University College Dublin

School of Electrical & Electronic Engineering

ME Electronic & Computer Engineering Interim Report

Fall Detection Using Wi-Fi Signals

Robert Keenan
15333066

Supervisor:

Dr. Nam Tran

Dublin, January 2020

Table of Contents

1. Introduction

Falls are one of the leading causes of fatal and non-fatal injuries especially for elderly people in today's society. Quick response is vital to reducing the long-term effects on the victim physically, emotionally and mentally (Wild et al., 1981; Stokes, 2009). Falls, especially among the elderly, place a severe strain on health and financial systems around the world costing the US healthcare system $50bn in 2015 (Florence et al., 2018).

As a result, an efficient, accurate and cost effective detection system needs to be implemented for a variety of environments. Over 50% of falls for the 65+ age group (1/3 experience a fall at least once a year) occur in their home so a solution that is **cheap, accurate, privacy-focused** and **easily installed**, is needed (Florence et al., 2018; Stokes, 2009). Personally, I have seen the effects, falls can have on an elderly person and the fear induced in a fall victim post-treatment can deem them unable to live independently in the future. (Friedman et al., 2002)

There are a number of solutions on the commercial market today which can be divided into a broad number of categories such as wearables, visual devices and ambient environment sensors. Products on the market in these categories include the Apple Watch, cameras, accelerometer belts, floor vibration sensors and infrared devices. **Privacy** is a key issue that needs to be addressed across the world in the next decade. Therefore, it was a key aim of my project to keep it as privacy oriented as possible. Many of the current solutions on the market suffer from privacy issues especially in such a sensitive environment as somebody's home. Many of these off-the-shelf solutions require **Direct Line of Sight (LOS)** to the person in the room to detect a fall. No obstacles can be in the way of the detection apparatus or a fall may not be detected by the system. This is especially useless in a busy household environment with many obstacles. All of these existing solutions are **very expensive** to buy and implement such as the Apple Watch, specialist cameras and Man-Down alarms. Another issue arises as elderly people are not inclined to wear them as they hinder them from daily activities and can feel like a chore to keep them charged/around their neck or waist. This results in a lot of people refusing to wear them which is clearly unsatisfactory as a solution.

More complex solutions such as Computer Vision cameras require **high processing power** for the calculations and classification operations needed for the sheer amount of data recovered in any of these fall detection experiments.

Therefore, I propose a Wi-Fi based solution utilising commercial off the shelf (COTS) Wi-Fi network interface cards (NICs) providing a cheap, highly accurate and non-intrusive fall detection system for the home and other environments. A fall is detected accurately and privately due to the rich multi-path environment that exists between a number of transmitting and receiving antennas.

The aims of the project and the project plan are as follows:

- Gain a strong understanding and background of Channel State Information (CSI) in the *IEEE802.11n* Wi-Fi standard
- Build an initial system which can collect CSI using the Intel Wi-Fi Link 5300 NIC for a range of Transmitter-Receiver setups
- Obtain CSI data for a range of human activities which would be typical in home/work-place environments
- Design, build and test signal processing algorithms to clean the CSI data obtained for fall detection under various conditions
- Design, implement and test various Machine Learning/Classification algorithms for fall detection for a target fall detection rate of >90%

2. Literature Review

2.1 Initial Theory

In terms of a Wireless Channel, it is preferred for the receiver to be able to estimate the state of the wireless channel. This can allow for optimisations to be made for optimal propagation of signals from transmitter to receiver. For the receiver to understand the state of the wireless channel, many models exist which explain the properties and state of the channel.

2.1.1 WLAN Channel Model

A wireless channel model describes how the amplitude and phase of a signal change as it propagates from the transmitter to the receiver. Of particular importance to this project is the propagation model of a channel. The main propagation models are large-scale propagation known as large-scale path loss and small-scale propagation known as small-scale fading (Paul and Ogunfunmi, 2008). Doppler spread can also be considered.

Large-scale path loss describes the attenuation of a signal between the transmitter and receiver due to physical phenomena in the environment. As the physical environment constrains the wireless signals, the received signals convey information about the environment that they pass through. It is clear that as the distance between transmitter and receiver is increased, the signal and thus, signal power, is spread over a larger area suffering greater attenuation. Small scale fading occurs due to the scattering environment caused by obstacles between the transmitter and receiver changing with time (Paul and Ogunfunmi, 2008). Transmitted signals arrive at the receiver whereby they are added constructively/destructively, as a function of time. This demonstrates the phase-shift in the wireless channel caused by the scattering environment. The signal level changes or fading has two types: microscopic & macroscopic (Hari, 2011)

The scattering environment is composed of Line-of-Sight (LOS) and Non-Line-of-Sight (NLOS) paths introduced by various factors such as furniture, walls, ceilings and more im-

portantly in my case, people. In a MIMO (Multi Input-Multi Output) channel, these scattering effects are amplified. This is known as utilising the spatial diversity of the channel by sending symbols on different streams between one transmit antenna and another receive antenna. A transmitted symbol through the LOS path will clearly arrive at the receiver before the corresponding symbol through a NLOS path (See Figure 2.1). Microscopic fading occurs when the receiver receives many copies of the signal due to scattering near the receiver while Macroscopic fading occurs when the receiver receives multiple delayed copies of the signal due to scattering over a large distance and time period (frequency selective fading). This is characterised by the delay spread of a channel which estimates earliest and latest arrival times of significant copies of the transmitted symbol (Hari, 2011; Paul and Ogunfunmi, 2008).

As small-scale fading is a phenomenon when the scattering environment changes with time for smaller changes in the distance, it will be much more useful for this project. Transmitting distances are not long enough for large-scale path loss to affect signal power greatly.

A $N_{TX} \times N_{RX}$ MIMO channel can be modelled by the following equation at the receive antenna by a spatial vector $\mathbf{y}$ with a sampling period of T = 1/Bandwidth:

$$\mathbf{y}(t) = \mathbf{H}(t)\mathbf{x}(t) + \mathbf{n}(t) \tag{2.1}$$

where $\mathbf{x}(t) = \left[x_1(t) \cdots x_{N_{TX}}(t) \right]^T$ is the transmitter signal vector, $\mathbf{n}(t) = \left[n_1(t) \cdots n_{N_{RX}}(t) \right]^T$ is the AWGN vector and $\mathbf{H}(t)$ is the channel response matrix at each time t (Samardzija and Mandayam, 2003). N_{TX} is the number of transmit antennas and N_{RX}, the number of receive antennas in a MIMO channel.

The majority of today's Wi-Fi networks operate in the $2 - 5GHz$ range and thus suffer from the propagation losses mentioned on Page 3. Orthogonal Frequency Division Multiplexing (OFDM) was proposed to offer a robust solution at these frequencies to narrow-band interference. OFDM is the method of digital modulation whereby the signal to be transmitted is split into a number of narrow-band channels at frequencies above and below the centre frequency ($5GHz$ for example) (Chadha et al., 2013). These narrow-band channels at different frequencies are also known as *subcarriers* and the method to obtain higher data rates is MIMO-OFDM (Nee et al., 2006). OFDM uses a number of modulation schemes such

as QAM and PSK. For each OFDM symbol, there are a number of QAM values depending on the channel bandwidth. The application of the IFFT modulates the OFDM symbol onto a number of subcarriers. This is followed by adding cyclic prefix to make the signal robust to multipath propagation (Chadha et al., 2013; Nee et al., 2006).

