https://eprints.gla.ac.uk/277562/

Deposited on: 24 August 2022

Enlighten – Research publications by members of the University of Glasgow
https://eprints.gla.ac.uk

An Implementation of Real-time Activity Sensing using Wi-Fi: Identifying Optimal Machine Learning Techniques for Performance Evaluation

William Taylor, Muhammad Zakir Khan, Ahsen Tahir, Ahmad Taha, Qammer H. Abbasi, Muhammad Imran.

Abstract—The elderly population is growing, and the health care system is experiencing a strain on services provided to the elderly. The recent COVID-19 pandemic has increased this strain and has resulted in an increased risk of exposure during visits to elderly homes. Increasing the desire to provide technological solutions to counteract this. Currently, there lack reliable real-time non-invasive sensing systems. This paper makes use of Radio Frequency sensing, where signal propagation is observed in Channel State Information (CSI) reports on Activities of Daily Living (ADLs). Real-time data has been collected for three classifications, "movement", "empty room", and "no activity". A filter is applied to reduce the noise of the CSI data. Then the mean, max, min, kurtosis, skew and standard deviation features are extracted from the CSI data. A machine learning model provides classification for the real-time monitoring system allowing detection of abnormalities in the expected ADLs of the elderly. The timing of classifications gives insight into the real-time capabilities of the system. The Random Forest algorithm is chosen to create the machine learning model based on accuracy and timing capabilities. The model was able to achieve an accuracy of 100 % on new unseen testing data with an average classification time of 7.31 milliseconds.

Index Terms—Channel State Information, Real-time, Activities of Daily Living, Elderly Care, Machine Learning, Python
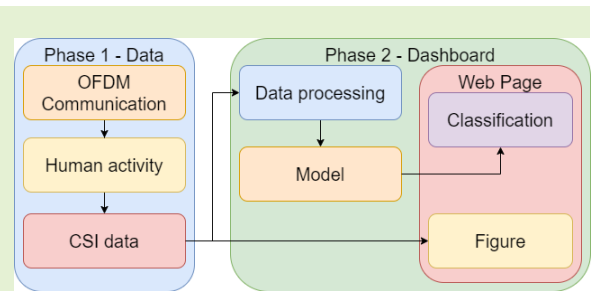
I. INTRODUCTION

The elderly population is increasing worldwide due an increase in life expectancy [1]. This leads to a strain on healthcare systems globally [2]. With the recent COVID-19 pandemic, it has now become dangerous for family members and care staff to visit elderly homes as elderly people are considered high-risk [3]. This contributes to the desired use of technologies to provide assistance to care and provide safety monitoring. Technology can be used to monitor the home and report irregularities in activities and vital signs. Supporting elderly people in daily routines and living within their residence [4]. Technologies can be divided into two subcategories, Invasive and non-invasive [5]. Invasive technologies include wearable devices, where the devices must be worn on the body for the sensors to report readings. Non-invasive techniques remove the need to wear devices and sensors can report on the environment and sense the move-

ments of humans. A wearable sensor that can detect human movements tends to use accelerometers [6]. Systems can make use of accelerometers, gyroscopes, and magnetometers to detect movements [7]. The disadvantage is that the sensor can be uncomfortable for individuals to wear and there is limited battery life. Additionally, elderly people may forget to wear the devices. This can be attributed to pre-existing conditions which can affect memory. Non-invasive techniques remove these disadvantages as the system is no longer dependent on the user wearing a device and/or device battery life. Non-invasive techniques can include ultrasonic sensing [8], camera technology [9], radar[10] and Radio Frequency (RF) sensing [11]. These different types of technologies can be used with AI for classifying different activities. Implementing systems such as ultrasonic sensing, vibration Sensing and radar can be expensive for purchasing additional equipment and the intrusion of the additional equipment around the home. Camera technology can also be expensive depending on the quality of the camera device and there may be issues with ambient lighting [12] and privacy concerns of cameras being present. RF sensing can be achieved by observing changes in the propagation of the radio signal caused by movements. The sensing aspect is concerned with continually monitoring the propagation information in the received signals and using this data to sense what is occurring between the transmitter and receiver. The use of RF sensing has clear benefits as it can be

implemented using ambient Wi-Fi signals already present in many homes [13]. This means that the cost is inexpensive and there is no intrusiveness of additional equipment in the home. It has become acceptable for homes to have Wi-Fi systems present. Currently, RF sensing research looks at how signals represent movements and how AI can classify movements. There is a lack of research on implementing a real-time working system. This paper looks to introduce the use of RF sensing into a working monitoring system.

