Determining the way of walking by leveraging CSI.

Jeroen Braks
University of Twente
PO Box 217, 7500 AE Enschede
the Netherlands

ABSTRACT

Due to the growing possibilities of the Internet of Things and the lasting importance of healthcare, these fields have been heavily investigated. This has, among other things, resulted in the use of monitoring techniques with e.g. sensors on patient's bodies. This thesis is focused on the opportunities of channel state information, an unobtrusive remote sensing technique. Channel state information can be used to monitor movements in-between its transmitters and receivers. While much has been investigated in the field of remote sensing, this specific topic has not been extensively explored yet. Therefore, this research is going to determine how accurate ways of walking can be determined using channel state information sensing in spacious rooms. In this thesis, deep learning will be used to determine the accuracy between data of single and multiple transmitter-receiver pairs. A comparison of datasets will be used to draw a conclusion on how accurately ways of walking can be determined in spacious rooms. This paper will conclude that ways of walking can be classified fairly accurate using channel state information in spacious rooms.

Keywords

Channel state information; Healthcare; Internet of Things; Convolutional neural network; Deep learning

1. INTRODUCTION

In healthcare, there is an increasing demand to monitor patients' state and progress [5, 13]. By monitoring patients, more appropriate care can be provided, and patients' overall wellbeing can be improved. Different monitoring techniques have been used in the past years, such as audiovisual monitoring techniques. These techniques are proven to be accurate, but very privacy-sensitive, which makes them inappropriate to use in most cases. Other techniques such as on-body sensing are less privacy-sensitive, but make compromises on user comfort. By using remote sensing instead, unobtrusive sensing can be achieved, improving privacy as well as user comfort [3]. For this purpose, channel state information can be considered; channel state information uses the multipath effect that occurs when multiple transmitters send Wi-Fi signals to the receivers. By analyzing the signal that arrives at the receivers, human activity can be recognized without wearables required.

The goal of this research project is to determine how accurate ways of walking can be monitored in spacious rooms. Both the use of a single and the use of multiple transmitter-receiver pairs will be investigated.

Experimental environments are set up by researchers and for initial data collection, small experiment rooms are often sufficient. However, for the technology to be of use in e.g. healthcare, it must be scalable. If scalability is possible and protocols have been written on how to set up channel state information technology in larger rooms, the possibilities of use will largely increment.

The following research questions are defined:

RQ1:

How accurately can normal walking be differentiated from disturbed walking by using channel state information in a spacious apartment?

RQ1.1:

By using a single transmitter-receiver pair.

RQ1.2:

By using multiple transmitter-receiver pairs.

To answer the question of how accurately normal and disturbed walking can be differentiated in spacious rooms by using channel state information, the accuracy results of both single and multiple transmitter-receiver pairs must be compared. By analyzing these results, an approximation can be made on how accurately normal and disturbed walking can be differentiated in spacious rooms.

First, related work is evaluated. In this section, the used materials are briefly stated. Thereafter the research methodology is given, as well as the data and its annotation. After this, the design of the machine learning algorithm is elaborated on. Lastly, the subsequent test results will be included and discussed, ending with the conclusion.

2. RELATED WORK

Since the possibilities of the Internet of Things are growing, various monitoring techniques in healthcare have been introduced and even more techniques such as video monitoring [26], on-body sensing with wearables [7, 10, 15] and even implanted sensors [2, 6, 21] have been investigated. To eliminate privacy concerns, researchers have been investigating other techniques as e.g. blob trackers [4]. However, even though there were fewer, privacy concerns remained among the elderly [4].

Other less privacy-sensitive techniques are signal-based techniques as received signal strength (RSS) and channel state information (CSI) [3, 8, 9, 25]. Multiple papers indicate degraded performance of RSS monitoring in indoor rooms, while CSI is stated as more stable, sensitive, and robust [12, 14].

There are multiple different machine learning variants, which can be found in research executed by I. G. Maglogiannis et al. [16]. Each variant has multiple different techniques, as described in e.g. [11, 16, 22, 24].

3. DATA COLLECTION

3.1 Dataset

The dataset that will be used is a dataset collected via an experiment conducted by Nikita Sharma et al. [23]. This experiment is executed in the e-health house at the University of Twente in The Netherlands. The complete experimental setup is shown in figure 3.

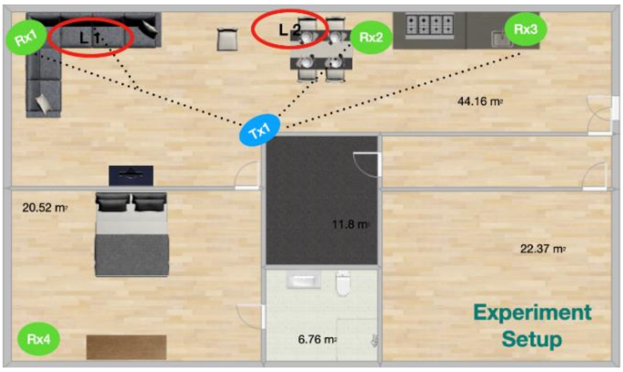

Figure 3. Experimental setup [23].

