Sleep Respiratory Rate Monitoring with Portable Wi-Fi

Name1 Surname[1,2◑], Name2 Surname[2◑], Name3 Surname[2,3¤], Name4 Surname[2], Name5 Surname[2‡], Name6 Surname[2‡], Name7 Surname[1,2,3*], with the Lorem Ipsum Consortium[¶]

1 Affiliation Dept/Program/Center, Institution Name, City, State, Country
2 Affiliation Dept/Program/Center, Institution Name, City, State, Country
3 Affiliation Dept/Program/Center, Institution Name, City, State, Country

* m6222040385@g.siit.tu.ac.th

Abstract

Introduction

Wi-Fi is one of the most common network mediums nowadays. Pervasively, it is used for establishing a wireless network to connect to the internet. But, there are still many more functions Wi-Fi is good at. Wi-Fi can also be applied in fields beside connecting to the internet according to its stability being upgraded continuously. Decent Wi-Fi connectivity can extract more data other than the data to be transmitted like concentration, speed, obstacle between the transmission. Those can be composed to be many useful applications like Localization, Activity Classification and etc.

In order to achieve the applications like mentioned, There are many works tried to extract deep features from Wi-Fi. But, they are mostly working with very specific tools and old Network Interface Card (NIC) connected to a laptop running Linux that is currently one of the ways allowing to obtain fine-grain Channel State Information (CSI), the descriptive data of the Wi-Fi propagating in that environment. Those limitations significantly decrease simplicity of implementation. It is hard for public demonstration and integration with many updated tools.

Actually, there are other existing ways for obtaining the CSI. One is from a ubiquitously used microprocessor, ESP32. which is still not much explored in Wi-Fi exploiting field. It is simple to implement and can be easily integrated with other tools in many platforms due to its massively produced external tools.

Human sleep monitoring is being observed for a whlie and recently applied on commercial products. Sleeping quality is significant to all persons and can be investigated to many symptoms e.g. Sleep Apnea, which is a very serious disease that is widely detected in the elderly. People are more concerned on observing their sleepness by using devices those are able to inform them their sleep information e.g. smartwatches, smart mattress, smart rings and etc. The main feature that helps to detect the Sleep Apnea is a respiration monitoring because those patient of the disease is unable to breath constantly at night. So, the respiration monitoring can show consistency of breathing at night through a “Respiratory Rate (RR)” which is in the unit of “Breath Per Minute (bpm)”. But, those devices can considerably affect sleepness since it needs a contact to users’ bodies.

So, this paper proposes a contactless sleep resipratory rate monitoring with Wi-Fi CSI from ESP32 where users can applied with easy-to-find microprocessor and monitor their respiration during sleep without contacting to their bodies. This allows sleep monitoring to be practical for other researchers and those who want to observe their health through sleepness.

Background

Wi-Fi

Wi-Fi is a well-known connectivity with no wire needed (wireless). It has been used as a medium for connecting to the internet for over 10 years. However, the Wi-Fi is the name covering IEEE 802.11 n/g/ac protocols. It delivers data through 2.4/5GHz frequency with multiple channels. The bandwidth in each channel is 22MHz. the data are to be transmitted parallelly with multiplexing technique named orthogonal frequency division multiplexing (OFDM). Each carrier may propagate to a receiver with encountering many obstacles. The effect of that situation is the Doppler Effect. So, Channel State Information (CSI) is represented as physical layer indicator that can be used to investigate how each channel propagate to the receiver or back to the transmitter.

If a sender sends data to a receiver through Wi-Fi, the data will be almostly not transmitted without any interference.

CSI data

As mentioned in Wi-Fi that data propagating to the receiver while touching surrounding environment, the CSI is a variation of the data. The CSI can be found at both sender and receiver since receiver may transmit data back. Let the sender use the modulation method of 16-quadrature amplitude modulation (16-QAM) which one carrier can carry 4 bits. When the sender needs to send a '1111', the modulation returns $x = 1 + 1i$. Then, transmit to the receiver. At the receiver, let the obtained data is $y = 0.8 + 0.9i$. So, the CSI can be computed by the variation $h = y/x = 0.2 + 3.4i$.

Human body is literally water which reflect radio wave like Wi-Fi. [6], [7] and [8] have proven that human body can affect the CSI.

ESP32

ESP32 is a very popular single-board computer (SBC). With its affordable price and many available additional tools, ESP32 is commonly used in Internet of Things field. Quantitative CSI can be obtained from Wi-Fi in ESP32 according to [26]. The number of available subcarriers in ESP32 is 64.

According to the detail about Wi-Fi mentioned in Wi-Fi, the Wi-Fi in ESP32 has some limitation. It supports only 2.4GHz frequency and can be set only one channel over a connection. The bandwidth of each channel is 22MHz. The CSI can be both obtain from Access point (AP) and Station (STA) as shown in Fig. 2. In this paper, we consider to mainly use CSI at the AP. The frequency of each channel is as 802.11 standard.

Human sleep respiratory rate monitoring

Human sleep respiratory rate monitoring is widely used and can help the investigation for identifying further incoming diseases as stated above. The common respiratory rate of human is various by age ranges and bodies (ref). But, it can be fluctuated during an abnormal situation. However respiratory rate during sleep of a certain person should be consistant (ref). Otherwise, the person can be in danger of many serious diseases e.g. Bradypnea, Sleep Apnea and etc.

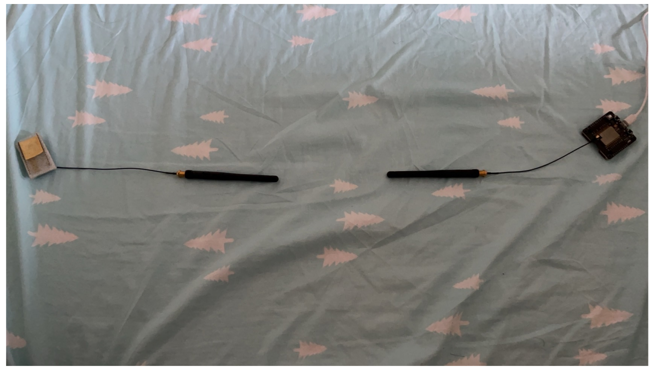

Fig 1. 2 ESP32s with external antenna. One act as a sender and another act as a receiver.

