A Deep Neural Network-Based Indoor Positioning Method using Channel State Information

Guan-Sian Wu and Po-Hsuan Tseng
Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan
Email: t104368031@ntut.org.tw and phtseng@ntut.edu.tw

***Abstract*—The channel state information (CSI) measurement, which characterizes the multipath channel between the transmitter and the receiver, can serve as a fingerprint for the receiver position. Recently, a series of deep learning-based indoor positioning fingerprinting (FP) methods using CSI have been proposed to enhance the localization performance. In this paper, we present a deep neural network (DNN)-based indoor positioning FP system using CSI, which is termed DNNFi. The proposed DNNFi, which maintains a single DNN instead of multiple deep autoencoders at different reference points, allows a faster computation for the online inference and a lower memory usage for the weights/biases. A stack of autoencoders is utilized to pre-train the weights layer-by-layer. The softmax function is utilized to decide the probabilities of the receiver position being on these reference points, which can be used to estimate the receiver position. Experimental results are presented to confirm that DNNFi can effectively reduce location error compared with the conventional CSI positioning FP approaches.**

***Index Terms*—Deep Neural Network (DNN), Channel State Information, Fingerprinting, Indoor Positioning.**

I. Introduction

Location-based services (LBSs) have drawn attention due to their huge potential market values. Unlike outdoor LBSs has a 10-m positioning accuracy, which is achieved by the global positioning system (GPS), the accuracy of indoor LBSs is expected to within a couple of meters level. In general, indoor positioning decides receiver's location with the information of distances or angles extracted from the signals emitted by the radios. With the firmware and software modifications, indoor positioning can be implemented using the commercial radios, such as cellular radios [1], Wi-Fi [2], and Bluetooth.

The trilateration-based and the fingerprinting (FP)-based indoor positioning methods are two major categories to estimate the indoor position of the devices. The trilateration-based method [3] estimates the device's positions based on their measured distances/angles to the base stations. In line-of-sight (LOS) scenarios, distances/angles can be measured accurately. However, in non-line-of-sight (NLOS) scenarios, when the direct paths are obstructed, the positioning accuracy is limited due to the estimation biases of distances/angles. Nevertheless, the FP-based method does not necessarily have the same performance trend in LOS/NLOS scenarios. The FPs of the received signals, e.g., the received signal strength (RSS) and the channel state information (CSI), provide richer features in the NLOS scenario. RSS-based FP methods [4], [5] are effective and simple to handle a diverse indoor propagation scenarios. In these methods, the K-nearest-neighbor (KNN) algorithm can smooth out the RSS errors induced by small-scale fading.