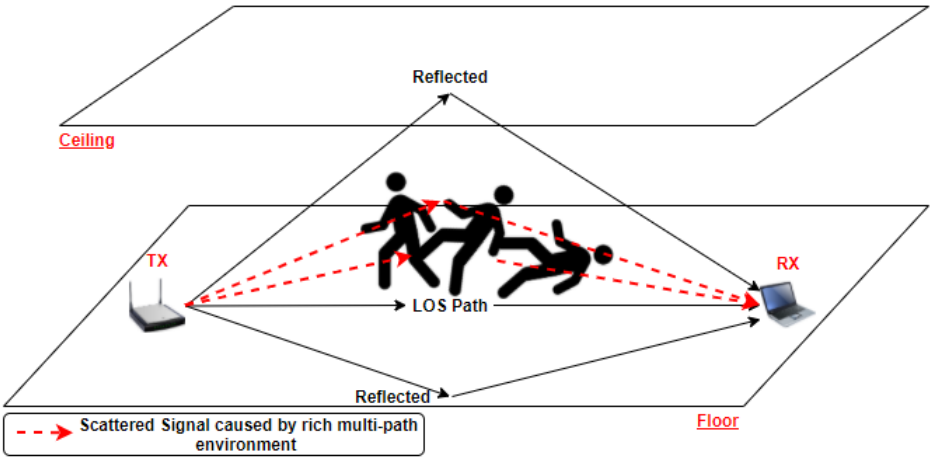


Figure 2.1: The LOS/NLOS paths between the Transmit and Receive antennas in a MIMO channel. Note how the scattering environment is created due to the environment structure and the person. Over a short LOS path, small-scale fading characterises the received signal and environment. (Taken from my Interim Presentation)

In a MIMO-OFDM 802.11n channel, the number of subcarriers is dependent on the channel bandwidth and grouping constant. The grouping constant is used to group adjacent subcarriers and report them as a single value to reduce the size of the report field (IEEE802, 2009). Thus, Equation 2.1 can be rewritten as the following for the subcarrier k at one time instance for a MIMO-OFDM channel:

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{n}_k \tag{2.2}$$

where $\mathbf{x}_k = \begin{bmatrix} x_{k,1} & \cdots & x_{k,N_{TX}} \end{bmatrix}^T$ is the transmitted signal vector, $\mathbf{n}_k = \begin{bmatrix} n_{k,1} & \cdots & n_{k,N_{RX}} \end{bmatrix}^T$ is the noise vector. However, the channel matrix $\mathbf{H}_k$ can be used to describe the channel response for each transmitter-receiver pair:

$$\mathbf{H}_k = \begin{bmatrix} h_{k,11} & \cdots & h_{k,N_{TX}1} \\ \vdots & \ddots & \vdots \\ h_{k,1N_{RX}} & \cdots & h_{k,N_{TX}N_{RX}} \end{bmatrix} \tag{2.3}$$

As described by the 802.11n standard, there are 56 subcarriers for a $20MHz$ bandwidth and 114 for a $40MHz$ bandwidth with a grouping constant $N_g = 1$. (IEEE802, 2009).

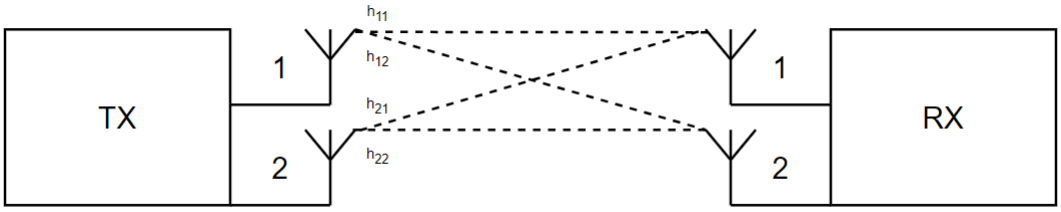


Figure 2.2: A 2×2 MIMO-OFDM channel is shown. The channel response in each stream is shown as h_{11}, h_{12}, etc. between each TX and RX antenna

2.1.2 Channel State Information (CSI)

The Channel State Information (CSI) describes the Channel Matrix **H** and thus, the wireless MIMO-OFDM channel itself. It can be described in 3-D matrix form for one packet with N_{TX} transmit antennas, N_{RX} receive antennas for a transmitter-receiver pair (TX antenna i and RX antenna j)

The depth of the 3-D matrix is dependent on the number of subcarriers. Each subcarrier k for a TX-RX pair (ij) conveys the amplitude/gain and the phase response of the channel at that time instance, $h_{k,ij} = |h_{k,ij}|e^{j\theta}$ (Chadha et al., 2013). Any change in the channel introduced by either path loss or multi-path fading (See Figure 2.1) will result in channel distortion (amplitude distortion and phase shift).

COTS Wi-Fi devices do not collect CSI data readily available for the user. With the arrival of the 802.11n WLAN Standard in 2009, transmit beamforming was able to be utilised to estimate the channel over which a beamformee (RX) and beamformer (TX) are communicating. It estimates the channel through 2 methods: implicit feedback and explicit feedback. In implicit feedback, the beamformer receives long trained symbols from the beamformee which it uses to estimate the channel. In explicit feedback, the beamformee makes a direct estimate from the training symbols sent by the beamformer (IEEE802, 2009). The goal is to focus energy towards the receiver to increase the SNR of the wireless channel (Bjrnson et al., 2014). The beamformer can maximise the signal power at the receiver depending on the current state of the channel. In a LOS scenario, it is the transmitter forming a beam to the receiver directly. (Bjrnson et al., 2014). This can give great insight into the changing environment between transmitter and receiver.

Currently, only Intel and Atheros Wi-Fi NICs can return CSI data through open source tools developed by a number of groups Halperin et al. (2011) (Intel NICs) & Xie et al. (2015) (Atheros NICs). The CSI of every subcarrier in a MIMO-OFDM wireless channel is presented as a complex number $a + jb$. The CSI matrix will be of the dimensions $N_{TX} \times N_{RX} \times 56$ for a $20MHz$ bandwidth and $N_{TX} \times N_{RX} \times 114$ for $40MHz$ bandwidth in theory. In practice, a Wi-Fi NIC uses several bits to represent a & b (10 bits for each in an Atheros NIC and 8 bits in an Intel NIC) (Xie et al., 2015). This allows the channel to be represented in a number of complex numbers at a range of subcarrier frequencies. Due to transmit beamforming, we can obtain an accurate model of the channel and the environment. This makes it extremely useful for fall detection, activity detection and indoor localisation where there is a reasonably short distance between transmitter and receiver. The scattering multi-path environment introduced by an individual will be reported by each data packet's CSI matrix and thus, we can classify the human activity. This will be useful for my project for when an individual enters a room, falls, creating a large multi-path scattering environment which will be seen in the CSI data (See Figure 2.1).