The experiment is being conducted in an apartment using 4 transmitter-receiver pairs. The combination of a spacious apartment and multiple transmitter-receiver pairs makes this dataset useful for the research.

The data collected by Nikita Sharma's experiment [23] are stored as '.dat' files. Each participant executed normal and disturbed walking both for 120 seconds.

3.2 Technology

The technology used for the experiment done by Nikita Sharma et al. [23] consists of Wi-Fi sensor nodes, a mini-PC, access points and a video recorder. The Wi-Fi sensor nodes are self-contained devices, they are used for data gathering, processing, and communicating the data wirelessly to a control unit [23]. The mini-PC uses an Intel Ultimate Wi-Fi Link 5300 NIC to send a signal over an 802.11n 5.32 GHz network. The channel state information is received and stored at access points. The received packets will be transmitted using 48 Mbps, 3x3 MIMO (Multiple Input Multiple Output), and 64 QAM (Quadrature amplitude modulation)." [23].

Channel state information combines the effect of time delay, energy attenuation and phase shift to monitor movements in a room [8, 25]. Channel state information uses thirty subcarriers, which each have various sensitivity due to selective fading of existing frequency, resulting in fine-grained information [8]. One of the advantages of channel state information compared to other Wi-Fi-based activity recognition techniques as RSS is that channel state information can capture more precise changes like gestures, heartbeat, speaking etc. [8].

In total the experiment makes use of 5 Wi-Fi sensor nodes, four receivers with 3 antennas each and one transmitter node, also with 3 antennas. This results in 270 (3 times 3 antennas times 30 subcarriers) channels per transmitter-receiver pair. Per second, 100 data points are stored for each channel. This results in a total of 100 times 270 data points per transmitter-receiver pair per second.

For baseline data, four video cameras are mounted on the ceiling of the e-health house [23]. The recorded videos are used to annotate the data.

3.3 Experiment

During the experiment, participants were asked to perform tasks on the places L1 and L2. Examples of performed tasks were clapping, hand gestures etc. Furthermore, the participants were asked to walk normally and disturbed through the room. Participants were not allowed to walk or sit in the bedroom what makes receiver Rx4 useless for this research.

The channel state information data of two participants, walking both normally and disturbed, obtained by Nikita Sharma et al. [23] will be used to perform this research. In a total, 8 minutes of data, alongside baseline data; video recordings will be used.

This research makes use of 3 of the 4 receivers (receiver Rx1, Rx2 and Rx3). The fourth receiver, Rx4, will not be used as no tasks were performed in the bedroom.

4. METHODOLOGY

4.1 General methodology

As discussed in section 2, multiple machine learning variants such as supervised, unsupervised, semi-supervised exist. Unsupervised machine learning algorithms are used by researchers to discover new, unknown classes [16]. While supervised learning is often used for classification. Supervised learning requires a dataset with known classes, while unsupervised learning can be executed with unclassified data [16].

According to J. A. Richards et al [11] "Supervised classification is the technique most often used for the quantitative analysis of remote sensing image data." Therefore, as this research investigates how accurate ways of walking can be classified using CSI, supervised classification will be used.

Supervised learning techniques are e.g. logic-based algorithms, perception-based techniques, and statistical learning algorithms [16]. Logic-based algorithms consist of 'decision trees' and 'learning set of rules' [16]. While perception-based techniques are e.g. neural networks. Statistical learning algorithms include e.g. Bayesian networks and instance-based learning [16].

In this thesis is chosen to use a perception based neural network technique: the convolutional neural network (CNN). According to other research, it is "one of the most popular deep neural networks" [1]. Convolutional neural network models process images and have the advantage of training its own features [1].

To ensure the reliability of the network, a pre-trained convolutional neural network is used: ResNet-50. This network must be changed slightly, so that it can be retrained using the dataset conducted by Nikita Sharma et al. [23].

The ResNet-50 model will be trained with 60 per cent of each dataset, as 20 per cent of the data will be used for validating and the remaining 20 per cent for testing the trained model. The validating data will estimate the skill of the model during the training. The test data will be used to calculate the classification accuracy of the final model.

The test accuracy will be stated in a table for comparison of the different datasets. The average validation accuracy will be stated to compare to the average test accuracy, if these values differ significantly, it could either indicate over/underfitting or incorrect split datasets. To illustrate the test results in more detail, confusion matrixes will be computed. These confusion matrixes show the predicted and true class of the classified test data on the trained models.

5. DATA

5.1 Annotation

Supervised machine learning requires classified datasets. Therefore, the data conducted by the experiment [23] must be classified. Two different classes are defined: ‘Disturbed’ and ‘Normal’ (walking).