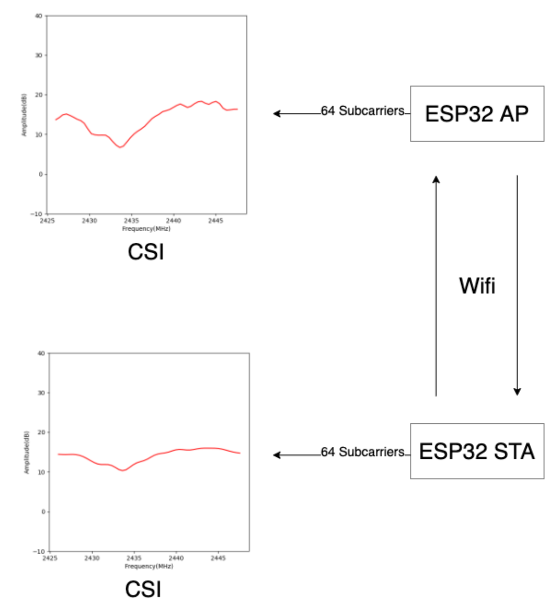

Fig 2. CSI from ESP32s with channel 6.

The simplest unit of respiratory rate is *bpm*, which tells how many time a certain person breaths in a minute. Many sensoring devices can log such information in time sequence. We use Vernier Respiration Monitor Belt, a device from the reliable sensoring manufacturer and commonly used in acadamic field, as our ground truth (ref). The belt is set to log RR at resolution of 4 Hz and The ESP32 is set to log CSI at resolution of 60 Hz. So, we herein create a mapping rule from a sequence of 60 CSI data to a sequence of 4 RR data for each second.

Fig 3. Vernier Respiration Monitor Belt as ground truth.

Materials and methods

Fig 4. Experimental setup.

Concept

Other famous proposed works for Activity Classification like [6] [7] and [25] use omnidirectional antennae and focus at the line of sight between the antennae. our work does likewise as shown in Fig. 4.

The CSI is not only affected by human body but also by overall environment. This means that 2 identical human poses can result obviously different CSIs if the environment around are not exactly the same as shown in Fig. 6. In short, CSI is suitable for moving target since we can focus on its change.

The example of mapping CSI to Activity Classification can be found in [8] and [27]. In the normal environment of sleeping, we assume that nothing in the room is moving besides an alive sleeping human on the bed. So, the corresponding changing of RR

Fig 5. Fitbit sleep tracker.

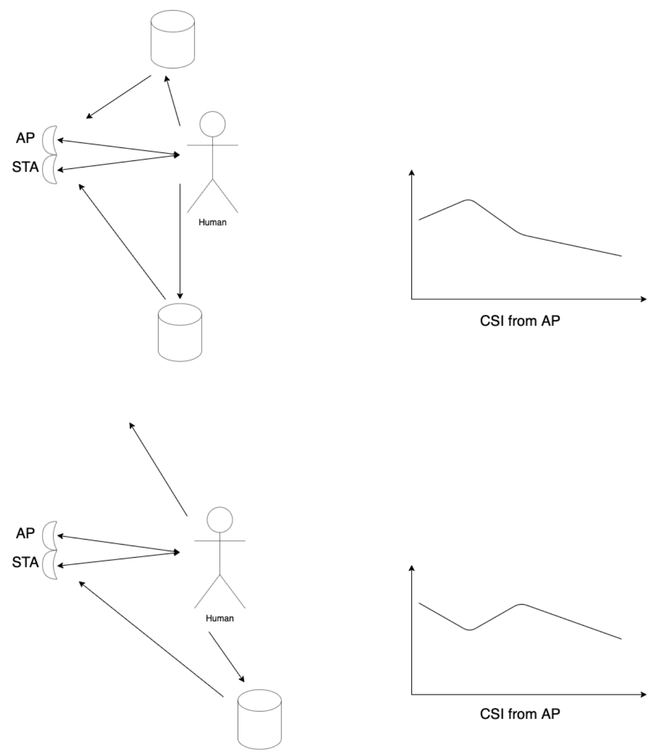

Fig 6. 2 different CSIs resulted from corresponding human poses.

during sleep may affect to the same changing pattern of the CSI. This hypothesis is investigated in the upcoming parts.

All steps of the mapping rule are shown in Fig. 7.

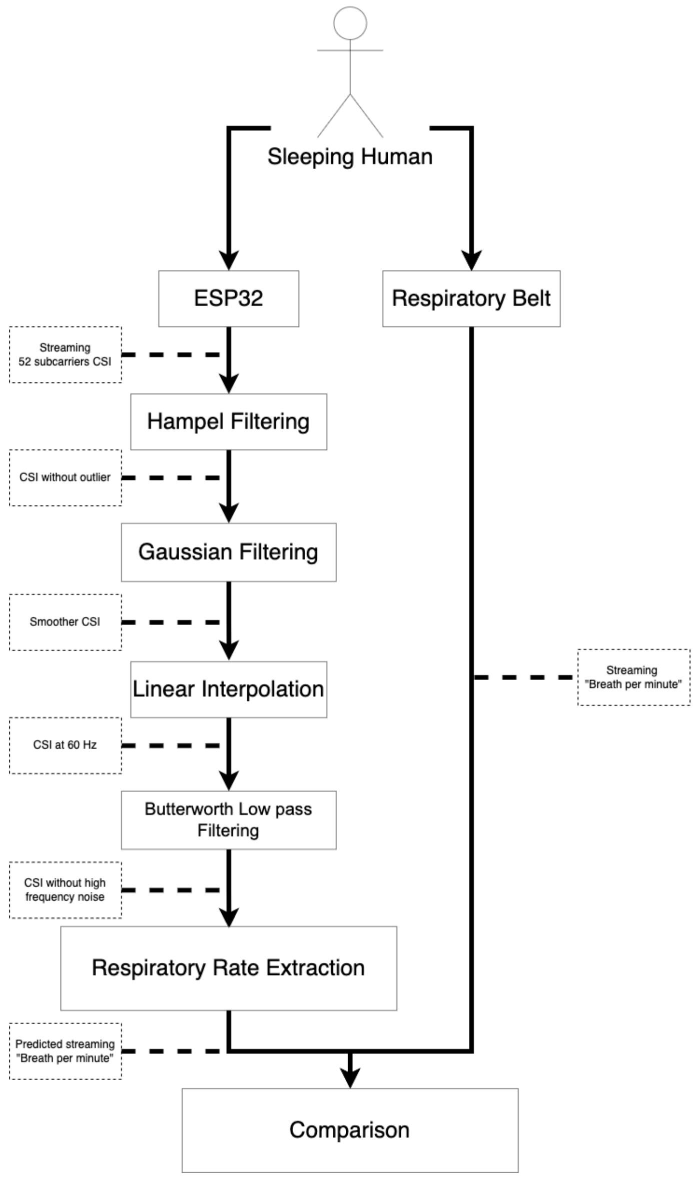

Fig 7. All the mapping rule.

