A DNN-based WiFi-RSSI Indoor Localization Method in IoT

Bing Jia[1], Zhaopeng Zong[1], Baoqi Huang[1(✉)], and Thar Baker[2]

[1] Inner Mongolia University, Hohhot 010021, China
{jiabing,cshbq}@imu.edu.cn, 31809008@mail.imu.edu.cn
[2] SLiverpool John Moores University, Liverpool, UK
t.baker@ljmu.ac.uk

Abstract. Indoor automatic localization technology is very important for the Internet of Things. With the development of wireless technology and the diversification of location service requirements, especially in complex indoor scenarios, users are increasingly demanding location-based services. Traditional Global Positioning System (GPS) location technology is difficult to solve some positioning problems in indoor environments, and WiFi is now available in most indoor environments. Therefore, using WiFi for positioning does not require additional deployment of hardware devices, which is a very cost-effective method. However, WiFi-based indoor positioning requires a large amount of data, so we can use artificial intelligence methods to analyze the data and obtain a positioning model. The traditional indoor positioning methods based on WiFi signals have some problems such as long positioning time and poor accuracy. In order to solve the above problems, this paper proposes an indoor localization method based on Deep Neural Networks (DNN) for WiFi fingerprint. In particular, a DNN-based WiFi-RSSI positioning method is proposed for indoor automatic localization. Besides, in the process of DNN training, a joint training method based on unsupervised learning and supervised learning is adopted and the special loss function is defined. Extensive experiments are carried out in both the UJIIndoorLoc public database and a real scenario, and a thorough comparison with several existing approaches indicates that the proposed scheme improves the localization accuracy on average.

Keywords: Indoor localization · Deep neural networks · WiFi-RSSI

1 Introduction

Location information is the basis of various Internet of Things (IoT) application systems to achieve service functions. It is an important issue to obtain location information through various location technology. Location issues include

Thanks to the National Natural Science Foundation of China (Grants No. 41761086 and 41871363,), the Natural Science Foundation of Inner Mongolia Autonomous Region of China (Grant No. 2017JQ09), and the Grassland Elite Project of the Inner Mongolia Autonomous Region (Grant No. CYYC5016).

H. Gao et al. (Eds.): ChinaCom 2020, LNICST 352, pp. 200–211, 2021.
https://doi.org/10.1007/978-3-030-67720-6_14

both outdoor localization and indoor localization. Outdoor localization has been widely used, especially Global Positioning System (GPS), which can enable people to travel the world freely. Traditional GPS location technology is difficult to solve some positioning problems in indoor environment due to a lack of line of sight (LoS) transmission channels between satellites and indoor receivers. With the booming development of IoT industry in recent years, the demand for indoor positioning technology has been increased by the various IoT application scenarios have greatly increased. Great efforts have been devoted to developing Indoor Positioning Systems (IPSs) to enable reliable and precise indoor positioning. Nowadays, with the popularity of the Internet, there are many WiFi signals around people. By measuring the received signal strength indicator (RSSI) of the WiFi signal, makes the indoor localization based on RSSI possible. In general, the WiFi-based IPS may estimate the position based on Time of Arrive (TOA) [4], Time Difference of Arrival (TDOA) [3] or Arrival of Angle (AOA) [16]. However, additional specialized equipments must be required to measure the round trip or angle of the WiFi, because the ordinary signal receiving equipment is not accurate enough. In contrast, fingerprint-based methods [1,15] do not require special device and are therefore easier to implement.

The original fingerprint-based positioning system [1] used K Nearest Neighbors (KNN) to find the closest match from the fingerprint database. In order to improve the robustness of the positioning system, a Bayesian-based [6,8,12] filtering method is proposed. Subsequently, the RSSI samples are associated with the fingerprint database by using Support Vector Machine (SVM) [15] and Compressed Sensing (CS) [5,10]. In order to reduce the burden of collecting fingerprint database, Pan et al. [11] proposed transfer learning. Recently, Li et al. [9] proposed a method of shunting short-term memory to solve the problem of location in a dynamic environment. De et al. [13] proposed a multi-hop approach to solve the problem of precise positioning.