2.1.3 Obtaining CSI information

As I will be using the Intel Wi-Fi link 5300 Wireless NIC, the open source Linux CSI tool, Halperin et al. (2011), is relevant. It operates on Ubuntu 12.04-14.04.4 using customised versions of Intel's close-source firmware and the open-source *iwlwifi* wireless driver (Halperin et al., 2011). They have also developed user-space measurement tools, access point functionality for both transmitter and receiver and MATLAB scripts for data analysis and pre-processing (Halperin et al., 2011). The CSI data is passed to the kernel driver of the receiver computer which passes the CSI to the user-space program for processing. In total, 30 groups of subcarriers evenly across the channel bandwidth are obtained by the Linux tool. The tool returns the CSI in a data structure which can be interpreted in MATLAB using provided scripts (Halperin et al., 2011). For example, *get_scaled_csi*() returns the CSI data structure in absolute units rather than Intel's reference level (Halperin et al., 2011).

2.2 Related Work & Existing Fall Detection Systems

This section deals with current work in the area of CSI and device-free location/activity detection and a literary investigation of the future steps of my project.

2.2.1 CSI and Human detection

Using the CSI tool, Halperin et al. (2011), it is clear to see there are many applications of CSI data for a wireless channel. The majority of these were carried out by research groups in the areas of indoor localisation (Wu et al., 2012), gesture/gait recognition (Nandakumar et al., 2014; Wang et al., 2016) and human activity detection (Wang et al., 2014b). As mentioned in Chapter 1, various wearable approaches have been investigated such as using smartphones for fall detection (Dai et al., 2010). These solutions have obvious disadvantages in terms of needing the device on your person at all times with a suitable level of battery charge. People in most cases either forget to wear them or don't want to. They can be very accurate when all sensors are worn but these may hinder a person's quality of life (Wang et al., 2014a).

As mentioned in Section 2.1, CSI can tell us a lot about the characteristics of the environment. There are a number of attributes in the CSI data across 30 subcarriers which can tell us something about the environment such as the rate of change of the CSI amplitude for the same subcarrier across packets. These attributes can be linked to human activities through a series of supervised learning algorithms in training. The attributes of the CSI data could be matched with a *fall* or *no-fall* activity. Of course, there will be areas where there are fall-like activities such as a person sitting down. This can lead to accurate classification of the activities. Clearly using CSI is a useful method for determining human activities in an environment as Wi-Fi access points are readily available around us in today's world.

2.2.2 Fall Detection using CSI

There are a number of research groups that have completed work in this field such as one of the first systems, WiFall (Wang et al., 2017c). They achieved 90% and 94% fall detection

accuracy with two different classifiers making this project a useful reference for me.

The WiFall system consists of three parts: sensing, learning & alerting. The sensing phase is the simple transmission of signals from transmitter to receiver. The learning phase is comprised of data processing, profile construction and an activity decision module. The alerting phase triggers an alarm when a fall is detected (Wang et al., 2017c). As in Section 2.1, the CSI packet is received as a $N_{TX} \times N_{RX} \times 30$ matrix where there are a number of transmitter-receiver pairs which WiFall calls *streams*. To simplify the calculations, each stream of 30 values is averaged into one single CSI value for each stream. To do this, they take the CSI value across the full frequency range of subcarriers as in Wu et al. (2012). However, they found that this led to a lack of frequency diversity among CSI subcarriers especially for rich, multi-path environments. As a result, they implemented a moving average filter at each time instance t. They have only used the amplitude for activity/anomaly detection and fall detection at $5GHz$ but not the CSI phase.

RT-Fall, Wang et al. (2017a), improves on the methods of WiFall by utilising the phase difference between antennas in a MIMO-OFDM channel. They claim that WiFall can only detect falls for four types of predefined activities of walking, sitting, standing up and falling. With WiFall, these activities cannot be performed continuously either (Wang et al., 2017a). They proposed to use the phase difference over 2 antennas at $5GHz$ as a more sensitive indicator for activity segmentation and fall detection. They find that human activities affect different subcarriers in a similar way and adjacent subcarriers behave similarly. They also find that the variance of the phase difference across 2 antennas is the sum of the variance on each antenna. This implies the phase difference is more sensitive to environment changes than the CSI amplitude making it a more suitable indicator of a fall (Wu et al., 2015). They also discovered a sharp power profile decline pattern of a fall in the time-frequency domain. Using these findings, they can segment human activities shown in the CSI data (See Section 2.2.3). They achieve a 14% higher sensitivity and 10% higher specificity on average than WiFall (Wang et al., 2017a).

2.2.3 Activity Segmentation

Taking the CSI data and segmenting it based on different human activities taking place in the environment is crucial. There are many different methods to this such as the one implemented by CARM (Wang et al., 2017b). They used two models to determine a human activity from CSI data: CSI-*speed model* and a CSI-*activity model*. The speed model quantifies the correlation of CSI value dynamics with human movement speeds. The activity model quantifies the movement speeds of different human body parts and a human activity. These 2 models build a correlation between CSI data and the specific activity classifying it appropriately. This ties in with the activity segmentation aspect of my project prior to classification.

One of the key steps in activity segmentation is *feature extraction*. In RT-Fall, they use the state transition of the CSI phase difference variance along with the sharp power profile decline pattern to determine the stationary activities and then to further determine fall from fall-like activities (Wang et al., 2017a). They also found that interpolation is needed in the pre-processing of data. For other pre-processing, they have followed PhaseU and WiFall. As a wireless channel is a shared channel, the devices use random access to share the channel. The received packets are not evenly spaced in the time domain as the transmitter and receiver make no attempt to synchronise timing before transmitting. As the samples are not continuous or evenly spaced in the time domain, Time Frequency analysis for feature extraction cannot take place. To solve this, they follow Nandakumar et al. (2014) in performing a 1-D linear interpolation algorithm on the raw CSI data. This is followed by band-pass filtering to clean the CSI data of noise and irrelevant frequency components such as breathing or slight movement which occurs in the range of $[0,4]Hz$ (Wang et al., 2017a). To separate fall and fall-like activities, they determine a proper trace back window size for a falling activity.

2.2.4 Feature Extraction & Classification

The first step to reliable classification is feature extraction. In Wang et al. (2017a), they use the first six features found by Wang et al. (2017c) and add the *TimeLag* and *Power*

Decline Ratio as their own. *TimeLag* is the time delay of the state transition point of an activity between the band-pass filtered and the raw CSI phase difference. In WiFall (Wang et al., 2017c), they only used the CSI amplitude to extract their six features while in RT-Fall, they extract the first six features from *both the CSI amplitude and phase difference* and extract the last two mentioned features from *CSI phase difference only*.

