Moving 3D Pose Estimation with ESP32 Wi-Fi

1st Ratthamontree Burimas
School of Information, Computer and Communication Technology
Sirindhorn International Institute of Technology
Pathum Thani, Thailand
bank23232525@gmail.com

2nd Teerayut Horanont
School of Information, Computer and Communication Technology
Sirindhorn International Institute of Technology
Pathum Thani, Thailand
teerayut@siit.tu.ac.th

***Abstract*—**
***Index Terms*—Channel State Information**

I. Introduction

Wifi is a common medium in many kinds of field nowadays. Normally, It is used for establishing a wireless network to connect to the internet. But there are still many more functions Wifi is good at. Wifi can also be applied in fields beside connecting to the internet according to its stability being upgraded continuously. Decent Wifi 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 data on their own like localization, activity prediction and etc.

Camera is a very good tool for monitoring things and is being used as a very effective data collector for mapping to the ground truth to create many popular machine-learning-based usable model like pose estimation, text segmentation, object detection and many more. However, camera are unavoidably judged as a serious privacy infringement since the data obtained like a photo or video are too clear and possess too much information that might be used in a bad way.

There are many works tried to extract those extractable features like camera does from Wifi. But, they are mostly working with very specific tools and Network Interface Card (NIC) connected to a labtop running Linux that is currently one of the ways allowing to obtain fine-grain Channel State Information (CSI), the descriptive data of the Wifi propagating in that environment. Those limitation significantly decrease steamline of implementation. It is hard for public demonstration and intregation with many updated tools in operating system like Windows or OSX.

Actually, there are other existing ways for obtaining the CSI. One is from a ubiquitously used microprcessor ESP32. which is still not much be explored in exploiting Wifi field. It is simple to implement and can be easily integrated with others tools in many platforms due to its massively produced external tools. This paper proposes a machine-learning-based model to create a mapping rule from Wifi CSI to 3D moving human pose estimation by using ESP32.

II. Background

A. Pose Estimation

There many machine-learning-based human body pose detection tools had been proposed and available online. Those can de found both 2D and 3D. In our paper, we chose a light weight 3D pose detection as an annotation due to its simplicity and the hypothesis that 3D should suit the most in our work. The project can be found at [github-lw3d] and is based on [paper-Lightweight OpenPose] and [paper- Single-Shot Multi-Person 3D Pose Estimation From Monocular RGB]. Its job is to simply create 3D human pose annotation from an image. Then, feed to our works training process as an annoation.

B. Wifi

Wifi 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 Wifi is the name covering IEEE 802.11 n/g/ac protocols. It mostly deliver data through 2.4/5GHz frequency with multiple channels. The bandwidth in each channel is 22MHz. the data are to be transmitted pararelly with multiplexing technique named orthogonal frequency division multiplexing (OFDM). Each carrier may propagate to a reciever 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 reciever or back to the transmitter.

If a sender sends data to a reciever through Wifi, the data will be surely not transmitted without any loss.

C. CSI data

As mentioned in II-B that data propagating to the reciever while touching surrounding environment, the CSI is a variation of the data. The CSI can be found at both sender and reciever since reciever may transmit data back.In this paper, We consider to mainly use CSI at the transmiter. 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 reciever. At the reciever, 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 Wifi. [] and [] have proven that human body can affect the CSI.

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

According to the detail about Wifi mentioned in II-B, the Wifi 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 which each frequency in the band is represented in 64 subcarriers The CSI can be both obtain from Access point (AP) and station (STA) as shown in Fig. 1. The frequency of each channel is as 802.11 standard.

