Received July 18, 2019, accepted August 1, 2019, date of publication August 8, 2019, date of current version August 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2933921

A Novel Convolutional Neural Network Based Indoor Localization Framework With WiFi Fingerprinting

XUDONG SONG[1,5], XIAOCHEN FAN[1], (Student Member, IEEE), CHAOCAN XIANG[2], QIANWEN YE[1], LEYU LIU[5], ZUMIN WANG[5], XIANGJIAN HE[1,4], (Senior Member, IEEE), NING YANG[3], AND GENGFA FANG[1]

[1]School of Electrical and Data Engineering, University of Technology Sydney, Ultimo, NSW 2007, Australia
[2]College of Computer Science, Chongqing University, Chongqing 400044, China
[3]School of Automation, Northwestern Polytechnical University, Xi'an 710072, China
[4]School of Software and Microelectronics, Northwestern Polytechnical University, Xi'an 710072, China
[5]College of Information Engineering, Dalian University, Dalian 116622, China

Corresponding authors: Zumin Wang (wangzumin@163.com), Xiangjian He (xiangjian.he@uts.edu.au), Ning Yang (ningyang@nwpu.edu.cn), and Gengfa Fang (Gengfa.Fang@uts.edu.au)

This work was supported in part by the Internet of Things (IoT) Laboratory of University of Technology Sydney, in part by the Liao Ning Science Public Welfare Research Fund of China under Grant 20170053, in part by the Key Research and Development Program of Liao Ning Province of China under Grant 2017104014, in part by the Liao Ning Provincial Ph.D. Startup Foundation of China under Grant 20170520438, in part by the Natural Science Foundation of Liao Ning Province of China under Grant 20180551247, and in part by the Science and Technology Innovation Fund Project of Dalian of China under Grant 2018J12GX049 and Grant 2018J13SN088.

ABSTRACT With the ubiquitous deployment of wireless systems and pervasive availability of smart devices, indoor localization is empowering numerous location-based services. With the established radio maps, WiFi fingerprinting has become one of the most practical approaches to localize mobile users. However, most fingerprint-based localization algorithms are computation-intensive, with heavy dependence on both offline training phase and online localization phase. In this paper, we propose CNNLoc, a Convolutional Neural Network (CNN) based indoor localization system with WiFi fingerprints for multi-building and multi-floor localization. Specifically, we devise a novel classification model and a novel positioning model by combining a Stacked Auto-Encoder (SAE) with a one-dimensional CNN. The SAE is utilized to precisely extract key features from sparse Received Signal Strength (RSS) data while the CNN is trained to effectively achieve high accuracy in the positioning phase. We evaluate the proposed system on the UJIIndoorLoc dataset and Tampere dataset and compare the performance with several state-of-the-art methods. Moreover, we further propose a newly collected WiFi fingerprinting dataset UTSIndoorLoc and test the positioning model of CNNLoc on it. The results show CNNLoc outperforms the existing solutions with 100% and 95% success rates on building-level localization and floor-level localization, respectively.

INDEX TERMS Indoor localization, deep learning, convolutional neural network, WiFi fingerprinting.

I. INTRODUCTION

Indoor Location-Based Services (ILBSs) have become an essential component for various indoor applications, such as location based wireless advertising, information retrieval and pedestrian navigation [1]. With an explosive demand on high-accuracy and low-cost localization, indoor positioning has attracted a lot of interests from industrial community to research literature. Indeed, knowing the accurate indoor position of a mobile user is particularly useful to a variety of applications in multi-building and multi-floor environment [2]. Hence, various indoor localization techniques have been proposed using different types of modalities, including WiFi, visible light, acoustic, cellular network and their combinations [3]. The majority of localization techniques utilize Received Signal Strength (RSS) from Wireless Access Points (WAPs) to deduce the locations of mobile users with the pre-constructed fingerprint database.

In general, there are two phases in fingerprinting localization, *i.e.*, the offline phase (training phase) and online

The associate editor coordinating the review of this article and approving it for publication was Guan Gui.

phase (positioning phase). In the offline phase, the fingerprint dataset (*i.e.*, radio map) is constructed by collecting RSS fingerprinting data at pre-known reference points of interested areas. In this way, the RSS fingerprints are labelled with the location information for training and matching purposes [4]. In the online phase, a mobile user can simply send a query to the system with his current RSS measurement. The localization system will preprocess the data with different localization techniques and estimate the user's current location based on fingerprint database. Finally, the localization results will be sent back to the requester [2].

Accordingly, literature studies confront with two major issues in fingerprint-based localization, *i.e.*, dataset construction and localization accuracy. In addressing the first issue, numerous recent studies embrace crowdsourcing-based techniques [4] to empower automatical fingerprint collection, avoiding labour-intensive site surveys. Meanwhile, despite the efforts to improve the localization accuracy, most existing works are based on unique environmental settings [5], so the experiments are unrepeatable. Even worse, as some localization algorithms are tested with personal dataset, it is very hard to compare the localization results of different techniques [6]. Therefore, some open-source RSS fingerprint databases are constructed and released as standard datasets for indoor localization, including UJIIndoorLoc [7], Tampere dataset [4] and IPIN dataset [8].

A key challenge in locating indoor target based on a WiFi fingerprinting dataset is how to achieve high-accuracy and low-cost localization under the fluctuation of signal and noise from multi-path effects. Traditional approaches, including probabilistic, K-nearest-neighbor (KNN) and Support Vector Machine (SVM), are computation-intensive and time-consuming with complex filtering and parameter tuning. Recently, with the rising of deep learning, Deep Neural Network (DNN) [9], [10] based localization approaches have been proposed. Nevertheless, the performance of DNN-based methods is still subject to the sufficiency of input training data. Since DNN is fully connected, its complexity in computation is directly related to the depth of the neural network (i.e., number of layers), which could directly affect the accuracy of localization results.

To address the above issues, we propose **CNNLoc**, a **CNN**-based indoor **Loc**alization System with WiFi fingerprinting. By leveraging the CNN, the convolution can replace the general matrix multiplication in neural networks thus reducing the computation complexity. Compared with the existing indoor localization approaches, the main contributions of this work are summarized as follows.

1) We propose an innovative deep-learning model for multi-building and multi-floor indoor localization. Our model leverages a SAE network to reduce the data dimension and combines a one-dimensional CNN to extract key features from fingerprinting dataset thus improving localization accuracy.
2) We present a novel algorithm to extract a verification set from the training dataset, resolving the instability problem brought by conventional random selection method, especially when the volume of dataset is small.
3) We evaluate CNNLoc on two open-source datasets and one self-collected dataset. The experimental results show the proposed CNNLoc outperforms the state-of-the-art methods with 100% and 95% success rates on building-level localization and floor-level localization, respectively.