WiFall uses two classifiers, a one-class Support Vector Machine (SVM) and the Random Forest Decision Tree classifier (Wang et al., 2017c). RT-Fall uses a different type of SVM classifier called a v-SVM classifier (Scholkopf et al., 2000). There needs to be an *objective* class (falling) and a *non-objective class* (other activities) which you want to classify correctly and train the model using training (segmented & labelled) data. In WiFall, falling belongs to the objective class while other human movement belongs to the non-objective class. In RT-Fall, most non-fall like human movements have been removed and the segmentation methods were performed on the fall and fall-like activities. In this case, falling belongs to the objective class and fall-like movements (e.g., standing up) belong to the non-objective class (Wang et al., 2017c,a). With user feedback, the classifier model can be tuned by relabelling incorrect classifications and updating the model. Both WiFall and RT-Fall have used LibSVM to develop their classification models (Chang and Lin, 2011).

Other CSI fall detection projects have used SVMs (Zhang et al., 2015), k-NN (Nearest Neighbour) algorithms (Cao et al., 2017), Decision Trees (Wang et al., 2017c; Dayal et al., 2016) and Neural Networks (Dayal et al., 2016). There are a number of differences and consequences for using different algorithms. For example, Decision Trees are useful because they do not assume any inherent relationship between each of the features while SVMs with a Gaussian Kernel assume a Gaussian distribution of the feature vector. (Dayal et al., 2016). For rich multi-path environments with many classes, one-class SVMs are not suitable and perform poorly where Decision Trees do well.

3. Completed Work

I will describe the technical work completed to date in order of completion and present any results obtained before moving on to future work to be completed.

3.1 Installing the CSI tool

The first step was to install an Intel Wi-Fi Link 5300 Wireless NIC into a suitable laptop. The card itself has a mPCIe connection and many computers with this connection for their NIC are discontinued. They now use the newer M.2/NGFF connection. I installed the Intel NIC into a Toshiba Satellite L750 laptop connecting two antenna cables to the card (two antennas in Toshiba laptop). At the same time, I purchased a mPCIe to PCIe adapter which could be installed into a desktop computer. This was installed into a computer in the Project Lab in UCD with all three antennas connected for another Intel NIC. The Linux CSI tool is built to work with the Intel Wi-Fi Link 5300 using customised Intel firmware and customised versions of the *iwlwifi* open-source driver (Halperin et al., 2011).

To install the Linux CSI tool, Halperin et al. (2011), I first loaded Ubuntu 14.04 LTS on both the laptop and desktop. Following the instructions on the tool's website and reloading the *iwlwifi* driver each time the laptop turns on, enables the CSI gathering functions. Running the `log_to_file` command as instructed on the tool's README while also sending ICMP (ping) requests to an access point or associated router (without encryption) leads to CSI data being gathered. The CSI data is added to a .dat file whereby the CSI matrices are of the form as mentioned in Section 2.

I used my smartphone's Wi-Fi hotspot initially to verify the tool worked correctly. The phone's $2.4GHz/5GHz$, two antenna wireless card was able to transmit data at 802.11n rates. I created a 1×2 SIMO network initially. I then tested the CSI tool with a commercial Virgin Media access point at home with three antennas and a $20MHz$ bandwidth. I demonstrated that the tool did not work with an access point with security enabled but it worked successfully with the access point not connected to the internet, thus keeping the network safe. This is due to not enough code space on the NIC for both beamforming software paths and encryption

software paths which means CSI data can only be retrieved from non-encrypted access points.

I purchased a TP-Link Archer C6 (AC1200) which I could use without an Ethernet connection to act as a transmitter for more antennas/customisability. This would allow me to perform experiments at $2.4GHz/5GHz$, channel bandwidth of $20MHz/40MHz$ and a range of WLAN Channels. The laptop would act as a receiver creating a 2×2 MIMO network. I attempted to set up *hostapd* on both the desktop and laptop but unfortunately, I found that the Intel NIC was geo-location locked to not allow AP (Access Point) mode at $5GHz$ in its EEPROM which I was unable to change. As I would need the CSI phase at $5GHz$, I abandoned this work direction.

3.2 Collection of CSI Data

CSI data is gathered when the client (laptop) with the modified *iwlwifi* drivers reloaded is connected to an un-encrypted AP. In the Ubuntu terminal, the `log_to_file` command is run. In a new terminal tab, a ping command is entered using the AP's gateway IP address. To maintain compatibility with other projects, I implemented a packet/sampling rate of *100pkts/sec.* and a packet size of 2000 bytes. The command to send an ICMP request to the AP from the client computer is `sudo ping −i 0.01 −s 2000 <AP IP Address>`. When the packets are transmitted back to the receiver at 802.11n rates, the CSI tool command logs the data to the accompanying .dat file as shown in Fig. 4.1.

3.2.1 MIMO Data Collection

In terms of the 2×2 and 2×3 MIMO wireless networks using the Archer C6 router, I set the experiment up in a variety of environments usually separating transmitter and receiver by 4-5m. Both transmitter and receiver were located in a laboratory/classroom environment to mimic the intended locations as closely as possible. I needed useful data for a variety of activities for the data pre-processing stage of the system. I initially started with separate CSI data files where each activity was performed separately. Many human activities were carried out to see if there was any patterns in the CSI data such as standing, walking, sitting

down, falling and standing still like RT-Fall. At this initial stage I was only concerned with identifying patterns in the CSI data.

3.3 CSI Data in MATLAB

I used MATLAB to process and analyse the obtained CSI.dat file. From here, I needed to unpack the binary format of the CSI data file using some of the provided scripts/MATLAB files by the tool's creators (Halperin et al., 2011). To unpack the binary data, I used a **MEX-file** compiled from the provided `read_bfee.c` file. Using the `read_bf_file` function I could store the CSI data in a variable called *csi_trace* which would contain every packet's CSI data in cell structures. Each cell structure contains a timestamp, the number of beamforming measurements, RSS of each antenna, noise and many more. The *csi* field contains the $N_{TX} \times N_{RX} \times 30$ CSI matrix. Examining the CSI data, I can see there are a number of transmitter-receiver pair values for each subcarrier in complex number form $a + jb$. I noticed that there is some cross-talk/interference on antenna 3 (not connected) from antenna 1 and 2 in the receiver NIC. The creators of the Linux CSI tool experienced this also and it is not present in the desktop computer which has all 3 antennas connected. The CSI data at this point is in Intel's internal reference level and as a result, I needed to convert the CSI to absolute units (Halperin et al., 2011).

