Human Activity Recognition: Speed Estimation using SVM and CSI

Hannan Adil *1233384*
Mohamed Amine Allani *1249462*
Jörg Schäfer
Frankfurt University of Applied Sciences
Faculty of Computer Science and Engineering
Nibelungenplatz 1 D-60318 Frankfurt am Main

***Abstract*—Human Activity Recognition (HAR) is a field of study that aims at predicting actions or movements made by a person. Multiple technologies can help achieve this goal from wearable sensors, to cameras as well as wireless signals. In recent years, Wifi signals have gained increasing interest among the scientific community. Indeed human bodies interfere in the generated signals, creating identifiable patterns. One metric, namely Channel State Information (CSI) is particularly sensitive to human bodies in its immediate environment. In the scope of this work, an attempt at exploiting CSI data to estimate the speed of a person was made. CSI data corresponding to three different velocities was collected and used to train an SVM classifier. Mutiple preprocessing steps were performed on the data in order to facilitate the training, among which, Discrete Wavelet Transform (DWT), Power Spectral Density (PSD), Haart wavelet analysis and Principal Component Analysis (PCA). Unfortunately, the experiment didn't yield good results. Indeed, on the same data and using the same preprocessing steps, the accuracy of the classifier was ranging from 25% to 70%.**

I. Introduction

Human Activity Recognition (HAR) technologies aims to monitor human behavior in an indoor or outdoor environment. These technologies have multiple fields of applications, all of which fall under the Internet of things paradigm, such as the monitoring of disabled and elderly people in healthcare or to optimize energy consumption in smart homes or even in security. [1]

Current HAR systems, rely on multiple technologies to monitor humans. These can be wearable devices equipped with sensors, such as gyroscopes to measure the orientation or accelerometers to estimate the speed. The data harnessed by the sensors is then processed and the activity of the subject is predicted using machine learning algorithms. The solution proposed in [9] has produced high accuracy results, with more than 90% accuracy using Support Vector Machine (SVM). However, using wearable sensors makes relying on the subjects to constantly wear the device crucial in order for the system to accurately predict the activity of a system at all time. This may not be feasible due to the fact that the subject can forget to wear his device or not want to wear it simply because such devices can be cumbersome. [1]

Other solutions require the use of cameras and image recognition techniques instead of wearable devices. However these techniques require the subject to be in the line of sight (LOS) of the camera. Furthermore, using a camera can violate the privacy of the subject and may not be possible in certain environments.

Due to the limitation of the mentioned techniques, other solutions are being investigated. Such solutions rely on radio signals which do not rely on wearable devices and do not violate the privacy of the subject. Furthermore, radio waves work both when the subject is in the line of sight or non-line of site of the device. Thus radio frequency based devices, such as Wifi-devices, have been the subject of extensive research in recent years in the field of HAR. Indeed Wifi-enabled devices are found in every household nowadays and therefore, such setup would require minimal to no new hardware. A typical setup requires setting up an transmitter device and and an emitter device to detect human activity. The signal transmitted between these two devices can be disturbed by the presence and movement of humans, due to the fact that a human body is 70% water. The fluctuation in the signal is then used by the system to recognize human activity.

In previous research, the Received Signal Strength Indicator (RSSI) has been used in HAR. RSSI is a metric that measures signal strength. [10] Although, RSSI is simple to use and available in most hardware device, it has it's disadvantages. Indeed RSSI remains highly variable, even in static indoor environments and for stationary devices. This can introduce recognition errors. [2]

Alternatively, Channel State Information (CSI) can be used in this context. CSI is provided by a WiFi network interface cards (NIC). In contrast to RSSI, which offer a right MAC layer information, CSI offers detailed physical-layer information including the amplitude and phase information of each subcarrier, making it more robust in complex environments [3]. Therefore, CSI is far superior to RSSI when applied. Indeed, CSI has seen intensive research in its application in HAR in the recent years; including localisation, gesture recognition [4] and the analysis of customer's behavior [6].