Next to classifying the data, the data must be annotated so that a dataset with multiple transmitter-receiver pairs can be created. To annotate the data properly, three spaces are made. The experiment room is divided into the sections Rx1 (red), Rx2 (green) and Rx3 (blue), figure 4 is an illustration of these spaces.

For transparency, the data annotation files are attached in appendix 1 and 2.

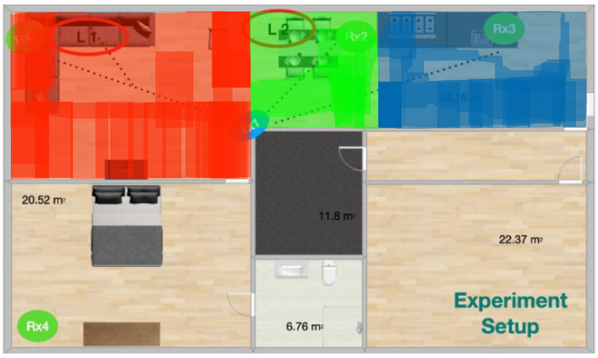

Figure 4. Spaces in experimental setup.

5.2 Data preparation

To be able to plot images, the data is packaged. Each package contains 270 images of 200 consecutive data points (I.e. two seconds of data) with a 50 per cent overlap. Each package thus consists of 270 images: every channel plotted over the same two seconds. The packages are generated out of the original data collected using channel state information, the images are plot using the spectrogram function of MatLab [20].

To access and process the images more quickly, the ‘imageDatastore’ function of MatLab is used [18]. This function returns an object that manages the image files and links the corresponding classes (Disturbed/Normal), which are later used to classify the images.

To prepare the data for the training, validating, and testing the model the data is split randomly among classes. 60 Per cent of the data is used for training, 20 per cent for validating and 20 per cent for testing.

6. MACHINE LEARNING

6.1 Convolutional neural network

A convolutional neural network can be divided into two sections: feature learning and classification. In the feature learning section, the image is converted using a convolutional operation. Convolutional filters are matrixes that are determined by training. These filter matrixes are multiplied by the image matrix, which results in images with different features, see figure 5.

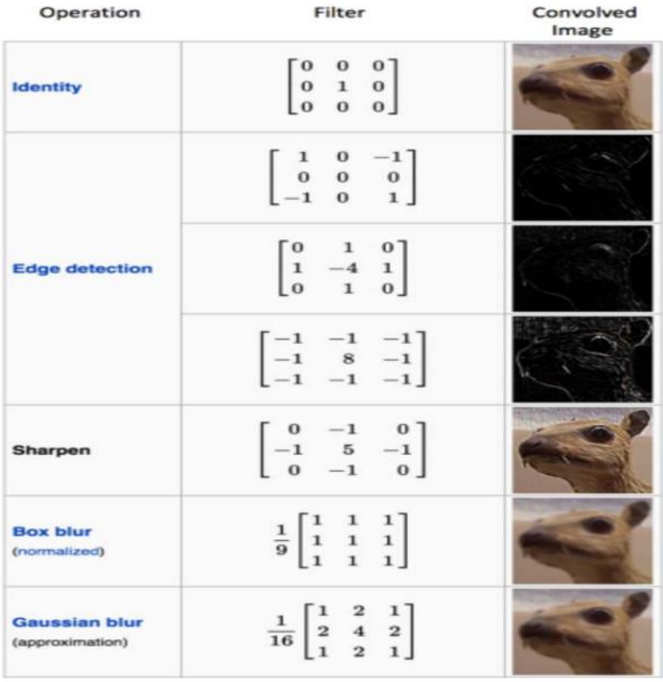

Figure 5. Effects of different convolution matrixes [1].

Besides the convolution layer, a non-linearity layer is applied. This layer is used to adjust or cut off the generated output [1]. ReLu is often used as it has a simpler definition in both function and gradient than other non-linearity layers [1].

In the pooling layer, the dimension of the image is reduced by combining multiple clustered pixels into single pixels, often max pooling is used. Max pooling combines 2x2 matrixes to single values by picking the maximum value of the original 2x2 matrix [1].

Lastly, the fully connected layer; a layer where all nodes are connected to its previous and next layer, is used to classify the image using the trained dataset.

6.2 Design

For the design of the convolution neural network, the deep network designer of MatLab is used [17]. The advantage of using the deep network designer is that a pre-trained network can be adjusted instead of the need of designing a whole new network. As well as the fact that the pre-trained network already is proven to be reliable. For this research, the 50 layered ResNet-50 network is used [19].

The network is slightly adapted, the first layer is adapted from an image input size of 224 x 224 x 3, to 128 x 128 x 3. As well as the fully connected layer at the end, the output size is changed to two, as only two classes are used in our case. Besides this, the weight – and bias learn rate factors are set to ten so that new layers are learned faster while retraining the network.