The contributions in this paper can be summarised in the following points:

1) A real-time analysis of the monitoring system looking at the timing of machine learning algorithms. Fast classification timing is essential in real-time AI applications as the AI model needs to be able to perform classifications before the next sample of data is received from the continuous sensing system so that warnings of critical events can be reported in real-time.

    The development of a monitoring system that makes use of RF signals to accurately sense if a person is present in a room and performing movements within the monitored room. Machine learning is used to recognise patterns of signal amplitude associated with the signal reflection of the human body. This will serve to provide a significant monitoring system that can be used to detect disruptions to elderly people's expected activities. If there are changes in expected activities, then this can be indicative of further issues and health care professionals or family members can take further actions if required.

2) A dashboard which displays the current classification from the real-time system. A dashboard is an important feature of any monitoring system as it must communicate the classification of the AI model to family members and healthcare professionals so that appropriate action can be taken to care for any irregularities detected by the system. The dashboard is designed with simplicity in mind. There are minimal elements that show only the important aspects required to report. The classification is shown as a heading and there is an image of the RF signal which can show how much disruption the event has caused to the RF signals.

II. RELATED WORK

The authors in [14] developed a real-time activity monitoring system using wristbands paired with mobile devices. The work makes use of data fusion to combine data from both wristbands and mobile devices. Data fusion provides a contingency for when users forget to wear the wristband, it is still possible that the user will have their mobile device on their person and vice versa. The work included a diverse dataset of many different activities which is possible by using the many sensors included in the mobile device and the wristbands. The smartphone is also able to identify the location of the person and thus provides the presence or non-presence of the person. However, the paper still relies on wearable devices and will be ineffective without users wearing the devices. Our proposed system seeks to provide the monitoring aspect

of this work but removes the need for wearable devices. The authors of [15] used RF sensing to detect four people performing activities. The work was able to have the highest accuracy of 91.25 % accuracy when using a single subject. Our work will seek to implement real-time sensing using a single subject. The paper [16] detailed how falls in the homes of elderly people can be detected by using Ultra Wide Band (UWB) radar technology. This, therefore, provides a contactless sensing fall detection system and the results of the paper show that this proposed approach can be applied in real-time by streaming incoming data from the UWB device. The paper aims to distinguish between normal activities and falling. The system is beneficial for fall detection but is not designed to provide broader monitoring of elderly people's activities which may be indicative of a fall risk rather than providing notification of a fall that occurs when harm may have already been caused to the individual. In our paper, we seek to make use of the cheaper more available WiFi technology compared to this work using radar technology. Aziz Shah, Syed, et al. [17] made use of contactless sensing using Wi-Fi for real-time classification of human activities picking up an object; walking; sitting on a chair; jogging and leaving a room. The work takes the amplitude of the Wi-Fi subcarriers and converts them to scalogram images and passes these images through deep learning algorithms. The deep learning provided an accuracy result of 91.1 %. The paper mentions the use of real-time but does not detail the process of how a real-time approach is carried out. One of the main focuses of this paper is the privacy factor of activity classification. Our paper expands on the ideas detailed in this work and implements a working real-time system to provide continuous monitoring. Zhang, Daqing, et al. [18] present a real-time fall sensing system using Channel State Information (CSI) of RF signals. The classifications are made using Support Vector Machine (SVM). The set-up uses a camera to serve as ground truth while the CSI stream records the data being collected. The data is then labelled using the camera footage as a reference. This shows that the data is collected in real-time and classifications are made on the real-time data. The paper does not detail how long it takes for a classification to be made following all the data processing steps detailed in the paper. Therefore our paper seeks to provide an analysis of machine learning algorithms to select an appropriate algorithm to provide a real-time classification of new incoming data. The authors of [19] made use of RF signals to sense sit down, stand up and static stand activities in real-time. The paper made use of Linear Discriminant Analysis and SVM machine learning algorithms. The authors of this paper considered the timing component of the algorithms concerning the sample rate. The results concluded the AI model created could make classifications within 300 milliseconds with an accuracy of 70 %. The work of our paper will seek to increase the accuracy of a real-time system by reducing the complexity of detected movements and keeping the classification time in milliseconds.