In the scope of this paper, CSI data was used to estimate the speed of a person.

A. Related Work

In the scope of [8], CSI information was exploited in order to recognise human activity. The goal of this experiment was to predict when a human sits, stands, walks, runs or when the room is empty. The CSI data harnessed in the scope of their experiments included a line of sight (LOS) setting and a non-line of site (NLOS) setting. Their experiments included a solution using SVM and another one using long short-term memory (LSTM) neural networks. Before applying SVM, a series of preprocessing techniques, including denoising and feature reduction techniques, were used on the data before feeding it to the classifier. These include a Discrete Wavelet Transform (DWT), a Power Spectral Density (PSD) analysis, the calculation of the frequency of center of energy and a Haart wavelet analysis for the processing and denoising of the signal. To reduce the dimensionality of the data, a Principal Component Analysis (PCA) was applied. The statistical properties of this processed data is then calculated and the result is fed to the SVM classifier. For the second approach, only denoising using DWT was applied on the data before training the LSTM network. Both experiments yielded good results with an accuracy of over 90% with both the SVM classifier and the LSTM network.

B. The Proposed Method

In the scope of this paper, an attempt in exploiting CSI data to estimate the speed of a person in an indoor environment. The performed experiment aimed to train an SVM classifier on CSI data samples into three classes, each symbolizing a different velocity. This experiment is meant as a proof of concept; an intermediate step in building a regression model that would estimate the estimate the speed of person with a high enough accuracy.

The preprocessing algorithm applied in the experiment was based on the preprocessing algorithm of [8]. The algorithm included applying packet wise DWT of the amplitudes of the CSI data. After that PCA was applied on the result to reduce the dimensionality of the data. On the first components, a PSD analysis was applied and the Frequency of center of energy matrix was calculated. After that a Haart wavelet analysis was performed, from which the statistical properties were calculated. As a last step, PCA was applied again in order to reduce the dimensionality of the data. The resulting processed data was then used to train an SVM classifier.

II. The Experiments

This section discusses the hardware that was used to collect the data, the data and human activity that was simulated as well as the location and framework of the experiment.

A. The Hardware

Collecting tailored measurement of CSI data is primordial in this experiment. In most NIC, only high level information on network conditions, such as RSSI, is available. Fortunately, the solution proposed by [7] provides a practical solution to harness CSI data.

Generally, 802.11n standard NICs measures the channel state for every packet received in order to compensate for channel effects before demodulation. Normally, this CSI information is not reported to kernel of the host machine. However, the proposed solution uses a modified version of the IWL 5300 driver, initially used for the Intel WiFi Link 5300 wireless NIC to access the data. By activating the debug more, it is possible to dump the CSI data into the kernel for each successfully received 802.11n packet. Then, using the modified driver developed by [7], this data is dumped into the main memory which makes it accessible to higher level applications, either online, or offline, by saving the results into a file. The solution provides a MATLAB library which allows users to process the data. The provided CSI represents the signal strength i.e amplitude and phase information for each packet sent from each subcarrier.

In the scope of this project, three Lenovo laptops equipped with the Intel 5300 NIC. On each laptop, The kernel version is 4.2.0-42 and a 64 Bit Ubuntu with the version 14.04 LTS is used as operating system. Furthermore, the modified driver implemented in [7] is used by the kernel. The toolkit can be accessed on all machines through a command line interface (CLI). Of the used laptops, two were as receivers and one was used as transmitter. In the scope of this experiment they were set to send and receive on a 40 MHz channel. For each measurement i.e sample, a 1000 packets were sent at an interval of 50ms, resulting in a total time of 1sec for each measurement. The size of each packet is 100 bytes. The resulting CSI data was then collected in a file from both receivers. The generated files are then ready for further processing by the MATLAB library.

