Design and Implementation of a CSI-Based AI Human Moion Detection System for Next-Generation Healthcare

*

Yukai Song*
Glasgow College UESTC
University of Glasgow
Glasgow, UK
2357612S@student.gla.ac.uk

William Taylor
James Watt School of Engineering
University of Glasgow
Glasgow, UK
2536400t@student.gla.ac.uk

Yao Ge
James Watt School of Engineering
University of Glasgow
Glasgow, UK
2288980G@student.gla.ac.uk

Kia Dashtipour
James Watt School of Engineering
University of Glasgow
Glasgow, UK
kia.dashtipour@glasgow.ac.uk

Muhammad Ali Imran
James Watt School of Engineering
University of Glasgow
Glasgow, UK
Muhammad.Imran@glasgow.ac.uk

Qammer H. Abbasi
James Watt School of Engineering
University of Glasgow
Glasgow, UK
Qammer.Abbasi@glasgow.ac.uk

***Abstract*—In Artificial Intelligence driven healthcare system, human motion detection is becoming increasingly popular as it can be applied to give remote healthcare for vulnerable people. This paper aims to develop a contactless AI enabled Healthcare system, aimed to detect human motion using RF (Radio Frequency) signals. Although human motion detection systems have been developed using wearable devices like smart watches have been widely deployed, it still leaves many issues that cannot be solved. For some disabled and elderly people, it is difficult and easily forgotten to wear the devices. Thus, in order to tackle those issues, we propose a novel method using non-wearable methods. We first produced a dataset of radio wave signals that contain patterns of human motion by using software-defined radios. Next, machine learning algorithms such as Neural Network (NN), K Nearest Neighbors (KNN), Random Forest and Support Vector Machine (SVM) were applied to the CSI data to classify different human activities. Then, we used each algorithm to build an independent classifier and ensemble three best performing classifiers to the healthcare system to reduce the possibility of False Positive cases or True Negative cases. The ensemble classifier is able to achieve an accuracy of around 98% using 70% data for training and 30% data for testing. This is much higher in contrast with a benchmark dataset collected by accelerometers in wearable devices with an accuracy of around 93%, proving the effectiveness of non-invasive methods.**

***Index Terms*—Random Forest, ensemble machine learning, Support Vector Machine (SVM), Neural Network (NN), K Nearest Neighbors (KNN), Channel State Information (CSI)**

I. INTRODUCTION

Recent decades have witnessed the development of the Internet of Things (IoT), which allows human motion detection to be applied in many scenarios [1,2]. It has been proved that human motion detection has already played a growing crucial role in those areas, especially in healthcare. Motion detection technologies mainly have two types: vision-based detection based on computer vision and sensor-based detection based on traditional machine learning algorithms. The importance of human healthcare systems about human motions for elders can be clearly demonstrated in the World Health Organization (WHO) reports that over 37 million serious injuries are related to falls and 646,000 deaths every year and most of them are related to elderly [3,4]. Additionally, the elderly population keeps on rising and according to statistics from the United Nations, the elderly population will be around 2.1 billion in the year 2050, which can be a serious challenge to the elderly healthcare system. Falling-related injuries can cause direct financial losses related to curing those patients and indirect loss to society productivity [5,6]. If caregivers know that patients fall through the human motion detection system, many tragedies can be avoided as the caregivers can offer assistance in time. More importantly, this can allow vulnerable populations like the elderly to live a more independent lifestyle while still being ensured their safety as they are being monitored by the system. As each elder will require a lower level of caring, one caregiver can allow time for more elders as the technology can lessen their burdens and in turn save medical resources.

Detecting human motion through wearable devices like smart phones or smart watches with accelerometers is a popular method of building human healthcare systems of human motion. When patients fall down, the smart phones or smart watches can detect it and pass information to the caregivers for help [7,8]. It will fail to work when people forget to wear their watches, carry their phone or fail to charge their device. Additionally, wearable devices also might lead to concerns of privacy. As data of wearable device is usually stored in the cloud and the data in the cloud can be easily attacked by hackers nowadays [9]. Another method of detecting human motion that does not require patients to wear any devices is by using RF signals. In this method, we detect human motions through wireless signals. From [10,11], we know that Channel State Information (CSI) can be used for detecting human movements. The CSI describes how the wireless signals propagate between the Radio Frequency Source and Radio Frequency Receiver [12] and it will perform changes in certain patterns when people perform certain movements, implying the potential of human motion detection. In this project, we recorded the amplitude of CSI, using Universal Software-defined Radio Peripheral (USRP), when volunteers performed certain human movements to construct a dataset of CSI of human activities. USRPs were used because they provide a simple framework that can enable us to collect data easily [13,14]. Besides, USRPs can transfer and receive frequencies in different bands. In this project, we used 64 subcarriers that were generated by OFDM with 64 points Fast Fourier Transform [15]. Since higher frequencies can detect large motions while lower frequencies can be used for small motions [16], using USRPs with 64 different frequencies might enable us to detect both large and small movements easily. After using the noise filter to clean the CSI data, we adjusted parameters of chosen machine learning models to study features of CSI and ensemble three best-performed machine learning models to be the ensemble classifier. The ensemble classifier and Random Forest were integrated into local and real time healthcare systems that might be used in real conditions.

The reason that we apply machine learning models to classify CSI signals is that many works related to human motion classification have been done through machine learning. The KNN model was applied to the data and the result shows that it can achieve an accuracy of 95.5%. The article [17] demonstrates pros and cons of several existing patients monitoring technologies, such as sound-based, motion-based, vision-based, sensor-based, Radio-Frequency (RF) sensing methods. It shows that the noise filtering and feature extraction are important parts of RF sensing. In [18], SVM is applied for detecting normal walking and abnormal walking based on S-band sensing. The S-band sensing works by collecting the wireless channel data operated at 2.4 GHz, which is similar to our non-invasive method. The result shows that SVM can achieve an accuracy of 93%. However, in this work, the writers did not make a comparison between SVM and other machine learning models. Without comparison, we do not know whether SVM is the best method or not, which can be improved in our work. Apart from motion detections, CSI data has also been applied to monitor breathing beats of people [19]. In this study, people used WiFi devices to estimate breathing frequencies for multiple people by measuring CSI phase difference between antennas at the WiFi receiver. After that, they applied Canonical Polyadic decomposition to obtain the required breathing signals. Finally, a stable signal matching model was applied to find the breathing rates for different people. This work proves that CSI data can not only classify motions but also estimate breathing rates of people, showing the potential of CSI data in wireless health sensing. [20] illustrates that SVM classifier yields a precision of 90% while Random Forest Classifier generates a precision of 94% in CSI data. This result shows that Random Forest Classifier might be a better machine learning algorithm in CSI data in contrast with SVM. Works [21-23] show the potential of CSI data in the area of motion tracing, recognizing gestures and detecting smoking behavior when light conditions are not ideal. Deep learning is currently one of the most successful and popular methods in machine learning and the paper [24] attempts to apply Deep learning to multi-frequency CSI signals. In this work, they constructed CSI-Net, a unified Deep Neural Network which can be applied to hand sign recognition and falling detection. Apart from CSI-Net, they presented methods about how to encode and process CSI signals in DNNs. The result shows that DNN can achieve an accuracy of 100% in sign recognition and 96.67% in falling detection, which are nearly the highest in similar works. They also applied several traditional machine learning algorithms like SVM and Naïve Bayes to their CSI data. SVM results in an accuracy of 90.24% in sign recognition and 81.46% respectively while those for Naïve Bayes are 81.00% and 73.01%. In comparison with DNNs, it is clear that SVM and Naïve Bayes perform worse in terms of classifying CSI data, proving the amazing classifying ability of DNN. Similarly, another work [25] shows the possibility of using fully convolutional neural network (FCN) to recognize human motions based on Wi-Fi data and it also obtains a great result. In comparison with [24], the network of this work only contains convolutional layer while [24] has both convolutional layer and fully connected layer. The work [26] combines three machine learning models: Classical Machine Learning-based Multi-Class classifier, Deep Learning-based Multi-Class classifier, and Deep Learning-based Binary-Class classifier to decrease the possibility of misdiagnosis. This work shows the power of ensemble learning in dropping the possibility of False Positive, which is crucial in healthcare. Those previous works provide the theoretical foundation for this project.