Prediction (ESP32)

We applied (ref-steve-ESP32-CSI) to the ESP32 at receiver side to obtain real-time CSI.

However, original CSI data is complex number so, we can either convert it into phase or amplitude. As mentioned in CSI data, the CSI is originally in form $h = y/x = v + wi$ so, they are parsed. For amplitude, $\sqrt{v^2 + w^2}$ and $\arctan2(v, 2)$ for phase. In this paper, we use amplitude.

$$\text{Amplitude} = \sqrt{v_{sc}^2 + w_{sc}^2}, sc \in [1, 52] \tag{1}$$

There are 52 subcarrier in ESP32 (ref). We simply pick one.

Hampel Filtering

The CSI amplitude parsed from raw CSI of ESP32 is originally noiseful and contain many outliers, a value which is significantly far from beside data. The CSI sequence is ideally a continuous data, so outliers are considered as noises. Hampel Filtering is a filtering process to remove obvious outliers. An example of the result is shown in Fig. 8 where the top is an input that is having many outliers and the bottom is an output that is having lower outliers.

Parameters of the process are window size and σ which are redefined as α and β respectively in this paper.

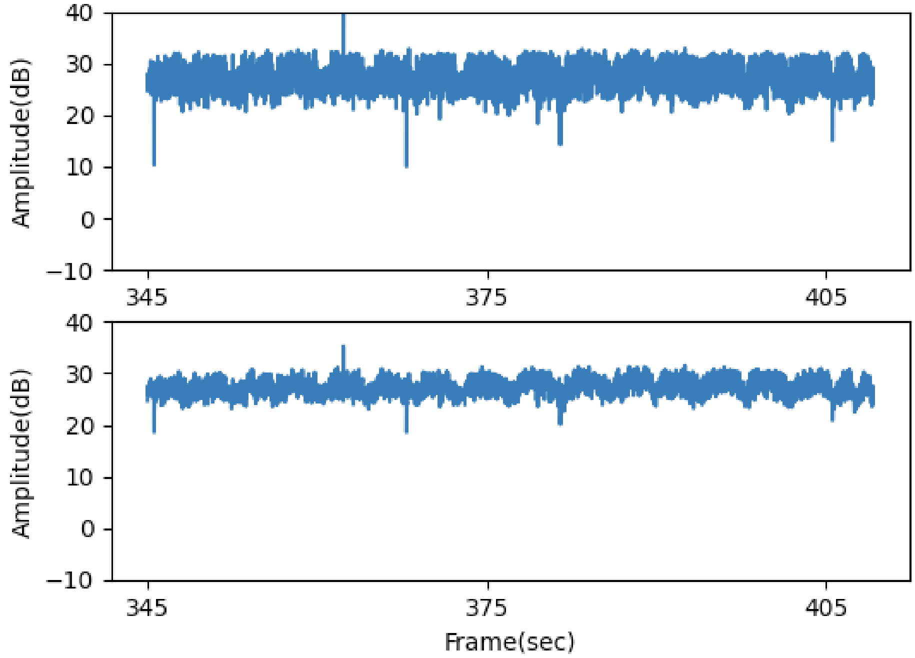

Fig 8. An example of applying Hampel filtering when $\alpha = 1$ and $\beta = -100$.

Gaussian Filtering

The CSI after applying Hampel filtering is more clear but, still lag smoothness ostensibly. To smoothen the CSI, we used Gaussian filtering as (ref Activity Class). An example of the result is shown in Fig. 9 where the top is an input that is very rough and the bottom is an output that is smoother.

A parameter of the process is σ which is redefined as γ in this paper.

Linear Interpolation

The sampling rate for CSI from ESP32 is originally unpredictable and not constant but it is running around 70Hz. So, we do a process called "Linear Interpolation" to the CSI data in order to maintain the dimension of each sequence. The CSI data can be resampled into any size so, we determined it as 60 Hz.

An example of CSI Linear Interpolation is shown in Fig. 10 where both look identical but the top is logged at $70 - 80$ Hz unstably while the bottom is at 60 Hz stably.

To recreate a CSI data at rate 60Hz, we calculate each with 2 data at the closet timestamps from the original with a simple mathematical weight equation as in Eq. 2 in order to recreate a CSI data with new frequency for all sequences.