B. The Data

The goal of this experiment is to estimate the velocity of a person indoor using CSI data. Therefore, each sample represents one of three different velocities. A total of 300 samples per receiver laptop were collected with 100 samples for each velocity. However, it is very difficult to generate consistent data using human subjects, because of the fact that it is near impossible to run at the exact same speed during each measurement. Therefore, a different system, inspired from the pendulum, was constructed. The core of the idea is that by swinging the pendulum from angle α, is it possible to calculate the velocity of the pendulum by applying the following formula [11]:

$$v = \sqrt{2 \times g \times L \times (1 - cos(\alpha))}$$

Where g is the acceleration of gravity, L is the length of the rope of the pendulum and α is the angle made by the rope before releasing the pendulum. The following figure represents a pendulum:

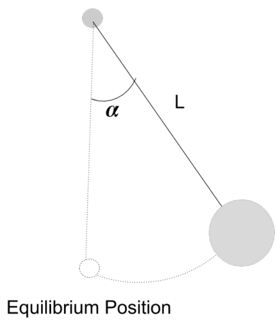


Fig. 1: Pendulum

By using different values for α, the pendulum swings at different velocities. The velocity v is reached at the equilibrium position during the first oscillation. The velocity of the pendulum keeps decreasing after each swing. In ideal conditions; i.e in order to have precise results, it is important for the center of the pendulum to be friction-less.

To simulate a human body, a water container was used and was filled with 30kg of water (see fig. 2). Indeed, the interference generated by the human body is in fact due to the water present in the body.

Fig. 2: Water container

As mentioned earlier, the measurements were done for three different velocities. Therefore, the pendulum was swung at three different angles, names at 30°, 60° and 85°. To facilitate the lifting of the container, another rope was used along with a pulley system. The measurement was started at the same time the pendulum was released. Each measurement is performed for roughly the same duration that the pendulum takes to perform one swing.

Unfortunately, the experimental setting was rudimentary and therefore the quality of generated data was not optimal. Indeed, swinging the pendulum generated a great amount of friction a the center, especially with the 85° angle. Therefore, to make the experiments consistent; i.e to pull the the pendulum at the same angle each time, the length of the second rope was measured and marked for each angle. The rope is then pulled until the marked position in the rope for all next measurements of the same angle.

C. The Location

The experiments took place at the training center in fire and rescue station "Zentrale Leitstelle Frankfurt". The data was collected in an apartment in the third floor of a dummy building used by trainee firemen to simulate rescue operations. The plan of the apartment is displayed int the following figure:

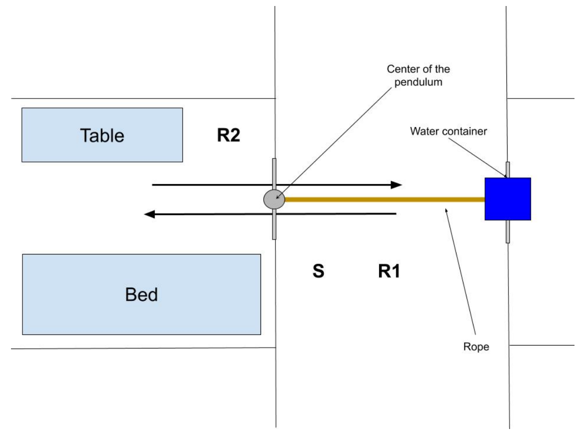


Fig. 3: Apartment

The positions of the laptops are displayed in Fig. 3. The transmitter (or sender), marked by S in the figure, are the second receiver, marked by R2, are positioned so that the pendulum swings between them. However, and in order to generate different samples, the first receiver, marked by R1, is positioned near the sender.

III. Algorithms

In the scope of this project, the following algorithm was implemented:

Input: Time Series d containing raw data

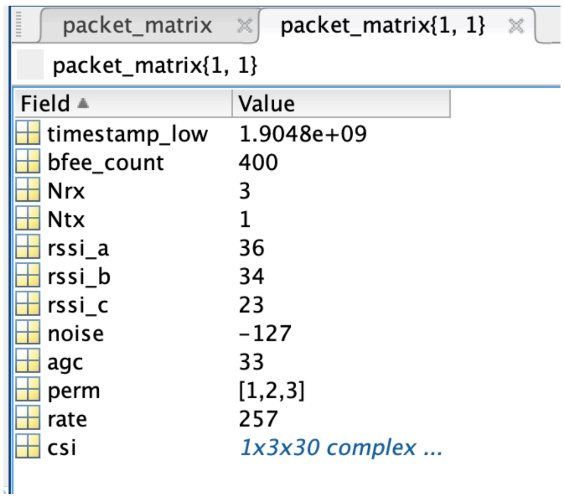
Output: The predicted activity with label L.