In this paper, the principle that we adopt is to train our machine learning models based on the CSI for classification. Firstly, we used USRP to collect CSI and apply a low pass filter to clean the noise of CSI. Next, we utilized some statistical standards, such as f1-score, precision and recall, and plot learning curves to exam whether models suffer from under fitting or over fitting. If we found models that could not fit the CSI data well, we would adjust parameters of machine learning models until those models achieve the best performances. Then, we ensemble three best fitted machine learning models. After that, we made a comparison with a benchmark dataset, which was constructed by data of accelerators attached to smart phones. Finally, real-time and local healthcare AI systems were constructed based on both Ensemble Classifier and Random Forest. The Ensemble Classifier should be applied when there are sufficient computing sources while Random Forest should be used when there are limited computing sources. The paper is organized in such a structure: Methodology part consists of Data Collection and Signal Processing, Machine Learning and Local and Real Time Classification, Analysis and Discussion part and Conclusion and Further Work part. In Data Collection and Signal Processing, we describe how we collect CSI data and clean it. Machine Learning demonstrates the principles of different machine learning models, their performances in this project and compares results with a benchmark dataset based on data from mobile accelerators. Local and Real Time Classification focuses on how we integrate machine learning models into Local Classification and Real Time Classification. In the Analysis and Discussion, how to achieve the balance between the accuracy and computing source is discussed. In the Conclusion and Further Work, we summarize what we have achieved in this paper and what can be done in the future by applying the strategy of online learning.

II. METHODOLOGY

A. Data Collection and Signal Processing

In this paper, CSI samples of sitting and standing were collected while USRP devices communicate between antennas. Two USRPs were used in this project, one as the transmitter and another as the receiver. The devices are connected to two computers through cables and they were set to transmit signals from one antenna to the other for 10 s. During the experiment of collecting CSI data, two USRP devices were kept at a distance of 4 m. The experiment was performed in an office containing tables, chairs, paintings, etc, to simulate real life situations. Volunteers were asked to complete sitting down and standing up between the two USRPs. When volunteers perform the actions, the CSI represents the propagation of the radio signal. Although people can never perform the exact same motion and the interference from ambient factors also contribute to the difference, their motions should follow the same patterns which can be extracted by the machine learning model. All volunteers had signed ethical approves provided by the University of Glasgow ethic review committee. This process helped us collect CSI for different human motions and we made labels on the CSI samples while collecting them. After 30 samples of each activity are collected, a CSI dataset is successfully built. Figure 1 shows the process of data collection.

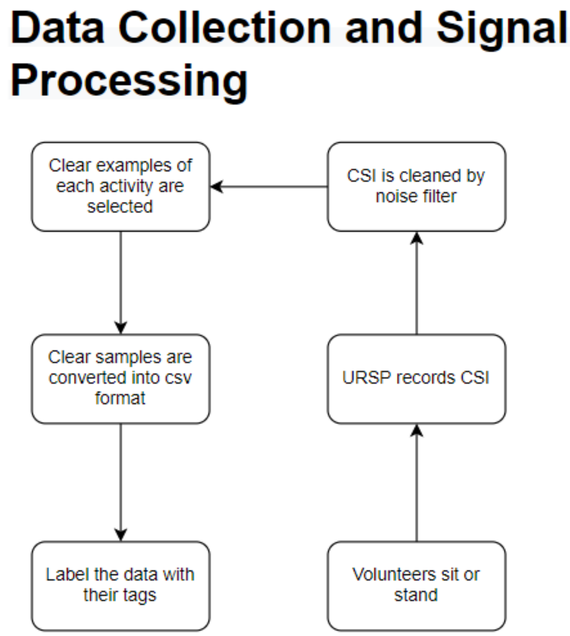

Fig. 1. Flow chart of Data Collection and Signal Processing.

In an attempt to explore features of CSI and the possibility to use filtering to limit the influence of noise, we first worked on signal processing. To be more specific, we first analyzed the CSI through comparison between sitting CSI and standing CSI in both time domain and frequency domain through MATLAB programming. Since collected data was measured in the time domain, we just directly showed them in the MATLAB figure while in the frequency domain, we used DFT (Discrete Fourier Transform) to display the frequency information. Results of comparison are shown in Figure 2 and Figure 3 below made by MATLAB and collected CSI data.

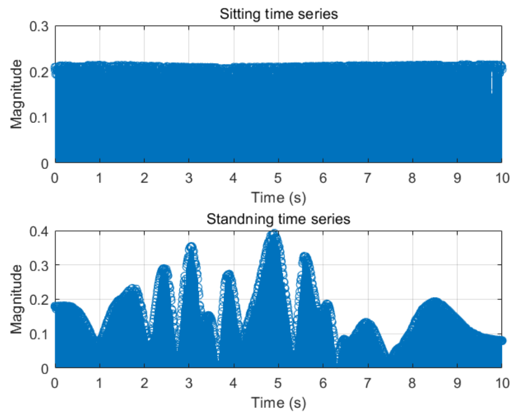

Fig. 2. Sitting time series and standing time series.

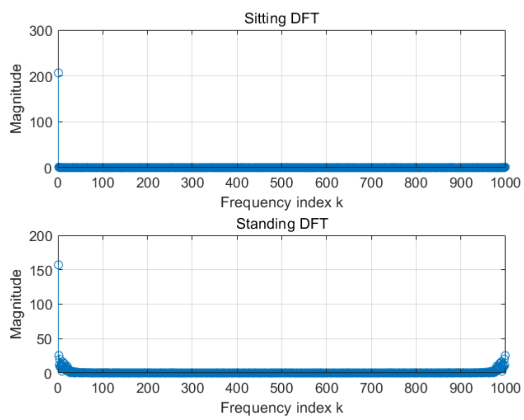

Fig. 3. Sitting DFT and standing DFT.

From Fig. 3, it is clear that the noise signals of both sitting and standing are high frequency. Therefore, a Butterworth lowpass filter is chosen as the noise filter to clean our data. After collecting and processing our data, we entered the machine learning stage.