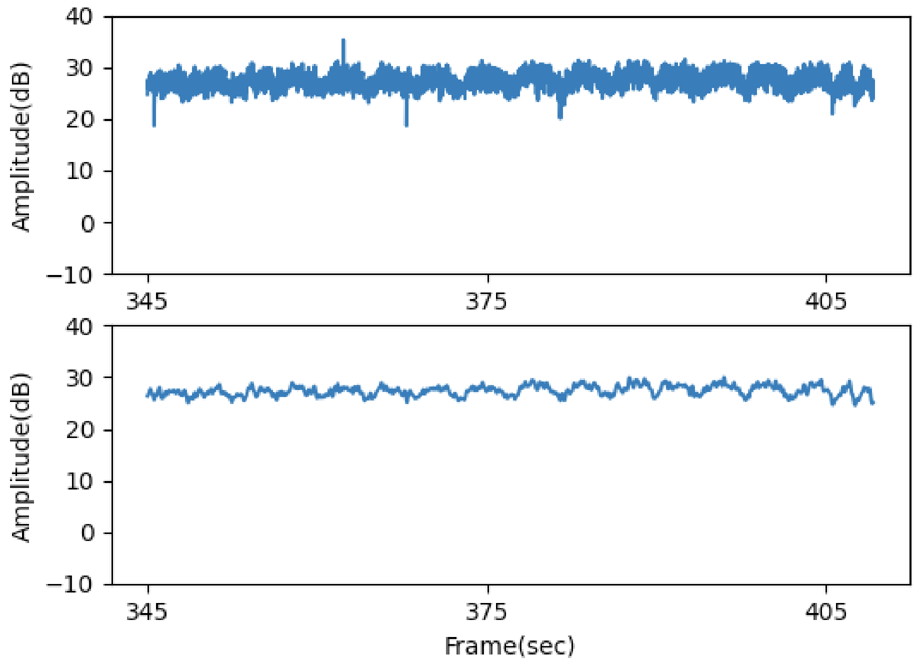

Fig 9. An example of applying Gaussian filtering when $\gamma = 5$.

$$CSI_{now} = CSI_{before} + \left(\frac{ts_{now} - ts_{before}}{ts_{after} - ts_{before}} \times (CSI_{after} - CSI_{before}) \right), \quad (2)$$

where ts_{now}, ts_{before}, ts_{after} are desired timestamp, timestamp at the closest CSI before the desired timestamp and timestamp at the closest CSI after the desired timestamp respectively. And, CSI_{now}, CSI_{before}, CSI_{after} are CSI at the desired timestamp, CSI before the desired timestamp and CSI after the desired timestamp respectively.

Butterworth Low Pass Filtering

After all those filtering, the CSI can implicitly show a wave at respiration frequency. Meanwhile, there is also many high frequency wave attached. Since we assume that nothing is moving in sleeping environment other than a sleeping person taking breath, the wave with frequency higher than normal person respiratory rate is discardable.

So, we applied Butterworth low pass filtering that can remove the wave with frequency higher than a certain threshold. By using the frequency slightly higher than common human respiratory rate, we can obtain only the wave of a human breathing as shown in Fig. 11 where the top is an input that is containing high frequency wave and the bottom is an output without frequency wave higher than common human respiratory rate.

Parameters of the process are the order of Butterworth filtering and cutoff frequency which are redefined as δ and ϵ respectively in this paper.

Respiratory Rate Extraction

The philosophy we used to extract RR from both data of CSI from ESP32 and pressure from the ground truth is "First to hit the mean" as (ref) have proven its reliability.

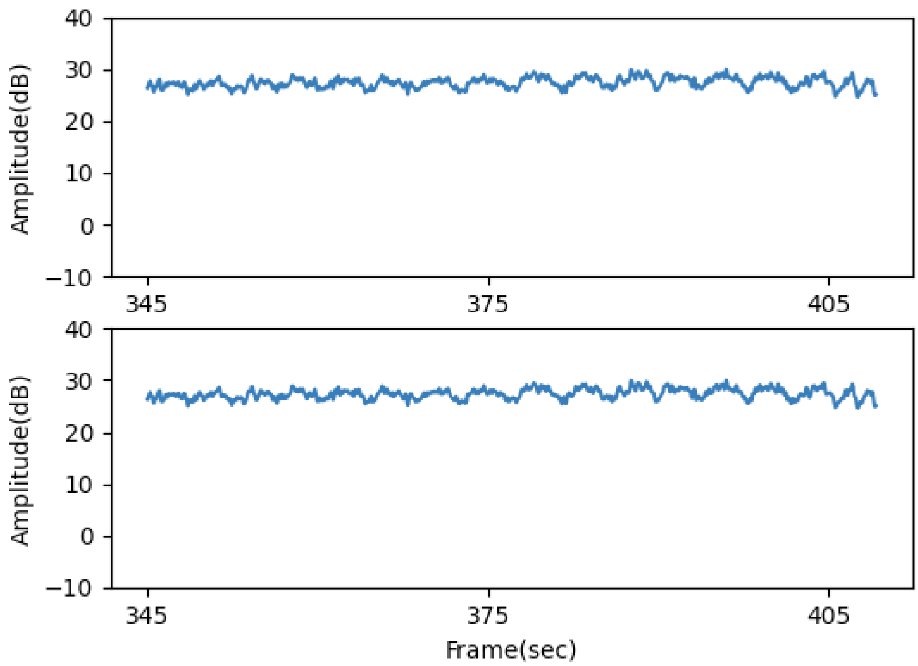

Fig 10. An example of applying CSI linear interpolation.

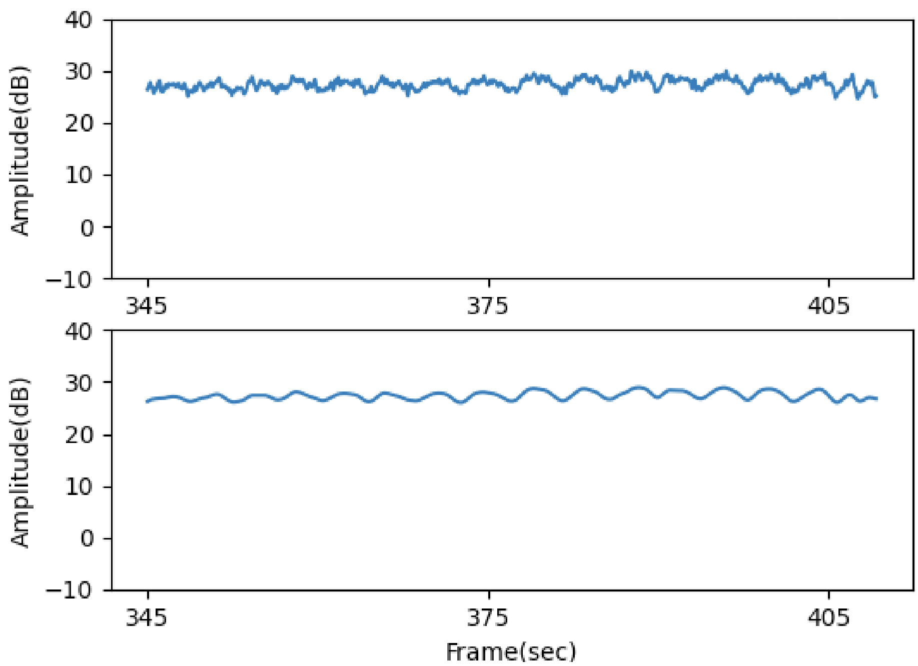

Fig 11. An example of applying Butterworth low pass filtering when $\delta = 2$ and $\epsilon = 30$.