The paper [20] made use of Wi-Fi to sense that social distancing is present in elevators and office space with an accuracy of 92 % for the elevator and 97 % in the office space. This is achieved by using the CSI to calculate the number
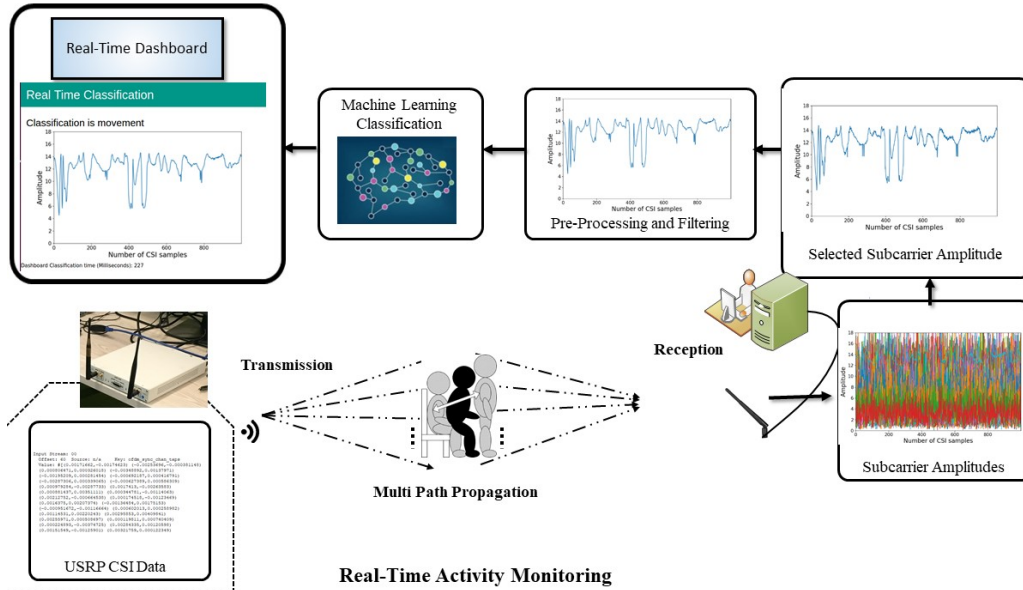
Fig. 1: Process of Real-Time Activity Monitoring

of occupants within an elevator and office space. The paper made use of description features of the CSI amplitude like the work of our paper. The features they used to describe the CSI amplitude were mean, standard deviation, MAD, min, max, skewness, kurtosis, and entropy. The paper did not compare the difference in results between using describing features against the full CSI amplitude data which the work of our paper looks to add to current literature. The authors of [21] recorded the timings of each algorithm in the application of detecting diseases by use of medical records as datasets. Algorithms tested like this paper was Support Vector Machine, K Nearest Neighbours and Random Forest. Results showed that on average Support Vector Machine was the fastest at classifying the data. Our paper provides a similar analysis in the field of activity detection using RF sensing.

is processed for machine learning classification and output. Figure 2 details the system architecture and data flow including data collection, data preprocessing, classification and output display on the web interface.

A. Hardware Setup

This work makes use of an X300 Universal Software-defined Radio Peripheral (USRP). As seen in Figure 3, the person performs activities in front of the USRP and the CSI is transferred to the computer using 1 Gigabyte Ethernet. Table I lists the parameters used to configure the USRP device for communication.
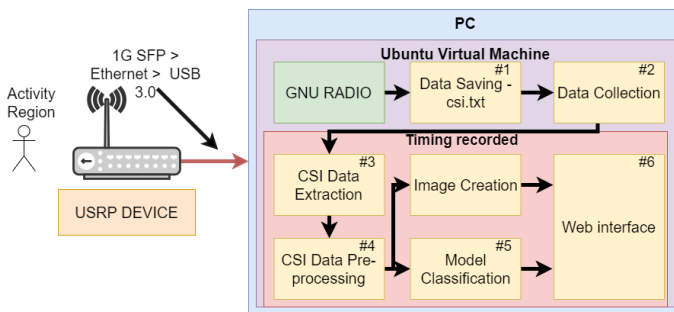
Fig. 2: Full system diagram

Fig. 3: Experimental setup of USRP placement with chair

III. METHODOLOGY

This section details the hardware setup, software configuration, data pre-processing and dashboard design. Figure 1 displays the process of real-time activity monitoring with the human movement taking place between the transmitter and receiver. The amplitude CSI of a selected subcarrier

B. Software Configuration

This paper makes use of GNU radio open-source software running in an Ubuntu virtual environment. GNU radio is used to create a python script using the parameters listed in Table I, to define the USRP operation. The output of this script

TABLE I: Parameters set for USRP OFDM Communication

Parameter	Value
Number of Subcarriers	64
Pilot carriers	12,26,40, 54
Sample Rate	400 kHz
Centre Frequency	2.4 GHz
Bandwidth	20 MHz

contains the CSI information. The Raw output is piped to a buffer. A second python script is used to read the buffer and extract the CSI information.