1. d $\leftarrow CSI\_Value\_Extraction(d)$
2. $d \leftarrow Denoise(d)$
3. $d \leftarrow PCA(d)$
4. $f \leftarrow Feature\_Extraction(d)$
5. $L \leftarrow SVM\_Classification(f)$

A. CSI Value Extraction

There are approximately 150 packets in each sample file. TO standardize the data, all samples were cropped to 130 packets. For this experiment, 90 sub-carriers groups for three antennas were used per transmitter and a receiver pair. The CSI values are recorded for each sub-carrier. From each sample file, the CSI values are extracted in the form of 3-D matrix of NT*NR*30 for 40MHz channel, where NT is a number of transmitter antennas and NR is a number of receiver antennas.

This data is then parsed using the MATLAB library developed by [7]. After the parting, the resulting matrix has 90 columns, representing the sub-carriers and 150 rows representing the packets. Each column of a matrix forms a time series for each 90 sub-carriers. The CSI value stored in each cell of the matrix is a complex number, where the amplitude of the signal is the real part and the phase of the signal is the imaginary part. Only the amplitude of the CSI value is considered for this project.



packet_matrix | packet_matrix{1, 1}

packet_matrix{1, 1}

Field	Value
timestamp_low	1.9048e+09
bfee_count	400
Nrx	3
Ntx	1
rssi_a	36
rssi_b	34
rssi_c	23
noise	-127
agc	33
perm	[1,2,3]
rate	257
csi	*1x3x30 complex ...*

Fig. 4: CSI Values for one sample

B. Denoising

The goal of this step is to remove the noise, while preserving the signal information. The common procedure to remove the noise is Discrete Wavelet Transform (DWT). In DWT the samples pass through banks of high pass and low pass filter at each level. DWT produce detailed and approximation coefficients. The first level coefficient is used to calculate a threshold. The sharp changes in human activity and noise are contained by the first level coefficient. After calculating the threshold, then it is applied to detailed coefficients of different levels. A new signal is reconstructed using new detailed coefficients. The denoising algorithm was implemented using MATLAB. Indeed, MATLAB provides the function `wdenoise`, which was used with the parameters level 3 and sym6 wavelet.

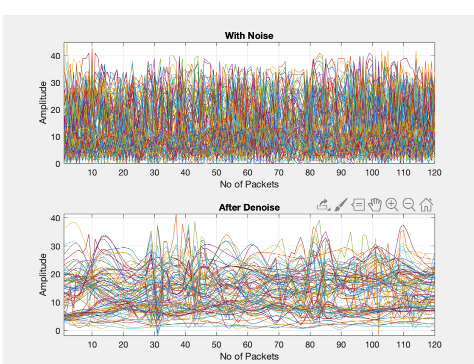


Fig. 5: Denoise Sample for Angle 30

The function `wdenoise` performs the discrete wavelet transform. First, it finds the threshold value using the first level coefficient and then applies this threshold value to all detailed coefficients of each level. Then a new signal is reconstructed. In Fig. 5, the first graph represents a data sample with noise and second graph represents the same sample after the noise removal.

C. Principle Component Analysis (PCA)

Principle Component Analysis (PCA) is used for feature reduction [12]. PCA is applied on the denoised data. The result are the so called components which represent the variance in the data. The first three components represent the 60% to 80% contribution to the variance in the data. The first component has the most contribution because of static objects (wall, furniture etc) in the room. To analyze components we use the MATLAB function `pca`, which computes the coefficient, score, latent, tsquared, explained values. The coefficients are the loading for each principle component, the variance of each principle component calculated by latent, explained are used to calculate the percentage of the total variation in each PC. For classification, we use the first three components and 10 components.