The remainder of this paper is organized as follows. We review the related indoor localization work in Section II. In Section III, we present the system architecture of CNNLoc based on open-source dataset. In Section IV, we devise an innovative algorithm for extracting the verification set and introduce the data preprocessing and model pre-training process. In Section V, we optimize our model through experimental studies and compare CNNLoc with several benchmarks in localization accuracy. Finally, we conclude this work with discussion on future work in Section VI.

II. RELATED WORK

With rapid development of mobile devices and advance in wireless networking technologies, wireless sensing have become more pervasive and ubiquitous in people's daily life [11]. For outdoor application, wireless sensing can support tracking [12], path planning [13], route prediction [14] and even Ride-on-demand revenue prediction [15]. For indoor application, wireless sensing is mainly leveraged for indoor localization [16], [17] and event detection [18]. Motivated by the development of artificial intelligence, deep learning techniques have been widely applied to efficiently process wireless signals [19]–[22]. In this work, we are the first to combine Convolutional Neural Network (CNN) and Stacked Auto-Encoder (SAE) to perform indoor localization task with WiFi signals.

WiFi based Indoor localization generally falls into two main categories: device-free and device-based localization [23]. In device-free localization, the target entities do not carry any wireless devices and the deployed system captures the presence and motion of each target entity via their reflection on the WiFi signals. Meanwhile, in device-based localization (as shown in Fig. 1), mobile users are located with WiFi-enabled mobile devices through various measurements, including Time-of-Arrival (ToA), Angle-of-Arrival (AOA), Received Signal Strength (RSS) and Channel State Information (CSI) [2]. Based on the above, WiFi fingerprinting has become a major approach because of its applicability in various indoor environments. In particular, deep learning has recently become attractive for WiFi-fingerprinting based indoor localization. In [24], a deep CNN was proposed to train the weights of AOA images derived from CSI information. In [25], a DNN-based indoor fingerprinting scheme was employed to address the labour-intensive and time-consuming issues in achieving reliable and accurate localization. In [26], a deep-learning based indoor fingerprinting system is proposed by combining RSS with magnetic filed. In [27], a CSI-based device-free localization algorithm

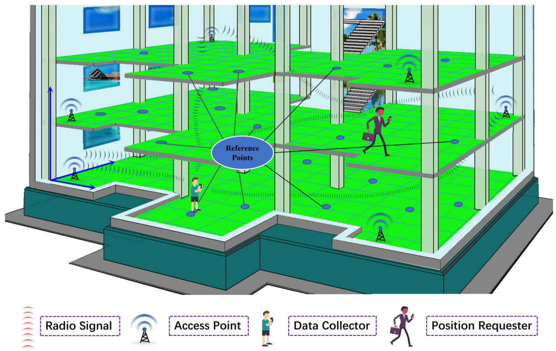


FIGURE 1. Fingerprinting based indoor localization scenario.

is proposed with deep neural networks. In [28], a CNN-based WiFi fingerprinting method was presented, and it outperformed the DNN-based methods. In this paper, we leverage deep learning by integrating a CNN with SAE for more accurate and efficient localization in a multi-building and multi-floor environment.

III. SYSTEM DESIGN

In this section, we first present the system architecture of CNNLoc, then we introduce the detailed model design of CNNLoc for offline phase and online phase, respectively.

A. SYSTEM ARCHITECTURE

The complete system architecture of CNNLoc is shown in Fig. 2, where the localization process consists of an offline training phase and an online positioning phase. In the offline phase, we adopt a fine-grain fingerprint database and then extract a verification set from the training dataset. Next, with the procedure of normalization, we can obtain the training data and testing data. In the online phase, CNNLoc will localize each location requester by matching received fingerprinting measurement and send back the localization position to the requester. In the following subsections, we first specify the data input and output of CNNLoc, then we present the floor classification model, building classification model and position estimation model in detail. In addition, we explain the cost function and earlystopping methods used in the CNNLoc model.

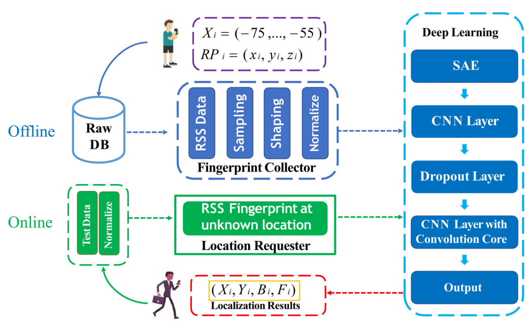


FIGURE 2. System architecture of CNNLoc.

B. INPUT AND OUTPUT SPECIFICATION

To specify the input and output structure of proposed model, we employ UJIIndoorLoc [7], the biggest open-access indoor localization database in literature. The UJIIndoorLoc database contains 21,049 fingerprint samples and covers 3 buildings with 4 to 5 floors. Accordingly, we specify the input of the CNNLoc model by $\vec{r} = (r_1, \ldots r_i, \ldots r_{520})$, where $\vec{r}$ is a 1-D vector with length of 520 and r_i denotes the RSS measurement on the i-th WAP. In the training phase, the RSS measurement $\vec{r}$ is combined with labels (x, y, f, b) to form a sample $(\vec{r}, x, y, f, b)$, where (x, y) is a sample reference point that is located in the building b on floor f. Hence, the output of the CNNLoc can be f, b or (x, y) when the trained model is a floor classification model, a building classification model or a position regression model. For training, verification and testing purposes, the UJIIndoorLoc dataset is divided into three parts, *i.e.*, a training set, a verification set and a testing set.

C. FLOOR CLASSIFICATION MODEL

In this subsection, we introduce the one-dimensional convolutional neural network (1D-CNN) classification model for multi-level indoor localization. The 1D-CNN model consists of self-encoding layers, a dropout layer, convolutional layers and an output layer. The architecture of the floor model is illustrated in Fig. 3 and we introduce the details of 1D-CNN model as follows.

1) SAE AND DROPOUT LAYER

For sparse data, the SAE network can effectively reduce the data dimension and still preserve the necessary feature information [25]. The SAE model is illustrated in Fig. 4, we leverage unsupervised learning to compress the input data to a feature layer. The output feature layer are further connected to CNN model for classification. To enable convolutional calculation, we further convert the output feature vector into a two-dimensional vector data. To avoid overfitting, we also add a Dropout layer between the SAE and CNN.