6.3 Training

The network is trained using the following combinations of data:

Participant 1	Rx1	Rx2	Rx3	Rx1, Rx2 and Rx3
Participant 14	Rx1	Rx2	Rx3	Rx1, Rx2 and Rx3
Participant 1 and participant 14	Rx1	Rx2	Rx3	Rx1, Rx2 and Rx3

7. RESULTS

The training, validation and test results are stated in the table of figure 5. More detailed classification information is visualized in figure 6 to 17, these figures exist of confusion matrixes that visualize the predicted classification of the test data in comparison with the true class.

Algorithm	Training data (60%)	Validation data (20%)	Testing data (20%)	Average validation accuracy (%)	Average testing accuracy (%)	Testing accuracy Rx1 (%)	Testing accuracy with Rx2 (%)	Testing accuracy Rx3 (%)	Testing accuracy Rx1, Rx2 and Rx3 (%)
CNN	P1	P1	P1	87.94	87.85	93.95	89.04	83.37	85.02
CNN	P14	P14	P14	83.27	83.32	85.29	82.97	79.80	85.23
CNN	P1 + P14	P1 + P14	P1 + P14	84.39	84.63	85.25	86.26	81.82	85.19

Figure 5. Deep learning results

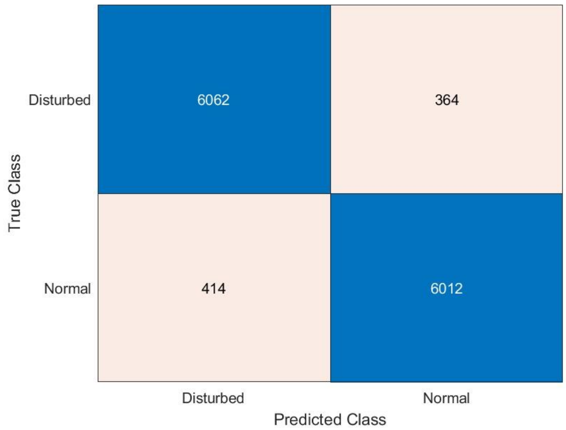

Figure 6. Confusion matrix Rx1 participant 1

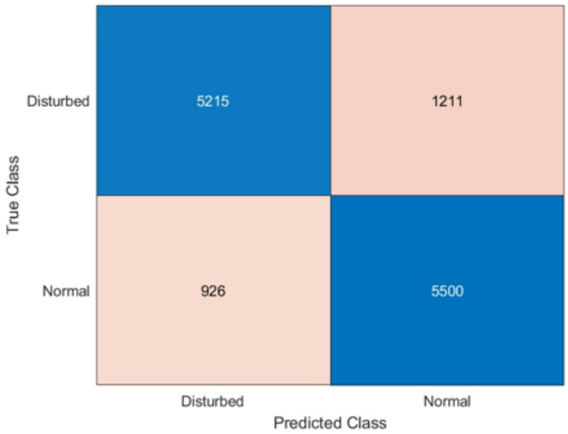

Figure 8. Confusion matrix Rx3 participant 1

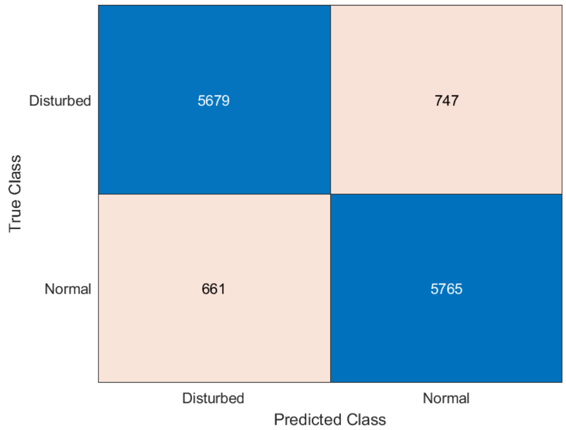

Figure 7. Confusion matrix Rx2 participant 1

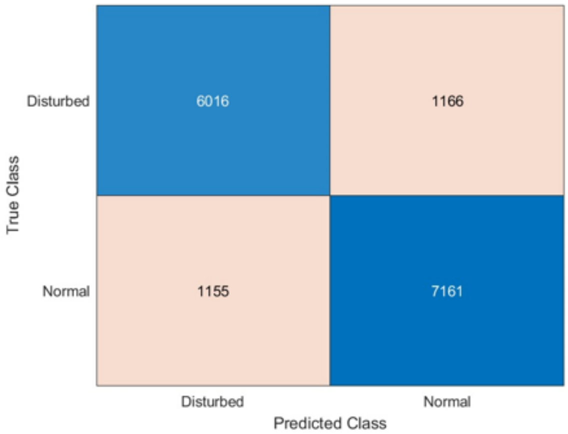

Figure 9. Confusion matrix Rx1, Rx2, Rx3 participant 1

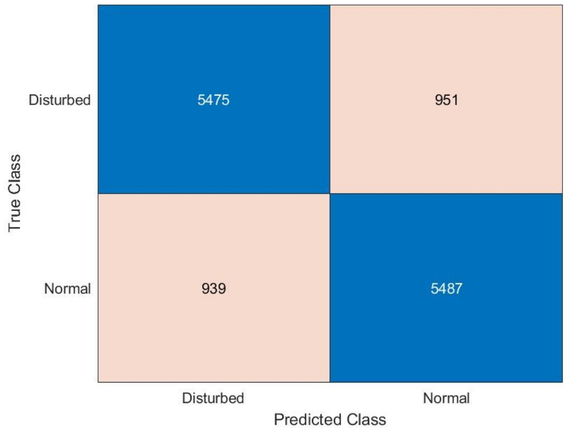

Figure 10. Confusion matrix Rx1 participant 14

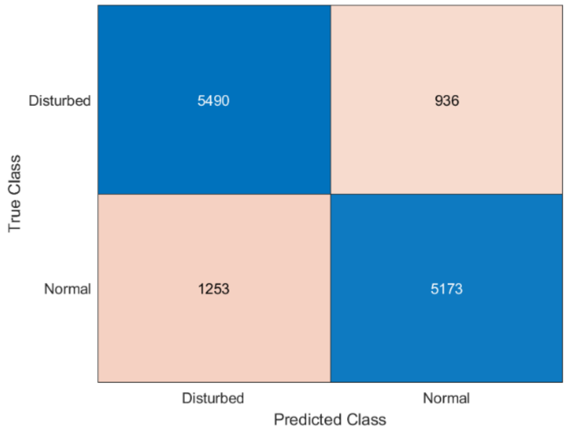

Figure 11. Confusion matrix Rx2 participant 14

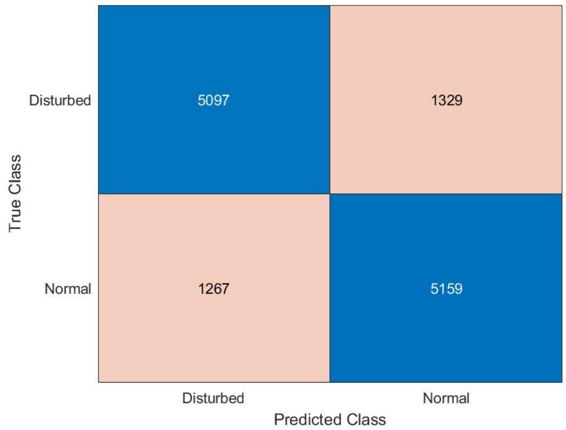

Figure 12. Confusion matrix Rx3 participant 14

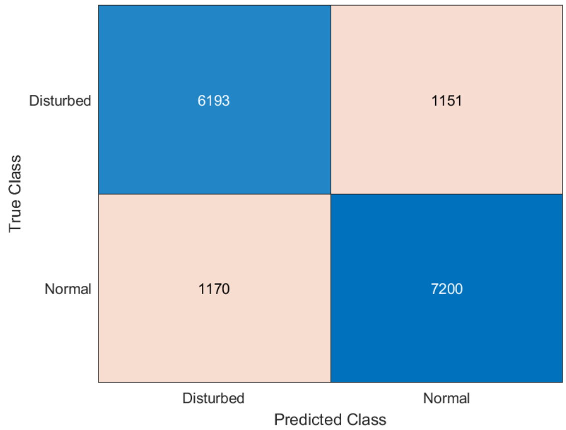

Figure 13. Confusion matrix Rx1, Rx2 and Rx3 participant 14