The CSI was converted using `get_scaled_csi()` which combines the RSSI (Received Signal Strength Indicator) and AGC (Automatic Gain Control) (gives constant output for range of inputs) to find the RSS in `dBm` and including noise to get the SNR. I built a MATLAB script which could take all of the received packets and perform this operation in order as `get_scaled_csi()` can only handle one cell structure at a time. This led to a CSI matrix in absolute units (voltage space) in the form $N_{TX} \times N_{RX} \times 30$. Obtaining the logarithm of one packet's CSI amplitude would allow me to plot the SNR and I could plot the CSI phase over the full range of 30 subcarriers also as shown in Fig. 4.2. Knowing that I was able to accurately obtain and simply process the CSI data for a 2×2 and 2×3 MIMO-OFDM system, I needed to implement the first stage of data processing for the amplitude and phase.

3.3.1 Data Processing

I found that the CSI amplitude is affected by human activities as seen in WiFall (Wang et al., 2017c). They affect different streams independently and affect different subcarriers in a similar way. As a result, I can average the CSI amplitude samples for each stream into one value instead of 30 using the following equation and the frequency of each subcarrier around the centre frequency of $f_0 = 5GHz$ using each subcarrier frequency f_j and $N = 30$ subcarriers:

$$CSI_{ij}(K) = \frac{1}{N}\sum_{j=1}^{N}\frac{f_j}{f_0}|CSI_{ij}(K)| \tag{3.1}$$

This equation was applied to the following matrix which I reconstructed in MATLAB for K packets for each transmitter-receiver pair (TX Antenna i & RX antenna j):

$$\mathbf{CSI}_{ij} = \begin{bmatrix} h_{ij,1}(1) & h_{ij,2}(1) & \cdots & h_{ij,30}(1) \\ h_{ij,1}(2) & h_{ij,2}(2) & \cdots & h_{ij,30}(2) \\ \vdots & \vdots & \ddots & \vdots \\ h_{ij,1}(K) & h_{ij,2}(K) & \cdots & h_{ij,30}(K) \end{bmatrix} \tag{3.2}$$

I then produced CSI amplitude plots using these equations/matrices to verify my methods in LOS/NLOS conditions (See Fig.4.5). The CSI measured phase of a subcarrier f is presented as shown as found in Sen et al. (2012):

$$\hat{\phi}_f = \phi_f + 2\pi f_f \Delta t + \beta + Z_f \tag{3.3}$$

where ϕ_f is the true phase, Δt is time lag at the antenna, β is an unknown constant phase offset and Z_f is measurement noise. Raw true phases are unusable as the timing lag is different for each packet. The phase difference is computed as:

$$\Delta\hat{\phi}_f = \Delta\phi_f + 2\pi f_f \epsilon + \Delta\beta + \Delta Z_f \tag{3.4}$$

where $\Delta\phi_f$ is the true phase difference, $\epsilon = \Delta t1 - \Delta t2$ which are time lags at each antenna and $\Delta\beta$ is the unknown constant phase difference offset. To mitigate random noise/outliers, I performed a linear transform on the measured phase to produce a calibrated phase for subcarrier f:

$$\tilde{\phi}_f = \hat{\phi}_f + \frac{\phi_N - \phi_1}{k_N - k_1}k_f - \frac{1}{N}\sum_{j=1}^{N}\phi_j \tag{3.5}$$

where k_N represents the coefficient of the N^{th} subcarrier (IEEE802, 2009, p.53), $\hat{\phi}_f$ is

the measured phase and ϕ is the phase of each numbered subcarrier (N, 1 or j). According to PhaseU (Wu et al., 2015), the phase difference between two antennas is the sum of the variance of each antenna (transmitter-receiver pair) as they are both independent:

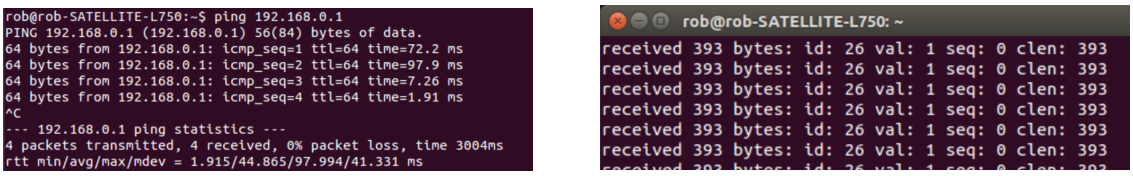
$$\sigma^2_{\Delta\tilde{\phi}_f} = \sigma^2_{\Delta\tilde{\phi}_{f,1}} + \sigma^2_{\Delta\tilde{\phi}_{f,2}} \tag{3.6}$$

The result of this linear transform on the phase difference can be seen in Fig. 4.7. I then carried out extensive experiments with these methods to prove that my methods were correct before performing one test where a video was also taken. The subject in the video was myself and I entered the area between the transmitter and receiver about 5m apart before standing still and falling. It would be easy to notice activities/patterns in the CSI data if I could compare them against the video. From here, I spent time gathering extensive periods of data greater than 10mins for my future work.

4. Initial Results

4.1 Initial CSI Data Gathering

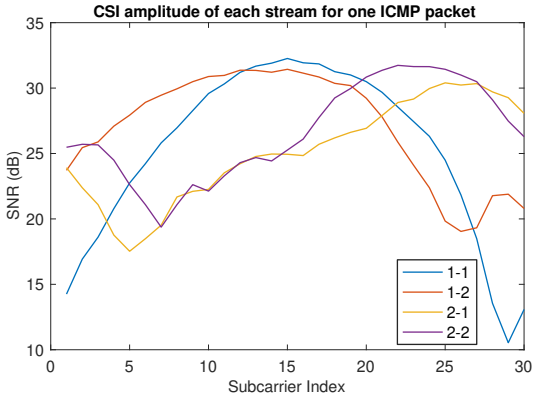
These are the results of gathering initial CSI data using Linux shell commands into a .dat file.



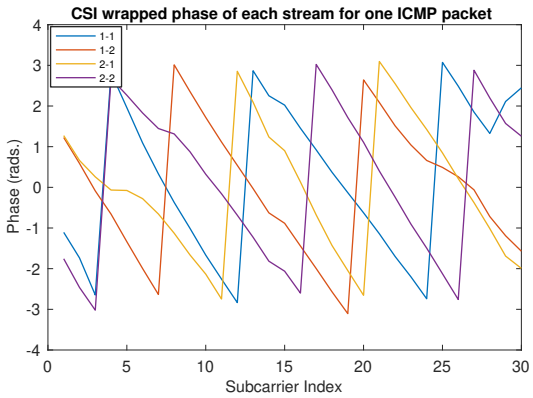
(a) `ping` gateway IP at 192.168.0.1 **(b)** Receiving data using `log_to_file`

Figure 4.1: Using `ping` & `log_to_file` to send and receive ICMP packets between Archer C6 router and laptop

Hence, I was able to present one packet's SNR and CSI-phase across the 30 subcarriers once the data was converted to absolute units in Figure 4.2. Each stream conveys similar information as seen by each coloured trace. However, there isn't enough information to understand anything about the surrounding environment.