However, due to multipath fading and occlusion of objects, the wireless signal has fluctuations, which in turn leads to changes in RSSI, which is a major problem for the positioning accuracy of the fingerprint-based. Even at the same location, the RSSI is different for different time periods. For more precise positioning, we need to collect as much Access Points (AP) information as possible, especially when the environment is particularly large. Therefore, the data of the fingerprint database will also become larger and larger. The challenge of the WiFi-based IPS is how to extract valid features from a large amount of data and find the best match.

The DNN has achieved significant success in solving many similar problems. In the fingerprint method, a machine learning method can be used to extract the main features of the signal. These features are treated as fingerprints and stored in a database. The fingerprint method can not only improve the positioning accuracy, but also reduce the computational complexity. This paper proposes an $8-layers$ DNN model, which consists of 7 fully connected hidden layers and an output layer to solve indoor positioning problems.

The rest of the paper is organized as follows. Section 2 introduces a general structure of DNN to predict the position, which is a $8-layers$ DNN model; The WIFI-RSSI positioning model based on DNN is described in Sect. 3. Section 4 gives the WIFI-RSSI database description and the result of experiments. Finally, We conclude this paper and shed light on future works in Sect. 5.

2 The Structure of DNN for Localization

DNN is very effective in the classification and regression of hidden feature extraction, and has been widely used in visual recognition and artificial intelligence in recent years. The popularity of DNN is attributed to the improvement of Graphic Processing Unit (GPU) processing power and the emergence of advanced DNN libraries, which simplify the realization of complex ideas. We can use deep neural networks to model the dependencies between location coordinates and WiFi fingerprints. In this paper, we propose an localization method based on $8-layers$ DNN to predict the position.

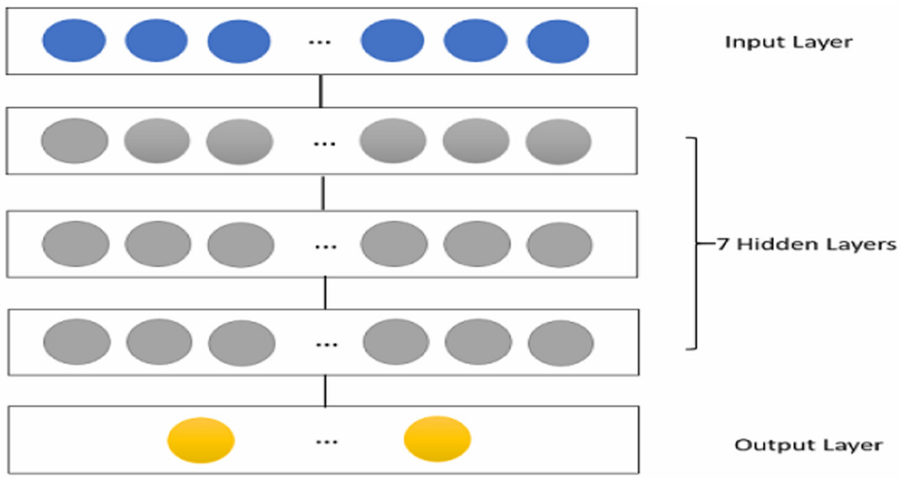


Fig. 1. The structure of the DNN for localization.

We use a fully connected DNN to implement the positioning task with WiFi fingerprint, as shown in Fig. 1. The DNN model has 7 fully connected hidden layers, as well as an input layer and an output layer. Each hidden layer has $k_1, k_2, ..., k_i, ..., k_7$ neurons. The input layer of the DNN is the WiFi signal value $R = (r_1, r_2, ..., r_n)$, and the output layer of the DNN is the predicted coordinates.