D. Feature Reduction

Power spectral density (PSD) analysis was used for feature extraction. It is a well-established and commonly used method for signal processing. When human do activities in WiFi-enabled environment, there are some variations in energy and power of a signal. Power Density Analysis is used to analyze these effects[13]. In PSD the value of the amplitude is squared and its normalizes the amplitudes by the frequency resolution to give the amplitudes a similar appearance. We use the following algorithm for Feature Extraction[8].

Algorithm for Feature Extraction
Input: The preprocessed time series d
Output: The extracted feature vector f
1. PSD,v $\leftarrow Compute\ Power\ Spectral\ Density\ from(d)$
2. $m_{psd}, m_v \leftarrow Haart\ Wavelet\ Transformation\ (PSD, v)$
3. $f \leftarrow Statistical\ Properties\ Extraction(m_{psd}, m_v)$

We perform the discrete Haar 1-D wavelet transformation in Haart wavelet transformation. We use Matlab function haart(x,level) and set the maximum level for haart transform that is 5. Finally, the statistical properties extraction algorithm computes for following data.

We calculate the statistical properties for each selected time series. such as Mean, Kurtosis, Skewness, Standard Deviation, Interquartile Range, Median, Third and Second Central Moment.

We calculate Mean, Mean, Kurtosis, Skewness, Standard Deviation, Interquartile Range, Median for each selected subcarrier's PSD matrix.

we calculate Mean, Max, Standard Deviation, and Interquartile Range for each of row of the selected subcarrier's frequency of center of energy.

E. Support Vector Classification (SVM)

The Support Vector Machine[14] is an efficient classification machine learning algorithm. The test data can be easily classified by maximizing the margin of hyperplane and feature vector. We use svm method with linear kernel to build a classification model and implement both in R and Matlab. We divide the data into training and testing set. 75% of the data used for training and 25% data for testing.

F. Data Set

For each angle, we collected 100 samples. We use 300 samples for classification. We merge all the data into one csv file. In that csv file the first column is the angle and the first row is a statistical properties of each sample.

Angle in degree	Number of Samples
30	100
60	100
85	100

Fig. 6: Data Samples

IV. Results

A. SVM Classification for First 3 Components with PSD

Fig.7 and Fig.8 shows prediction graph and the confusion matrix for first three components. Fig.13 shows that the accuracy is 62%.

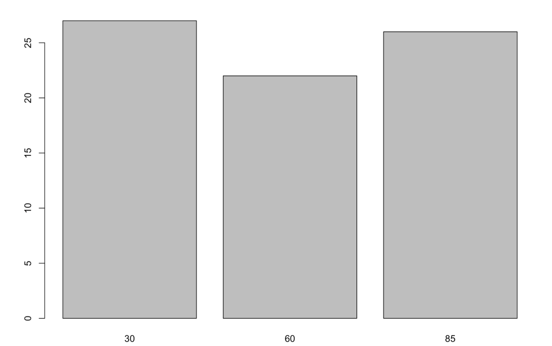


Fig. 7: Prediction Graph for first 3 Components

Angle Prediction	30	60	85
30	16	6	3
60	8	12	5
85	3	4	18
Total	27	22	26

Fig. 8: Confusion Matrix for First 3 Components

B. SVM Classification for First 3 Components with PSD

Fig.9 and Fig.10 shows prediction graph and the confusion matrix for first three components.

C. SVM Classification for Fist 3 Components

Fig.7 and Fig.8 shows prediction graph and the confusion matrix for first three components.