B. Machine learning

In the machine learning stage, scikit-learn library was selected since it contains many powerful machine learning algorithms as well as a large amount of useful data processing functions [27]. To be more specific, the scikit-learn library can provide different machine learning algorithms, which can satisfy different conditions, including classification, regression, dimensionality reduction and clustering. Besides, it is easy to do training and testing splitting as well as data pre-processing. In this paper, we adapted the simple imputer function to impute value NAN with 0 for normalization before putting the data into training models. The NAN values are a result of the differing sizes of data received while the USRPs communicated within the 10 second time frame between collected samples in the full dataset.

We split 70% of the data for the training and 30% of the data for the testing. In an attempt to assess the performances of those machine learning algorithms, we calculated four classification values, including False Positive (FP), True Positive (TP), False Negative (FN) and True Negative (TN). Then we got the performance metrics based on the above four classification values. The details of calculating performance metrics [28] can be found in equations (1)-(4).

$$Recall = \frac{TP}{TP + FN} \tag{1}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Accuracy = 2 \times \frac{Precision \cdot Recall}{Precision + Recall} \tag{4}$$

Four machine learning algorithms were selected to train and test the CSI dataset. The four machine learning algorithms are SVM, NN, Random Forest and KNN. In order to increase the precision, we designed an ensemble classifier based on the prediction of four machine learning algorithms. Specifically, the ensemble classifier will declare the result of prediction from votes of each algorithm prediction. If three algorithms predict “Sitting” and only one algorithm predicts other results, the ensemble classifier will take “Sitting” as its result of prediction.

The Random Forest consists of many decision trees and every decision tree predicts through finding features in training. The decision tree is similar to a tree structure, consisting of node and directed edge, which can be used for classification. The internal node and leaf node are two types of nodes of the tree. A leaf node represents a class while an internal node represents a feature. ID3 algorithm based on the principle of information gain is adapted to the training of decision tree. The advantages of the decision tree are: 1. It is easy to understand the reasoning process based on the format: "If Then", 2. The reasoning process depends on the values of features only, 3. It can ignore features that have no contribution, judge the importance of different features, which can help us simplify features. However, if we want to achieve the best structure of decision tree, we need to have the fewest nodes, have the lowest depth, fulfill the fewest nodes and lowest depth at the same time and this has been proven to be a NP issue. To be more specific, if the depth of the decision tree is too large, it will meet the issue of over fitting while if there are too few nodes, it will meet the issue of under fitting. Therefore, it is hard to find the best decision tree.

For Random Forest, the results of statistical standards are accuracy=0.99, precision=0.99, recall=0.99 and f1-score=0.99. Since these four standards for evaluation are all close to 1, it is clear that this model can have a good performance for our CSI dataset. However, it just gives average values of four standards of standing and sitting. In an attempt to explore more details of training effects in terms of standing and sitting, we constructed a confusion matrix to illustrate.

TABLE I
CONFUSION MATRIX OF RANDOM FOREST

True \ Predicted	Standing	Sitting
Standing	341	9
Sitting	1	349

From learning curves for Random Forest, it is clear that as the training samples increase, cross-validation score gradually becomes larger until nearly achieves 1 while training score stays around 1. This fact shows that at the beginning of training, Random Forest suffers from the problem of high variance, the sign of over fitting. But as more training samples are trained, the cross-validation score and the training score become closer. Therefore, we can conclude that the Random Forest model has successfully overcome the problem of over fitting when there are enough training samples.

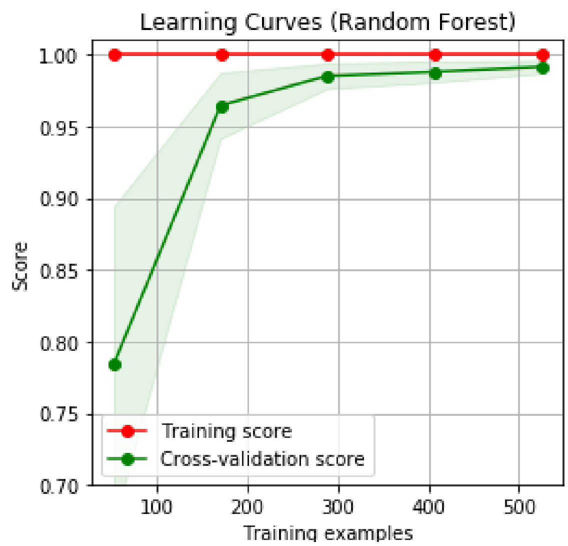

Fig. 4. Learning curves of Random forest.

K Nearest Neighbors (KNN) algorithm has the advantages of simplicity, high accuracy and not sensitive to abnormal value. The principle of KNN is: given a training dataset, for a new input data, we need to find the k nearest examples from the training dataset to this new input data. If most of the k nearest neighbors belong to class 1, the new input data will be predicted as class 1. However, it has a high computing complexity and spatial complexity and it is sensitive to selection of k. If we choose a small k, the approximation error will become smaller while estimation error will become larger, the model will be more sensitive to noise and the complexity will become larger, leading to the problem of overfitting. If we choose a lager k, we will have less estimation error while approximation error will become larger and the model will be simpler, contributing to under fitting problem [30]. Therefore, it is vital to find the best k to achieve a balance between under fitting and over fitting.

For KNN, the results of statistical standards are accuracy=0.78, precision=0.85, recall=0.78 and f1-score=0.77 when k=1. Because these four standards for evaluation are not ideal, KNN cannot fit our CSI data well. From the confusion matrix, we can find that the prediction for sitting is poor but standing is satisfactory.

TABLE II
CONFUSION MATRIX OF KNN

Predicted / True	Standing	Sitting
Standing	350	0
Sitting	153	197

From its learning curves, it can be observed that as the training samples increase, cross-validation score gradually becomes larger until nearly achieves 0.8 while training score stays around 1. This fact shows that the model suffers from over fitting since the cross-validation score and the training score still have a large distance after all training examples are trained. Therefore, the KNN model is not suitable to integrate into the ensemble classifier.

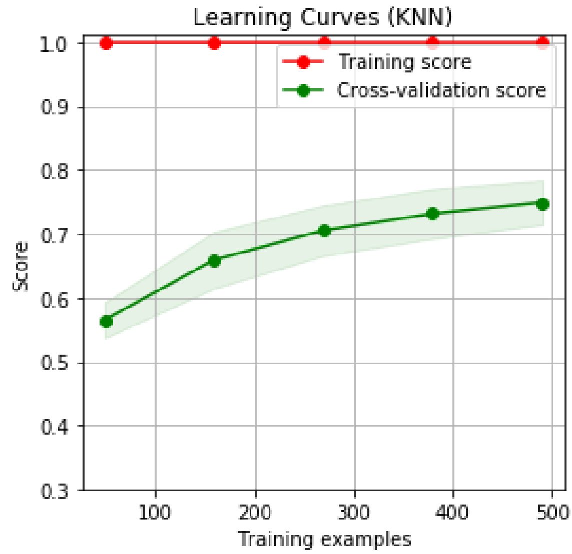

Fig. 5. Learning curves of KNN.