C. Data Collection

The training data is collected from a real-time stream. Three phases of data collection are completed for each of the three classifications. The three classifications used in this model are an "empty room", "no activity" and movement. Figure 3 is used for the initial demonstration of a real-time sensing system using RF signals. The concept of using RF signals to sense movements in real-time has been mentioned in works such as [22] but our work seeks to contribute to the real-time application of RF sensing monitoring. Our work looks to implement multiple classifications in real-time, which are missing from works [18] and [20]. For real-life scenarios, multiple devices can be used as a transmitter and receiver and placed 2 metres apart. This can allow for a larger monitoring area. The CSI data is collected and saved in CSV format. 300 CSV files for each classification are collected. The data collected for movement consisted of a person performing sitting and standing simultaneously while the USRP transmits and receives RF signals until 300 CSVs of the movement action are collected. For "no activity" the person remains still sitting for 150 samples and a further 150 samples of standing still are collected. The reason for this is to allow the real-time model to account for any differences between sitting still and standing still to remain as "no activity". For the "empty room" samples, participants leave the room while CSI data is collected. Steps 1, 2 and 3 in Figure 2 show the steps taken in the data collection stage. The extracted data is then saved into CSV format for the training stage.

D. Proposed Real-Time Activity Monitoring System

1) Sub carrier selection: During the training phase of the real-time system, the data must be processed to improve the machine learning classification accuracy. This was performed in two stages, the first is to select one subcarrier from the 64 captured ones. Subcarrier selection is used to remove redundant CSI subcarriers which can result in overfitting during machine learning processes [23]. Overfitting is when too much data is passed through the algorithm, and it causes the algorithm to memorise the training data rather than recognise patterns of the data. The selection of the subcarrier is decided in two steps; firstly by looking at the pilot subcarriers which show the variations caused by the human movement as they are used to carry the pilot symbols in the Channel Estimation process [24]. Subcarrier number 40 shows to have the most variation. This is calculated by taking the standard deviation

of each of the pilot subcarriers. standard deviation is selected as the metric to show this as it provides a measure of variation amongst the group of pilot subcarriers. subcarrier 40 was shown to have the highest standard deviation value in all collected movement samples.

2) Pre-processing of Selected Sub Carrier: The pre-processing stage consists of two steps to finalise the data for Machine Learning. The first step is to filter the data and then once a filter is taken, the features can be extracted. A butterworth low-pass filter with an order of 1 and a cut-off frequency of 0.5 is applied to the data to remove high-frequency noise from the signal. high-frequency noise is not the result of human activity as human activity is detected at low frequencies [25]. Figure 4 shows a comparison of a movement sample before and after the pre-processing.
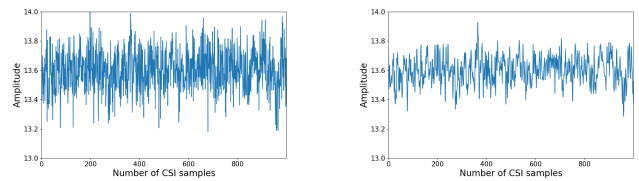
Fig. 4: Non-Filtered vs Filtered CSI data

The last step is to take the features of the filtered 1000 CSI data points. Six features are taken to summarise all the CSI amplitude data of subcarrier 40. The six features used are mean, max, min, kurtosis, skew, and standard deviation [26]. These features describe the wave of the signal including the peaks and size of peaks. Mean is used to show the mean of each value. CSI samples with peaks caused by movements will show a higher mean than samples with fewer peaks in amplitude. Similarly to the min and max values, where lower movement samples will have a lower max value and may display lower min values. Kurtosis and skew will also vary on the shape of the peaks observed in the CSI samples. Standard deviation will take note of the variance of the samples. These features provide an effective way of highlighting the differences between the 3 classifications. During the training phase, this process is carried out for each sample and the appropriate label is added. In the real-time process, the CSI data is processed and then fed via the AI model to give the prediction of the model.

3) Machine Learning: A selection of machine learning algorithms is tested against the data. The selected algorithms are selected based on the related work using these algorithms with CSI data. These algorithms are Random Forest [11], Bagged Trees [27], K Nearest Neighbours (KNN) and SVM [28].

4) Real-time Classification: This paper aims to provide users with the detection of movement as it happens. A real-time system is defined as a system in which input data is processed within milliseconds so that it is available virtually immediately.

In this experiment, the sensing data is the CSI and it is constantly being collected from received RF signals. The system works by using a window size of 1000 CSI data points. During the time of classifying the CSI window, another 1000

CSI data points are collected. 1000 CSI data points are equal to approximately 2.5 seconds due to the 400 kHz sample rate used in the RF communication. The real-time system must classify the data before the new data is received. Therefore the system must perform classification within the 2.5 seconds time frame as the next CSI window will be complete by the end of this time frame. Within this time frame, raw samples are passed to code which will carry out all pre-processing of the data and then pass the pre-processed data to the machine learning model. The entire process is recorded by taking timestamps before and after classification is provided. The time of classification is taken on a virtual machine with an i7 9700 3.00GHz CPU and 12GB of RAM.