2) 1D-CNN AND OUTPUT LAYER

Due to the small variations of RSS fingerprints between adjacent floors, a major challenge in improving localization accuracy is how to distinguish users from adjacent floors. For example, the benchmark approach proposed in [7] can only achieve the success rate of floor classification at 89.92%. In order to obtain better prediction results, we connect a 1D-CNN model to the pre-trained SAE to further supervise the entire model. The 1D-CNN operates three convolutional operations, and the output layer is two-dimensional. Finally, the two-dimensional layers come from 1D-CNN are flatten into one-dimensional features and then connected to the final output layer for classification.

D. BUILDING CLASSIFICATION MODEL

As we aim to localize mobile users in multi-building scenarios, we present the basic building classification model

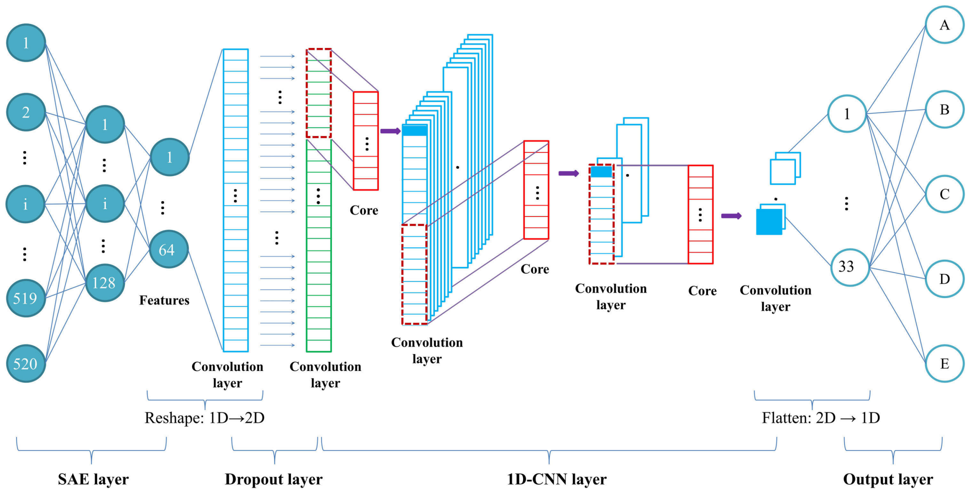


FIGURE 3. Floor model with SAE and CNN layers.

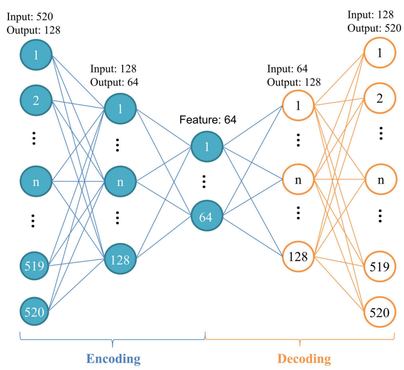


FIGURE 4. SAE Network.

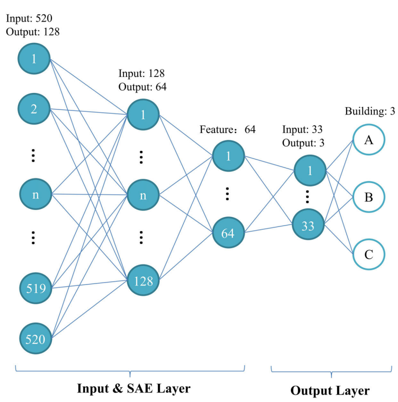


FIGURE 5. Building model with SAE.

in Fig. 5. Due to the relatively long distance between buildings, WiFi signals in each building vary with large difference. Hence, we adopt a fully connected neural network to solve the classification task in multi-building scenarios. While building classification is the first and most fundamental step for indoor localization, we use similar self-encoding layers in the floor classification model for building classification model, and these layers are further connected to a fully connected hidden layer (consisting of 33 neurons) and the final building classification layer.

E. POSITION ESTIMATION MODEL

Based on the two classification models, CNNLoc can determine the building-level and floor-level positions of mobile users. To further localize the absolute position of mobile users, we employ a position regression model to accurately measure the specific location of fingerprint collectors and requesters. The position estimation model is mainly based on the floor classification model, however, we also makes some changes in the structure of neural networks and we explain them as follows. First, we remove the dropout layer in the position estimation model. The dropout layer can solve the over-fitting problem in building-level and floor-level classification. However, for estimating the absolute position, a dropout layer could make CNNLoc miss some useful and important feature information. Second, we change the output layer as a pairwise result (X,Y). The floor-level localization is based on the classification results, however, the absolute position estimation is based on regression results. The two elements in pairwise positioning results (X,Y) represent the horizontal and vertical coordinates of a requester's position, where X represents the relative distance from the reference coordinate point to the westernmost side of the building and Y represents the relative position of the reference coordinate point to the southernmost side of the building. At last, we select the Rectified Linear Unit(ReLU) as the linear activation function for position estimation model, as the values on X and Y axis should be continuous.

F. COST FUNCTION

The cost function that we use is called quadratic cost function, which is defined by

$$C = \frac{1}{2n} \sum \|y(\vec{x}) - a^L(\vec{x})\|^2, \tag{1}$$

where n is the total number of verification samples, $y(\vec{x})$ is the corresponding desired output, L denotes the number of layers in the network and $a^L(\vec{x})$ is the vector of activations output from the network when $\vec{x}$ is the input. We adopt this cost function to calculate the degree of inconsistency between the localization results and the ground-truth. The smaller the value of the cost function is, the more accurate the corresponding localization result is.

G. EARLYSTOPPING STRATEGY

We further adopt the EarlyStopping strategy to monitor the performance of CNNLoc on verification set. During each training session, the trained model is evaluated by the verification set. For example, if the localization accuracies on the verification set shows no improvement within last p training sessions, the training process will be terminated. In the CNNLoc model, we use the patience parameter p to control the training process with the EarlyStopping. In this way, we can improve the training efficiency and localization performance simultaneously.

IV. DATA PRE-PROCESSING AND MODEL PRE-TRAINING

Before training the proposed deep learning model, we pre-process the data and pre-train the model in three folds. First, we devise a novel algorithm to extract a verification set from the training dataset. Second, we normalize the input data and restore it into the original representation, thereby improving the accuracy of the results. Third, the SAE model is pre-trained before we start to train the whole CNNLoc model.

A. EXTRACTING THE VERIFICATION SET

CNNLoc has several settings that can be leveraged to control the behavior of the learning process, and these settings are called *hyperparameters*. If learned on the training set, these hyperparameters would always choose the maximum possible model capacity, resulting in overfitting [29]. Therefore, we need to extract a verification set from the training dataset and use this verification set to update the hyperparameters accordingly. However, as most conventional methods extract the verification set by a random selection [25], the performance of trained model is not always stable. To solve this problem, we adopt the uniform sampling method [30] to extract the verification set from the training set evenly, thus improving the stability of localization results.