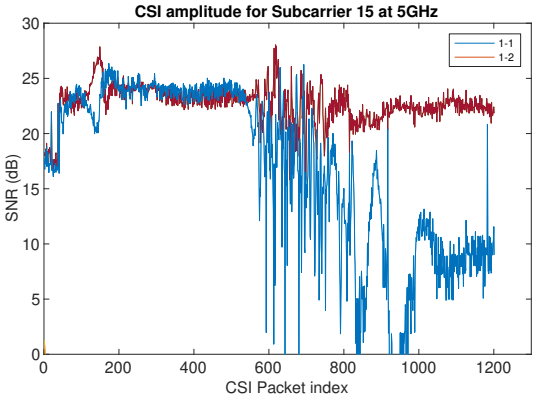
(a) CSI amplitude of each stream for one packet



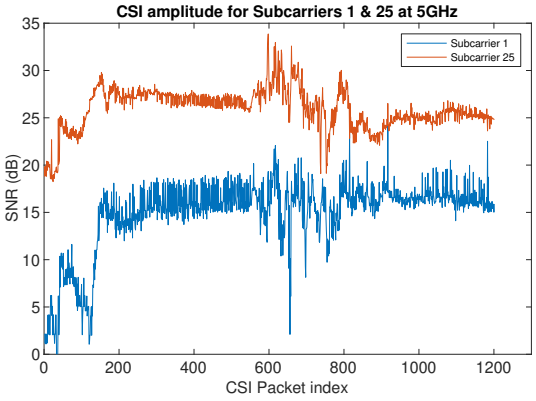
(b) Wrapped phase of each stream for one packet

Figure 4.2: (2×2 MIMO-OFDM channel) CSI amplitude experiences frequency dependent fading at the edges of the channel bandwidth (Subcarriers 1 and 30). CSI phase is wrapped between π and $-\pi$ rads. in a typical '$zig-zag$' pattern.

The CSI phase can be unwrapped using `unwrap` in MATLAB to verify the phase according to RT-Fall and Zhuo et al. (2017). It was decreasing for each consecutive subcarrier. However, each stream has 30 values per packet. A link between each of these needed to be found to simplify the calculations using streams such as 1-1 below (TX & RX = 1):



(a) Different Streams for Subcarrier 15



(b) Different subcarriers in Stream 1-1

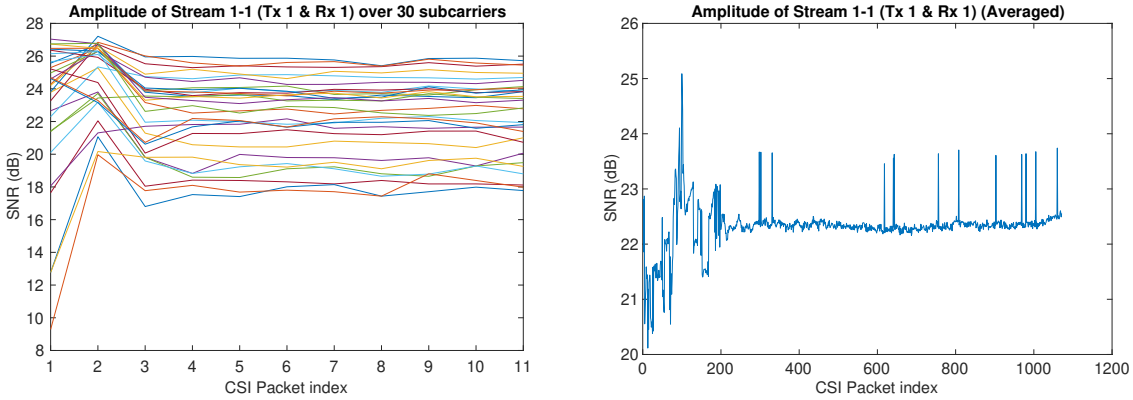
Figure 4.3: Demonstrating that different streams react differently to human activity in (a) but different subcarriers in the same stream act similarly in (b)

In Figure 4.3 (a), the fall occurring at the 600th packet affects streams 1-1, 1-2 differently due to the large change in the blue plot compared to red. However, human activities affect subcarriers similarly as seen in Figure 4.3 (b). This allows me to use the Equation 3.1 to average the subcarriers into 1 value for each stream.

4.2 Data Processing

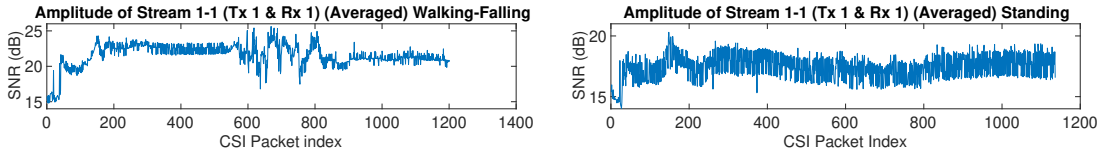
4.2.1 CSI Amplitude

The result of using Eqn. 3.1 is shown below where the 30 subcarriers shown in Fig. 4.4 (a) are calculated and converted to one value per packet in Fig. 4.4 (b). The overall shape of the subcarriers is preserved but the full SNR range is not (some subcarriers have values around 27dB while Fig. 4.4 (b)'s maximum point is 25dB). This implies a loss of frequency diversity in the calculation making it problematic for fall detection as seen by WiFall.

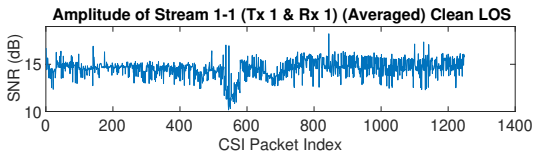


(a) Subcarrier amplitudes in Stream 1-1 **(b)** Averaged amplitude. One value per packet

Figure 4.4: Using the findings from Fig. 4.3, I apply Eqn. 3.1 to obtain a simpler amplitude plot in (b)



(a) Person falling (600th packet) (b) Person standing still



(c) Clean LOS from Tx-Rx

Figure 4.5: CSI amplitude with a person falling, standing and then being absent (Clean LOS) in separate tests

I carried out a number of tests using the averaged CSI amplitude as shown in Fig. 4.5.

Mobile and immobile activities obtain different CSI amplitude variance with time. It is clear in (a) that something has occurred but this "fall" could also be confused with the fluctuation in (c) caused by the small disturbance closer to the receiver with nobody in the environment. It is hard to differentiate and recognise patterns between each of the situations.