3 WiFi-RSSI Localization Model Based on DNN

3.1 The Structure of WiFi-RSSI Localization Model

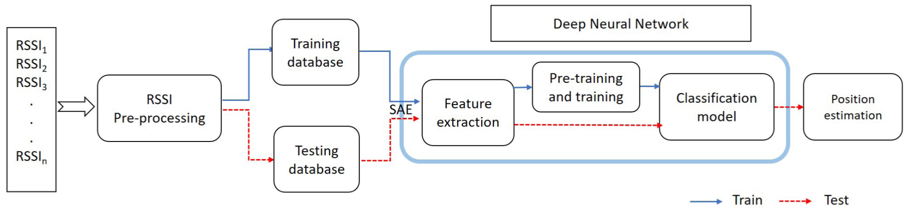


Fig. 2. The structure of WiFi-RSSI localization model.

The localization model, as shown in Fig. 2. The model consists of two parts: the offline-training phase and the online-localization phase. During the offline-training phase, the data of the fingerprint database is used to put into the $8 - layers$ of DNN for training. The training of the DNN is to determine the structure of the deep neural network and the weight of each neuron. First performing feature extraction on the data. Then performing pre-training to initialize network parameters. After a series of fine-turning and dropout, a classification model is finally obtained. In the online-localization phase, we use the testing database to evaluate the performance of the model. First, the WiFi-RSSI data needs preprocessing. Then, inputting the processed data into the trained WiFi-RSSI localization model, and finally outputting the current position coordinates and calculating the Average Localization Errors.

In the localization model, we use Pandas to read the data, Scikit-learn and Numpy to normalize and format the data, and the data flow diagram of Tensor Flow to calculate and update the parameters.

3.2 Encoding and Decoding Based on SAE

To reduce the dimensionality of the WiFi-RSSI data, we used Stacked AutoEncoders(SAE) [2], as shown in Fig. 3. Before encoding, we need to regularize the WiFi-RSSI data. Three hidden layers are used here with dimensions of 256, 128, and 64, and the feature dimensions can be continuously reduced. Finally, a 64-dimensional feature is obtained. This feature is used to perform reverse reconstruction to obtain the reconstructed feature. The reconstructed feature is the same as the original feature. By SAE, we can obtain data that contains many detailed and valuable feature information than the raw data, which is used to train classifiers with specific contexts and is more accurate than training with raw data.

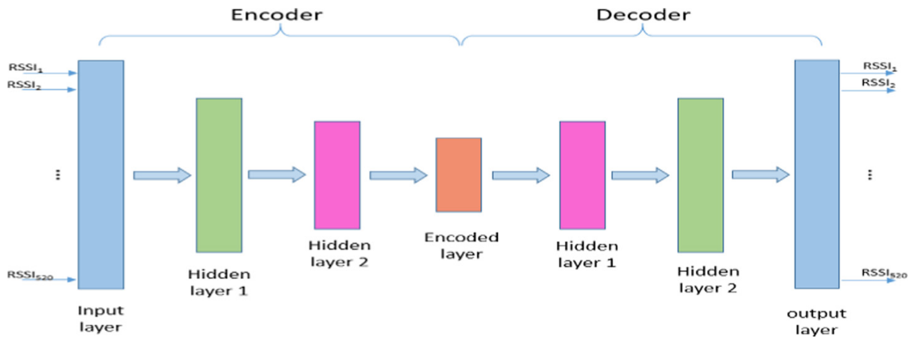


Fig. 3. Stacked AutoEncoder used in DNN.

3.3 The Selection of Activation Function

The activation function introduces non-linearity into the deep neural network so that the deep neural network can have hierarchical non-linear mapping learning ability, which is also an important factor affecting performance. So the activation function is an integral part of the neural network. In the localization model, we use the softmax function [7] as the activation function, the formula is as follows:

$$S_i = \frac{e^i}{\sum_j^n e^j} \tag{1}$$

where, S_i represents the probability that the current output belongs to i, e^i represents the i-th power of e.

3.4 The Definition of Loss Function

We use the loss function based on Back Propagation (BP) to train the DNN weights. In order to obtain better positioning results, it is necessary to continuously reduce the difference between the real coordinates and the predicted coordinates. So we redefine the loss function to calculate the difference between the real coordinates and the output of the DNN model. The formula is as follows:

$$f_{loss} = \sqrt{(x_p - x_t)^2 + (y_p - y_t)^2} \tag{2}$$

where (x_t, y_t) is the real coordinate of reference point sample i, and (x_p, y_p) is the estimated coordinates of reference point sample i. By using the BP algorithm to minimize the value of the loss function f_{loss}, the gradient descent algorithm is used to update the DNN weights until the value of f_{loss} converges.