The examples of applying the technique with different breathing pattern are shown in Fig. 12, Fig. 13, Fig. 14 and Fig. 15 where each orange X are counted as a breath.

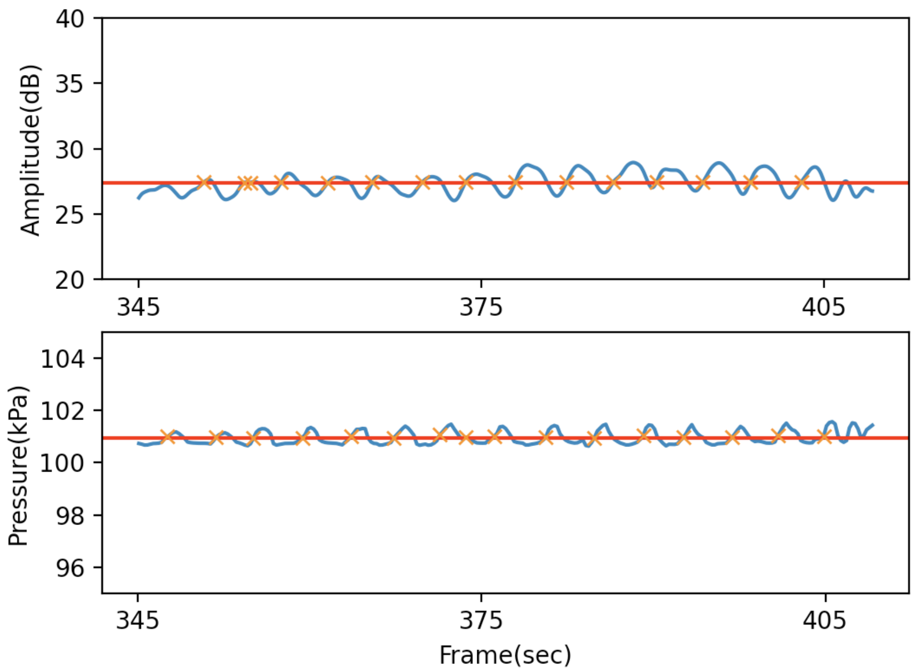

Fig 12. An example of Respiratory Rate Extraction from prediction (top) and ground truth (bottom) for normal-rate breathing (prediction BPM = 15, ground truth BPM = 16).

As you can see in Fig. 14, the straight behavior of the algorithm recognized the low pressure fluctuation as a breath which is incorrect. So, we need a further correction to distinguish if the state of the graph is a breath holding.

Breath missing Detection

A parameter to check if the Amplitude is too low to be a breath.

Ground Truth (Respiratory Belt)

The Respiratory Belt logs pressure data with a unit of kilopascal. As mentioned in Human sleep respiratory rate monitoring, The belt is set to log RR at resolution of 4 Hz. By connecting the belt to a microprocessor as shown in Fig. 3, we can set log resolution at any rate. However, based on our experiment, normal human RR can be determined at resolution of 4 Hz as shown in Fig. 16 (top).

As it can be simply seen that the raw data from the belt is clear but not precise, so we still need more filtering on it. 3 filterings, as shown in Fig. 7 (right), are applied. For more precision, we applied more Linear Interpolation to ensure the resolution is exactly 4 Hz, The result are shown in Fig. 16

Parameters of the Hampel Filtering process is window size and σ which is redefined as ζ and η in this paper. A parameter of the Gaussian Filtering process is σ which is redefined as θ in this paper.

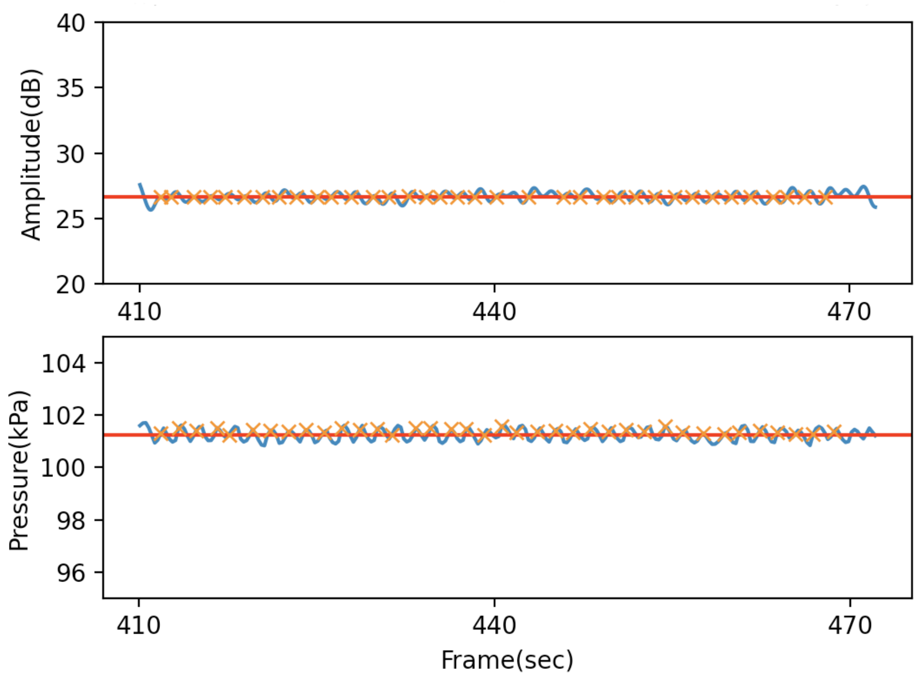

Fig 13. An example of Respiratory Rate Extraction from prediction (top) and ground truth (bottom) for fast-rate breathing (prediction BPM = 36, ground truth BPM = 38).