4.2.2 CSI Phase

I have noticed that CSI phase is a more granular indication of a fall in an environment. I plotted the phase difference, as noted in PhaseU (Wu et al., 2015), below:

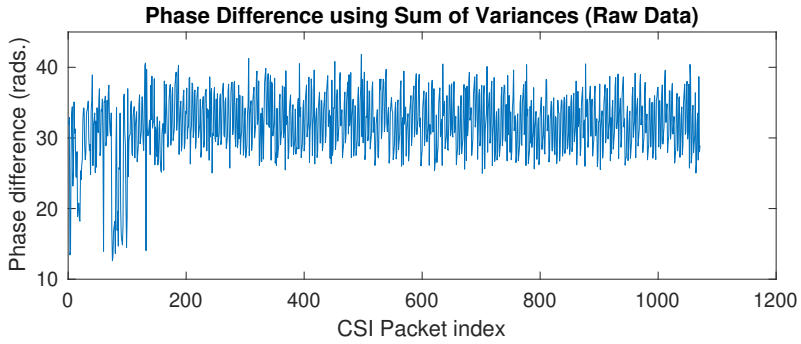


Figure 4.6: Phase difference containing phase offsets and outliers

This plot above contains the unknown phase offset and timing offset as alluded to in Eqn. 3.3 & 3.4. No recognisable patterns in the data are noticeable. Using Eqn. 3.5, I can obtain a cleaner plot:

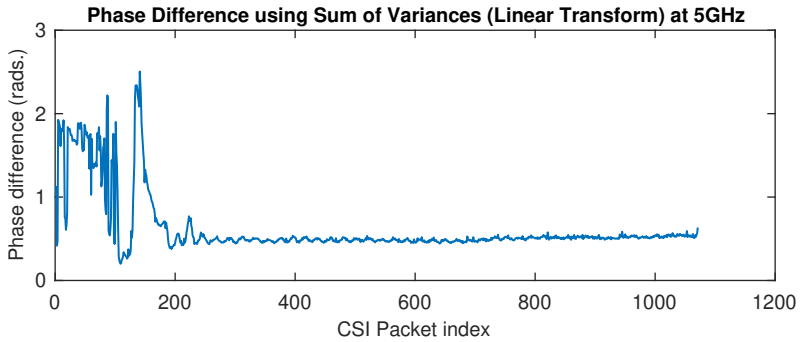


Figure 4.7: Phase difference from Fig. 4.6 after applying linear transform in Eqn. 3.5

This plot has all phase offsets and outliers removed. The data is now interpretable in the time domain (after interpolation) and human activities can be noticed.

4.2.3 Phase Difference & Amplitude for Fall Detection

Clearly, I have demonstrated the phase difference is a much more granular indicator of a human activity in an environment. I tested this using a "walking-fall" as shown below:

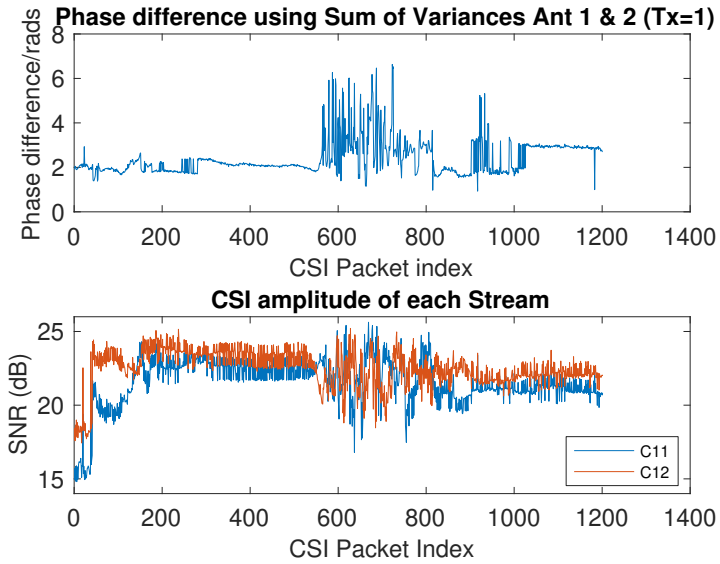


Figure 4.8: CSI amplitude and transformed phase difference for 4m environment between TX and RX with a person walking into the environment, falling and standing up again

Both the amplitude and phase difference experience a sudden disturbance at the start. This is a combination of the rate selection algorithm of the Archer C6 router adjusting and the test subject walking close to the receiver creating a multipath environment. The fall occurring just before the 600th packet creates a very large disturbance in the phase difference. The CSI amplitude suffers a disturbance also. However, the amplitude's response to the fall is very similar to the disturbance caused by the person walking between the 200th and 400th packet. The amplitude can be used for activity detection as in WiFall but isn't clear enough for fall detection. The phase difference clearly shows where the fall finished, save for a few oscillations. It is a much more sensitive indicator and the transformed phase is much less noisy than the amplitude. In a more complex MIMO-OFDM channel, the phase difference resolution will improve due to the introduction of more antennas into the system and thus, performance for LOS/NLOS environments will greatly improve.

5. Future Work

5.1 Signal Processing

Up until now, I have researched and implemented many different pre-processing techniques which has led to the CSI amplitude and phase data being more interpretable. The lack of synchronisation between transmitter and receiver as mentioned in Section 2.2.3 presents two issues: The sampled CSI data may not be continuous which is needed for feature extraction during a fall and I am unable to obtain a spectrogram of the data if they are unevenly spaced in the time domain. Feature extraction is extremely hard and inaccurate with this issue. To resolve this, I will implement a **1-D linear interpolation algorithm** on the raw CSI. This is done in MATLAB using `interp1` on each transmitter-receiver pair matrix (CSI_{ij}). This will allow me to see accurate receiving times for each packet.

To make the training of my classifiers easier, I can perform band-pass filtering on the interpolated CSI data. This can filter out the lower frequency components as seen by RT-Fall in the range [0,4]Hz as most fall or fall-like activities occur in the range [5,10]Hz (Wang et al., 2017a).

5.2 Activity Segmentation

Having obtained real-life CSI data from an indoor environment of over 10mins to mimic a real test environment such as a bedroom, I need to find the finishing point of a fall or fall-like activity in this large amount of CSI data. I will use a **threshold-based sliding window** firstly. I will need two stable signal streams across multiple sliding windows from which I can calculate their mean μ and normal standard deviation σ. I can calculate if an equation involving these is less than a threshold value detecting their current state (stable/fluctuating).

When there is a transition in signal state, I can mark this time as the *start* and the time of another transition as the *end* which gives me an activity window. From here, I can segment the activities by determining the appropriate trace back window size.

5.3 Feature Extraction & Classification

As seen in RT-Fall and WiFall, which use similar number and type of features for recognising a fall, I will need to gather and research further features that could be an indication of a fall occurring in the CSI data. All of these features will form the input to my classifiers for training the classification model.

I will then be able to design a binary classifier such as a SVM (Support Vector Machine) for detecting between a fall and a fall-like activity as these are the most similar activities and could lead to the highest missed/false detection rate. The training dataset will need to be created in a usual living room and the human activities will need to be segmented, labelled and passed into the SVM classifier along with the extracted features. With user feedback, classification errors will be relabelled to adjust the model. I can then test the model again with unseen data.

I am hoping to implement a similar method using a Decision Tree classifier such as RandomForest and perhaps, a Neural Network, if time allows and the classification accuracy from SVM is not high enough. Currently, I am in the process of extensive & prolonged CSI data gathering for the activity segmentation step and the enhanced data pre-processing involving interpolation and filtering.