1) METHODOLOGY

The uniform sampling algorithm for UJIIndoorLoc dataset is presented in Algorithm 1. The input includes all-training dataset AD, side-length of cell-grid L and number of samples N in each cell-grid. The output is the training set T and the verification set V.

Algorithm 1 The Uniform Sampling Algorithm for Verification Set All-training

Input: dataset AD, Side-length of cell-grid L, The number of samples in each cell-grid N.

Output: Training set T, Verification set V.

for all $SD_i \in AD$ **do**
  Create a rectangle area (consists of multiple cell-grid with side-length L) to cover all the coordinates in the SD_i.
  $C \leftarrow$The set of center coordinate for all cell-grid.
  for all $G \in C$ **do**
    $I \leftarrow$A set of data in the SD_i covered by the cell-grid whose center is G.
    if I not empty **then**
      $NP \leftarrow$(N samples closest to G center in I)
      Delete NP from I
      $V.append(NP)$
      $T.append(I)$
    end if
  end for
end for
return T, V

First, all-training dataset AD is divided into several sub-datasets by building labels and floor labels. Second, for each sub-dataset SD_i, we create a rectangle area (consisting of multiple cell-grids) to cover all the reference points in SD_i. Within each rectangle area, each cell-grid has a center coordinate G. We further create a set C that contains all Gs in each SD_i. Third, for each coordinate center G in set C, we select a set of data I covered by the grid-cell in SD_i. Finally, if I is not empty, the closest N samples to the center point G in I will be selected as a part of verification set V. The remaining of I as a part of training set T.

As shown in Fig. 6, subgraph (a) shows the relative position of AD, and subgraph (b) is the relative position of the verification set V, which is extracted from AD. It is worth mentioning that this dataset collects multiple samples at each position, so the points shown in the figure are the results of the superposition, and the number of real points is more than the points displayed.

B. DATA NORMALIZATION

We further preprocess the UJIIndoorLoc dataset for training process of CNNLoc model. In this dataset, the value of input RSS data ranges from -104 dBm to 0 dBm, and we convert these RSS values into (0, 1) range by using Equation 2 below. For any WAP that is not detected in one measurement, its RSS value is marked as 100 dBm in the UJIIndoorLoc dataset, and we denote these RSS values as 0. In [31], it is indicated that different data representations of RSS fingerprints can influence the success rate and error

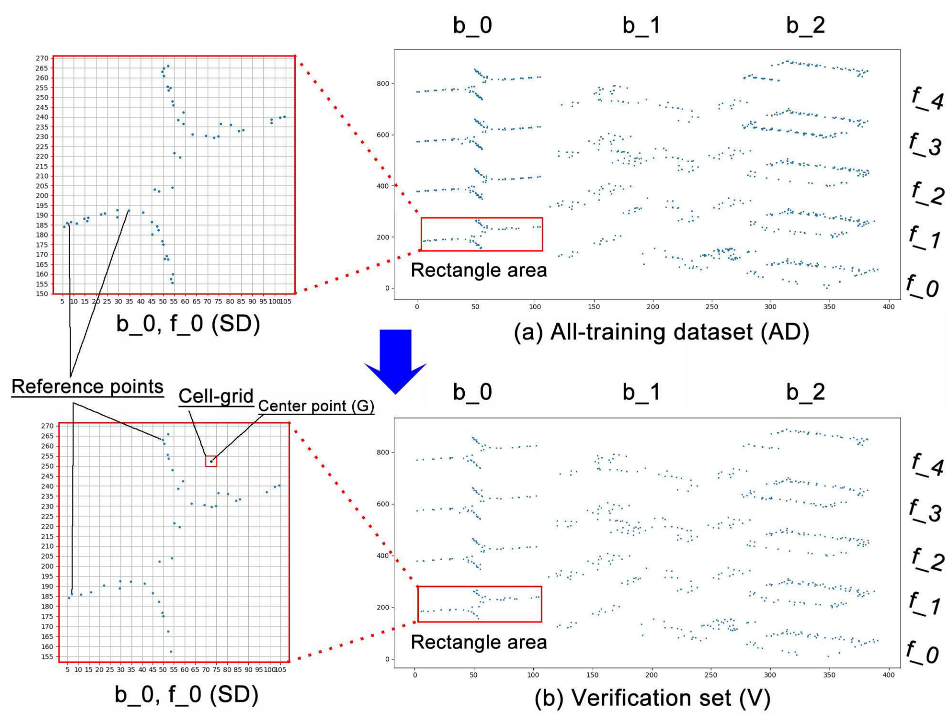


FIGURE 6. Extract the verification set from the training dataset.

rate during localization. Specifically, the authors compared the linear representation, exponential representation and the powed representation (Equation 2). As a result, the powered data tend to represent the RSS values with the best performance. Moreover, the powered data can better tame the fluctuation-existing RSS signals, so we use powered data for the representation of WiFi fingerprint in this work.

$$Powed = \begin{cases} 0, & RSS_i \text{ is None} \\ \left(\dfrac{RSS_i - min}{-min}\right)^{\beta}, & \text{otherwise} \end{cases} \tag{2}$$

where i is the WAP identifier, RSS_i is the actual intensity level provided by the i-th WAP, min is the lowest RSS value considering all of the fingerprints and WAPs of the database, and β is the mathematical constant e.

C. SAE MODEL PRE-TRAINING

As the input data for WiFi fingerprint positioning is very sparse, we import a SAE model before CNN layers to compress the dimension of the input data. In specific, we pretrain the SAE network [10] to obtain appropriate parameters (weights and biases) before we train the CNNLoc model. Ultimately, the whole CNNLoc model (consisting of encoder of SAE and CNN layers) will be further fine-turned.

V. PERFORMANCE EVALUATION

In this section, we evaluate the proposed CNNLoc by comparing its performance with the state-of-the-art approaches. Two public open-source datasets, UJIIndoorLoc [7] and Tampere [4], are employed for experimental studies. Moreover, we further propose a self-collected WiFi fingerprinting dataset, namely UTSIndoorLoc, to test the performance of CNNLoc. We implement the CNNLoc on a Computing Cluster with Quadro P5000 GPU, and the deep learning model is trained on Keras-2.2.2 (with Tensorflow-gpu-1.10.0) using Python-3.6.6.

TABLE 1. General parameter setting of the model.

Parameter	Values
SAE activation function	ReLU
SAE Optimizer	Adam (lr=0.001)
SAE loss	MSE
1D-CNN activation function	ReLU
1D-CNN Optimizer	Adam (lr=0.001)
1D-CNN loss	MSE
Output layer activation function	Softmax
Earlystopping parameter patience	3
Batch size	66

A. MODEL OPTIMIZATION

First, we explore how to optimize the CNNLoc model through experiments. The parameters used in the optimization experiments are shown in Table 1. The activation functions in both of the SAE and CNN models are the Rectified Linear Unit (ReLU), the optimizer is Adam (learning rate 0.001), and the loss function is Mean Squared Error (MSE). The output layer activation function is Softmax, the training batch is set to 66 and the patience parameter in earlystopping is set to 3.