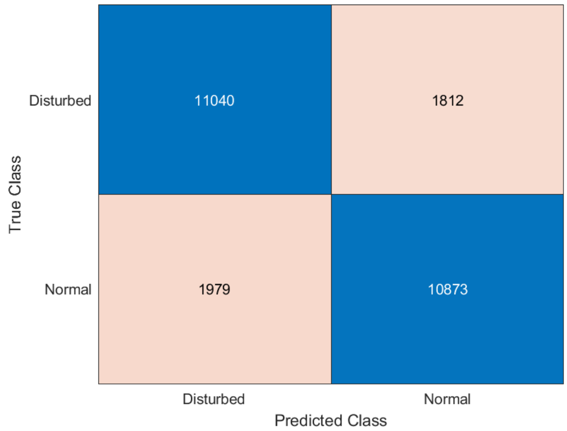

Figure 14. Confusion matrix Rx1 participant 1 and participant 14

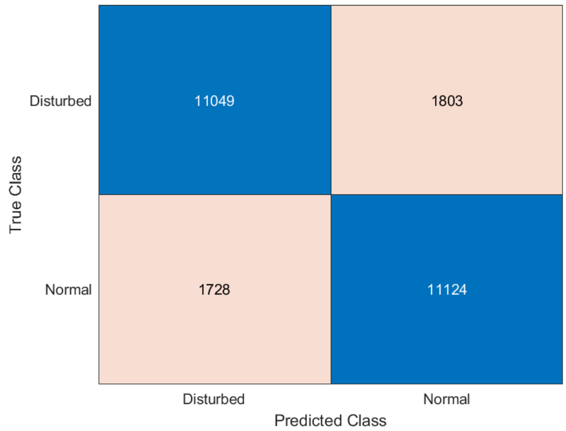

Figure 15. Confusion matrix Rx2 participant 1 and participant 14

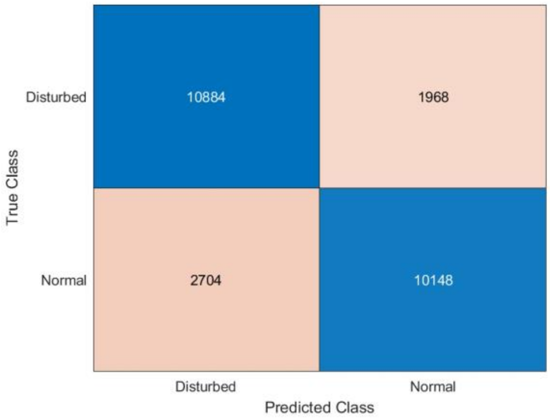

Figure 16. Confusion matrix Rx3 participant 1 and participant 14

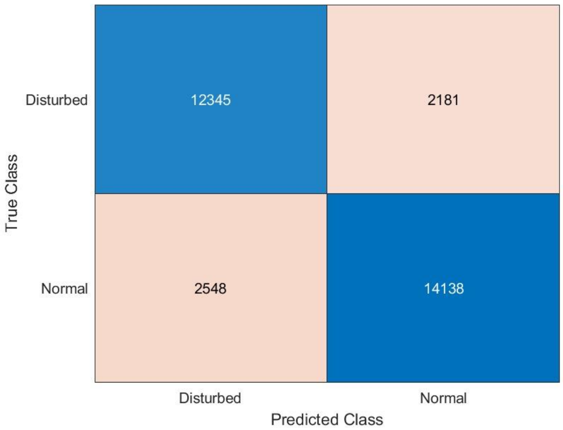

Figure 17. Confusion matrix Rx1, Rx2 and Rx3 participant 1 and participant 14

8. DISCUSSION

The test results show that with a single receiver-transmitter pair a high accuracy can be reached, as is especially the case when training the models with data from Rx1. Both the models with data from participant 1 and participant 14 individually have the highest test accuracy using Rx1. However, when combining the data, Rx2 has a slightly higher accuracy. Rx1 had, when using combined data from participant 1 and 14, in comparison with Rx2, more problems classifying normal data (figure 14 and 15).

Both participants spent a little more time in the red space (figure 4), the space closest to Rx1. In figure 18, an overview of the time spent in each area is stated. Looking at the time spent in each area, it seemed most logically that Rx1 would have the highest accuracy, Rx2 the second highest and Rx3 the lowest accuracy. However, Rx2 has a higher test accuracy than Rx3. This could be argued by the fact that Rx3 is on the far right, and thus Rx1 is further away from receiver Rx3 than from Rx2.

Time (%)	Rx1	Rx2	Rx3
P1	38.33	30.00	31.67
P14	39.58	25.00	35.42

Figure 18. Percentage of time spent in each area.