SVM constructs hyper planes, which can separate data of various types [31]. The optimizing aim of SVM is to construct hyper planes that can separate positive and negative samples with the largest margin. The points located at boundaries are referred to as Support Vector.

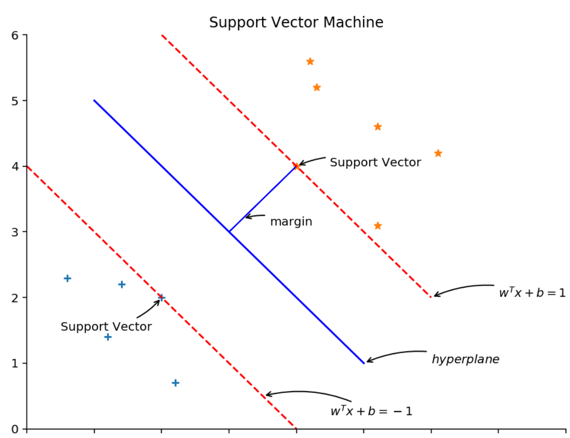

Fig. 6. SVM optimal hyperplane.

In this project, SVC was selected from sklearn.svm and set the parameter gamma=0.00015 and we got accuracy=0.94, precision=0.94, recall=0.94, f1-score=0.94. Similar to Random Forest, it is clear that this model can have a great performance over our CSI data. From the confusion matrix below, we can find that the prediction for standing is better than the prediction for sitting.

TABLE III
CONFUSION MATRIX OF SVM

True \ Predicted	Standing	Sitting
Standing	349	1
Sitting	18	332

When observing the learning curves, as the training samples increase, both cross-validation score and training score gradually become larger until nearly achieves 1. This fact shows that SVM has a high bias at the beginning and a low bias at the end of the training process, meaning that SVM suffers from under fitting and fits the model greatly when all training examples are imported. In comparison with the learning curves of Random Forest, SVM cross-validation score is smaller than that of Random Forest when there are few training examples. Therefore, we can conclude that SVM performs worse than Random Forest when there are few training examples and performs nearly as well as Random Forest after all training examples are trained.

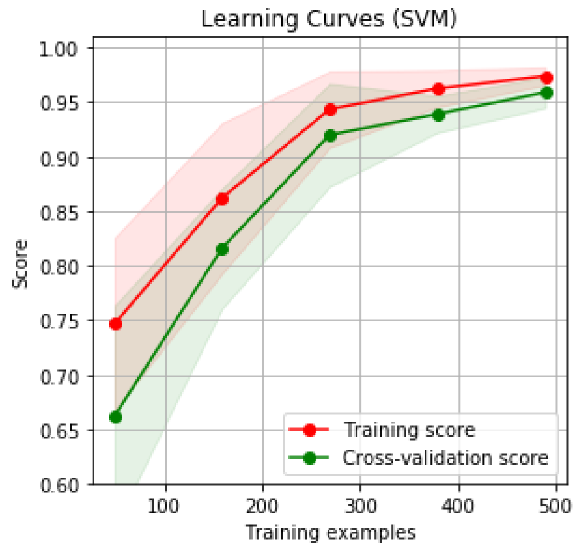

Fig. 7. Learning curves of SVM.

The Neural Network (NN) model is similar to the work principle of the human brain [32]. The deeper the NN, the model can extract more features of data, but this can also increase the possibility that the model will be over-fitting while a few layers network can lead to under-fitting. Thus, it is difficult to achieve a balance between under-fitting and over-fitting [33]. After a careful selection, we chose 5 hidden layers with 20 hidden nodes in each layer. Apart from that, learning strategies can be crucial in training the NN model and the best learning strategy should have the lowest loss at the end.

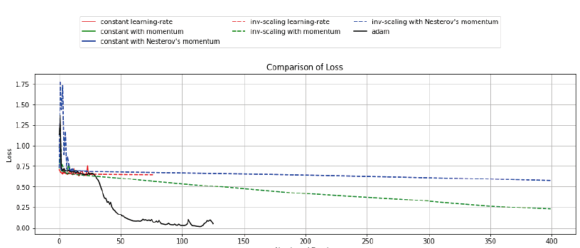

Fig. 8. Comparison of different learning strategies.

From comparison in Fig. 8, we can find that adam has the lowest loss after training and it achieves its optimal point quickly. Therefore, we select adam as our learning strategy.

After those optimizations, we finally achieved a great result in the Neural Network model with accuracy=0.94, precision=0.94, recall=0.94 and f1-score=0.94. From the confusion matrix, it is noticeable that performance over sitting is much worse in comparison with standing.

TABLE IV
CONFUSION MATRIX OF NEURAL NETWORK

True \ Predicted	Standing	Sitting
Standing	341	9
Sitting	32	318

From its learning curve, we can find that the training score stays around 1 while cross-validation score grows gradually until around 0.95. Therefore, the Neural Network model has low bias but high variation, meaning that the Neural Network has the problem of over fitting but it is not very serious as the difference is not large and we can still use Neural Network as part of our ensemble machine learning classifier.

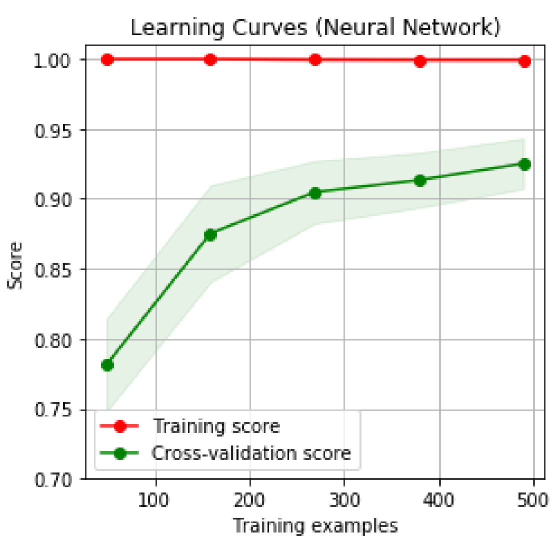

Fig. 9. Learning curves of Neural Network.

Based on the above analysis of 4 machine learning models, we ensemble three models: Random Forest, SVM and Neural Network (abandon KNN due to its serious over fitting problem) to be our ensemble classifier with 'hard vote' [34], meaning that only the prediction with most of votes will be the prediction. Assume that predictions made by Random Forest, SVM and Neural Network are independent, Random Forest=C1, SVM=C2, Neural Network=C3, Ensemble Classifier=C, Standing=S1, Sitting=S2.