3.5 Unsupervised Pre-training and Supervised Training

In order to solve the network parameter contribution is small and the update speed is slow, we use unsupervised pre-training. Supervised pre-training uses layer-by-layer greedy training strategy, that is, training the first hidden layer of the network, and then training the second hidden layer, until the last hidden layer is trained. Finally, we use these trained network parameter values as the initial values of the overall network parameters.

The input of the input layer is the n-dimensional preprocessed WiFi-RSSI vector $R_i = (r_1, r_2, ..., r_n)$ and the label L_i. First, the signal value is encoded and decoded through SAE to obtain the representation of the WiFi-RSSI data and reduce the dimensionality of the WiFi-RSSI data. In this way, the results are more accurate than training with raw data. Then, unsupervised pre-training is adopted. After the pre-training is completed, the obtained parameters are used as initial values of the entire network parameters. Finally, supervised training is performed. Using the BP algorithm and the gradient descent algorithm to fine-tuning continuously and obtain the optimal network parameters.

4 Experiments

In this section, extensive experiments have been done to verify the performance of WiFi-RSSI localization model.

4.1 Experimental Data

In order to test and evaluate the performance of the system, a large database containing location tags is required. In this experiment, two databases of self-collected laboratory database and UJIIndoorLoc [14] public database were used for training and testing. The main difference between the two databases is that the self-collected database is a dense database, and the reference points are uniformly distributed; the UJIIndoorLoc public database is a sparse database, the reference points are unevenly distributed, some places are dense, and some places are sparse.

The Self-collected Laboratory Database. The experimental data was collected at the 316 laboratory on the 3rd floor of the School of Computer Science, Inner Mongolia University. The laboratory covers an area of about 78 square meters(the length is 13.5 m, the width is 6 m).

The laboratory data is collected at a sampling interval of 1 m. First, taking the position of the door as the coordinate origin, then dividing the coordinates, from (1,1) to (5,13), a total of 65 points are setting as shown in Fig. 4.

Next, We use mobile phones to collect data at each point. The collecting data from 65 coordinate points as training data; then, at non-integer coordinate points, we randomly select some points to collect data as testing data. Each record of the collected data includes 5 attributes, WiFi name, Media Access Control (MAC) address, RSSI levels, X and Y. Some examples of the collected data are shown in the Table 1.

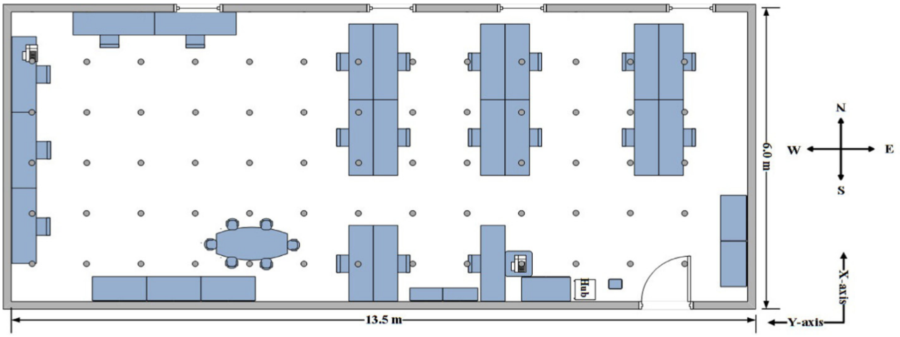


Fig. 4. The coordinate points of 316 laboratory.

Table 1. Some examples of 316 laboratory collected data.