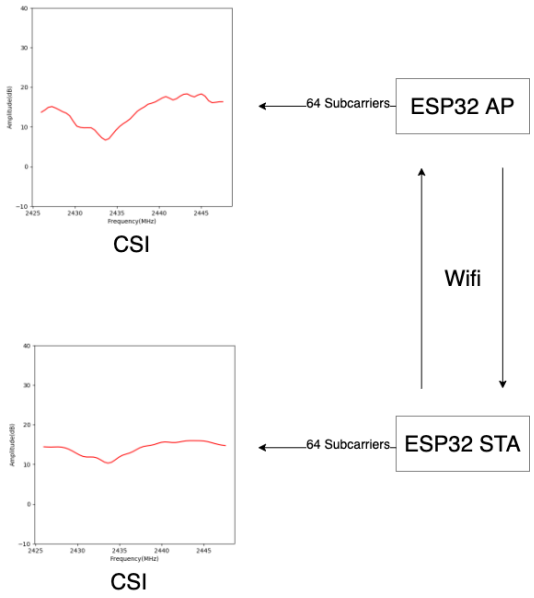

Fig. 1. CSI from ESP32s with channel 6.

III. Proposed Method

A. Concept

Other famous proposed works like [] [] and [] focus on line-of-sight (LOS) in between AP and STA while our work uses 2 directional Wifi antennae and focus on reflection from human body as shown in Fig. 2 on the left.

The reason we name "Moving Pose Detection" instead of "Pose Detection" is that CSI is not only affected by human body but mainly by overall environment. This means that 2 corresponding human poses can result obviously different CSIs if the environment around are not exactly the same as shown in Fig. 2.

So, the detection of human standing still in every environment is nearly impossible since the CSI of that situation may be found exactly matched to a CSI of the environment that a big bottle of water placed in front of ESP32. In short, if it does not move, we do not if it is human.

Meanwhile, the moving pose is totally different because we focus on its change instead. In different environment , the CSIs are different. But, the corresponding moving pose may affect to the same changing pattern of CSI. This hypothesis is investigated in the upcoming parts.

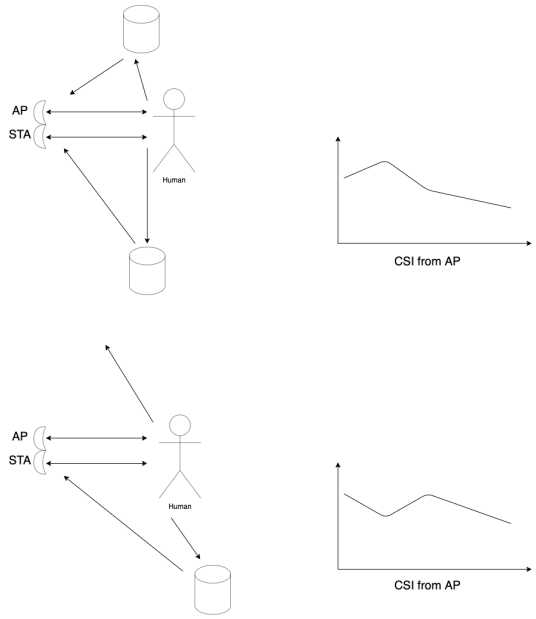

Fig. 2. 2 different CSIs resulted from corresponding human poses.

B. Pre-processing

A summation of all steps in our method is shown in Fig. 3.

1) CSI Preparation: As mentioned in II-C1, There are 64 subcarriers in CSI data from ESP32 but there are only 52 those are usable the rest are null. So, we can construst a tensor of 1×52 to represent each CSI. We are to map 3D human pose annotation from a camera to CSI from the ESP32. The sampling rate of the camera are set to 30Hz. So, we have 30 human pose annotations for one second. For the ESP32, the sampling rate is unpredictable and not constant but it is running around 120Hz. So we do a process called "Resampling" to obtain CSI at rate 30Hz in order to map to each human pose annotation.

An example of CSI Resampling is shown in Fig. 4. The top graph shows that the the original CSI is logged unstably. The bottom one is to pick a timestamp at rate 30Hz and calculated each with the closet data from the original with a simple mathamatical weight equation as in Eq. 1 in order to predict CSI at timestamp corresponding to each human pose annotation.

$$CSI_{now} = CSI_{before} + \left(\frac{ts_{now} - ts_{before}}{ts_{after} - ts_{before}} \times (CSI_{after} - CSI_{before}) \right) \quad (1)$$

, where ....