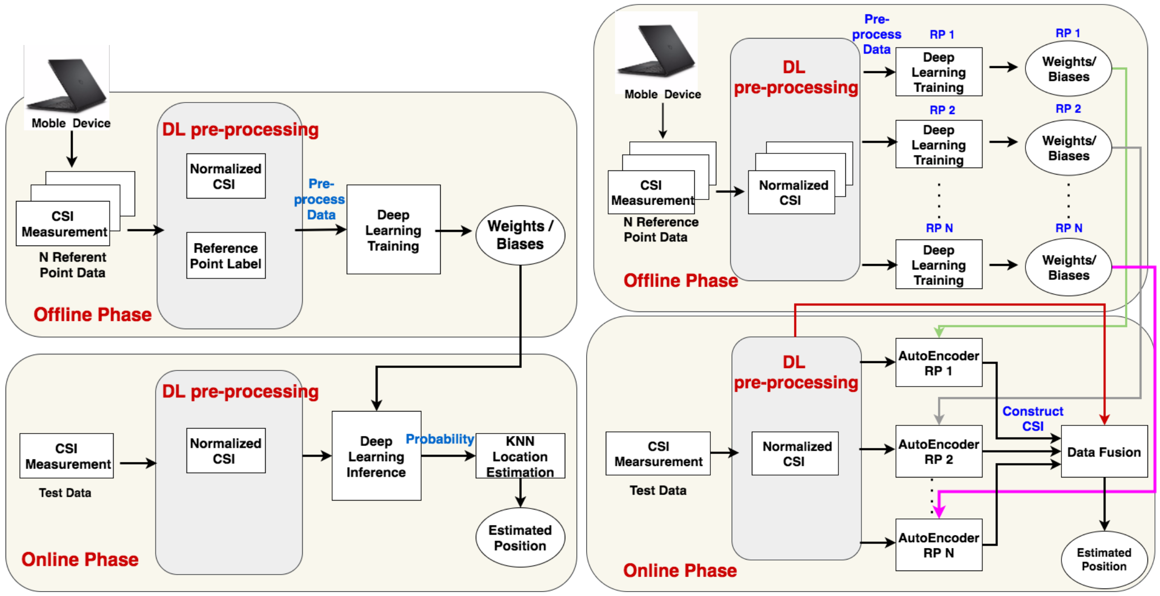
CSI, which characterizes the multipath channel between the transmitter to the receiver, is used as another positioning FPs [6]. Demanded by various broadband communication systems, e.g., Wi-Fi system, CSI is measured at the physical layer to achieve reliable communication. With the modification of firmware, CSI can be accessed by the upper layer. A CSI tool based on an Intel Wi-Fi 5300 network interface card [7] has been utilized to measure CSI while Atheros-CSI-Tool [8] is another example to access CSI. Containing richer information about receiver positions than RSS does, CSI is expected to improve the FP methods that only takes RSS as the signature.

Unlike the RSS, which can be described by a log-normal probability distribution based on the empirical path loss model, there is no general probability distribution for CSI. Although the stochastic channel model is often utilized to represent the behavior of different types of environment for communication performance assessment, e.g., the urban and the indoor WINNER models, this kind of model can not reflect the differences between the transmitter and different RPs. Statistical estimation theories cannot be applied for indoor positioning in the absence of a CSI probability model. In this situation, machine learning-based methods, which rely on a large amount of offline data being collected at known RPs, can be utilized to learn the classification rule. The classification rule is learned based on the similarity of the measured data between the measured object and the RP. Deep neural network (DNN), which regains its success in the last decade, mostly in the supervised learning, has the advantage to extract the features of CSIs. A series of works from *X. Wang et al.* [9], [10] adopt the concept of deep learning to classify the online measurements, and determines its position with the RPs of the most similar signatures. We refer to one of them, [9], in which DeepFi is proposed for the deep learning to extract features from the amplitude of CSI.

Although DeepFi successfully outperforms the KNN-based CSI in [11] and shows its advantage to tolerate the human effect, the major component of the DeepFi is based on the classic deep belief nets (DBNs) with a stack of restricted Boltzmann machines (RBMs) [12], [13]. Since there are lots of techniques being discussed recently, such as if pre-train is

[1]This work was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-027-019 and by the Technological University Paradigms.

978-15386-3652-7/18/$31.00 ©2018 IEEE



(a) Proposed DNNFi (b) DeepFi [9]

Fig. 1: Flowchart comparison between proposed DNNFi and DeepFi [9]

necessary or not in the current computer's computing power, the trend motivates us to study if there is any other DNN suitable for the indoor positioning problem. We propose a deep neural network-based fingerprinting (DNNFi) method, as shown in Fig. 1(a). The key differences of our proposed DNNFi and DeepFi [9], [10] illustrated in Fig. 1(b) are listed as follow

1) The deep learning part of proposed DNNFi is supervised learning, while that in DeepFi is unsupervised learning.
   a) Exact RPs' positions, which are required in the offline stage of FP-based method, can serve as the input labels available for deep learning. Based on the labels, the output of DNNFi estimates the probabilities of the measured position being at RPs.
   b) DeepFi requires training a deep autoencoder for each RP. Since the output of the autoencoder is the reconstructed CSI from each RP, the probabilities of the measured position being at RPs are calculated at the fusion stage before position estimation.
2) This architecture of DNNFi with one DNN allows a speed-up over DeepFi, which requires a number of autoencoders equal to the number of RPs.