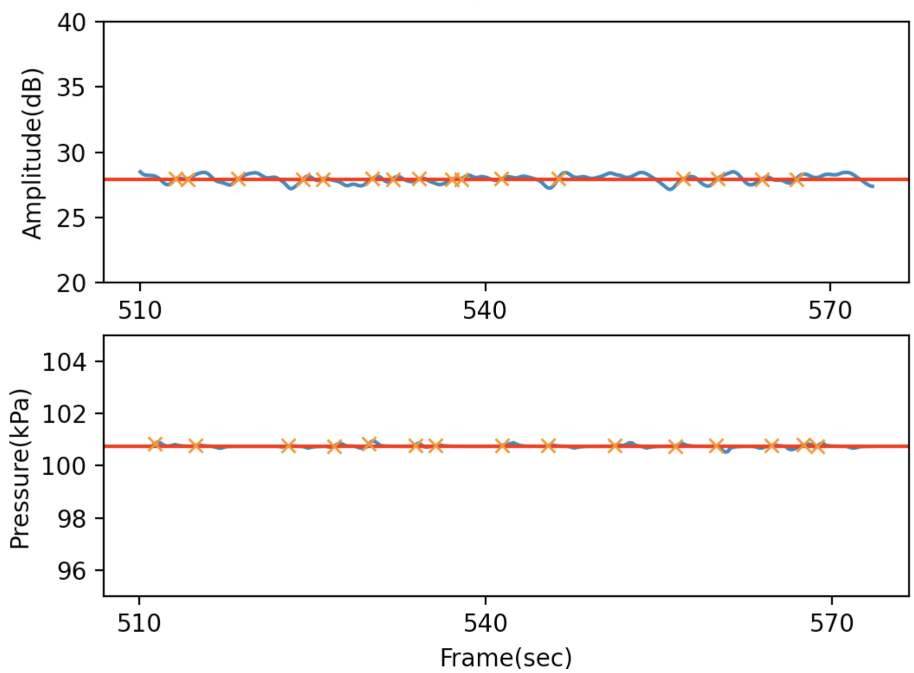

Fig 14. An example of Respiratory Rate Extraction from prediction (top) and ground truth (bottom) for holding-rate breathing (prediction BPM = 16, ground truth BPM = 15).

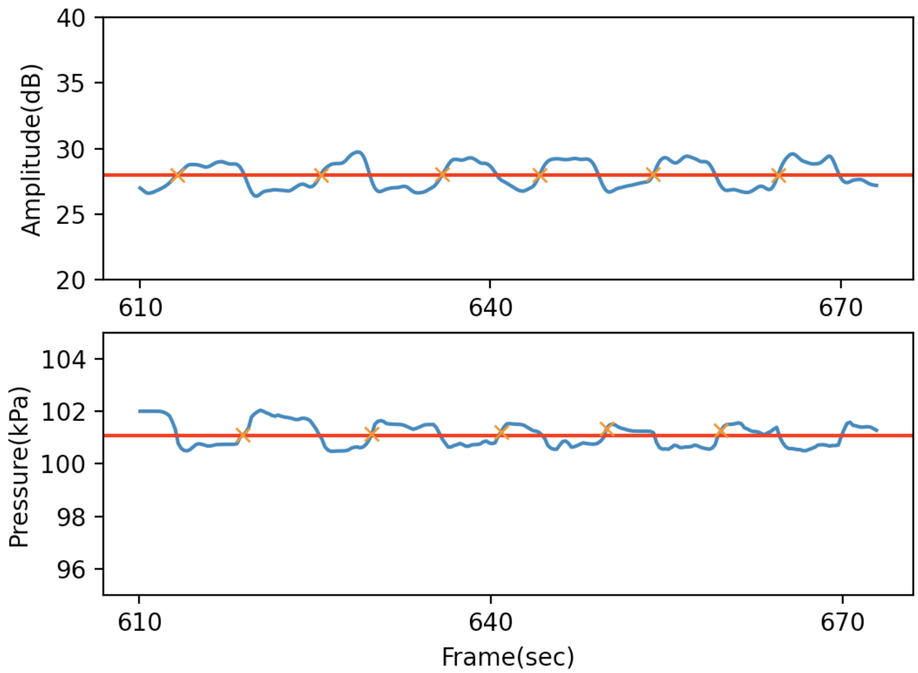

Fig 15. An example of Respiratory Rate Extraction from prediction (top) and ground truth (bottom) for slow-rate breathing (prediction BPM = 6, ground truth BPM = 5).

Results

Data Collection

We recruited 10 volunteers to sleep in between the devices (2 WiFi anntennae) while wearing a Vernier belt.

The whole data collection is 50 hours of sleep which worth 3,000 data for 60 seconds/RR rate.

Experimental Result

An evaluation is achieved by the algorithm written in Python 3.8. The code is available on Github[1].

For error metrics, Mean Absolute Difference (MAD) is a decent indicator to tell how much RR from the ESP32 is different to RR from the belt. With windows of 60 s, MAD is computed by the following equation.

$$MAD = \frac{\sum_{i=1}^{n} |N_i^{gt} - N_i^{pd}|}{n}, \tag{3}$$

where n is a total number of dataset, N_i^{gt} is RR obtained from the belt at index i and N_i^{pd} is RR obtained from the ESP32 at index i.

Table 1, Table **??**, Table **??** and Table **??** show the MAD of 4 breathing pattern with different parameter fine-tunings.