1) CNN MODEL OPTIMIZATION

To make the comparison more reasonable, we evaluate the structure of a CNN model with the same SAE model by adopting the SAE(256-128-64) from [25], where the SAE contains three hidden layers of 256, 128 and 64 neurons. We vary the structure of CNN model by changing the number of layers from 1 to 3 with different filters and kernel sizes. For example, Conv(99-33,63-22) refers to a CNN model with two convolutional layers. The first layer has 99 output filters with a kernel size of 33 and the second layer has 63 output filters with a kernel size of 22. In Fig. 7, we show the performance of different CNN models and find that Conv(99-22,66-22,33-22) can achieve the best success rate of 95.14% in floor classification.

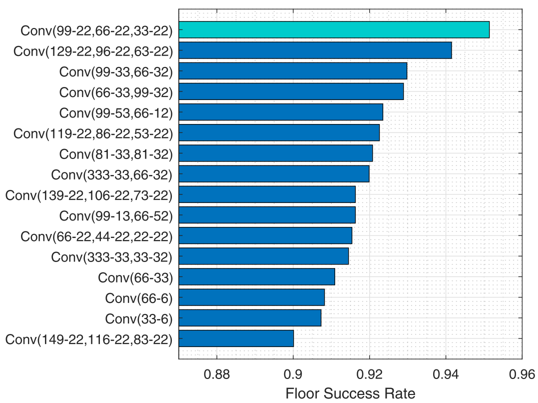


FIGURE 7. Effect of different CNN models on floor classification.

2) SAE MODEL OPTIMIZATION

Similarly, we evaluate the performance of different SAE models by using the same CNN model. Therefore, Conv(99-22,66-22,33-22) is connected to the SAE models with different combinations of hidden layers. The comparison results are shown in Fig. 8, where the best-performance SAE

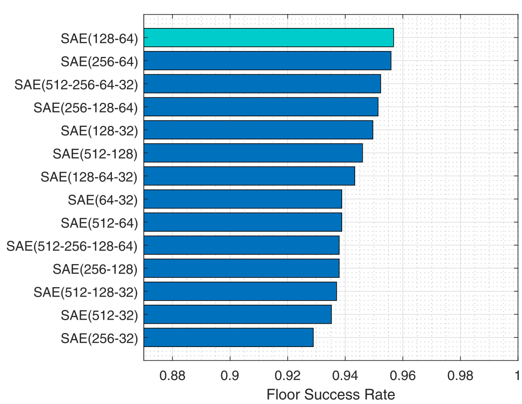


FIGURE 8. Effect of different SAE models on floor classification.

model is the two-layer SAE with 128 and 64 neurons in each layer (achieving a floor success rate of 95.68%).

3) DROPOUT LAYER OPTIMIZATION

To prevent convolutional neural networks from overfitting, we further adopt a dropout layer before the convolutional computation. According to a fraction *rate*, the dropout layer randomly sets the number of input units to 0 at each update during the training process. In this way, the sample data will not be trained too accurate to make the actual localization accuracy too low [28]. We evaluate CNNLoc with different dropout rates, using the optimized SAE model and CNN model obtained in previous optimization session.

We list the floor classification success rates under different dropout rates in Table 2. During each iteration of training, we test the CNNLoc model using the training set and verification set, respectively. It can be observed that the success rate reaches a peak of 0.9541 when the dropout rate equals 0.7. On this basis, we further evaluate the verification and training results without/with dropout layer (rate = 0.7). In Fig. 9, the blue curves represent the localization results using the verification set and the red curves represent the localization results using the training set. As the training time interval increases, the accuracy rates rise to over 95% for both training set and verification set. However, if CNNLoc does not have a dropout layer, the overfitting problem occurs, where the result of verification set is lower than the result of training set. In contrast, when we integrate a dropout layer to CNNLoc, the localization results of verification set are always better than training set and ultimately they converge to the same level. Hence, we adopt the dropout layer to address the overfitting problem in model training session.

TABLE 2. Results of different dropout rates.

Dropout rate	0.4	0.5	0.6	0.7	0.8	0.9
Success rate	0.948$\uparrow$	0.953$\uparrow$	0.952$\uparrow$	0.954$\uparrow$	0.951$\downarrow$	0.948$\downarrow$

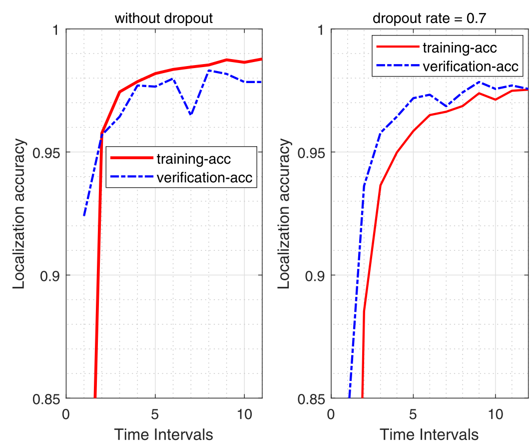


FIGURE 9. Floor success rate without/with dropout layer.

4) MODEL PARAMETER OPTIMIZATION

After the above optimization process, we have achieved the best model structure. Next, we consider to explore the best

TABLE 3. Effects of MSE Loss function and various optimizer with different learning rate on the performance of multi-building and multi-floor indoor localization.

Optimizer	Learning rate /%	Building hit rate /%	Floor hit rate /%	Positioning error /m
Adam	0.005	99.63	41.58	709.2
	0.001	99.72	41.22	709.2
	0.0005	99.72	94.95	12.64
	0.0001	99.54	95.22	17.88
Nadam	0.005	94.86	15.48	709.2
	0.001	**99.81**	15.48	709.2
	0.0005	99.72	95.31	14.28
	0.0001	99.72	95.22	15.07
RMSpro	0.005	99.18	3.510	709.2
	0.001	99.72	93.51	13.82
	0.0005	99.63	**95.31**	14.04
	0.0001	99.72	95.04	20.95
AdaMax	0.05	91.80	11.88	709.2
	0.01	99.63	77.58	13.46
	0.005	99.09	94.68	**11.78**
	0.001	99.72	94.59	21.30

TABLE 4. Effects of MSLE Loss function and various optimizer with different learning rate on the performance of multi-building and multi-floor indoor localization.