We conduct our experiment using Atheros CSI tool to verify that DNNFi performs better than DeepFi. On the other hand, DNNFi maintains lower computational complexity in the online stage, in which the computational time is N times faster than DeepFi, given N as the number of RPs. Notice that Wi-Fi supports CSI measurement, and the proposed DNNFi can be utilized in any Wi-Fi systems after the firmware modification.

The remainder of the paper is organized as follows. An overview of the system model is presented in Section II. The procedures of proposed DNNFi method are detailed in Section III. In Section IV, we state the indoor experiments with which we used for performance evaluation and then compare the performance of the proposed DNNFi, DeepFi and other FP methods. Finally, we summarize our work in Section V.

II. System Model

A. Channel Model

Provided that the bandwidth be infinite, the ideal channel impulse response (CIR) model with total L paths is represented as a deterministic tapped delay line model as

$$h(\tau) = a_0\delta(\tau - \tau_0) + \sum_{i=1}^{L-1} a_i\delta(\tau - \tau_i) \ . \tag{1}$$

i represents the index of the arriving multipath and τ_i is the arriving timing of the i-th path. For the notation simplicity, we denote τ_0, i.e., the arriving timing of the first path, as the transmission delay. a_i is the complex channel gain associated with the i-th path. In the considered indoor scenario, we considered a slow fading scenario, in which the variation rates of a_i and τ_i caused by the nearby motions of people and obstructer are very low. Therefore, we treat a_i and τ_i as time-invariant random variables. The relative delay of each path and its associated gain are determined by penetration, reflection, diffraction, and scattering of wave propagation. The multipath channel between the transmitter and the receiver is fully characterized by the CIR, which can be utilized as a signature of the position where the receiver locates.

By utilizing Wi-Fi system to measure CSI, the access point (AP) transmits a preamble sequence, and the CSI is measured at the receiver on a total of M orthogonal subcarriers. We represent the ideal CSI model of the k-th subcarrier as

$$H = \sum_{k=0}^{L-1} a_i e^{-j2\pi f_i \tau_k} , \quad (2)$$

$\forall i = 0 \sim M-1$, where f_i is the i-th subcarrier frequency. Let Δf denotes the frequency spacing. f_i can be expressed as $f_i - f_0 = i\Delta f$. In this model, the path information, i.e., the channel gain and the delay, can be observed in CSI. Therefore, CSI can reflect a unique channel characteristic between the AP and a certain received position. In practice, CSI is measured and collected in the presence of channel estimation error.

B. Fingerprinting System Architecture

With the Atheros CSI tool developed in [8], we utilize Atheros *AR9380* to measure the CSI between the AP and the receiver. We adopt 20-MHz system bandwidth using 2.4-GHz frequency band. In this mode, CSI values of $M = 56$ subcarriers can be measured. Each CSI value is with 10-bit resolution for both imaginary and real part. The proposed algorithm is based on the concept of fingerprinting and the adoption of DNN, with the system architecture that includes the offline and online phases, as shown in Fig. 1(a).

1) **Offline** phase: We establish a database for the indoor positioning area in the offline phase. One laptop with Atheros *AR9380* serves as the AP with the *hostapd* daemon to send the preamble, and the other one serves as the receiver to measure CSI. The fingerprints are collected at a total of N RPs, where the signals are transmitted from 1 AP in this paper. By placing the position of the receiver at the j-th RP in the two-dimensional (2-D) space as $\mathbf{x}_j^{RP} = [x_j^{RP} \ y_j^{RP}]$, the receiver is capable of measuring the CSI as a 56×1 vector $\mathbf{H}_j$ from the AP $\mathbf{x}^{AP} = [x^{AP} \ y^{AP}]$. We set up a label for each RP for supervised learning. The label of the j-st RP is represented as a $N \times 1$ vector $\mathbf{l}_j$, where

$$\mathbf{l}_j[i] = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

We utilize the collected CSI to train the DNN.

