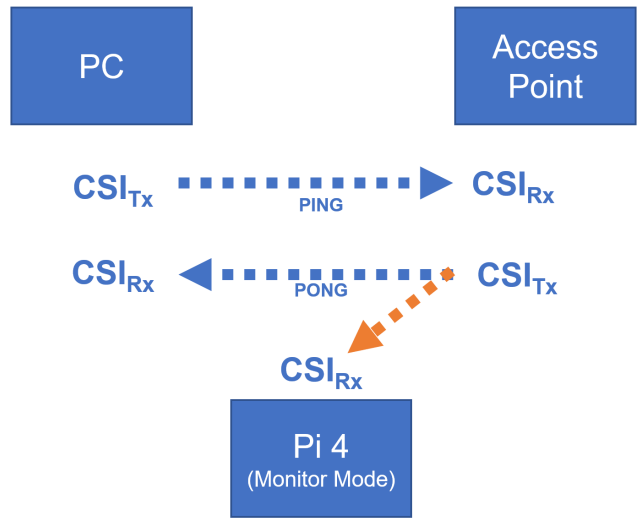


Fig. 1: Activity capture layout, showing CSI as captured at each device.

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 1 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. Traditionally, standard machine learning classification algorithms such as SVM have been used on CSI data for activity recognition [29]. However recent research has shown that CSI data is well suited for use with convolutional LSTMs [30]. We use a deep variant of this model 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 the Pi-based system will benefit more from a convolutional structure due to the increased number of available subcarriers.

V. 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.

A. Equipment

The Raspberry Pi 4 is configured with Debian 10 (Buster/Linux 4.19.97) with the main branch of nexmon_csi² installed. Nexmon was configured with the following filter options: Channel 36/80, Core 1, NSS mask 1, 30us Delay. The MAC address filter was set for the AP. Data collection was controlled from another device 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.

B. 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 [29]. 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 housecat which operated autonomously throughout and moved between rooms during activity performances. This was also deemed to be representative interference.



Fig. 2: Layout of the environment and activity locations.

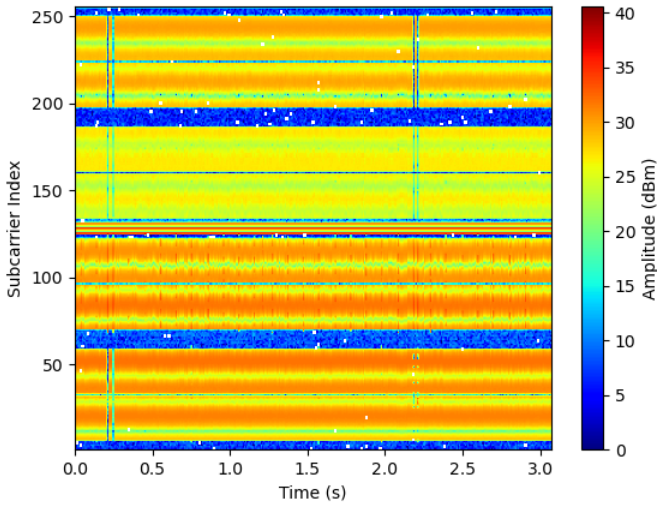
Figure 2 shows the placement of the Pi and AP in relation to where activities were performed in the apartment. As the largest room in the space, the devices were placed in there at opposing corners. Both devices were placed at the same height 1m above the ground to ensure there was an unobstructed signal path.

C. Activity Performances

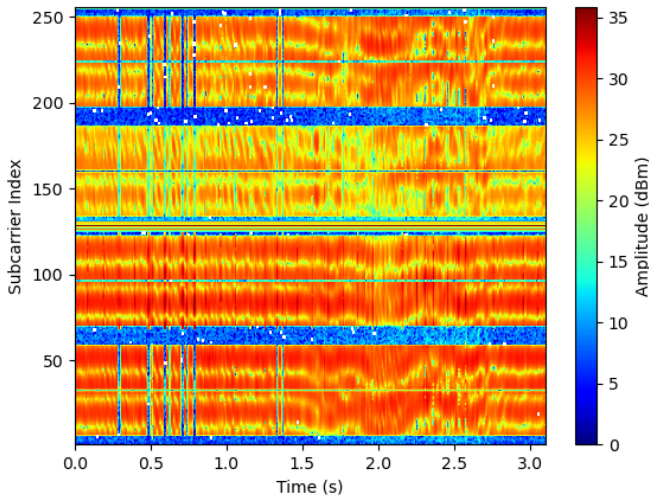
Each activity being performed in the dataset has been designed to both be easily repeatable and representative of realistic in-home behaviour. The selected activities were also chosen to provide a wide range of both similar and distinct activities to effectively assess the performance of the classifier, 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 be observed in Figure 3(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

²https://github.com/seemoo-lab/nexmon_csi/



(a) still



(b) walking

Fig. 3: Comparison of still and walking CSI readings.