WiFi name	MAC address	RSSI levels	X	Y
IMUDGES Pro	d4:a1:48:a4:87:3c	−83	1	1
network	f4:83:cd:91:16:62	−82	2	1
NETGEAR	00:0f:b5:35:32:a4	−48	3	1

The collected data is processed to establish a fingerprint database. Training database with a total of 10,339 sampled points, each of which includes 84 wireless access points, coordinate points (X, Y), and testing data with 6337 sampled points, the attributes of 316 laboratory database as shown in Table 2.

Table 2. Some examples of 316 laboratory collected data.

Index	Attribute
01–84	RSSI levels
85	X
86	Y

The UJIIndoorLoc Database. The UJIIndoorLoc database covers a surface of 108703 m^2 including 3 buildings with 4 or 5 floors depending on the building. The number of different places(reference points) appearing in the database is 933. The map of the UJI Riu Sec Campus, as shown in Fig. 5.

The database consists of 21049 sampled points, of which 19938 are used for training and 1111 are used for testing. Each record in the database contains 529 attributes, as shown in the Table 3, the first 520 attributes inform about the RSSI levels of these 520 wireless access points. The remaining 9 attributes contain longitude and latitude of measurement, floor number, building ID, space ID, relative position, user ID, phone ID and timestamp of measurement.

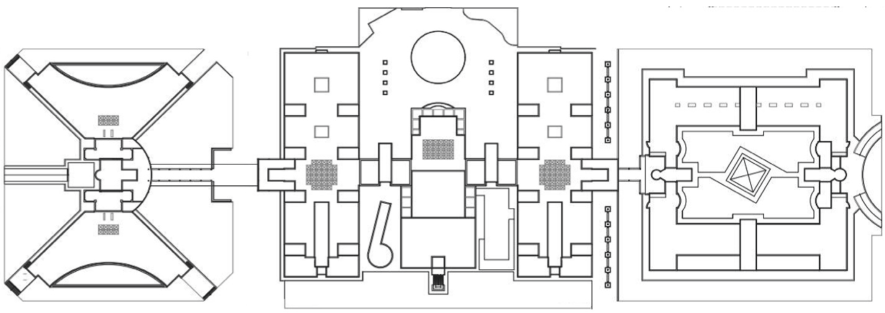


Fig. 5. The map of UJI Riu Sec Campus.

Table 3. The attributes of UJIIndoorLoc database.

Index	Attributes
001–520	RSSI levels
521–523	Real world coordinates of the sample points
524	BuildingID
525	SpaceID
526	Relative position with respect to SpaceID
527	UserID
528	PhoneID
529	Timestamp

To ensure the independence of the database, a validation (or testing) samples are obtained 4 months after the training samples. Therefore, using UJIIndoorLoc database for positioning is a challenge, but the results obtained can be used to estimate the actual performance of the system. For the use of UJIIndoorLoc public database, some methods are used to locate the building and the floor. The method proposed in this paper is used to locate coordinate points.

4.2 Setup

The localization system runs on a PC with the windows 10(64 bit) operating system, which CPU is i7-4690 and memory is 8 GB, the development tool is PyCharm.

This experiment uses Average Localization Errors to measure the performance of the positioning. First, we use the Euclidean distance to calculate the distance between two points, the formula is as follow.

$$d(i,k) = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}; \tag{3}$$

where i, k represent the real point and the predicted point, respectively.

Then we sum the distances of all the points and calculate the mean $\bar{d}$ as the average localization errors. The smaller the average localization errors, the better the positioning performance.

$$\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d(i,k) \tag{4}$$

4.3 The Comparison of 316 Laboratory

The experimental results of 316 are shown in the Table 4. The average localization errors of DNN is 2.7 m, and the average localization errors of other methods exceeds 3 m. The positioning effect of DNN is better than other methods. Since the database from the 316 laboratory was collected at a sampling interval of 1 m, the 316 laboratory database has a smaller coordinate range than the UJIIndoorLoc database. Therefore, the average localization errors calculated by these six methods is very close.