2) **Online** phase: The Atheros CSI tool is utilized to collect the online CSI measurement as a 56×1 vector $\hat{\mathbf{H}}$ at some position in the designated area. In our proposed method, we utilize the weights and biased from the offline-training to inference the probabilities of the receiver position being on the RPs. The 2-D position is estimated as $\hat{\mathbf{x}} = [\hat{x} \ \hat{y}]$.

III. Proposed Deep Neural Network-Based Fingerprinting (DNNFi)

As shown in Fig. 1(a), the proposed DNNFi adopts a supervised learning architecture with the softmax function to find the probability of position being on the RP. Before the DNN training, the offline data preprocessing is required to normalize the input between 0 to 1. The weights/biases of the DNN are trained based on the data collected at N RPs in the offline phase and can be utilized for the online inference. The inferred probabilities are then combined with the concept of KNN, which estimates the receiver's position based on the weighted average of the top-K probabilities and their RP positions.

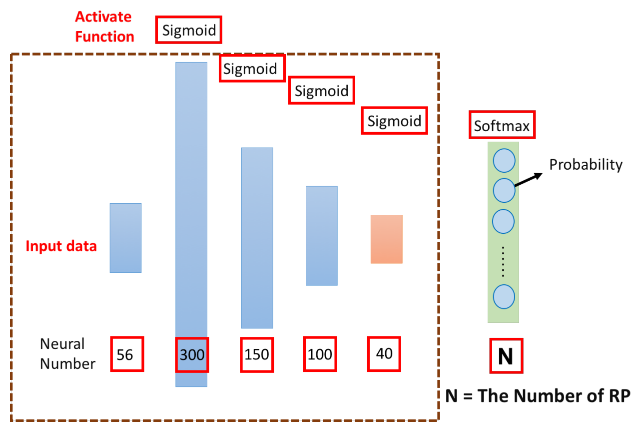


Fig. 2: Network Structure of Proposed DNN

A. Offline-Phase: Preprocessing/DNN Training

For the effective learning of DNN, the input is set from 0 to 1. Therefore, we first normalize the input of our DNNFi, $\mathbf{H}^n$, to the value of 0 to 1.

The network structure of the proposed DNN is illustrated in Fig. 2. We collect 10,000 CSI measurements at N RPs. We considered four hidden layers before the softmax for the output layer. The numbers of neurons are 300, 150, 100, and 40 for these four layers, which are consistent with DeepFi [9]. Both architectures have the same number of neurons and the same activation function. However, in the online stage, the number of autoencoders in DeepFi is equal to the number of RPs, while a single DNN is utilized in DNNFi, as shown in Fig. 2. Therefore, the computational time of DNNFi in the online stage is N times faster than DNNFi. We choose the activation functions of the four hidden layers, $f_h(.)$, as the sigmoid function, in which its output is in range (0, 1). The activation function of the output layer, $f_o(.)$, is the softmax function, which outputs the probabilities of being on the RP positions. After the preprocessing step, we train the weights $\mathbf{w}_1 \sim \mathbf{w}_4$ and biases $\mathbf{b}_1 \sim \mathbf{b}_4$ for this DNN based on the input of normalized CSI measurements $\mathbf{H}_j^n$ and their associated labels $\mathbf{l}_j$ for $j = 1$ to N. The output of the 1-st hidden layer can be represented as $\mathbf{Y}_1 = f_h(\mathbf{w}_1\mathbf{H}^n + \mathbf{b}_1)$ and the output of the remaining i-th hidden layer is $\mathbf{Y}_{i+1} = f_h(\mathbf{w}_i\mathbf{Y}_i + \mathbf{b}_i)$. The loss function is to find the minimum norm between the input label and the output function $\mathbf{Y}_o = f_o(\mathbf{Y}_4)$ as $\arg\min ||\mathbf{Y}_o - \mathbf{l}||$.