For the classification rate of the models, using data of multiple participants seems to affect the testing accuracy positively for participants that have a lower accuracy themselves. However, to know this for sure, further research must be done as our testing data contained data from both participants. For this statement to be confirmed, the model should be trained with data from both participants and tested with data from individual participants.

While the accuracy using data of individual receivers fluctuates quite a lot, the models trained with data of receiver Rx1, Rx2 and Rx3 shows a rather consistent accuracy. The accuracy of all three datasets tested (participant 1, participant 14 and participant 1 and 14) is about 85%, even though the average testing accuracy is not consistent.

Overall, the trained models have little, but not significantly, more trouble classifying normal walking. Of the wrongly classified data, 51.74% is normal walking classified as disturbed walking.

9. CONCLUSION

This paper investigated how accurately channel state information can be used in spacious rooms to correctly classify normal and disturbed walking.

The research concludes that by the use of a good placed single transmitter-receiver pair, high accuracy can be achieved when using CNN to classify CSI data. However, data of participants differ and not in all cases an equally high accuracy can be reached. When using a single transmitter-receiver pair, it is important to investigate the optimal placement since accuracy can significantly drop when transmitter-receiver pairs are placed inefficiently. For single transmitter-receiver pairs to be of use in spacious rooms, further research should be done on the placement of the transmitter and receiver, as well as research on the effect of movement in different areas of the room.