Based on those assumptions and statistical knowledge, we can find the probabilities:

$P(C = S1|S1) = P(C1 = S1|S1) \cdot P(C2 = S1|S1) \cdot P(C3 = S1|S1) + P(C1 = S2|S1) \cdot P(C2 = S1|S1) \cdot P(C3 = S1|S1) + P(C1 = S1|S1) \cdot P(C2 = S2|S1) \cdot P(C3 = S1|S1) + P(C1 = S1|S1) \cdot P(C2 = S1|S1) \cdot P(C3 = S2|S1) = \frac{341}{350} \cdot \frac{349}{350} \cdot \frac{341}{350} + \frac{9}{350} \cdot \frac{349}{350} \cdot \frac{341}{350} + \frac{341}{350} \cdot \frac{1}{350} \cdot \frac{341}{350} + \frac{341}{350} \cdot \frac{349}{350} \cdot \frac{9}{350} = 0.9992$

$P(C = S2|S2) = P(C1 = S2|S2) \cdot P(C2 = S2|S2) \cdot P(C3 = S2|S2) + P(C1 = S1|S2) \cdot P(C2 = S2|S2) \cdot P(C3 = S2|S2) + P(C1 = S2|S2) \cdot P(C2 = S1|S2) \cdot P(C3 = S2|S2) + P(C1 = S2|S2) \cdot P(C2 = S2|S2) \cdot P(C3 = S1|S2) = \frac{349}{350} \cdot \frac{333}{350} \cdot \frac{318}{350} + \frac{1}{350} \cdot \frac{333}{350} \cdot \frac{318}{350} + \frac{349}{350} \cdot \frac{17}{350} \cdot \frac{318}{350} + \frac{349}{350} \cdot \frac{333}{350} \cdot \frac{32}{350} = 0.9952$

From the law of total probability, we can get the result:

$\because P(S1) = P(S2) = P(C = S1) = P(C = S2) = 0.5, P(C = S1) = P(S2) \cdot P(C = S1|S2) + P(S1) \cdot P(C = S1|S1), P(C = S2) = P(S1) \cdot P(C = S2|S1) + P(S2) \cdot P(C = S2|S2) \therefore P(C = S1|S2) = 1 - P(C = S1|S1) = 1 - 0.9992 = 0.0008, P(C = S2|S1) = 1 - P(C = S2|S2) = 1 - 0.9952 = 0.0048$

From the above analysis, we find that using the ensemble classifier can increase the precision of the model, which will help us improve performances of the machine learning model. However, in practice, the three machine learning models are not independent. Thus, we cannot have an ensemble classifier as good as the ideal one. But it can still achieve $accuracy = 0.987 \approx 0.99, precision = 0.987 \approx 0.99, recall = 0.987 \approx 0.99 and f1 - score = 0.987 \approx 0.99$ and the confusion matrix below shows that the ability of prediction for standing and sitting are nearly the same for ensemble classifier.

TABLE V
CONFUSION MATRIX OF ENSEMBLE CLASSIFIER

True \ Predicted	Standing	Sitting
Standing	346	4
Sitting	5	345

The learning curves show that the training score of the ensemble classifier stays around 1 while the cross-validation score grows from around 0.78 to near 0.99 as the training examples increase. Therefore, we can conclude that ensemble classifier has low bias but high variation when there are around 200 training examples and it overcomes this issue when all 490 training examples are trained. This fact proves that ensemble classifier really works.

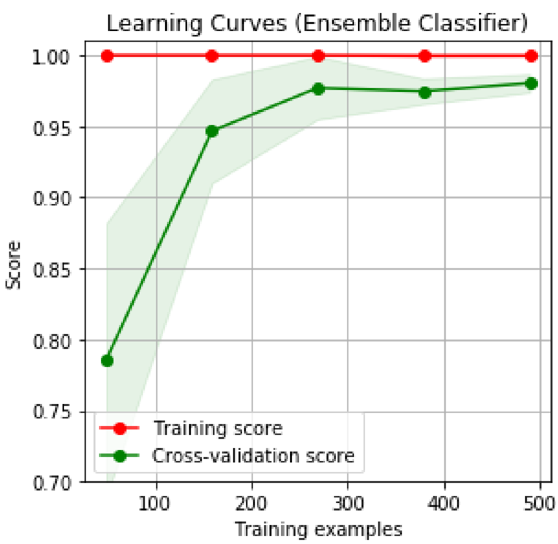

Fig. 10. Learning curves of Ensemble Classifier.

Before using our machine learning model, we made a comparison between the performance of our CSI Dataset versus a Benchmark Dataset measured by accelerators of smart phones in an attempt to verify the effectiveness of our CSI dataset. The benchmark dataset is obtained from the dataset constructed by UCI [37], where human motions are measured by accelerators of smart phones.

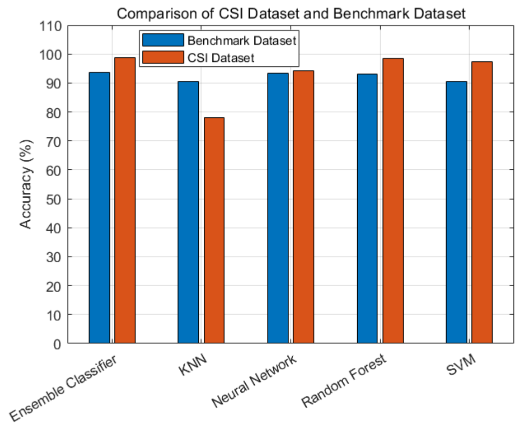

Fig. 11. Comparison of Benchmark Dataset and CSI Dataset.

From Fig. 11, we can find that our CSI Dataset performs better in 4 machine learning algorithms while Benchmark Dataset gains a better performance in terms of KNN. This comparison shows that CSI Dataset gains better performance compared with Benchmark Dataset in most machine learning models used in this project. Therefore, it is reasonable to adapt CSI Dataset when we are trying to classify human motions.

TABLE VI
COMPARISON OF BENCHMARK DATASET AND CSI DATASET

Machine Learning Model	***Benchmark Dataset Accuracy (%)***	***CSI Dataset Accuracy (%)***
Random Forest	93.01	98.57
K Nearest Neighbors	90.40	78.14
Support Vector Machine	90.48	97.29
Neural Network	93.42	94.14
Ensemble Classifier	93.64	98.71

After applying Random Forest, Benchmark Dataset can get an accuracy of 93.01% while that of CSI dataset is 98.57%. Compared with other machine learning models, the performance can rank 3rd in Benchmark dataset and 2nd in CSI dataset with only a little difference to the best model (0.63% in benchmark dataset and 0.14% in CSI dataset). Thus, it is reasonable to apply Random

Forest when there is only a limited source of computation. The KNN model acquires accuracies of 90.40% and 78.14% in Benchmark Dataset and CSI Dataset respectively. It is noticeable that the KNN model has the worst performance in both Benchmark Dataset and CSI Dataset. Accuracy of 90.48% in Benchmark Dataset and accuracy of 97.29% are gotten by Support Vector Machine. According to the accuracies, Support Vector Machine can rank 4th in Benchmark Dataset and rank 3rd in CSI Dataset. This fact proves that SVM can perform well in the task of human motion classification. The Neural Network model gets an accuracy of 93.42% in Benchmark Dataset, ranking 2nd and gets an accuracy of 94.14% in CSI dataset, ranking 4th. Neural Network has proved to be a useful model over the problem of classification and this project proves this fact again, but it might suffer from its high cost of computation source and issue of over fitting. Lastly, Ensemble Classifier is best in both Benchmark Dataset and CSI Dataset with an accuracy of 93.64% in Benchmark Dataset and 98.71% in CSI Dataset due to the principle of probability behind it. However, since it depends on predictions of three machine learning models to make its own decision, it will take the largest amount of computing source. Thus, it is not recommended for embedded devices with limited computing source.