Using unsupervised pre-training to initialize neural networks and fine-tuning them in a supervised task [14] which can be dated back to DBN [12], was one of the essential steps for deep learning at that time. However, with sufficient data,

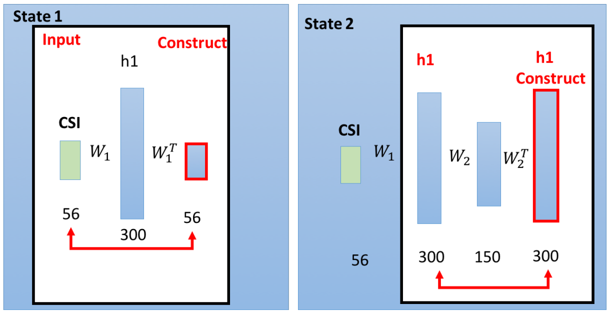


Fig. 3: Stacked Autoencoder Pre-Training

improvement of initialization and computational power, the popularity to pre-train a supervised DNN using unsupervised NN as initialization is no longer a must. A decent discussion on this subject can be referred to [15], in which the authors propose a method that unsupervised pre-training with DNN can regularize the supervised training for a better generalization performance. In the indoor localization case, we observe that if without pre-training, the proposed DNN relies on the shape of CSI is more than the absolute value of CSI. To explicitly consider its absolute value, we utilize the stacked autoencoder [16] for pre-training, as shown in Fig. 3. The loss function of the stacked autoencoders is designed as $\ell = ||\mathbf{H} - \hat{\mathbf{H}}||^2 + \alpha \cdot \Big| ||\mathbf{H}||^2 - ||\hat{\mathbf{H}}||^2 \Big|$, where $\mathbf{H}$ is the input of each layer and $\hat{\mathbf{H}}$ is the associated output. The left part considers the norm between the input and the output CSIs; the right part is to maintain the normalization to the input data of different layers. The weights and biases of each layer are pre-trained to serve as the initial values of DNN, which retain a better description of unlabeled data to be more discriminative for labeled data. Finally, the softmax output layer is trained, and then back propagation to fine-tune the weights and biases for each layer.

B. Online Phase: DNN Inference for Location Estimation

Given the online measured CSI, we utilize the trained DNN for the online inference. The output of the DNN $\mathbf{Y}_o$ is the probability of the receiver being on each RP. The weight of the i-th RP can be determined based on the inferred probability in DNN as $w_i = \mathbf{Y}_o[i]$. We select out the indices of RPs with the first K largest w_{i_l} and drop out the rest. Without loss of generality, we denote these indices as $i_1, i_2, \ldots, i_K$ so that $w_{i_1} \geq w_{i_2} \geq \ldots \geq w_{i_K}$. Finally, we can estimate the receiver position as

$$\hat{\mathbf{x}} = \frac{\sum_{l=1}^{K} w_{i_l} \cdot \mathbf{x}_{i_l}^{RP}}{\sum_{l=1}^{K} w_{i_l}} . \tag{4}$$

IV. Experimental Results

We tested the proposed algorithm in the room 211 of the Integrated Complex Building of National Taipei University of Technology, Taipei, Taiwan. Fig. 4 shows a layout of the floor plan. We placed 1 AP and measured the CSI measurements at 40 different received positions which are separated by 50 cm. For the offline DNN training, we chose $N = 31$ RPs. We used the same offline/online measurement data with DNNFi to explore the positioning performance of the proposed DNNFi with other state-of-art methods, i.e., DeepFi [9] and KNN-CSI [11].