[1] https://github.com/rtmtree/CSPS

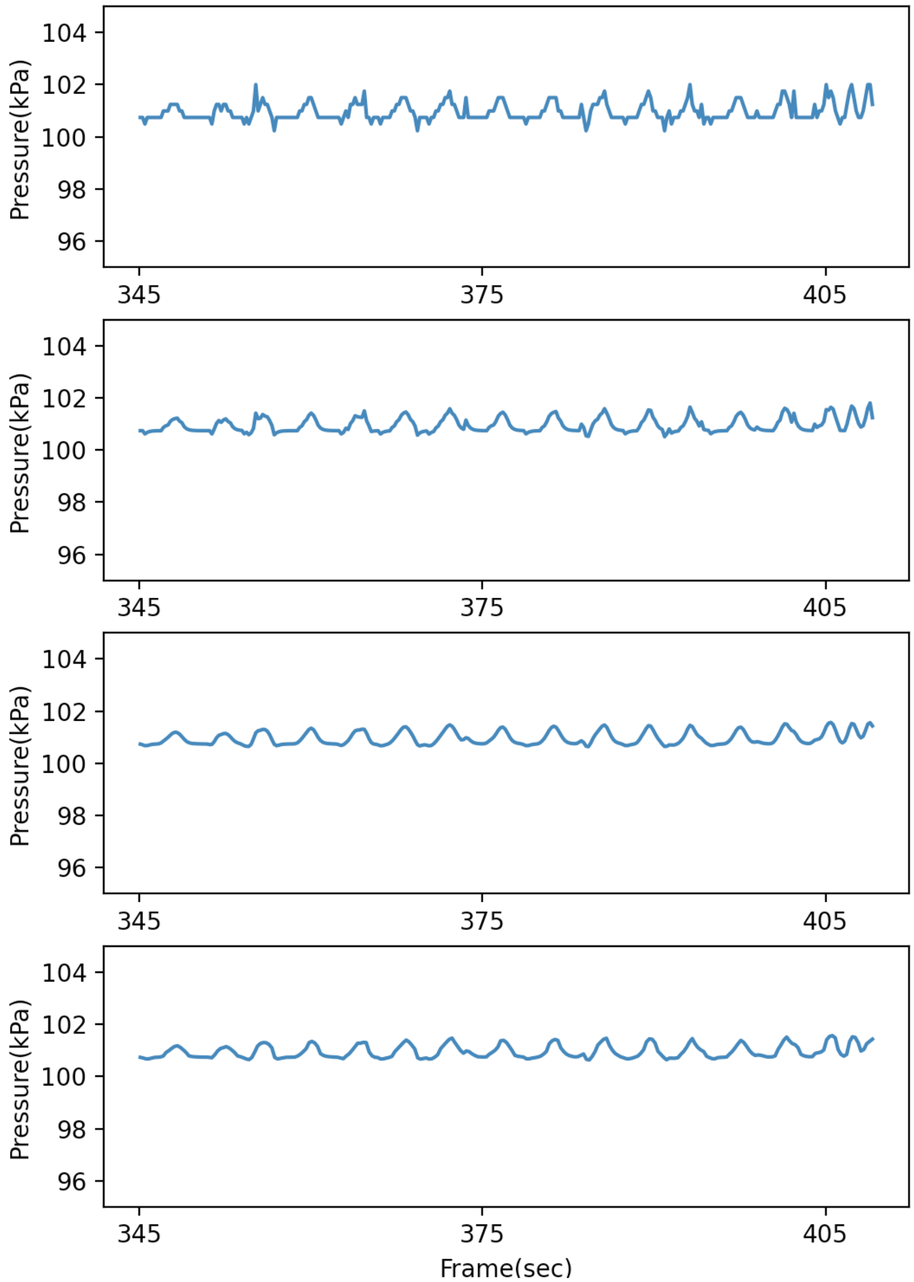

Fig 16. An example of pressure logged from Vernier Respiratory Belt for normal human taking breath. Without filtering (top), with Hampel Filtering (2nd top) when $\zeta = 1$ and $\eta = -100$, with Gaussian Filtering (3rd top) when $\theta = 1$ and with Linear Interpolation (bottom) with 4 Hz.

Table 1. Table of evaluation result when $\alpha = 1$.

Breathing \ Result	MAD
Normal	2.0
Fast	6.03
Hold	1.97
Slow	0.39

Discussion

Environment Installation

The Installation of WiFi antennae can affect tremendously in data. We fixed the positions in both training and testing process as shown in Fig. 4. 2 antennae are connecting to ESP32s and to the PC afterward for the AP antennae for CSI collection. Lastly, the human sleeping on the bed need to wear Fitbit sleep tracker as shown in Fig. 5.

CSI Extraction Method

There are many solutions for extracting WiFi CSI from the ESP32 as mentioned in ESP32. This paper picked the solution from ESP32-CSI-Tool[1] since it is considerably well-written and simple to organize. To change the method of extracting WiFi CSI would affect the result significantly.

Multiple Person Estimation

The model does a mapping rule from WiFi CSI with a specific dimension to human sleep stage matrix. So, it is able to only detect single person pose in a range. The annotations originally can result all human sleep stage by using the devices for all specimen. If we do training with multiple sleep stage data instead, we do not believe the result will be well since CSI of a person moving is rarely separatable form others.

Conclusion

Currently, there are many concern on sleepness in all group of people. Especially, those who works in extraordinary time period are facing problem of having deficient sleepness. They are pleased to pay for a device that can inform if their sleepness is well enough e.g. smartwatch, smart ring and smart mattress. Those options come with additional and uncomfortable since users' bodies need a contact to an external device. So, we aim for proposing a contactless Sleep Monitoring device.

After having a research, we discoverd that a variation in WiFi called WiFi CSI can tell whether the area is having an activity or moving objects. Moreoever, [8], [27] and many other works had been done very well on detecting even what kind of activities is happening in the area. We do not believe that this is the limitation of WiFi CSI. In order to solve the above problem and prove if WiFi CSI is precise enough, we tried to overcome this by extracting a deeper information like Sleep Stage Estimation from it.

We controlled environments and enhanced WiFi antennae stability as much as possible then mapped it to the Sleep Stage annotated by Fitbit technology. The result was very poor since the WiFi CSI is very vague. It penetrates through most of the

[1] https://github.com/StevenMHernandez/ESP32-CSI-Tool

things. We learned from [20] that WiFi CSI value does not matter than its change. The works used Long-short term memory (LSTM), a neural network where focusing on sequence of the data, and obtained a very good result.

We applied the idea of LSTM instead and obtained a lot better result with more than 80% of overall accuracy. Then, we adjusted the model to be suitable for our type of data and did fine-tuning for the frame size to be fit the most for normal human speed of moving while sleeping. So, This paper proposed a model of mapping rule that can takes a sequence of 30-second WiFi circumstance as an input and return an according human sleep stage as an output. The work can help people to monitor Sleep Stage of elderlies and those who are in need of sleep concerning. Moreover as a contact is not needed, users can not differentiate thier sleeping with or without the device. Significantly, the WiFi CSI obtained from ESP32 which is affordable and widely reachable so, people can simply apply the method on their own. The result of various fine-tuned environments and parammeters is acceptable and shown in Experimental Result.