Optimizer	Learning rate /%	Building hit rate /%	Floor hit rate /%	Positioning error /m
Adam	0.005	**100.0**	27.54	709.2
	0.001	99.27	14.22	13.48
	0.0005	99.90	94.86	16.05
	0.0001	99.54	95.04	16.71
Nadam	0.005	27.63	3.510	709.2
	0.001	99.90	27.54	709.2
	0.0005	99.72	95.04	12.68
	0.0001	99.90	94.68	16.34
RMSpro	0.005	99.36	41.58	709.2
	0.001	99.27	**96.03**	18.10
	0.0005	99.72	94.50	**12.60**
	0.0001	99.63	95.58	21.87
AdaMax	0.05	99.90	11.88	709.2
	0.01	99.00	74.16	14.83
	0.005	99.27	92.07	12.87
	0.001	99.45	94.59	19.45

loss function, optimizer and corresponding learning rate for the training session. Here, the experimental parameters are all based on Table 1 and Table 5. The verification sets are extracted by using uniform sampling method proposed in Algorithm 1. The loss functions that we selected for the comparison experiments include Mean Squared Error (MSE) and Mean Squared Logarithmic Error (MSLE). We vary the learning rate in training session with a number of optimizers, including Adam, Nadam, RMSprop and AdaMax [32], *etc.*. The experimental results based on MSE and MSLE are presented in Table 3 and Table 4, respectively. We can observe from the experimental results that CNNLoc achieves differential localization results with different combinations of optimizer and learning rates. Most optimizers can achieve high building-level hit rate over 99%, where the highest building hit rate is 100% when the learning rate of Adam optimizer is 0.005. Meanwhile, the floor-level localization results are more fluctuant, with AdaMax achieving 96.03 on learning rate 0.001. With MSE loss function and MSLE loss, CNNLoc can achieve absolute positioning with errors down to 11.78m and 12.60m, with 94.68% and 94.5% floor hit rates, respectively. Note that, the lowest positioning in the state-of-the-art study [10] is 9.29m, but with 91.27% floor hit rate. Meanwhile, CNNLoc takes floor hit rate as a priority so it can enhance the final localization success rate. Based on Table 3 and Table 4, RMSprop optimizer has better performance in building-level and floor-level localization, while AdaMax achieves lower position estimation error.

B. EVALUATION ON DIFFERENT VERIFICATION SETS

To validate the proposed Algorithm 1, we further evaluate the impact of different verification sets. In this simulation, the side-length of cell-grid L is set to 3, and the number of samples in each cell-grid N is set to 5. The parameters set for CNNLoc are shown in Table 1 and Table 5. Here, we use the same pre-trained model but test its performance with two different verification sets, *i.e.*, a randomly selected set and a uniformly selected set.

TABLE 5. Parameter values for model.

Parameter	Values
SAE hidden layers	(128-64-128)
1D-CNN hidden layers	(99-22, 66-22, 33-22)
SAE/1D-CNN	Adam (lr=0.0001)
Dropout rate	0.7
Max training iterations	40

We evaluate the performance of CNNLoc for 4 times with different verification sets and the results are presented in Table 6. It is easy to observe that the testing results on

TABLE 6. Comparison of results on different methods on sampling verification sets.

Data type	1st /%	2nd /%	3rd /%	4th /%	Average /%
uniformed	95.41	94.69	95.41	95.32	95.21
random	94.96	93.97	95.14	92.08	94.04

verification sets selected by Algorithm 1 are more stable, with an average success rate of 95.21% in localization.

C. EXPERIMENTS ON UJIINDOORLOC DATASET

In this subsection, we evaluate the performance of CNNLoc model by comparing it with the state-of-the-arts on the UJIIndoorLoc dataset. We apply the uniform extraction algorithm on the UJIIndoorLoc dataset and obtain 2,198 samples for the verification set, 17,739 samples for the training set, and 1,111 samples for the testing set. The benchmark methods include MOSAIC [33], 1-KNN [7], 13-KNN [31], DNN [25], 2D-CNN [28] and scalable DNN [10]. We illustrate the comparison results in Fig. 10 in terms of building success rate and floor success rate. CNNLoc and most of the benchmark approaches achieve 100% success rates in building localization, showing that the proposed SAE can effectively handle building classification. Meanwhile, CNNLoc outperforms other benchmarks with a highest floor localization rate of 96.03%.

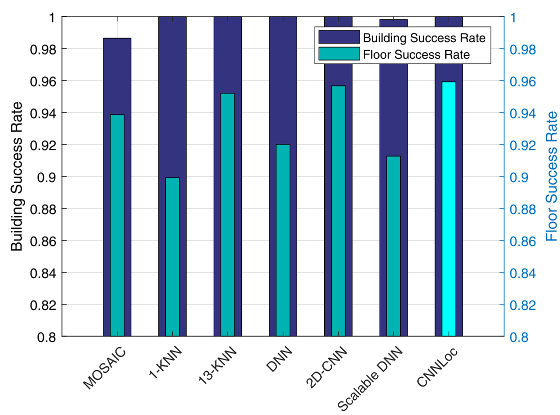


FIGURE 10. Localization result comparison on UJIIndoorLoc Dataset.

D. EXPERIMENTS ON TAMPERE DATASET

To test the scalability of CNNLoc, we further evaluate its performance with a newly released WiFi RSS fingerprint database, *i.e.*, Tampere dataset [4]. We compare the key features of the UJIIndoorLoc dataset and Tampere dataset in Table 7. The number of test samples is a lot more than that of training samples in the Tampere dataset.

TABLE 7. Feature comparison of two datasets.

Features	UJIIndoorLoc	Tampere
Number of training samples	19,938	697
Number of test samples	1,111	3,951
Number of WAPs	520	992
Default value of missed RSS	100	100
Floor representation	floor number	floor height

In Table 7, the number of training samples of Tampere dataset is less than the number of its test samples. That was arranged on purpose by the collector of the dataset for testing the comprehensive performance of our methods. Because the Tampere dataset contains 992 WAPs, CNNLoc has an input layer with a length of 992 in the SAE model. As the Tampere dataset uses floor height as the floor representation instead of the floor number, we preprocess the dataset using the z-coordinates to represent the floor levels. The benchmarks include the coverage area-based algorithm, rank-based algorithm, RTLS@UM system, UJI kNN algorithm, RSS clustering algorithm, log-Gaussian probabilistic algorithm and weighted centroid algorithm [4]. The experimental results are shown in Fig. 11. CNNLoc achieves an average floor hit rate of 94.22% and outperforms the state-of-the-art methods by 2.54% to 12.34%. Although the result just surpasses 2.54% compare to UJI kNN. However, this is the result of CNNLoc didn't go through a heavy parameter adjustment process. Therefore, CNNLoc demonstrates its scalability in multi-building and multi-floor localization.