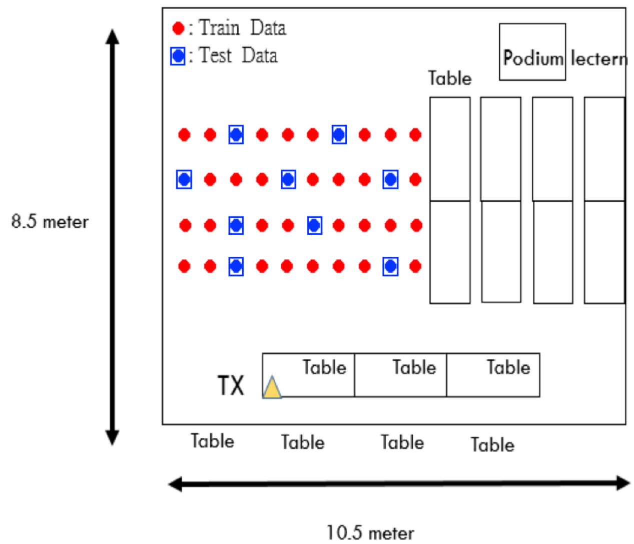


Fig. 4: Test Environment: Yellow Triangle: AP Position, Red Point: RP Position, Blue Point: Test Position

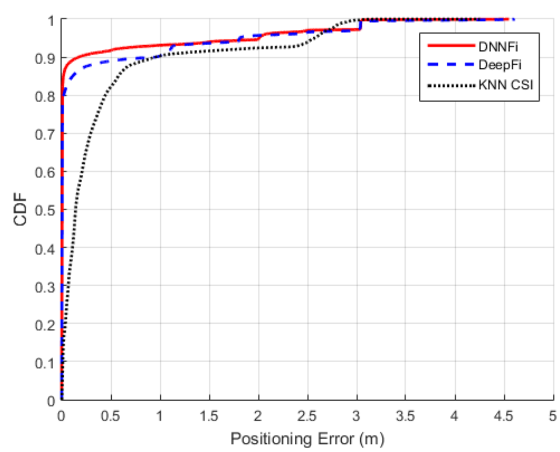


Fig. 5: Performance of fingerprinting algorithms when the test points are the same with the RPs.

The first part is the validation of the testing performance when the test points are chosen from the RPs, but the CSI measurements are collected in different time instants. In other words, although the online receiver position is the same as one of the offline receiver positions, the time-varying channel effect might result in different offline and online CSI data. Notice that this is usually the testing performance considered for machine/deep learning to see if the training based on the collected measurement is also robust for the new measurement. As shown in Fig. 5, we can find that our proposed DNNFi has

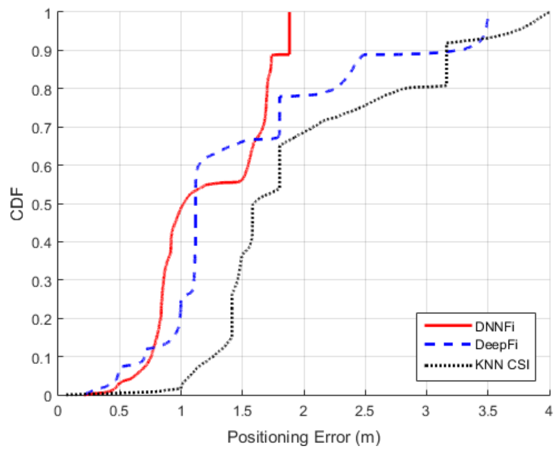


Fig. 6: Performance of fingerprinting algorithms when the positions of test points are the blue circles in Fig. 4.

a similar performance with DeepFi, where 85% of the time the classification for RPs is correct without any error.