To evaluate the timing performance 20 samples from each classification are removed from the training data. The remaining 240 samples are used to train a model and that model is used on the new testing samples and the time and accuracy are recorded. Additionally, a real-time comparison is made between the different algorithms using full CSI data and features to justify a choice in algorithm and between complete CSI data and the extracted features of the CSI data.

5) Dashboard Design: The Dashboard is created using the Flask python package. Flask is a micro framework that uses python to create web pages. The web page is designed with simplicity in mind. The reason for a simplistic design is to refrain from distractions as this system is designed with alerts in mind. The web page is comprised of a title banner, a heading declaring the classification of the data and a visual image of the current CSI to which the classification relates. Additionally, for this paper, the web page displays the total time of the classification at the bottom of the web page. This is used to calculate the total time between stages 3, 4, 5 and 6 shown in Figure 2. Python is used to take a timestamp at the start of stage 3 and JavaScript is used on the web page in stage 6 to take the end timestamp and duration can be calculated by using these two timestamps.

IV. RESULTS

A. Comparison of Machine Learning Algorithms

Figure 5 displays the accuracy of each of the algorithms when using 10-fold cross-validation. Table II shows the results achieved after all data has been tested during each of the 10 folds of the k-fold cross validation method including the Accuracy, Precision, Recall and F1 score which are common indicators of how machine learning algorithms perform.

The results show that Random Forest performed the best with an accuracy of 99.33 %. KNN had an accuracy result of 99.11 % and SVM had an accuracy score of 98.22 %. Bagged trees showed the lowest accuracy score of 93.11 %. All the algorithms however were able to achieve an accuracy score of over 90 %. This, therefore, indicates that the data of the three classifications are distinctive. Figure 6 shows the confusion matrix of the best-performing Random Forest algorithm. The figure shows how all 300 samples were classified in each of the 10-folds during cross-validation testing. The 10-fold cross-validation testing method splits the data into 10 groups, with each group taking a turn as the testing data and the remaining
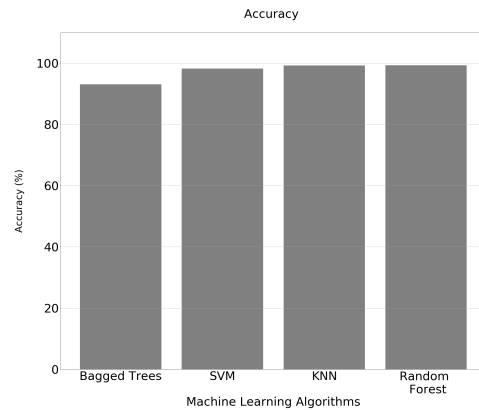

Fig. 5: Accuracy comparison of selected algorithms

TABLE II: Results from Comparison of Machine Learning Algorithms

Algorithm	Accuracy (%)	Classification	Precision	Recall	F1 Score
Random Forest	99.33	Empty	0.99	0.99	0.99
		Movement	1	1	1
		No Activity	0.99	0.99	0.99
Bagged Trees	93.11	Empty	0.86	0.95	0.90
		Movement	1	1	1
		No Activity	0.94	0.85	0.89
KNN	99.11	Empty	0.99	0.99	0.99
		Movement	1	1	1
		No Activity	0.99	0.99	0.99
SVM	98.22	Empty	0.99	0.96	0.97
		Movement	0.97	1	0.99
		No Activity	0.98	0.99	0.99

9 groups are used as training data. Once all 10 groups have been used as testing data once, then the classification results for all samples can be presented.

The Random Forest confusion matrix shows all the movement samples were correctly classified. 299 out of 300 samples were correctly classified as "no activity" samples. The "empty room" correctly classified 295 samples. As expected, most of the algorithm results have been able to establish the movement. The difficulty lies with differentiating between "empty room" and "no activity" as this is where most of the errors in the classification occur. The algorithms have classified most of the samples correctly for all three classifications. hence the high accuracy score achieved. For the implementation of a real system, some inference can be implemented which will work on the assumption that a movement must take place before a room can be empty. This can eliminate the small error rate in classifying between "empty room" and "no activity".

1) CSI data vs Feature selection: The above results are achieved by using features from the CSI data described in Section III-D.2. To justify the use of features in this paper a comparison is made between the processed CSI data and extracted features. Figure 7 shows the comparison of accuracy between using the CSI data vs the Features.