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.

Figure 2 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.

D. Data Representation

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.

E. 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.

VI. RESULTS

nothing	517	0	0	0	0	0	0	0	0	0	
standup	0	265	158	1	0	0	0	1	0	0	
sitdown	0	122	321	0	0	0	0	3	0	0	
intobed	0	0	1	436	0	1	0	1	34	0	
cook	0	0	0	0	477	0	0	1	4	0	
washdish	0	0	0	0	0	527	0	0	0	0	
brushteeth	0	0	0	0	0	0	475	1	0	0	
drink	0	0	3	0	0	0	0	318	0	0	
petcat	0	0	0	3	39	0	0	0	179	0	
sleeping	0	0	0	0	0	0	0	0	0	516	
walk	0	0	0	0	0	0	0	0	0	0	
	nothing	standup	sitdown	intobed	cook	washdish	brushteeth	drink	petcat	sleeping	walk

Fig. 4: Confusion Matrix for 100Hz results.

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

TABLE I: Classification report.

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

4 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 “brushsteeth” 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.

VII. 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 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 4, 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.

A. 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 [31]. 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 [21], [29]. Downsampling our activity capture data to 10Hz, we repeated our experiments with a similar model structure. In Figure 5, 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 would allow for the removal of the separate PC for traffic generation in the system, and the Pi could capture CSI for the beacon frames ambiently generated by the AP which are produced at 9.5Hz. This would allow for deployed Pi systems to utilise the existing WiFi infrastructure in smart homes.

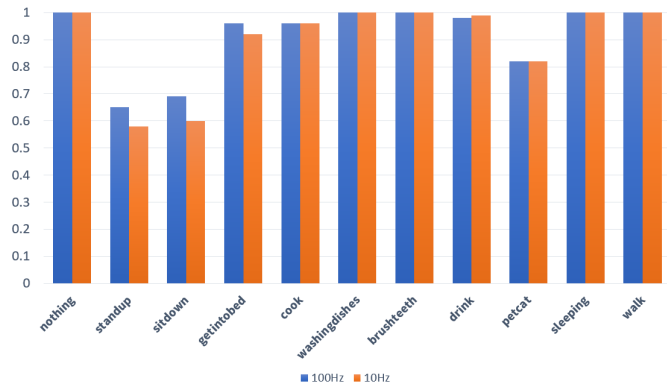


Fig. 5: Comparison of per-class performance on data captured at 100 and 10Hz (macro F1 scores).

VIII. CONCLUSION

Our results confirm the Raspberry 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.

Planned future work is centred on performing direct comparisons between IWL5300 and the Pi hardware in activity recognition performance. Given the extensive research available regarding activity recognition on the IWL5300, establishing the differences in performance in LOS and NLOS scenarios will better facilitate discussion on deployment opportunities for both hardware solutions.

Finally, many non-standard applications have 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 [23] and crowd counting [24]. 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.