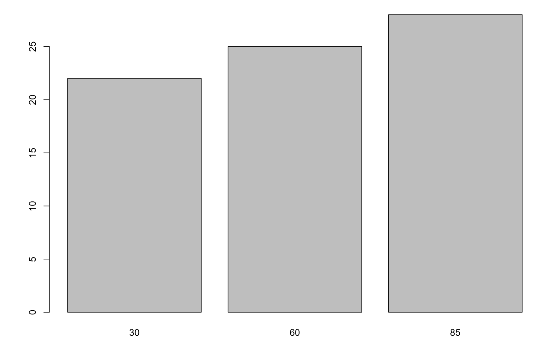


Fig. 9: Prediction Graph for first 3 Components with PSD

Angle Prediction	30	60	85
30	16	6	3
60	5	11	9
85	1	8	16
Total	22	25	28

Fig. 10: Confusion Matrix for First 3 Components with PSD

D. SVM Classification for fist 10 Components

Fig.11 and Fig.12 shows prediction graph and the confusion matrix for first 10 components.

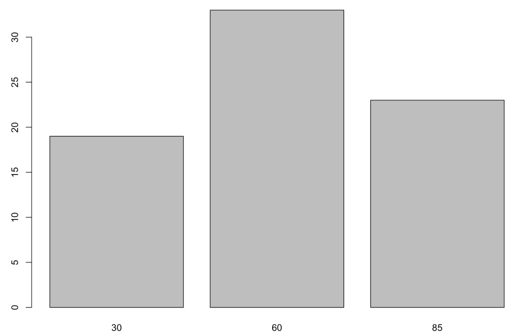


Fig. 11: Prediction Graph for 10 Components

Angle Prediction	30	60	85
30	15	7	3
60	4	18	3
85	0	8	17
Total	19	35	23

Fig. 12: Confusion Matrix for 10 Components

E. Comparison between different Components

Fig. 13 shows that the accuracy of first 3 components is 62% without PSD. The accuracy for the first 3 components with PSD is 58% and for the first 10 components is 67%.

Components	True Samples	Fasle Samples	True %	False %
Three Components	29	46	62 %	38 %
Three Components (PSD)	43	32	58 %	42 %
Ten Components	25	50	67 %	33 %

Fig. 13: Comparison between different Components

F. Explanation

Unfortunately the classification results were not good. In order to investigation and find the reason behind these results, the different statistical values for all the samples were observed. The following plots displays the results of the investigation:

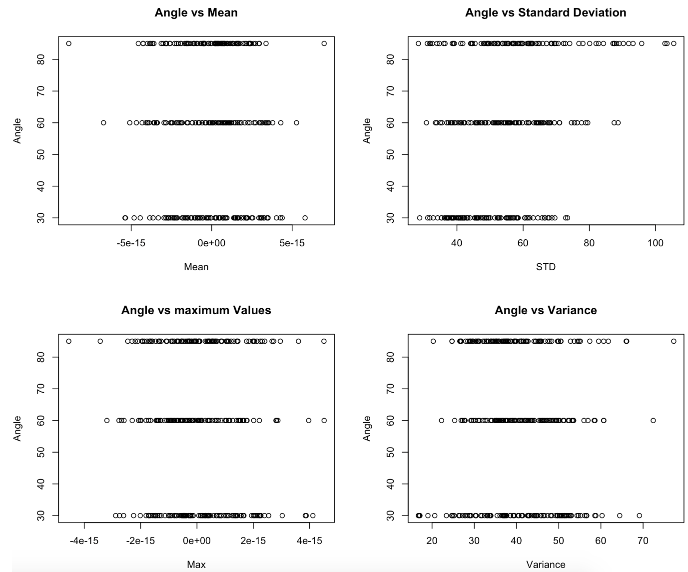


Fig. 14: Scatter Plot for Statistical Properties with their corresponding samples

Fig.14 shows a scatter plot representing some of the statistical properties, mean, max, standard deviation and variance, for the complete data set. It is clear from the plots that these properties are not suitable for classification. Indeed, the values for the samples are similar for different angles. They are normally distributed and most of them overlap values from samples having different angles, suggesting low correlation between the samples. More statistical properties were investigated and all of them had a very similar distribution: overlapping values between samples from different angles. Ideal values would be clusters of samples having the same angle. In order to validate the proposed theory, that the statistical properties are poorly correlation, PCA was applied as a last step before classification. This yielded the same poor results.