The algorithms perform better when using the features of the CSI data except for SVM. Random Forest had the highest accuracy out of all results of 99.11 % when using features.

Random Forest

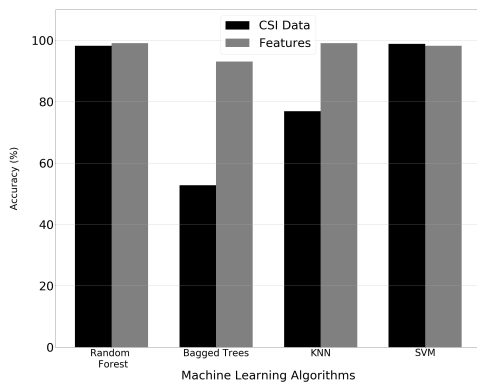Fig. 6: Confusion Matrix's of the Random Forest Algorithm

Fig. 7: Accuracy comparison of CSI data vs features

The Bagged Trees and KNN had significantly better results using extracted features. This improved accuracy is used as justification in this paper to use features in the real-time system.

B. Evaluation of Real-time Classification

Due to the real-time nature of the proposed system, the section will evaluate the real-time aspect of this work. To evaluate real-time classification the data is split into 80 % training data and 20 % testing data. Models are trained with the training data and then used to classify and record the time of the testing data. The results are displayed in the below table III. Table III presents the results of the average classification time and accuracy of 20 % testing samples of the data with full data vs feature extraction.

The results show that all algorithms can classify the data in milliseconds which would therefore be considered real-time classification. The results of the full features show the Random Forest algorithm as being the most accurate at 100 % and the fastest time at 7.31 milliseconds. The Bagged Trees algorithm has the lowest accuracy of 50 % and took the longest time of 940.13 milliseconds. KNN had an average time of 215.77 milliseconds and an accuracy of 85 %. The SVM algorithm had the second-best accuracy of 96.67 % with an average time of 49.30 milliseconds.

TABLE III: Results from Comparison of Machine Learning Algorithms

Algorithm	Average CSI runtime cost (Milliseconds)	Average Features runtime cost (Milliseconds)	CSI Accuracy	Extracted Features Accuracy
Random Forest	7.31	12.63	100.00%	95.00%
Bagged Trees	940.13	17.81	50.00%	91.67%
KNN	215.77	10.86	85.00%	95.00%
SVM	49.30	8.47	96.67%	95.00%

When the features were reduced Random Forest, KNN and SVM all had the same accuracy of 95 % with SVM performing the fastest at 8.47 milliseconds, KNN with 10.86 milliseconds and Random Forest with 12.63 milliseconds. The Bagged Tree algorithm was still the lowest-performing algorithm with an accuracy score of 91.67 % and an average time of 17.81 seconds.

The results show that the Bagged Trees, KNN and SVM show improvement in timing when the features are reduced. The best time in all of the results is the Random Forest algorithm when using full features. This suggests that Random Forest classification time is not affected by the size of the data and the additional time is due to the extra steps of producing the features of the full CSI data. To prove this the classification time is recalculated by not including the pre-processing steps and purely the classification time. The results showed that with full features the Random Forest was able to obtain an average classification time of 6.42 milliseconds and with the features, it was 6.48 milliseconds. This is a negligible difference of 0.02 milliseconds.

From these results, the authors have chosen Random Forest with features as the algorithm to create the AI model for the real-time system. This is justified by the fast classification time paired with the highest accuracy using a 10-fold cross-validation shown in Table II. All algorithms can perform real-time classification, but Random Forest had the highest accuracy when using the 10-fold cross-validation Therefore has been selected as the algorithm for the real-time model.

C. Classification of Unseen Data

As presented in the previous section, Random Forest with reduced features is best suited to our proposed system. We can collect new unseen data and test how the trained model can classify new unseen data using the selected methods. This will indicate how the model will classify new incoming data during the operation of the real-time system. A total of 100 additional samples are collected and compiled into a new testing dataset. All the Data collected and tested in previous sections is used in the creation of an AI model which is applied to each sample and results are recorded. Figure 8 shows the confusion matrix and it shows that all new unseen testing samples were correctly classified. This provides sufficient evidence that our model will