In positioning performance, the second case is when the rest nine received positions in Fig. 4 are the test points for the online performance comparison. The online test points for the performance comparison are marked as the blue points in Fig. 4. Unlike the robustness testing in the previous case, this case is more important for the comparison of the performance of FP-based methods. Notice that both the proposed DNNFi and the KNN-based FP [4], [11] are classified as supervised learning; however, instead of using the mean and variance statistic for the collected CSI in the offline database, the introduction of deep learning allows a better feature extraction to distinguish the RP compared to only using the statistical information. As shown in Fig. 6, the deep learning-based methods outperform the KNN-based CSI, which can only rely on the statistics from the offline data, such as means and standard deviations. Moreover, our proposed DNNFi has a 20% performance improvement over DeepFi at 90% of positioning error while maintains a similar performance at the rest percentage. The utilization of a single DNN in DNNFi considers not only the feature extraction of the channel responses at different RPs but also the feature discriminations between them, while parallel autoencoders in DeepFi consider feature extraction independently. With the input of all the measurements into a single DNN, the features extracted from the measured CSIs of different RPs can be more discriminative compared to those from the proposed DNNFi. The result demonstrates the effectiveness of the use of labels in the supervised architecture of DNNFi. Besides, DNNFi has a lower computational complexity for training and inferring in a DNN, which requires $1/N$ computational time in the online stage.

V. Conclusion

We propose a deep neural network (DNN)-based fingerprinting (DNNFi) method. With a pre-processing of CSI and an unsupervised pre-training to consider the channel state information (CSI) and its normalized amplitude, a supervised DNN extracts the features of CSI at different reference points (RPs). The output of DNN, probabilities of received position being on the RPs, can be used to estimate the weighted average position of the receiver. We test our proposed method by using only one access point (AP). The results show that the proposed DNNFi has a performance compared to the state-of-art methods such as DeepFi and KNN-CSI. Compared to DeepFi, the proposed DNNFi not only saves online inferring time but also saves the memory usage for the storage of the weights and biases.

References

[1] P. H. Tseng and K. T. Lee, "A Femto-Aided Location Tracking Algorithm in LTE-A Heterogeneous Networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 748–762, Jan 2017.

[2] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, "Received-signal-strength-based Indoor Positioning using Compressive Sensing," *IEEE Transactions on Mobile Computing*, vol. 11, no. 12, pp. 1983–1993, 2012.

[3] W. Yuan, N. Wu, B. Etzlinger, H. Wang, and J. Kuang, "Cooperative joint localization and clock synchronization based on gaussian message passing in asynchronous wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7258–7273, 2016.

[4] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*. ACM, 2005, pp. 205–218.

[5] H.-X. Liu, B.-A. Chen, P.-H. Tseng, K.-T. Feng, and T.-S. Wang, "Map-aware Indoor Area Estimation with Shortest Path Based on RSS Fingerprinting," in *IEEE Vehicular Technology Conference (VTC) Spring*, May 2015, pp. 1–5.

[6] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: Indoor Localization via Channel Response," *ACM Comput. Surv.*, vol. 46, no. 2, pp. 1–32, Dec. 2013.

[7] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool Release: Gathering 802.11n Traces with Channel State Information," *ACM SIGCOMM CCR*, vol. 41, no. 1, p. 53, Jan. 2011.

[8] Y. Xie, Z. Li, and M. Li, "Precise Power Delay Profiling with Commodity WiFi," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 53–64.

[9] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based Fingerprinting for Indoor Localization: A Deep Learning Approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan 2017.

[10] X. Wang, L. Gao, and S. Mao, "CSI Phase Fingerprinting for Indoor Localization with a Deep Learning Approach," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2016.

[11] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. Ni, "CSI-Based Indoor Localization," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 7, pp. 1300–1309, July 2013.

[12] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why Does Unsupervised Pre-training Help Deep Learning?" *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 625–660, 2010.

[15] J. Dong, X. Mao, C. Shen, and Y. Yang, "Unsupervised feature learning with symmetrically connected convolutional denoising autoencoders," *CoRR*, vol. abs/1611.09119, 2016. [Online]. Available: http://arxiv.org/abs/1611.09119

[16] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.