Using multiple transmitter-receiver pairs, the research concluded that a fairly high accuracy of 85% seems feasible using channel state information in spacious rooms. The use of multiple transmitter-receiver pairs also seems to decrease the importance of transmitter-receiver placement and movement areas.

This research concludes that CSI can determine normal and disturbed walking in spacious rooms with a consistent accuracy of approximately 85% per cent, using multiple transmitter-receiver pairs. Optimally placed transmitter-receiver pairs could potentially improve the accuracy, using either a single or multiple transmitter-receiver pairs. However, to conclude this, further research in this field is needed.

10. REFERENCES

1. Albawi, S., T.A. Mohammed, and S. Al-Zawi. *Understanding of a convolutional neural network*. in *2017 International Conference on Engineering and Technology (ICET)*. 2017.
2. Andreu-Perez, J., et al., *From Wearable Sensors to Smart Implants--Toward Pervasive and Personalized Healthcare.* IEEE Transactions on Biomedical Engineering, 2015. **62**(12): p. 2750-2762.
3. Brinke, J.K. and N. Meratnia, *Dataset: Channel state information for different activities, participants and days*, in *Proceedings of the 2nd Workshop on Data Acquisition To Analysis*. 2019, Association for Computing Machinery: New York, NY, USA. p. 61–64.
4. Caine, K., A. Fisk, and W. Rogers, *Benefits and Privacy Concerns of a Home Equipped with a Visual Sensing System: A Perspective from Older Adults.* Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 2006. **50**.
5. CapMinds. *Future Of Remote Patient Monitoring In Healthcare 2021*. 2020 [cited 2021 25/05]; Available from: https://www.capminds.com/blog/future-of-remote-patient-monitoring-in-healthcare-2021/.
6. Darwish, A. and A.E. Hassanien, *Wearable and Implantable Wireless Sensor Network Solutions for Healthcare Monitoring.* Sensors, 2011. **11**(6).
7. Fafoutis, X., et al., *Designing wearable sensing platforms for healthcare in a residential environment.* EAI Endorsed Transactions on Pervasive Health and Technology, 2017. **3**(12).
8. Guo, L., et al., *Wiar: A Public Dataset for Wifi-Based Activity Recognition.* IEEE Access, 2019. **7**: p. 154935-154945.
9. Halperin, D., et al., *Tool release: gathering 802.11n traces with channel state information.* SIGCOMM Comput. Commun. Rev., 2011. **41**(1): p. 53.
10. Heikenfeld, J., et al., *Wearable sensors: modalities, challenges, and prospects.* Lab on a Chip, 2018. **18**(2): p. 217-248.
11. J.A., R., *Supervised Classification Techniques*, in *Remote Sensing Digital Image Analysis*. 2013, Springer, Berlin, Heidelberg: Springer. p. 247-318.
12. Lee, H., C.R. Ahn†, and N. Choi, *The Impacts of CSI Temporal Variations on CSI-based Occupancy Monitoring Systems: An Exploratory Study*, in *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. 2020, Association for Computing Machinery: Virtual Event, Japan. p. 278–281.
13. Lishman, L.J. *Remote patient monitoring: the future of healthcare?* 2021 [cited 2021 25/05]; Available from: http://www.pharmatimes.com/web_exclusives/Remote_patient_monitoring_the_future_of_healthcare_1365643.
14. Liu, X., et al., *Contactless Respiration Monitoring Via Off-the-Shelf WiFi Devices.* IEEE Transactions on Mobile Computing, 2016. **15**(10): p. 2466-2479.
15. Liu, Y., M. Pharr, and G.A. Salvatore, *Lab-on-Skin: A Review of Flexible and Stretchable Electronics for Wearable Health Monitoring.* ACS Nano, 2017. **11**(10): p. 9614-9635.
16. Maglogiannis, I.G., *Emerging artificial intelligence applications in computer engineering: real word ai systems with applications in ehealth, hci, information retrieval and pervasive technologies*. Vol. 160. 2007: Ios Press.
17. MathWorks. *Get Started with Deep Network Designer*. 2021 [cited 2021 10/06]; R2021a:[Available from: https://www.mathworks.com/help/deeplearning/gs/get-started-with-deep-network-designer.html.
18. MathWorks. *imageDatastore*. 2021 [cited 2021 10/06]; R2021a:[Available from: https://www.mathworks.com/help/matlab/ref/matlab.io.datastore.imagedatastore.html.
19. MathWorks. *resnet50*. 2021 [cited 2021 10/06]; R2021a:[Available from: https://www.mathworks.com/help/deeplearning/ref/resnet50.html.
20. MathWorks. *spectrogram*. 2021 [cited 2021 10/06]; R2021a:[Available from: https://www.mathworks.com/help/signal/ref/spectrogram.html.
21. Misra, S., P.K. Bishoyi, and S. Sarkar, *i-MAC: In-Body Sensor MAC in Wireless Body Area Networks for Healthcare IoT.* IEEE Systems Journal, 2020: p. 1-8.
22. Prakash, J. and D. Nithya, *A Survey On Semi-Supervised Learning Techniques.* International Journal of Computer Trends and Technology, 2014. **8**.
23. Sharma, N. 2021.
24. Usama, M., et al., *Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges.* IEEE Access, 2019. **7**: p. 65579-65615.
25. Wang, Y., K. Wu, and L.M. Ni, *WiFall: Device-Free Fall Detection by Wireless Networks.* IEEE Transactions on Mobile Computing, 2017. **16**(2): p. 581-594.
26. Zhongna, Z., et al. *Video-based activity monitoring for indoor environments*. in *2009 IEEE International Symposium on Circuits and Systems*. 2009.