Fig. 8: Model Classification of New Unseen Data

be able to classify the 3 classifications during the real-time operation of the monitoring system.
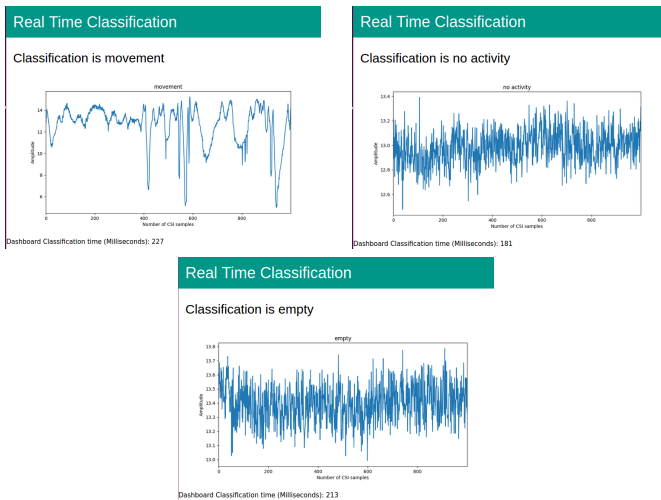
D. Dashboard

Fig. 9: Dashboard Output for all Three Classifications

The Dashboard is used to provide visualisation of the classification produced from the AI model. The result of the AI must be presented, and it is not expected that family members or care staff to read from terminal outputs. The result of the AI model is passed to the dashboard python code as a variable. Then the dashboard loads the web page based on the contents of the variable. Rules can also be set to create alerts if any activity is detected which is not considered regular and then raise an alert. These alerts can be used to assist. The dashboard displays the detected classification and a visual representation of the CSI. Figure 9 shows all of the classes outputted on the dashboard with images and time for classification. It can be seen how the amplitude is affected in the movement sample. The difference between "empty room" and "no activity" is less clear to visualise but it can be noted that the "empty room" sample has a slightly higher amplitude than the "no activity". This is the reason why most errors in classification

were caused by the "empty room" and "no activity" classes as shown in Figure 6.

Due to the current system's constraint of range from the use of a single USRP shown in Figure 3, as soon as the person leaves the small monitoring area, the system will then trigger the empty classification. The timing recording is done by taking a UNIX timestamp from python when the web page is first loaded then the processes of the web app take place and then when the web page is displayed on the browser, JavaScript is applied to take a new UNIX timestamp which is then subtracted from the first timestamp and thus providing a time for how long the entire process took. This time is displayed on the bottom of the web page shown in Figure 9. The time is still in milliseconds but much larger than the times shown for Random Forest model classification in Section IV-B. This extra time is due to the extra functions of the web page such as creating the CSI image displayed on the web page additionally time is taken for retrieving the data before data processing and model classification can take place. However, the time is still in milliseconds and clearly in line with the definition of real-time. It is worth considering that more complex web pages may result in higher latency and will thus affect the real-time functionality of the web page. It is also worth considering the processing power of web servers that will host the dashboard.

V. CONCLUSION

In this paper, a novel real-time sensing system is presented. The system makes use of AI to identify the moving or presence of a person in a room using Wi-Fi. The use of Wi-Fi ensures a non-contact method of movement detection where a wearable device is not required. A USRP device is set up to use Wi-Fi between a transmitting antenna and a receiving antenna. The Wi-Fi signals are then transmitted through the atmosphere and when a person is present or moves the Wi-Fi signal is disrupted. These disruptions are present in the CSI. Every 1000 CSI data points received at the receiver side is then processed and passed to AI models to make a classification on whether the disruption indicates an "empty room", the presence of a person or movement. 300 training samples are collected for each classification and the accuracy and computational time of 4 machine learning algorithms are compared. The results are presented, and the justification is given for the choice of algorithm to create an AI model. The AI model which achieved 100 % accuracy on new unseen testing data is used in a dashboard to create a real-time monitoring system. Future work will seek to create an alert system where irregularities in movements can be flagged up as an alert. Which will create a more effective monitoring system which will not require constant monitoring of a dashboard.

REFERENCES

[1]　Maria Vittoria Conti et al. "Bioactive compounds in legumes: Implications for sustainable nutrition and health in the elderly population". In: *Trends in Food Science & Technology* (2021).

[2] Soe Ye Yint Tun, Samaneh Madanian, and Farhaan Mirza. "Internet of things (IoT) applications for elderly care: a reflective review". In: *Aging clinical and experimental research* 33.4 (2021), pp. 855–867.

[3] Francisco Luna-Perejón et al. "IoT garment for remote elderly care network". In: *Biomedical Signal Processing and Control* 69 (2021), p. 102848.

[4] Eleni Boumpa and Athanasios Kakarountas. "Home supporting smart systems for elderly people". In: *Convergence of ICT and Smart Devices for Emerging Applications*. Springer, 2020, pp. 81–98.

[5] Gokmen Ascioglu and Yavuz Senol. "Design of a wearable wireless multi-sensor monitoring system and application for activity recognition using deep learning". In: *IEEE Access* 8 (2020), pp. 169183–169195.