Overall, a better performance can be gained by CSI Dataset in most machine learning models used in this project, proving the effectiveness of CSI Dataset. Among 5 machine learning models used in this project, Random Forest model is suitable for using in a place with limited computing resources like mobile phones, while Ensemble Classifier should be used in a place with abundant computing sources such as cloud computing.

C. Local Classification and Real time Classification

After we finish the machine learning process, we can finally enter into the testing process. In this project, two modes of testing are created, one is a local test and the other is a real time test. The local test can be applied when there are limited or no internet sources while the real time test serves for the conditions with internet sources.

The following flow chart explains how the local test and real time test work. Most of the processes of two tests are the same and the differences are 1. The local test uses data from local files to predict while the real time test applies data from online dataset 2. The local test will display the result on the local GUI and the real time test will display the result on the real time web interface. At the beginning of the process, volunteers sit or stand when the USRP will record its CSI data and this data will be stored in the MATLAB. After that, Python will extract CSI data from MATLAB. Compared with the training and testing process, we use all data for training machine learning models in an attempt to increase the accuracy of the model. Thanks to the convenience of scikit library, both ensemble classifier model and Random Forest model can be saved and reused by using the joblib library for the situations when there are sufficient computing sources and limited computing sources respectively. After the saved machine learning model is imported into python, the model can make predictions based on the extracted CSI data and this prediction will be displayed on the local GUI or real time web interface.

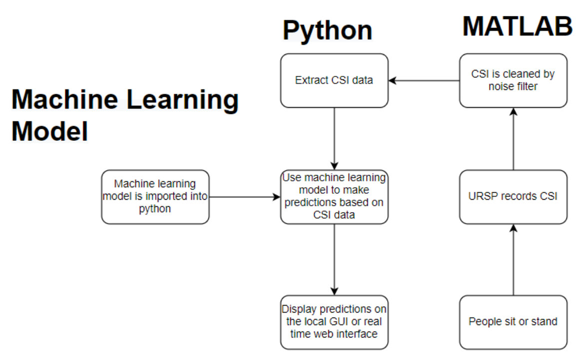

Fig. 12. Flow chart of local test and real time test process.

The following two graphs of GUI show how the local test is performed when the human movement is sitting and standing. For testing, we print the tags of data as the real result and put features into the integrated machine learning model to predict. After the result has been established, the predicted result will be printed on the GUI as prediction. It is clear that our machine learning model can predict the right result as its tag shows, no matter whether the tag is standing or sitting.

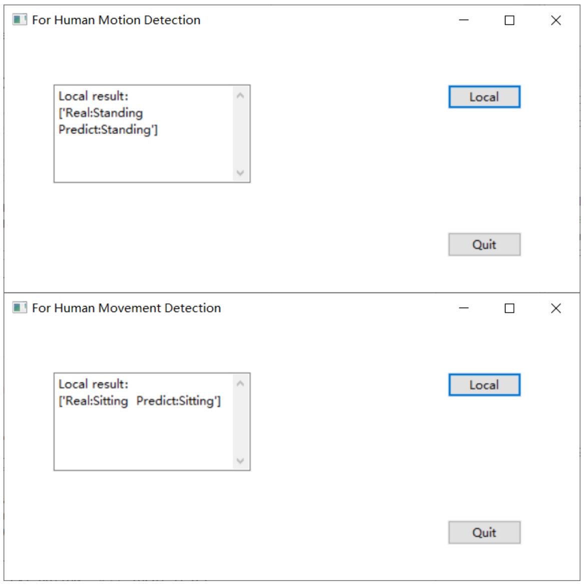

Fig. 13. Local classification demo.

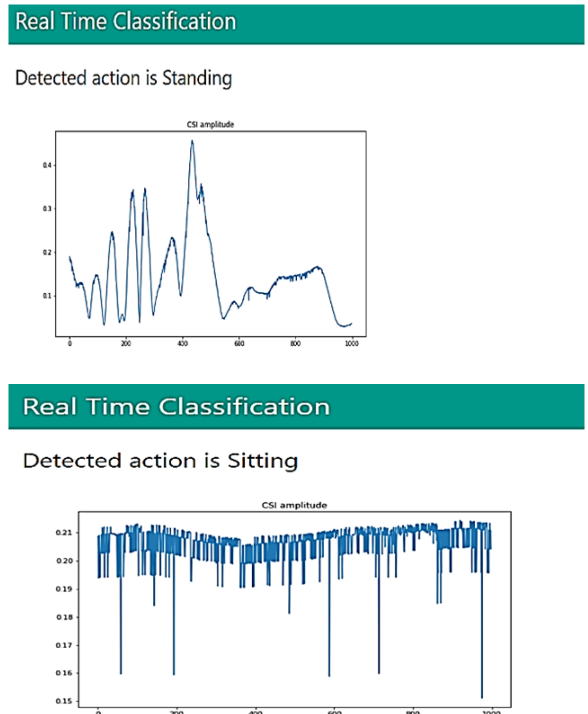

Fig. 14. Real time classification demo.

The real time test can show both CSI amplitude graph and predicted result on the web interface. From Fig. 14, we can find that the CSI amplitude of standing has a large difference in contrast with that of sitting. The CSI amplitude of standing experiences huge changes from around 0.05 to 0.43 as time goes while that of sitting vibrates around 0.19 to 0.21 most of the time. Thus, they have distinguished features that can be recognized by our integrated machine learning models, which is the foundation of our machine learning classification.

III. ANALYSIS AND DISCUSSION

The most important part of this work is how we achieve a balance between increasing accuracy and saving computing resources in the machine learning process. What we first planned to do is by testing different machine learning algorithms and choosing the best one based on accuracy, precision, recall and f1-score, four statistical standards of machine learning models. However, we soon realized that these statistical standards cannot reflect whether the model is under fitting, over fitting or just fitting during the training process. In an attempt to explore deeper, we plotted the learning curves of Random Forest, KNN, SVM and Neural Network. It turns out that Random Forest, KNN and Neural Network suffer from over fitting while SVM has the problem of under fitting. Among those machine learning algorithms, KNN has the worst performance while Random Forest is the best in terms of accuracy but its performance does not exceed SVM and Neural Network much. Based on this, we were inspired that it might be a great idea to combine the advantages of different models to make predictions. Then, we created the ensemble classifier which makes prediction based on the votes of three models: Random Forest, SVM and Neural Network (KNN is abandoned due to its worst performance). The result of ensemble classifier is even better than Random Forest from both theoretical analysis and the real test. However, because it requires votes from three machine learning models to make its own decision, it will use the most computing sources, which are not suitable for conditions with limited computing sources. Therefore, we applied the Random Forest model to make predictions when computing sources are limited and Ensemble Classifier when there are sufficient computing sources for both the local test and the real time test.

After making all these modifications to balance accuracy and computing source, we made a comparison between our CSI Dataset and a traditional embedded device Benchmark Dataset in terms of accuracy with the same machine learning models to test whether our CSI Dataset is effective or not. In this comparison, we kept parameters of machine learning models the same to ensure fairness. The result of comparison demonstrates that our CSI Dataset can gain a better result in Random Forest, SVM, Neural Network and Ensemble Classifier while Benchmark Dataset can perform better in KNN. This result proves the effectiveness of our CSI Dataset in terms of predicting human activities and our CSI Dataset can gain a better performance in most machine learning models used in this project.