V. Conclusion

As the results show that we can detect the speed of human while falling down in WiFi-enabled network. The results are not that much efficient as we expected. As we already stated that the classification of the activities totally depends on the quality of the data. CSI value has a lot of noise and abrupt values.

References

[1] S. Yousefi, H. Narui, S. Dayal, S. Ermon and S. Valaee, "A Survey on Behavior Recognition Using WiFi Channel State Information," in IEEE Communications Magazine, vol. 55, no. 10, pp. 98-104, Oct. 2017. doi: 10.1109/MCOM.2017.1700082

[2] Wang, X., Gao, L., Mao, S., Pandey, S.: CSI-based fingerprinting for indoor localization: a deep learning approach. CoRR https ://arxiv .org/abs/1603.07080 (2016b)

[3] Y. Cheng and R. Y. Chang, "Device-free indoor people counting using wi-fi channel state information for internet of things," in GLOBECOM 2017 - 2017 IEEE Global Communications Conference, Dec 2017, pp. 1–6.

[4]] W. He, K. Wu, Y. Zou, and Z. Ming, "WiG: WiFi-based gesture recognition system," in 24th Int'l Conf. on Computer Commun. and Networks (ICCCN), 2015, pp. 1–7

[5] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of WiFi signal based human activity recognition," in Proc. of the 21st Annual Int'l Conf. on Mobile Comput. and Netw., 2015, pp. 65–76.

[6] Y. Zeng, P. H. Pathak, and P. Mohapatra, "Analyzing shopper's behavior through WiFi signals," in Proc. of the 2nd Workshop on Physical Analytics, 2015, pp. 13–18.

[7] Halperin, D., Hu,W., Sheth, A.,Wetherall, D.: Tool release: Gathering 802.11n traces with channel state information. SIGCOMM Comput. Commun. Rev. 41(1), 53–53 (2011). DOI 10.1145/1925861.1925870. URL http://doi.acm.org/10.1145/1925861.1925870

[8] Damodaran, N., Haruni, E., Kokhkharova, M. et al. Device free human activity and fall recognition using WiFi channel state information (CSI). CCF Trans. Pervasive Comp. Interact. 2, 1–17 (2020). https://doi.org/10.1007/s42486-020-00027-1

[9] O. Politi, I. Mporas and V. Megalooikonomou, "Human motion detection in daily activity tasks using wearable sensors," 2014 22nd European Signal Processing Conference (EUSIPCO), Lisbon, 2014, pp. 2315-2319.

[10] Y. Gu, F. Ren and J. Li, "PAWS: Passive Human Activity Recognition Based on WiFi Ambient Signals," in IEEE Internet of Things Journal, vol. 3, no. 5, pp. 796-805, Oct. 2016. doi: 10.1109/JIOT.2015.2511805

[11] Eric W. Weisstein. "Pendulum" From Eric Weisstein's World of Physics–A Wolfram Web Resource. http://scienceworld.wolfram.com/physics/Pendulum.html

[12] W. Wang, A. X. Liu, M. Shahzad, K. Ling and S. Lu, "Device-Free Human Activity Recognition Using Commercial WiFi Devices," in IEEE Journal on Selected Areas in Communications, vol. 35, no. 5, pp. 1118-1131, May 2017. doi: 10.1109/JSAC.2017.2679658

[13] P. Stoica and R. Moses, Spectral Analysis of Signals. Pearson Prentice Hall, 2005. [Online]. Available: https://books.google.de/books?id=h78ZAQAAIAAJ

[14] K. G. Manosha Chathuramali and R. Rodrigo, "Faster human activity recognition with SVM," International Conference on Advances in ICT for Emerging Regions (ICTer2012), Colombo, 2012, pp. 197-203. doi: 10.1109/ICTer.2012.6421415