Bibliography

E. Bjrnson, M. Bengtsson, and B. Ottersten. Optimal Multiuser Transmit Beamforming: A Difficult Problem with a Simple Solution Structure [Lecture Notes]. *IEEE Signal Processing Magazine*, 31(4):142–148, July 2014. ISSN 1558-0792. doi: 10.1109/MSP.2014.2312183. 6

W. Cao, X. Liu, and F. Li. Robust device-free fall detection using fine-grained wi-fi signatures. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 1404–1408, March 2017. doi: 10.1109/IAEAC.2017.8054245. 11

A. Chadha, N. Satam, and B. Ballal. Orthogonal Frequency Division Multiplexing and its Applications. *International Journal of Science and Research*, 2:325, 01 2013. 4, 5, 6

C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 11

J. Dai, X. Bai, Z. Yang, Z. Shen, and D. Xuan. PerFallD: A Pervasive Fall Detection System Using Mobile Phones. pages 292–297, 10 2010. doi: 10.1109/PERCOMW.2010.5470652. 8

S. Dayal, H. Narui, and P. Deligiannis. Human fall detection in indoor environments using channel state information of wi-fi signals. 2016. 11

C. S. Florence, G. Bergen, A. Atherly, E. Burns, J. Stevens, and C. Drake. Medical costs of fatal and nonfatal falls in older adults. *Journal of the American Geriatrics Society*, 66(4): 693–698, 2018. doi: 10.1111/jgs.15304. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/jgs.15304`. 1

S. M. Friedman, B. Munoz, S. K. West, G. S. Rubin, and L. P. Fried. Falls and fear of falling: which comes first? a longitudinal prediction model suggests strategies for primary and secondary prevention. *Journal of the American Geriatrics Society*, Aug 2002. URL `https://www.ncbi.nlm.nih.gov/pubmed/12164987`. 1

D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Tool Release: Gathering 802.11n Traces with Channel State Information. *ACM SIGCOMM CCR*, 41(1):53, Jan. 2011. 7, 8, 12, 14

K. Hari. *Channel Models for Wireless Communication Systems*, volume 158, pages 47–64. 10 2011. doi: 10.1007/978-1-4419-6111-2_3. 3, 4

IEEE802. Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac)and physical layer (phy) specifications amendment 5: Enhancements for higher throughput. *IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009)*, pages 1–565, Oct 2009. ISSN null. doi: 10.1109/IEEESTD.2009.5307322. 5, 6, 15

R. Nandakumar, B. Kellogg, and S. Gollakota. Wi-Fi Gesture Recognition on Existing Devices. 11 2014. 8, 10

R. Nee, V. Jones, G. Awater, A. Zelst, J. Gardner, and G. Steele. The 802.11n MIMO-OFDM standard for wireless LAN and beyond. *Wireless Personal Communications*, 37:445–453, 05 2006. doi: 10.1007/s11277-006-9073-2. 4, 5

Paul and Ogunfunmi. Wireless LAN comes of age: Understanding the IEEE 802.11n amendment. *Circuits and Systems Magazine, IEEE*, 8:28 – 54, 04 2008. doi: 10.1109/MCAS.2008.915504. 3, 4

D. Samardzija and N. Mandayam. Pilot-assisted estimation of mimo fading channel response and achievable data rates. *IEEE Transactions on Signal Processing*, 51(11):28822890, 2003. doi: 10.1109/tsp.2003.818158. 4

B. Scholkopf, A. Smola, R. Williamson, and P. Bartlett. New support vector algorithms. *Neural computation*, 12:1207–45, 06 2000. doi: 10.1162/089976600300015565. 11

S. Sen, B. Radunovic, R. R. Choudhury, and T. Minka. You are facing the mona lisa: spot localization using phy layer information. In *MobiSys '12*, 2012. 15

J. M. Stokes. Falls in older people: Risk factors and strategies for prevention (2nd edn) - by stephen lord, catherine sherrington, hylton menz, and jacqueline close. *Australasian Journal on Ageing*, 28(1):47–47, 2009. doi: 10.1111/j.1741-6612.2009.00347.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1741-6612.2009.00347.x. 1

H. Wang, D. Zhang, Y. Wang, J. Ma, Y. Wang, and S. Li. Rt-fall: A real-time and contactless fall detection system with commodity wifi devices. *IEEE Transactions on Mobile Computing*, 16(2):511–526, Feb 2017a. ISSN 2161-9875. doi: 10.1109/TMC.2016.2557795. 9, 10, 11, 21

J. Wang, Z. Zhang, L. Bin, S. Lee, and R. Sherratt. An Enhanced Fall Detection System for Elderly Person Monitoring using Consumer Home Networks. *Consumer Electronics, IEEE Transactions on*, 60:23–29, 02 2014a. doi: 10.1109/TCE.2014.6780921. 8

W. Wang, A. X. Liu, and M. Shahzad. Gait Recognition Using Wifi Signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp 16, page 363373, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450344616. doi: 10.1145/2971648.2971670. URL https://doi.org/10.1145/2971648.2971670. 8

W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu. Device-free human activity recognition using commercial wifi devices. *IEEE Journal on Selected Areas in Communications*, 35(5):1118–1131, May 2017b. ISSN 1558-0008. doi: 10.1109/JSAC.2017.2679658. 10

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*, 09 2014b. doi: 10.1145/2639108.2639143. 8

Y. Wang, K. Wu, and L. M. Ni. WiFall: Device-Free Fall Detection by Wireless Networks. *IEEE Transactions on Mobile Computing*, 16(2):581–594, Feb 2017c. ISSN 2161-9875. doi: 10.1109/TMC.2016.2557792. 8, 9, 10, 11, 15

D. Wild, U. S. Nayak, and B. Isaacs. How dangerous are falls in old people at home? *British medical journal (Clinical research ed.)*, Jan 1981. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1504022/. 1

C. Wu, Z. Yang, Z. Zhou, K. Qian, Y. Liu, and M. Liu. Phaseu: Real-time los identification with wifi. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 2038–2046, April 2015. doi: 10.1109/INFOCOM.2015.7218588. 9, 16, 19

K. Wu, J. Xiao, Y. Yi, M. Gao, and L. Ni. FILA: Fine-grained indoor localization. *Proceedings - IEEE INFOCOM*, pages 2210–2218, 03 2012. doi: 10.1109/INFCOM.2012.6195606. 8, 9

Y. Xie, Z. Li, and M. Li. Precise Power Delay Profiling with Commodity WiFi. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, page 5364, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3619-2. doi: 10.1145/2789168.2790124. URL http://doi.acm.org/10.1145/2789168.2790124. 7

D. Zhang, H. Wang, Y. Wang, and J. Ma. Anti-fall: A non-intrusive and real-time fall detector leveraging csi from commodity wifi devices. 07 2015. doi: 10.1007/978-3-319-19312-0_15. 11

Y. Zhuo, H. Zhu, H. Xue, and S. Chang. Perceiving accurate csi phases with commodity wifi devices. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, May 2017. doi: 10.1109/INFOCOM.2017.8056964. 17