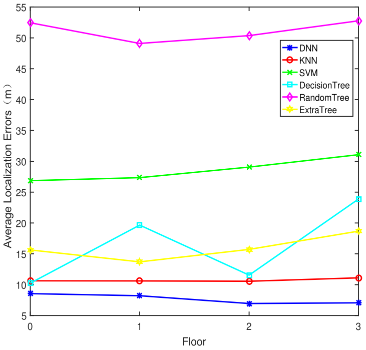
Table 4. The comparison of six methods in 316 laboratory.

Method	Average localization errors(m)
DNN	2.7
KNN	3.8
SVM	4.6
DecisionTree	3.4
RandomTree	4.85
ExtraTree	3.65

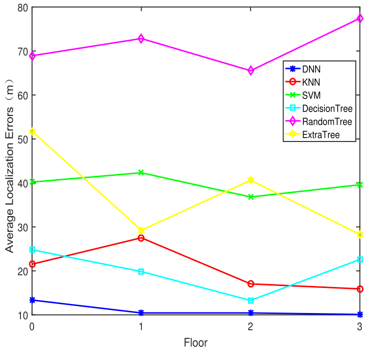
4.4 The Comparison of UJIIndoorLoc

The UJIIndoorLoc database includes 3 buildings, the building 0 and building 1 with 4 floors and the building 2 with 5 floors. We used the single floor and multiple floors data of three buildings to test the system performance respectively. The main difference between single floor and multiple floors is that single floor only uses the data of a certain floor without considering the influence of the height of the floor. Multiple floors uses the data of multiple floors, and the positioning result will be affected by the height of the floor.

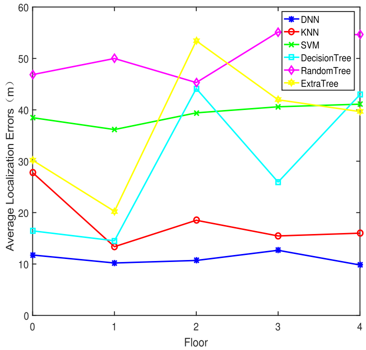
Single Floor. The experimental results of the UJIIndoorLoc database are shown in the Fig. 6. The number 0 to 4 in the figure correspond to different floor of the building. In general, the DNN is relatively stable and has the best positioning performance. The average positioning errors corresponding to the three buildings are 7.53 m, 11.07 m and 10.98 m, respectively. Other methods have large fluctuations and large positioning errors.



(a) The comparison of building 0



(b) The comparison of building 1

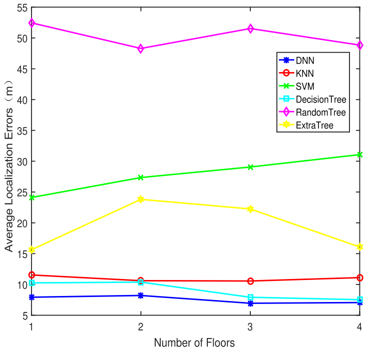


(c) The comparison of building 2

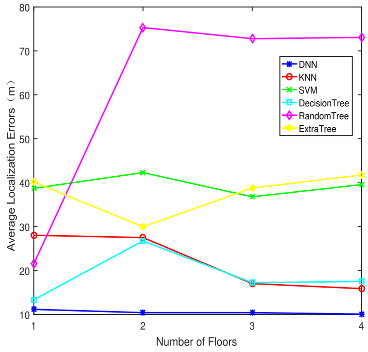
Fig. 6. The comparison of six methods for single floor in UJIIndoorLoc database.

Multiple Floors. The experimental results of the UJIIndoorLoc database are shown in the Fig. 7. The number from 1 to 5 in the figure correspond to the number of floors of the building in the UJIIndoorLoc database. In conclusion, the DNN is stable and has the best positioning performance. The minimum localization errors corresponding to the three buildings are 7.06 m, 10.09 m, and 9.81 meters, respectively. Other methods have large fluctuations and large localization errors.