REFERENCES

- [1] H. Anumala and S. M. Busetty, "Distributed device health platform using internet of things devices," in *IEEE International Conference on Data Science and Data Intensive Systems*. IEEE, 2015, pp. 525–531.
- [2] S. Massie, G. Forbes, S. Craw, L. Fraser, and G. Hamilton, "Monitoring health in smart homes using simple sensors." K. Bach and R. Bunescu, Eds. CEUR Workshop Proceedings, 2018, pp. 33–37.
- [3] Q. Bu, G. Yang, X. Ming, T. Zhang, J. Feng, and J. Zhang, "Deep transfer learning for gesture recognition with WiFi signals," *Personal and Ubiquitous Computing*, 2020.
- [4] J. K. Brinke and N. Meratnia, "Scaling activity recognition using channel state information through convolutional neural networks and transfer learning," *AIChallengesIoT - Proceedings of the International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things*, pp. 56–62, 2019.
- [5] H. Yan, Y. Zhang, Y. Wang, and K. Xu, "WiAct: A Passive WiFi-based Human Activity Recognition System," *IEEE Sensors Journal*, vol. PP, no. c, pp. 1–1, 2019.
- [6] Linksys, "Linksys aware," Online, 2019. [Online]. Available: <https://www.linksys.com/us/linksys-aware/>
- [7] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 1, p. 53, 2011.
- [8] M. Schulz, D. Wegemer, and M. Hollick, "Nexmon: The c-based firmware patching framework," 2017. [Online]. Available: <https://nexmon.org>
- [9] E. M. Tapia, S. S. Intille, and K. Larson, "Activity recognition in the home using simple and ubiquitous sensors," in *Pervasive Computing*, A. Ferscha and F. Mattern, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 158–175.
- [10] M. Ogawa, R. Suzuki, S. Otake, T. Izutsu, T. Iwaya, and T. Togawa, "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, 2002.
- [11] S. Vestergaard, K. V. Patel, S. Bandinelli, L. Ferrucci, and J. M. Guralnik, "Characteristics of 400-Meter Walk Test Performance and Subsequent Mortality in Older Adults," *Rejuvenation Research*, vol. 12, no. 3, pp. 177–184, 2009.
- [12] T. Michikawa, Y. Nishiwaki, T. Takebayashi, and Y. Toyama, "One-leg standing test for elderly populations," *Journal of Orthopaedic Science*, vol. 14, no. 5, pp. 675–685, 2009.
- [13] E. E. Stone and M. Skubic, "Unobtrusive, continuous, in-home gait measurement using the microsoft kinect," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 10, pp. 2925–2932, 2013.
- [14] P. Pierleoni, A. Belli, L. Palma, M. Pellegrini, L. Pernini, and S. Valenti, "A High Reliability Wearable Device for Elderly Fall Detection," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4544–4553, 2015.
- [15] F. J. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors (Switzerland)*, vol. 16, no. 1, 2016.
- [16] R. Rana, D. Austin, P. G. Jacobs, M. Karunanithi, and J. Kaye, "Gait velocity estimation using time-interleaved between consecutive passive IR Sensor Activations," *IEEE Sensors Journal*, vol. 16, no. 16, pp. 6351–6358, 2016.
- [17] R. Aipperspach, E. Cohen, and J. Canny, "Modeling human behaviour from simple sensors in the home," *Pervasive*, vol. 6030, no. 783, pp. 62–79, 2010.
- [18] F. Adib and D. Katabi, "See through walls with WiFi!" *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 75–86, 2013.
- [19] M. S. Seyfioğlu, S. Z. Gürbüz, A. M. Özbayoğlu, and M. Yüksel, "Deep learning of micro-doppler features for aided and unaided gait recognition," in *Radar Conference*. IEEE, 2017, pp. 1125–1130.
- [20] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via RF body reflections," *Proceedings of the 12th USENIX Conference on Networked Systems Design and Implementation*, pp. 279–292, 2015.
- [21] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, "Tracking vital signs during sleep leveraging off-the-shelf WiFi," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing - MobiHoc '15*, 2015, pp. 267–276.
- [22] A. Khamis, C. T. Chou, B. Kusy, and W. Hu, "Cardiofi: Enabling heart rate monitoring on unmodified COTS WiFi devices," *ACM International Conference Proceeding Series*, pp. 97–106, 2018.
- [23] X. Zheng, J. Wang, L. Shangguan, Z. Zhou, and Y. Liu, "Smokey: Ubiquitous smoking detection with commercial WiFi infrastructures," *Proceedings - IEEE INFOCOM*, vol. 2016-July, no. Cv, 2016.
- [24] S. Liu, Y. Zhao, F. Xue, B. Chen, and X. Chen, "DeepCount: Crowd counting with wifi via deep learning," pp. 1–13, 2019. [Online]. Available: <http://arxiv.org/abs/1903.05316>
- [25] Y. Xiao, "IEEE 802.11n: enhancements for higher throughput in wireless LANs," *IEEE Wireless Communications*, vol. 12, no. 6, pp. 82–91, 2005.
- [26] X. Wang, C. Yang, and S. Mao, "PhaseBeat: Exploiting CSI Phase Data for Vital Sign Monitoring with Commodity WiFi Devices," *Proceedings - International Conference on Distributed Computing Systems*, pp. 1230–1239, 2017.
- [27] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: Device-free location-oriented activity identification using fine-grained WiFi signatures," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 617–628, 2014.
- [28] F. Gringoli, M. Schulz, J. Link, and M. Hollick, "Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets," in *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization*, ser. WiNTECH '19, 2019, p. 21–28.
- [29] H. Lee, C. R. Ahn, N. Choi, T. Kim, and H. Lee, "The effects of housing environments on the performance of activity-recognition systems using wi-fi channel state information: An exploratory study," *Sensors (Switzerland)*, vol. 19, no. 5, 2019.
- [30] X. Wang, "WiFi Fingerprinting based Indoor Localization: When CSI Tensor meets Deep Residual Sharing Learning," *Journal of Chemical Information and Modeling*, vol. 53, no. 9, pp. 1689–1699, 2019.
- [31] H. Narui, R. Shu, F. F. Gonzalez-Navarro, and S. Ermon, *Domain Adaptation for Human Fall Detection Using WiFi Channel State Information*. Cham: Springer International Publishing, 2020, pp. 177–181.