Although our designed Real-Time Human Activity Recognition System has been proved to have a great performance through testing its statistical performances and comparing its result with a Benchmark Dataset, we still cannot ensure that it can perform well in real life. Since different people might generate different CSI, the CSI signals in real life might follow different patterns in contrast with the CSI Dataset that we created, which is called concept drift formally. The concept drift might result in a decline of accuracy in the real time test. Apart from concept drift, it is hard for us to improve the recognizing ability of our recognition system after the training process is finished. Implementing an online machine learning method into this CSI Dataset might be meaningful to improve accuracy in this project. There are mainly two reasons for adapting to online machine learning or incremental learning: 1. incremental learning can adapt to concept drift, like changes in data distribution 2. online machine learning is able to process an infinite data stream with finite resources [36]. Those two advantages can help our Human Activity Recognition System gradually adapt to different CSI signals and improve its accuracy. Due to the limit of time, we decided to apply online machine learning in the future.

IV. CONCLUSIONS AND FUTHERWORK

A. Conclusions

What we have achieved is to provide a dataset of human motion based on USRP CSI, test effects of several traditional machine learning algorithms on this dataset, propose an ensemble classifier from three best performed machine learning algorithms and ensemble the Random Forest and Ensemble classifier into local GUI and real time web interface for the conditions with limited and sufficient computing resource respectively.

The CSI Dataset records two types of human motion: standing and sitting, which gets an accuracy over 98% for both Random Forest and ensemble classifier. This high accuracy implies the fact that CSI of standing and sitting have their distinguished features that can be found by machine learning algorithms, Random Forest and Ensemble Classifier can distinguish their differences best. From learning curves of Random Forest, KNN, Neural Network and Ensemble Classifier, we can find that they have high variance at the beginning of training but only KNN still has this high variance after the training process is finished, while other machine learning models overcome this over fitting problem successfully. Unlike other machine learning models, SVM suffers the problem of high bias at the beginning of training but it gradually overcomes after more training examples are trained.

In an attempt to compare the effect of our CSI Dataset and the effect of traditional embedded device dataset, we found a Benchmark Dataset from UCI and tested it with the same machine learning models. The result shows that our dataset performs better in Random Forest, SVM, Neural Network and Ensemble Classifier while UCI Benchmark Dataset gains a better performance in KNN. Since our CSI Dataset performs better in most machine learning models, we can verify the effectiveness of our CSI Dataset. This comparison shows that KNN has the worst performance in both CSI Dataset and Benchmark Dataset, which might imply that KNN is not suitable for human motion classification.

Random Forest and Ensemble Classifier are selected as models that are imported into the local test and the real time test. Although Ensemble Classifier can perform best among all models, it needs to take the largest amount of computing source, which is not beneficial for places with limited computing source. In an attempt to tackle this limit in computing source, Random Forest with the second-best performance can be applied. Though it cannot perform as good as the Ensemble Classifier, it still can gain a relatively satisfactory result in terms of accuracy and, most importantly, it can utilize the precious computing source well.

B. *Suggestions for further work*

One of the biggest issues of traditional offline machine learning is that it can perform well in the training set but its performance will become worse once tested in real scenarios. This is because there might be a concept drift between data in the training set and testing set in real scenarios. In contrast, online machine learning will use the data in the testing set to update its parameters and gradually adapt to the special features of user and the features learned from previous training data will gradually be forgotten. This can be very useful to improve the accuracy in real scenarios. As different people have different features when they perform their motions, enabling the machine learning model to learn corresponding user's features is important for the progress of the machine learning model. The explanation of an online machine algorithm is shown below:

Algorithm 1 Online Machine Learning Algorithm

Input: Get (x,y) corresponding to the current user.....;

Output: Parameters of online machine learning model θ......

0: Repeat forever as long as the website is running......

0: **for** j=0:n **do**

0: θ=$\theta_j - \alpha($ h_θ(x)-y)

Real life is composed of infinite data stream while traditional offline machine learning can only use finite data to train. However, online machine learning can continually learn from the infinite data stream. Although online machine learning can only have a small difference compared with traditional machine learning at the beginning, as time goes by, online machine learning will outperform traditional machine learning since online machine learning keeps learning from new data stream while traditional machine learning just stops learning.

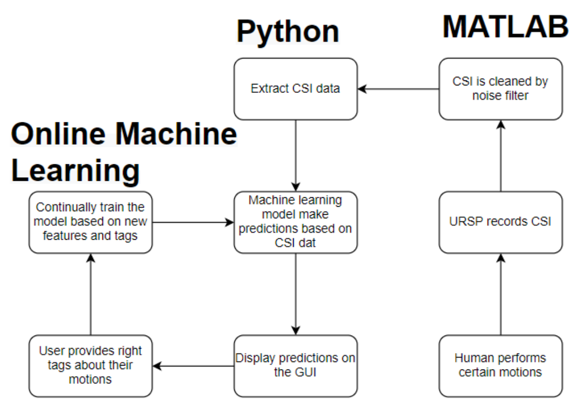

Fig. 15. Planned Online Machine Learning based GUI.

In our plan, a GUI based on the online machine learning model will be constructed. This GUI can intellectually adapt to the concept drift that happened in the real scenario and gain a better performance towards certain users as time goes by, while the previous GUI cannot behave better as time elapses. This is a great improvement compared with previous designs. The overall design of the Online Machine Learning based GUI can be found in Fig. 15. It consists of three parts: MATLAB, Python and Online Machine Learning Model. The MATLAB part is responsible for detecting motion, recording and storing CSI. The Python part mainly works for making predictions based on the Online Machine Learning Model and presenting the result of prediction to people and the Online Machine Learning Model should receive users' feedback and adjust its parameters based on users' feedback.

ACKNOWLEDGMENT

The author would like to thank buddy pair PhD and Director, who provide useful guidance in this paper. The author is grateful to Glasgow College UESTC for its assistance during this research.

REFERENCES

[1] Yang, X.; Ren, X.; Chen, M.; Wang, L.; Ding, Y. Human Posture Recognition in Intelligent Healthcare. J. Physics: Conf. Ser. 2020, 1437, 012014.

[2] Abbasi, Q.H.; Rehman, M.U.; Qaraqe, K.; Alomainy, A. Advances in Body-Centric Wireless Communication: Applications and State-of-the-Art; Institution of Engineering and Technology: London, UK, 2016.

[3] Santos, G.L.; Endo, P.T.; Monteiro, K.H.d.C.; Rocha, E.d.S.; Silva, I.; Lynn, T. Accelerometer-based human fall detection using convolutional neural networks. Sensors 2019, 19, 1644.

[4] Jilani, S.F.; Munoz, M.O.; Abbasi, Q.H.; Alomainy, A. Millimeter-wave liquid crystal polymer based conformal antenna array for 5G applications. IEEE Antennas Wirel. Propag. Lett. 2018, 18, 84–88.