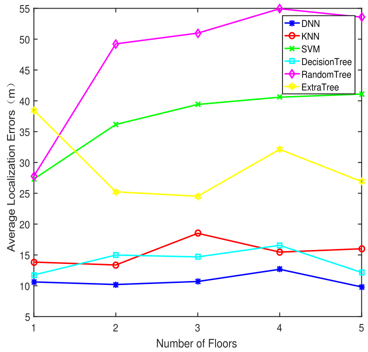
The positioning effect of DNN is better than other methods. The reason for the high localization errors is include two parts: one is that the position of the UJIIndoorLoc data point is represented by the latitude and longitude of the map, and the value is very large; the second is that when the predicted result is not accurate, it will lead to a large distance between the predicted point and the real point.



(a) The comparison of building 0



(b) The comparison of building 1



(c) The comparison of building 2

Fig. 7. The comparison of six methods for multiple floors in UJIIndoorLoc database.

5 Conclusion

This paper proposes an DNN-based WiFi-RSSI Indoor Localization Method. The average localization errors on the self-collected database is 2.7 m, which is better than the other five methods. The average localization errors obtained on the UJIIndoorLoc database is 9 m, which is also better than the other methods. Therefore, the DNN-based WiFi-RSSI Localization Method for Indoor Automatic Localization is accurate and adaptable. However, for the positioning of multiple buildings, further researched need to be conducted to improve positioning accuracy.

References

1. Bahl, P., Padmanabhan, V.N.: Radar: an in-building RF-based user location and tracking system. In: Infocom Nineteenth Joint Conference of the IEEE Computer & Communications Societies. IEEE (2000)
2. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: Advances in Neural Information Processing Systems, pp. 153–160 (2007)
3. Cheng, L., Wu, C.D., Zhang, Y.Z.: Indoor robot localization based on wireless sensor networks. IEEE Trans. Consum. Electron. **57**(3), 1099–1104 (2011)
4. Ciurana, M., Barceló, F., Cugno, S.: Indoor tracking in WLAN location with TOA measurements. In: Proceedings of the 4th ACM International Workshop on Mobility Management and Wireless Access, pp. 121–125 (2006)
5. Feng, C., Au, W.S.A., Valaee, S., Tan, Z.: Received-signal-strength-based indoor positioning using compressive sensing. IEEE Trans. Mob. Comput. **11**(12), 1983–1993 (2012)
6. Fox, D., Hightower, J., Liao, L., Schulz, D., Borriello, G.: Bayesian filtering for location estimation. Pervasive Comput. IEEE **2**(3), 24–33 (2003)
7. Gao, B., Pavel, L.: On the properties of the softmax function with application in game theory and reinforcement learning. arXiv preprint arXiv:1704.00805 (2017)
8. Hightower, J., Borriello, G.: Particle filters for location estimation in ubiquitous computing: a case study. In: International Conference on Ubiquitous Computing (2004)
9. Li, Y., Shuai, L., Ge, Y.: A biologically inspired solution to simultaneous localization and consistent mapping in dynamic environments. Neurocomputing **104**(104), 170–179 (2013)
10. Nikitaki, S., Tsakalides, P.: Localization in wireless networks via spatial sparsity. In: 2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers, pp. 236–239. IEEE (2010)
11. Pan, S.J., Zheng, V.W., Yang, Q., Hu, D.H.: Transfer learning for wifi-based indoor localization. In: Association for the Advancement of Artificial Intelligence (AAAI) Workshop, vol. 6. The Association for the Advancement of Artificial Intelligence Palo Alto (2008)
12. Paul, A.S., Wan, E.A.: Wi-fi based indoor localization and tracking using sigma-point kalman filtering methods. In: IEEE/ION Position, Location & Navigation Symposium (2008)
13. Sá, A.O.D., Nedjah, N., Mourelle, L.D.M.: Distributed efficient localization in swarm robotic systems using swarm intelligence algorithms. Neurocomputing **172**(C), 322–336 (2016)
14. Torres-Sospedra, J., et al.: Ujiindoorloc: a new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems. In: 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN) (2014)
15. Wu, C.L., Fu, L.C., Lian, F.L.: WLAN location determination in e-home via support vector classification. In: IEEE International Conference on Networking (2004)
16. Xu, J., Ma, M., Law, C.L.: Cooperative angle-of-arrival position localization. Measurement **59**, 302–313 (2015)