Acknowledgments

We thank to Sirindhorn International Institute of Technology for providing technical environment and supportive information.

References

1. Conant GC, Wolfe KH. Turning a hobby into a job: how duplicated genes find new functions. Nat Rev Genet. 2008 Dec;9(12):938–950.

2. Ohno S. Evolution by gene duplication. London: George Alien & Unwin Ltd. Berlin, Heidelberg and New York: Springer-Verlag.; 1970.

3. Magwire MM, Bayer F, Webster CL, Cao C, Jiggins FM. Successive increases in the resistance of Drosophila to viral infection through a transposon insertion followed by a Duplication. PLoS Genet. 2011 Oct;7(10):e1002337.

4. Osokin D. Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose. arXiv:1811.12004 [cs.CV]. 2018 Nov;18.

5. Mehta D, Sotnychenko O, Mueller F, Xu W, Sridhar S, Pons-Moll G, Theobalt C. Single-Shot Multi-Person 3D Pose Estimation From Monocular RGB. arXiv:1712.03453 [cs.CV]. 2018 Aug;28.

6. Wang F, Panev S, Ziyi D, Han J, Huang D. Can WiFi Estimate Person Pose?. arXiv:1904.00277 [cs.CV]. 2019 Apr;2.

7. Liu J, Liu H, Chen Y, Wang Y, Wang C. Wireless Sensing for Human Activity: A Survey. IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 22, NO. 3, THIRD QUARTER 2020.

8. Chowdhury TZ, Leung C, Miao CY. WiHACS: Leveraging WiFi for Human Activity Classification using OFDM Subcarriers' correlation. IEEE, GlobalSIP 2017.

9. Guo L, Wang L, Liu J, Zhou W, Lu B. HuAc: Human Activity Recognition Using Crowdsourced WiFi Signals and Skeleton Data. Wireless Communications and Mobile Computing Volume 2018, Article ID 6163475.

10. Wang F, Feng J, Zhao Y, Xiaobin Zhang, Zhang S. Joint Activity Recognition and Indoor Localization with WiFi Fingerprints. arXiv:1904.04964 [cs.CV]. 2019 Jul;18.

11. Al-qaness MAA, Li F, Ma X, Zhang Y, Liu G. Device-Free Indoor Activity Recognition System. Appl. Sci. 2016, 6, 329; doi:10.3390.

12. Wang W, Liu AX, Shahzad M, Ling K, Lu S. Device-free Human Activity Recognition Using Commercial WiFi Devices. IEEE Journal on Selected Areas in Communications. DOI 10.1109/JSAC.2017.2679658.

13. Zhao T, Li F, Tian P. A Deep-Learning Method for Device Activity Detection in mMTC Under Imperfect CSI Based on Variational-Autoencoder. IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 69, NO. 7, JULY 2020.

14. Liu J, Teng G, Hong F. Human Activity Sensing with Wireless Signals: A Survey. Sensors 2020, 20, 1210; doi:10.3390/s20041210.

15. Yousefi S, Narui H, Dayal S, Ermon S, Valaee S. A Survey on Behaviour Recognition Using WiFi Channel State Information. arXiv:1708.07129 [cs.AI]. 2017 Aug;23.

16. Chen Z, Zhang L, Jiang C, Cao Z, Cui W. WiFi CSI Based Passive Human Activity Recognition Using Attention Based BLSTM. IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 18, NO. 11, 2019 Nov.

17. Li B, Cui W, Wang W, Zhang L, Chen Z, Wu M. Two-Stream Convolution Augmented Transformer for Human Activity Recognition. Association for the Advancement of Artificial Intelligence, 2021.

18. Luo Y, Ren J, Wang Z, Sun W, Pan J, Liu J, Pan J, Lin L. LSTM Pose Machines. arXiv:1712.06316 [cs.CV].

19. Lee K, Lee I, Lee S. Propagating LSTM: 3D Pose Estimation based on Joint Interdependency. Computer Vision – ECCV 2018. ECCV 2018. Lecture Notes in Computer Science, vol 11211. Springer, Cham.

20. Du X, Vasudevan R, Johnson-Roberson M. Bio-LSTM: A Biomechanically Inspired Recurrent Neural Network for 3D Pedestrian Pose and Gait Prediction. arXiv:1809.03705 [cs.RO]. 2019 Sep;13.

21. Hossain MRI, Little JJ. Exploiting temporal information for 3D human pose estimation. arXiv:1711.08585 [cs.CV]. 2018 Sep;12.

22. Pavllo D, Feichtenhofer C, Grangier D, Auli M. 3D human pose estimation in video with temporal convolutions and semi-supervised training. arXiv:1811.11742 [cs.CV]. 2019 Mar;29.

23. Chen T, Fang C, Shen X, Zhu Y, Chen Z, Luo J. Anatomy-aware 3D Human Pose Estimation with Bone-based Pose Decomposition. arXiv:2002.10322 [cs.CV]. 2021 Jan;26.

24. Ruiz AH, Porzi L, Bulo SR, Moreno-Noguer F. 3D CNNs on Distance Matrices for Human Action Recognition. MM ’17, Mountain View, CA, USA. 2017 Oct;23–27.

25. Hernandez SM, Bulut E. Lightweight and Standalone IoT based WiFi Sensing for Active Repositioning and Mobility. IEEE 21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), 2020.

26. Atif M, Muralidharan S, Ko H, Yoo B. Wi-ESP—A tool for CSI-based Device-Free Wi-Fi Sensing (DFWS). Journal of Computational Design and Engineering, 2020, 7(5), 644–656.

27. Zou H, Zhou Y, Yang J, Jiang H, Xie L, Spanos CJ. DeepSense: Device-free Human Activity Recognition via Autoencoder Long-term Recurrent Convolutional Network. 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 2018, pp. 1-6, doi: 10.1109/ICC.2018.8422895.

28. Hochreiter S, Schmidhuber J. LONG SHORT-TERM MEMORY. Neural Computation 9(8):1735-1780, 1997.