APPENDIX

A.5 Data

A.5.1 Disturbed walking data annotation.

PARTICIPANT 1		
Disturbed walking		
Begin time	End Time	Cane?
0	120	Y
Rx1		
Begin time	End Time	
0	16	Y
55	82	Y
Rx2		
Begin time	End Time	
16	25	Y
44	55	Y
82	93	Y
105	120	Y
Rx3		
Begin time	End Time	
25	44	Y
93	105	Y

PARTICIPANT 14		
Disturbed walking		
Begin time	End Time	Cane?
0	120	Y
Rx1		
Begin time	End Time	
0	17	Y
53	74	Y
112	120	Y
Rx2		
Begin time	End Time	
17	25	Y
47	53	Y
74	82	Y
105	112	Y
Rx3		
Begin time	End Time	
25	47	Y
82	105	Y

PARTICIPANT 1		
Disturbed walking		
Point	Point	Cane?
0	12000	Y
Rx1	Node 1	
Point	Point	
0	1600	Y
5500	8200	Y
Rx2	Node 0	
Point	Point	
1600	2500	Y
4400	5500	Y
8200	9300	Y
10500	12000	Y
Rx3	Node 4	
Point	Point	
2500	4400	Y
9300	10500	Y

PARTICIPANT 14		
Disturbed walking		
Point	Point	Cane?
0	12000	Y
Rx1	Node 1	
Point	Point	
0	1700	Y
5300	7400	Y
11200	12000	Y
Rx2	Node 0	
Point	Point	
1700	2500	Y
4700	5300	Y
7400	8200	Y
10500	11200	Y
Rx3	Node 4	
Point	Point	
2500	4700	Y
8200	10500	Y

A.5.2 Normal walking data annotation.

PARTICIPANT 1		
Normal walking		
Begin time	End Time	Cane?
0	120	N
Rx1		
Begin time	End Time	
0	7	N
23	30	N
46	66	N
81	88	N
102	110	N
Rx2		
Begin time	End Time	
7	9	N
20	23	N
30	33	N
43	46	N
66	70	N
78	81	N
88	90	N
99	102	N
110	113	N
Rx3		
Begin time	End Time	
9	20	N
33	43	N
70	78	N
90	99	N
113	120	N

PARTICIPANT 14		
Normal walking		
Begin time	End Time	Cane?
0	120	N
Rx1		
Begin time	End Time	
0	2	N
19	30	N
48	57	N
73	80	N
84	94	N
109	119	N
Rx2		
Begin time	End Time	
2	5	N
16	19	N
30	36	N
46	48	N
57	60	N
70	73	N
80	84	N
94	97	N
106	109	N
119	120	N
Rx3		
Begin time	End Time	
5	16	N
36	46	N
60	70	N
97	106	N

PARTICIPANT 1		
Normal walking		
Point	Point	Cane?
0	12000	N
Rx1	Node 1	
Point	Point	
0	700	N
2300	3000	N
4600	6600	N
8100	8800	N
10200	11000	N
Rx2	Node 0	
Point	Point	
700	900	N
2000	2300	N
3000	3300	N
4300	4600	N
6600	7000	N
7800	8100	N
8800	9000	N
9900	10200	N
11000	11300	N
Rx3	Node 4	
Point	Point	
900	2000	N
3300	4300	N
7000	7800	N
9000	9900	N
11300	12000	N

PARTICIPANT 14		
Normal walking		
Point	Point	Cane?
0	12000	N
Rx1	Node 1	
Point	Point	
0	200	N
1900	3000	N
4800	5700	N
7300	8000	N
8400	9400	N
10900	11900	N
0	0	N
Rx2	Node 0	
Point	Point	
200	500	N
1600	1900	N
3000	3600	N
4600	4800	N
5700	6000	N
7000	7300	N
8000	8400	N
9400	9700	N
10600	10900	N
11900	12000	N
Rx3	Node 4	
Point	Point	
500	1600	N
3600	4600	N
6000	7000	N
9700	10600	N