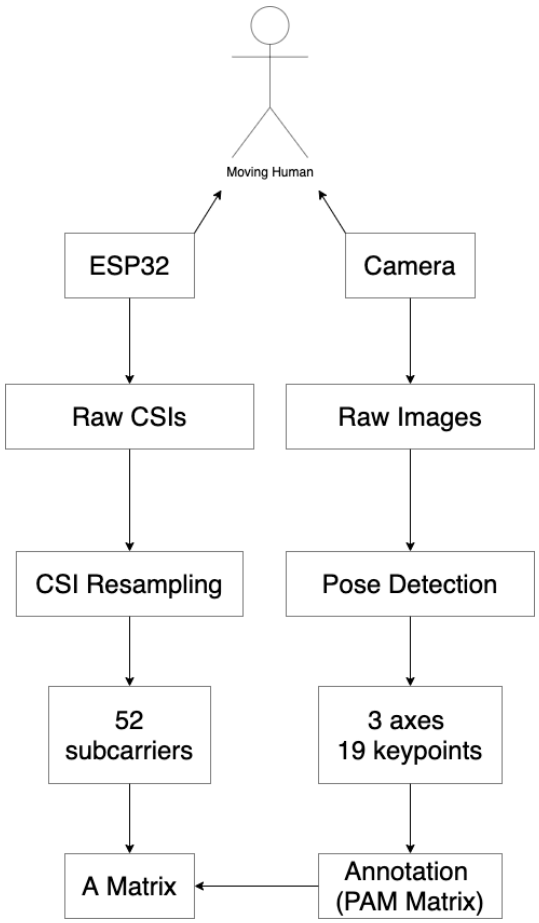

Fig. 3. A summation of all steps.

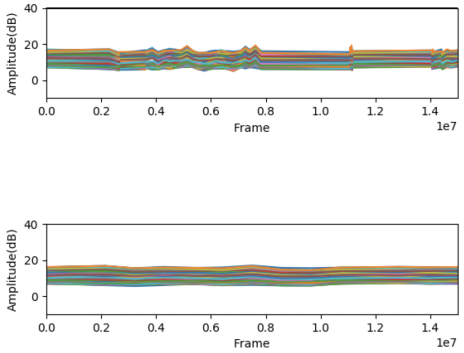

Fig. 4. An example of CSI resampling.

Now we can map one CSI samples to one 3D human pose annotations from one images with the corresponding timestamp.

2) Human-Pose Preparation: A matrix of 19×3 are used as an annotation where 19 is for keypoints in human body and 3 is for 3 axes coordination as shown in Fig. **??** But, the annotation can still possess too much independency. Some alignment of keypoints may lead to some impossible pose e.g. body keypoint found very far from neck keypoint or right shoulder keypoint attached to left knee keypoint. We assume those poses are quite not possible for normal human pose. To preserve those constraint, we form pose adjacent matrix (PAM) from an original 19×3 matrix. the PAM is applied for all x, y and z axes. Each are to be form their 19×19 matrix by the following equations.

$$x'_{i,j} = \begin{cases} x_i - x_j & i \neq j \\ x_i & i = j \end{cases} \tag{2}$$

,

$$y'_{i,j} = \begin{cases} y_i - y_j & i \neq j \\ y_i & i = j \end{cases} \tag{3}$$

and

$$z'_{i,j} = \begin{cases} z_i - x_j & i \neq j \\ z_i & i = j \end{cases} \tag{4}$$

.

The PAM is finally a $3 \times 19 \times 19$ matrix created from 3 matrices of x',y' and z' stacked. Apparently, one PAM represent one human pose.

Conclusively, we are making a model by mapping a tensor of 1×52 from CSI to each tensor of $19 \times 3 \times 3$ PAM from moving human pose annotation with the corresponding timestamp as shown in Fig. 5.

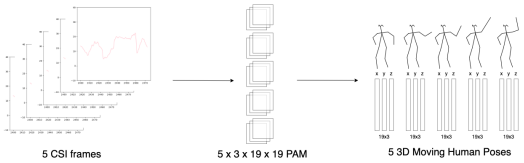

Fig. 5. A mapping rule from 5 CSI frames to 5 human poses.