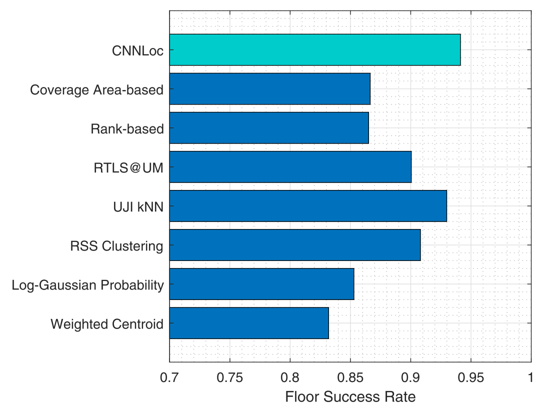


FIGURE 11. Localization result comparison on Tampere Dataset.

E. EXPERIMENTS ON UTSINDOORLOC DATASET

To further explore the scalability and capability of CNNLoc, we propose a new WiFi fingerprinting dataset, namely UTSIndoorLoc. In this subsection, we first introduce the details of UTSIndoorLoc dataset and then evaluate the CNNLoc on it.

1) INTRODUCTION OF UTSIndoorLoc DATASET

The UTSIndoorLoc dataset was collected in the FEIT Building at University of Technology Sydney (UTS), which is shown in Fig. 12. This building has 18 levels, including 4 basement levels. Since the bottom and top floors are not public active areas, UTSIndoorLoc dataset covers 16 floors of this building.

The main features of the UTSIndoorLoc dataset are as follows. 1) It covers an area of approximately 44,000 square meters and contains WiFi fingerprinting data for 16 floors. 2) The total number of sample points at different locations is 1840. 3) In a total of 9494 sample data, 9107 samples are used for the training session and the rest 387 are used for the testing session. 4) There are totaly 589 different Wireless Access Points (WAPs) included in the dataset.

FIGURE 12. Appearance of FEIT Building at UTS.

2) TEST RESULTS OF CNNLoc ON UTSIndoorLoc

Table 8 shows the benchmark results of CNNLoc on UTSIndoorLoc dataset compared with other two open dataset. On UTSIndoorLoc dataset, CNNLoc can achieve an average floor hit rate of 94.57%, while it achieves 96.03% and 94.22% average floor hit rate on UJIIndoorLoc and Tampere, respectively. Moreover, CNNLoc has better positioning error under UTSIndoorLoc dataset, compared with 11.78m on UJIIndoorLoc dataset. As a result, CNNLoc achieves a balanced trade-off with 94.22% floor hit rate and 10.88m positioning error.

TABLE 8. The result of UTSIndoorLoc running on CNNLoc.

Dataset	UTSIndoorLoc	UJIIndoorLoc	Tampere
Floor hit rate	94.57%	96.03%	94.22%
Positioning error	7.60m	11.78m	10.88m

VI. CONCLUSION

In this study, we have proposed CNNLoc, a deep-learning framework for multi-building and multi-floor localization with WiFi fingerprinting. By combining the SAE with a 1D-CNN model, CNNLoc can precisely extract key features from sparse WiFi fingerprints and achieve a high localization accuracy. We have evaluated CNNLoc on three open-source datasets (*i.e.*, UJIIndoorLoc, Tampere and UTSIndoorLoc). The experimental results have demonstrated the superiority of CNNLoc, as it has achieved the highest success rates (96.03% and 94.22% on UJIIndoorLoc and Tampere, respectively) for multi-building and multi-floor localization compared with the state-of-the-art approaches. On the newly proposed dataset UTSIndoorLoc, CNNLoc also has a significant localization performance, with the floor classification hit rate of 94.57% and an average positioning error of 7.60m.

ACKNOWLEDGMENT

The preliminary version of this article was published in the 16th IEEE International Conference on Ubiquitous Intelligence and Computing (UIC 2019), Leicester, U.K.

REFERENCES

[1] A. Basiri, E. S. Lohan, T. Moore, A. Winstanley, P. Peltola, C. Hill, P. Amirian, and P. F. e Silva, "Indoor location based services challenges, requirements and usability of current solutions," *Comput. Sci. Rev.*, vol. 24, pp. 1–12, May 2017.

[2] C. Laoudias, A. Moreira, S. Kim, S. Lee, L. Wirola, and C. Fischione, "A survey of enabling technologies for network localization, tracking, and navigation," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3607–3644, 4th Quart., 2018.

[3] A. Yassin, Y. Nasser, M. Awad, A. Al-Dubai, R. Liu, C. Yuen, R. Raulefs, and E. Aboutanios, "Recent advances in indoor localization: A survey on theoretical approaches and applications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1327–1346, 2nd Quart., 2016.

[4] E. S. Lohan, J. Torres-Sospedra, H. Leppäkoski, P. Richter, Z. Peng, and J. Huerta, "Wi-Fi crowdsourced fingerprinting dataset for indoor positioning," *Data*, vol. 2, no. 4, p. 32, Oct. 2017.

[5] S. He and S.-H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 466–490, Jan. 2015.

[6] Y. Wen, X. Tian, X. Wang, and S. Lu, "Fundamental limits of RSS fingerprinting based indoor localization," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr./May 2015, pp. 2479–2487.

[7] J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, J. P. Avariento, T. J. Arnau, M. Benedito-Bordonau, and J. Huerta, "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2014, pp. 261–270.

[8] R. Montoliu, E. Sansano, J. Torres-Sospedra, and O. Belmonte, "IndoorLoc platform: A public repository for comparing and evaluating indoor positioning systems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2017, pp. 1–8.

[9] W. Zhang, K. Liu, W. Zhang, Y. Zhang, and J. Gu, "Deep neural networks for wireless localization in indoor and outdoor environments," *Neurocomputing*, vol. 194, pp. 279–287, Jun. 2016.

[10] K. S. Kim, S. Lee, and K. Huang, "A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting," *Big Data Anal.*, vol. 3, no. 1, p. 4, Dec. 2018.

[11] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, to be published.

[12] C. Chen, Y. Ding, X. Xie, S. Zhang, Z. Wang, and L. Feng, "TrajCompressor: An online map-matching-based trajectory compression framework leveraging vehicle heading direction and change," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[13] C. Chen, D. Zhang, X. Ma, B. Guo, L. Wang, Y. Wang, and E. Sha, "Crowddeliver: Planning city-wide package delivery paths leveraging the crowd of taxis," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1478–1496, Jun. 2017.

[14] C. Chen, S. Jiao, S. Zhang, W. Liu, L. Feng, and Y. Wang, "TripImputor: Real-time imputing taxi trip purpose leveraging multi-sourced urban data," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 10, pp. 3292–3304, Oct. 2018.