[6] Nahed Jalloul. "Wearable sensors for the monitoring of movement disorders". In: *Biomedical journal* 41.4 (2018), pp. 249–253.

[7] Ahmad Jalal, Majid Ali Khan Quaid, Kibum Kim, et al. "A study of accelerometer and gyroscope measurements in physical life-log activities detection systems". In: *Sensors* 20.22 (2020), p. 6670.

[8] David Caicedo, Ashish Pandharipande, and Frans MJ Willems. "Detection performance analysis of an ultrasonic presence sensor". In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2013, pp. 2780–2784.

[9] Francy Shu and Jeff Shu. "An eight-camera fall detection system using human fall pattern recognition via machine learning by a low-cost android box". In: *Scientific reports* 11.1 (2021), pp. 1–17.

[10] William Taylor et al. "Radar sensing for activity classification in elderly people exploiting micro-doppler signatures using machine learning". In: *Sensors* 21.11 (2021), p. 3881.

[11] William Taylor et al. "An intelligent non-invasive real-time human activity recognition system for next-generation healthcare". In: *Sensors* 20.9 (2020), p. 2653.

[12] Shigeyuki Tateno et al. "Privacy-preserved fall detection method with three-dimensional convolutional neural network using low-resolution infrared array sensor". In: *Sensors* 20.20 (2020), p. 5957.

[13] Yu Gu et al. "MoSense: An RF-based motion detection system via off-the-shelf WiFi devices". In: *IEEE Internet of Things Journal* 4.6 (2017), pp. 2326–2341.

[14] Božidara Cvetković et al. "Real-time activity monitoring with a wristband and a smartphone". In: *Information Fusion* 43 (2018), pp. 77–93.

[15] Aboajeila Milad Ashleibta et al. "5g-enabled contactless multi-user presence and activity detection for independent assisted living". In: *Scientific Reports* 11.1 (2021), pp. 1–15.

[16] Ghassem Mokhtari, Qing Zhang, and Amir Fazlollahi. "Non-wearable UWB sensor to detect falls in smart home environment". In: *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE. 2017, pp. 274–278.

[17] Syed Aziz Shah et al. "Privacy-preserving non-wearable occupancy monitoring system exploiting Wi-Fi imaging for next-generation body centric communication". In: *Micromachines* 11.4 (2020), p. 379.

[18] Daqing Zhang et al. "Anti-fall: A non-intrusive and real-time fall detector leveraging CSI from commodity WiFi devices". In: *International Conference on Smart Homes and Health Telematics*. Springer. 2015, pp. 181–193.

[19] Yao Ge et al. "Real-Time Human Activity Recognition System Exploiting Ubiquitous Wi-Fi Signals". In: *2021 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting (APS/URSI)*. IEEE. 2021, pp. 1033–1034.

[20] Aryan Sharma et al. "Passive WiFi CSI sensing based machine learning framework for COVID-Safe occupancy monitoring". In: *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE. 2021, pp. 1–6.

[21] Munder Abdulatef Al-Hashem, Ali Mohammad Alqudah, and Qasem Qananwah. "Performance Evaluation of Different Machine Learning Classification Algorithms for Disease Diagnosis". In: *International Journal of E-Health and Medical Communications (IJEHMC)* 12.6 (2021), pp. 1–28.

[22] Syed Aziz Shah and Francesco Fioranelli. "Human activity recognition: Preliminary results for dataset portability using FMCW radar". In: *2019 International Radar Conference (RADAR)*. IEEE. 2019, pp. 1–4.

[23] Yu Taso et al. "Subcarrier selection for efficient CSI-based indoor localization". In: *IOP Conference Series: Materials Science and Engineering*. Vol. 383. 1. IOP Publishing. 2018, p. 012017.

[24] Mehran Soltani et al. "Deep learning-based channel estimation". In: *IEEE Communications Letters* 23.4 (2019), pp. 652–655.

[25] Yan Wang et al. "E-eyes: device-free location-oriented activity identification using fine-grained wifi signatures". In: *Proceedings of the 20th annual international conference on Mobile computing and networking*. 2014, pp. 617–628.

[26] Itaf Omar Joudeh et al. "Location Independence in Machine Learning Classification of Sitting-Down and Standing-Up Actions using Wi-Fi Sensors". In: *2021 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*. IEEE. 2021, pp. 1–6.

[27] Robert M Keenan and Le-Nam Tran. "Fall detection using Wi-Fi signals and threshold-based activity segmentation". In: *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE. 2020, pp. 1–6.

[28] Sheheryar Arshad et al. "Wi-chase: A WiFi based human activity recognition system for sensorless environments". In: *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE. 2017, pp. 1–6.