C. Processing

1) Data Alignment: Let D be a set synchronized images and CSI data package. Each pair has corresponding timestamp.

$$D = (I_t, C_t), t \in [1, n] \tag{5}$$

, where n is a number of pairs, I_t is for PAM annotation as the ground truth, C_t is for CSI data from ESP32, t is the timestamp when those 2 were collected and n is the number of data.

σ frames and CSIs are fed to the model concurrently.

2) Form Network Layer: feed each π of D

LSTM

IV. Evaluation

A. Data Collection

dance in ma room ((We collected data under an approval of Carnegie Mellon University IRB 4 . We recruited 8 volunteers, and asked them to do casual daily actions in two rooms of the campus, one laboratory room and one class room. Floor plans and data collection positions are illustrated in Fig. 8. During the actions, we run the system in Fig. 3 to record CSI samples and videos, simultaneously. For each volunteer, data of his first 80test the networks. The data size of training and testing are 79496 and 19931, respectively))

B. Experimental Result

Percentage of Correct Keypoint (PCK) is widely used to evaluate the performance of human pose estimation according to [].

$$PCK_i@a = \frac{1}{N}\sum_{i=1}^{N} I(\frac{||\ pd_i\ -\ gt_i\ ||_2^2}{\sqrt{rh^2+lh^2}} \leq a), \tag{6}$$

where $I(\cdot)$ is a binary indicator that outputs 1 while true and 0 while false, N is the number of frames, i is the index of keypoints that $i \in 1, 2, ..., 19$. The rh and lh are for the positions of the right shoulder and the left hip from the ground truth, respectively. So, the $\sqrt{rh^2+lh^2}$ is considered as the length of the upper limb from the ground truth, which is used to normalize the prediction error length $||\ pd_i\ -\ gt_i\ ||_2^2$, and pd_i and gt_i are coordinates of prediction and ground-truth at the keypoint i repectively.

[Table. I] shows the estimation performance of 19 body keypoint in PCK@5, PCK@10, PCK@20, PCK@30, PCK@40, and PCK@50 of the frame length 15, isActSDthreshold is 100 and minCSIthreshold is 40.

Github[1].

Conclusion and Future Development

This paper proposes 2 algorithms for 2 simplification types, Fixed-# and Fixed-LSSED. Classically, to achievement the best satisfying result for both types, the time complexity of $O(2^{N-2})$ is needed, where N is a number of trajectory point. By exploiting quantum mechanics from the QSMA, we can do the same job by consuming only the time complexity of $O(\frac{\pi}{2}\sqrt{\frac{2^{N-2}}{M}} \times c)$, where M is a number of the finest solution and c is an adjustable parameter.

Acknowledgment

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

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

References

[1] Y. Chen, S. Wei, X. Gao, C. Wang, J. Wu and H. Guo, "An Optimized Quantum Maximum or Minimum Searching Algorithm and its Circuits," arXiv:1908.07943 [quant-ph] 21 Aug 2019.

[2] G. L. Long, "Grover algorithm with zero theoretical failure rate," arXiv:quant-ph/0106071, 13 Jun 2001.

[3] A. Ahuja and S. Kapoor, "A Quantum Algorithm for finding the Maximum," arXiv:quant-ph/9911082, 18 Nov 1999.

[4] C. Dürr and P. Høyer, "A quantum algorithm for finding the minimum," arXiv:quant-ph/9911082, 18 Nov 1999.

[5] M. Chen, M. Xu and P. Fränti, "A Fast Multiresolution Polygonal Approximation Algorithm for GPS Trajectory Simplification," IEEE Transactions on image processing, col. 21, No. 5, May 2012.

[6] D. Zhang, M. Ding, D. Yang, Y. Liu, J. Fan, H. T. Shen, "Trajectory Simplification: An Experimental Study and Quality Analysis," Proceedings of the VLDB Endowment, vol. 11, No. 9, Rio de Janeiro, Brazil, August 2018.

[7] N. Meratnia and R. A. de By, "Spatiotemporal compression techniques for moving point objects," In EDBT, pages 765–782, 2004.

TABLE I
WIFI2POSE EVALUATION

Order	Keypoint	PCK@5	PCK@10	PCK@20	PCK@30	PCK@40	PCK@50
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							