[15] S. Guo, C. Chen, J. Wang, Y. Liu, X. Ke, Z. Yu, D. Zhang, and D.-M. Chiu, "Rod-revenue: Seeking strategies analysis and revenue prediction in ride-on-demand service using multi-source urban data," *IEEE Trans. Mobile Comput.*, to be published.

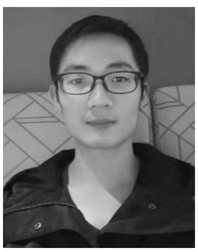
[16] F. Xiao, L. Chen, C. Sha, L. Sun, R. Wang, A. X. Liu, and F. Ahmed, "Noise tolerant localization for sensor networks," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1701–1714, Aug. 2018.

[17] F. Xiao, Z. Wang, N. Ye, R. Wang, and X.-Y. Li, "One more tag enables fine-grained RFID localization and tracking," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 161–174, Feb. 2018.

[18] L. Gong, Y. Zhao, X. Chaocan, Z. Li, C. Qian, and P. Yang, "Robust light-weight magnetic-based door event detection with smartphones," *IEEE Trans. Mobile Comput.*, to be published.

[19] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.

[20] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.
[21] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep-learning-based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3027–3032, Mar. 2019.
[22] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised learning-based fast beamforming design for downlink MIMO," *IEEE Access*, vol. 7, pp. 7599–7605, 2018.
[23] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Commun. Surveys Tuts.*, to be published.
[24] 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.
[25] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with WiFi fingerprints using deep learning," in *Proc. Int. Conf. Autom.* Springer, 2017, pp. 575–584.
[26] W. Zhang, R. Sengupta, J. Fodero, and X. Li, "DeepPositioning: Intelligent fusion of pervasive magnetic field and WiFi fingerprinting for smartphone indoor localization via deep learning," in *Proc. IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2017, pp. 7–13.
[27] R. Zhou, M. Hao, X. Lu, M. Tang, and Y. Fu, "Device-free localization based on CSI fingerprints and deep neural networks," in *Proc. 15th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2018, pp. 1–9.
[28] J.-W. Jang and S.-N. Hong, "Indoor localization with WiFi fingerprinting using convolutional neural network," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2018, pp. 753–758.
[29] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.
[30] H. Zheng, F. Yang, X. Tian, X. Gan, X. Wang, and S. Xiao, "Data gathering with compressive sensing in wireless sensor networks: A random walk based approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 35–44, Jan. 2015.
[31] J. Torres-Sospedra, R. Montoliu, S. Trilles, Ó. Belmonte, and J. Huerta, "Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems," *Expert Syst. Appl.*, vol. 42, no. 23, pp. 9263–9278, Dec. 2015.
[32] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*. [Online]. Available: https://arxiv.org/abs/1609.04747
[33] R. Berkvens, M. Weyn, and H. Peremans, "Localization performance quantification by conditional entropy," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2015, pp. 1–7.



XUDONG SONG received the B.S. degree in electrical engineering and automation from the Qingdao University of Science and Technology, Qingdao, China, in 2016, and the M.S. degree in control science and engineering from the Dalian University, Dalian, China, in 2019. He is currently a Visiting Scholar with the University of Technology Sydney. His research interests include the IoT, indoor localization, deep learning, and edge computing.

XIAOCHEN FAN (S'13) received the B.S. degree in computer science from the Beijing Institute of Technology, China, in 2013. He is currently pursuing the Ph.D. degree in computer science with the University of Technology Sydney (UTS), Australia. His research interests include mobile cloud computing, cyber security, the Internet of Things, and edge computing.

CHAOCAN XIANG received the B.S. and Ph.D. degrees in computer science and engineering from the Institute of Communication Engineering, PLA University of Science and Technology, China, in 2009 and 2014, respectively. From March 2017 to March 2018, he was a Visiting Scholar with the University of Michigan-Ann Arbor, USA. He is currently an Assistant Professor with the Chongqing University and Army Logistics University of PLA, China. He has published more than 30 papers in peer-reviewed journals and refereed conference, such as the IEEE TMC, IEEE TPDS, IEEE TVT and ACM Ubicomp. His current research interests include wireless sensor networks, crowd-sensing networks, and IOT.

QIANWEN YE received the B.S degree in electronic information engineering from the Hubei University of Technology, Wuhan, China, in 2013. She is currently pursuing the Ph.D. degree with the Faculty of Engineering and Information Technology, University of Technology Sydney, and the School of Information and Communication engineering, Beijing University of Posts and Communications. Her current research interests include the IoT, cooperation location, IFI, and channel models.

LEYU LIU received the B.S. degree in electronic information engineering from Anhui Xinhua University, Anhui, China, in 2014. She is currently pursuing the M.S. degree in control science and engineering with the Dalian University, Dalian, China. Her current research interests include the IoT, passive sensor networks, and indoor localization.

ZUMIN WANG received the M.S. degree in mechanical manufacturing and automation from the North University of China, in 2004, and the Ph.D. degree in physical electronics from the Chinese Academy of Sciences, in 2007. He has been a Professor with Dalian University, since 2014; and the Director of Dalian Key Laboratory of Smart Medical and Healthcare, since 2015. He was a Visiting Scholar with the University of Washington, from November 2016 to November 2017. His research interests include sensors and sensor application, wireless sensor networks, and smart city. He is also a Distinguished Member of CCF.

XIANGJIAN HE (M'99–SM'05) received the Ph.D. degree in computing sciences from the University of Technology Sydney, Australia, in 1999. Since 1999, he has been with the University of Technology Sydney. He is currently a Full Professor and the Director of the Computer Vision and Pattern Recognition Laboratory. He is also a Visiting Professor with Northwestern Polytechnical University.

NING YANG received the Ph.D. degree from Northwestern Polytechnical University (NWPU), China, in 2011, where she is currently an Associate Professor of control science and engineering with the College of Automation. Her research interests include computer vision, digital image processing, 3-D stereo structure, image and information fusion, image semantic, mobile computing, and search engine.

GENGFA FANG received the master's degree in telecommunications from Zhejiang University, in 2002, and the Ph.D. degree in wireless communications from the Institute of Computing Technology, Chinese Academy of Sciences, in 2007. From 2007 to 2009, he was a Researcher with the Canberra Research Laboratory, National ICT Australia on WCDMA Femtocell Project. In 2010, he was a Research Scientist with the Researching on Rural Broadband Access Project, CSIRO. From 2009 to 2015, he was with the Department of Engineering Macquarie University. In 2016, he moved to the University of Technology Sydney, where he is currently a Senior Lecturer.