[5] Liu, Y.; Zhang, L.; Yang, Y.; Zhou, L.; Ren, L.;Wang, F.; Liu, R.; Pang, Z.; Deen, M.J. A novel cloud-based framework for the elderly healthcare services using digital twin. IEEE Access 2019, 7, 49088–49101.

[6] Fan, D.; Ren, A.; Zhao, N.; Yang, X.; Zhang, Z.; Shah, S.A.; Hu, F.; Abbasi, Q.H. Breathing rhythm analysis in body centric networks. IEEE Access 2018, 6, 32507–32513.

[7] Yao, X.; Khan, A.; Jin, Y..Energy Efficient Communication among Wearable Devices using Optimized[A]. Symposium on Computers and Communications[C].Barcelona, Spain: IEEE, 29 June–3 July 2019:1–6.

[8] Yang, X.D.; Abbasi, Q.H.; Alomainy, A.; Hao, Y..Spatial correlation analysis of on-body radio channels[A]. Antennas Wirel[C].IEEE, 2011:780–783.

[9] Aveek K.;Parth H.;Chen-Nee C.;Prasant M..Uncovering Privacy Leakage in BLE Network Traffic of Wearable Fitness Trackers[A].HotMobile '16: Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications[C].Association for Computing MachineryNew YorkNYUnited States, 2016:99-104.

[10] Zhao, J.; Liu, L.; Wei, Z.; Zhang, C.; Wang,W.; Fan, Y..R-DEHM: CSI-based robust duration estimation of[J].Sensors, 2019(19):1421.

[11] Chopra, N.; Yang, K.; Abbasi, Q.H.; Qaraqe, K.A.; Philpott, M.; Alomainy, A..THz time-domain spectroscopy of human skin tissue for in-body nanonetworks.[J].IEEE Trans, 2016(6):803-809.

[12] Lolla, S.; Zhao, A..WiFi Motion Detection: A Study into Efficacy and Classification.[A].Integrated STEM Education Conference (ISEC),[C].IEEE, 16 March 2019:375–378.

[13] Demir, A.F.; Abbasi, Q.H.; Ankarali, Z.E.; Alomainy, A.; Qaraqe, K.; Serpedin, E.; Arslan, H.Anatomical region-specific in vivo wireless communication channel characterization.[J].IEEE J. Biomed. Health Inf, 2016(21):1254–1262.

[14] Christiansen, J.M.; Smith, G.E.Development and Calibration of a Low-Cost Radar Testbed Based on the[J].IEEE Aerosp. Electron. Syst. Mag, 2019(34):50–60.

[15] Ashleibta, A.; Shah, S.; Zahid, A.; Imran, M.A.; Abbasi, Q.H..Software Defined Radio Based Testbed for[J].IEEE Aerosp. Electron. Syst. Mag, 2020(Accepted for Publication).

[16] Zhang, J.; Xu, W.; Hu, W.; Kanhere, S.S..WiCare: Towards In-Situ Breath Monitoring.[A].the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and[C].Melbourne, VIC, Australia, 7–10 November 2017:126–135.

[17] Shah, S.A.; Fioranelli, F. RF sensing technologies for assisted daily living in healthcare: A comprehensive review. IEEE Aerosp. Electron. Syst. Mag. 2019, 34, 26–44.

[18] Yang,X et al.,.Band Sensing-Based Motion Assessment Framework for Cerebellar Dysfunction Patients[J].IEEE Sensors Journal, 1 Oct.1, 2019,(19):8460-8467.

[19] Wang, X;Yang, C;Mao, S.TensorBeat: Tensor Decomposition for Monitoring Multi-Person Breathing Beats with Commodity WiFi[J].ACM Transactions on Intelligent Systems and Technology, Sep, 2017,(8):https://doi.org/10.1145/3078855.

[20] Wang, Y; Wu, K; Lionel, M.WiFall: Device-Free Fall Detection[J].IEEE TRANSACTIONS ON MOBILE COMPUTING, 2017, 16(2).

[21] Kiran, J;Dinesh, B;Manikanta, K; Sachin K.WiDeo: Fine-grained Device-free Motion Tracing[J].USENIX Symposium on Networked Systems, May 4–6, 2015(Oakland, CA, USA):ISBN 978-1-931971-218.

[22] Heba, A;Moustafa, Y;Khaled A.WiGest: A Ubiquitous WiFi-based Gesture[A].IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)[C].IEEE, 2015:17-18.

[23] Zheng,X; Wang,J; Shangguan, L; Zhou, Z;Liu, Y.Design and Implementation of a CSI-Based Ubiquitous Smoking Detection System[J].IEEE/ACM Transactions on Networking, Dec. 2017, 25(6):3781-3793.

[24] Wang, F;Han, J; Zhang, S;He, X;Huang, D.CSI-Net: Uni-

fied Human Body Characterization and Pose Recognition[D].arXiv, Oct. 2018.

[25] Wang, F;Stanislav, P; Dai, Z;Han, J;Huang, D.Can WiFi Estimate Person Pose?[D].arXiv, Mar 30. 2018.

[26] Ali, I;Iryna, P;Haneya, N;Usama, M .AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app[J].Informatics in Medicine Unlocked, 2020, 20(https://doi.org/10.1016/j.imu.2020.100378).

[27] Hao, J.; Ho, T.K..Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming[J].J. Educ. Behav. Stat., 2019(44):348-361.

[28] Aditya Mishra.Metrics to Evaluate your Machine Learning Algorithm[EB/OL].2018[2021-4-1].https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234.

[29] Claudia Perlich.Learning Curves in Machine Learning[R].Thomas J.Watson Research Center:IBM Research Division, 2009.

[30] Saçlı, B.; Aydınalp, C.; Cansız, G.; Joof, S.; Yilmaz, T.; Çayören, M.; Önal, B.; Akduman, I..Microwave dielectric property based classification of renal calculi: Application of a kNN algorithm[J].Comput. Biol. Med., 2019(112):103-366.

[31] Jain, M.; Narayan, S.; Balaji, P.; Bhowmick, A.; Muthu, R.K.; Bharath, K.P.; Karthik, R..Speech emotion recognition using support vector machine[EB/OL].2020[2020-12-21].https://arxiv.org/abs/2002.07590.

[32] Hamid, Y.; Sugumaran, M.; Balasaraswathi, V..Ids using machine learning-current state of art and future directions[J].Curr. J. Appl. Sci. Technol., 2016(15):1-22.

[33] Biswas, S.K..Intrusion detection using machine learning: A comparison study[J].Int. J. Pure Appl. Math., 2018(118):101-114.

[34] Pedregosa, F; Varoquaux, G.; Gramfort, A.; Michel, V..Ensemble methods[EB/OL].2021[2021-4-1].https://scikit-learn.org/stable/modules/ensemble.html.

[35] Jorge-L. Reyes-Ortiz, Luca Oneto, Albert SamÃ , Xavier Parra, Davide Anguita. Transition-Aware Human Activity Recognition Using Smartphones. Neurocomputing. Springer 2015.

[36] Miguek, l.;Eduardo, G.;Carlos, M.;Juan, I..Online machine learning algorithms to predict link quality in community wireless mesh networks[J].Computer Networks, 2018(132):68-80.