IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 2, 2019, accepted August 21, 2019, date of publication August 27, 2019, date of current version September 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2937831

An Efficient Indoor Localization Method Based on the Long Short-Term Memory Recurrent Neuron Network

BO XU, XIAORONG ZHU, AND HONGBO ZHU

Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Corresponding author: Hongbo Zhu (zhuhb@njupt.edu.cn)

This work was supported in part by the Natural Science Foundation of China under Grant 61871237 and Grant 61871446, and in part by the Natural Science Foundation of the Higher Education Institutions of Jiangsu Province under Grant 16KJA510005.

ABSTRACT With the development of deep learning, fingerprints recognition based on neural networks is a widely used method in indoor localization. In this paper, we build a long short-term memory (LSTM) recurrent neuron network to make regression between fingerprints and locations in order to track the moving target. Simulations are in a BLE5.0 based environment and we use received signal strength indication (RSSI) as the element of fingerprints. Since the preparation of fingerprints is an inevitable and time-consuming process in the testing phase of LSTM, we propose two methods to improve the real-time performance of the localization without changing the structure of LSTM. A decentralized sorting algorithm is proposed to divide the received RSSI signals into multiple parts based on the MAC address of BLE5.0 equipment and use GPUs to sort each part. A complete fingerprint is a combination of these parts. Then, an optimization model aimed at maximum localization accuracy and minimal time used in the testing process of LSTM is proposed by changing the length of fingerprints. Many experiments simulated in different trajectories show that LSTM is more accurate in localization than many other neural networks. Further results demonstrate that using decentralized fingerprints preparation and finding an optimal fingerprint length can keep balance between the localization accuracy and real-time performance.

INDEX TERMS Indoor localization, BLE5.0, fingerprint, received signal strength indication, long short-term memory recurrent neuron network.

I. INTRODUCTION

Location-based Service (LBS) has received much attention in recent years. In the outdoor environment, Global Positioning System (GPS) is the most famous localization technology which has reached the accuracy of 5 meters [1]. But in the indoor environment, because of the deep shadowing effects [2] and higher precision requirements, indoor localization is a challenging problem.

Traditional localization methods are solving a multiple quadratic equations problem [3] in which RSSI [4], Time of Arrival (TOA) [5], Time Difference of Arrival (TDOA) [6] and Angle of Arrival (AOA) [7] are common features for computation.

With the development of deep learning, indoor localization based on fingerprints has become a popular method in many researches [8], [9]. Fingerprints can be seen as a combination of features in a form of sorted vectors. Fingerprints recognition is usually conducted into two phases: off-line phase and online phase [9]. In off-line phase, a location known mobile user (MU) collects features from anchors and sends them to the server. These vectors and locations are trained in server by neural networks. In the online phase, the MU repeats preparation process of fingerprints, and the server compares them with the previously trained model.

Various advanced fingerprints based localization methods have been proposed in recent papers. The existing innovation points can be divided into two categories. The first one is designing new hardware systems which provide some new features of the moving target. A sensor-based algorithm is addressed in [10], which uses the features of acceleration sensors, magnet sensors and gyroscopes from an inertial measurement unit (IMU) sensor of smart phone for position estimation. The IMU can be a form of

The associate editor coordinating the review of this article and approving it for publication was Jon Atli Benediktsson.

foot-mounted [11], which attaches the IMU with RFID equipments, and analyses the moving information to track the target. Maybe in the future, some new devices will assist localization, but there will be a long way to reduce the cost of these devices. The second one is designing some efficient algorithms to improve the accuracy and the positioning reliability of localization. In [12] the authors proposed a path-loss-based localization scheme and a dual-scanned localization scheme to overcome the fluctuation and delay of fingerprints. Collecting fingerprints based on multiple antennas is also a creative idea [13], which propose a feasible channel selection when the non-line-of-sight propagation happens. [14] derived a convolutional neural networks based algorithm to exclude some non-line-of-sight (NLoS) channels from anchors before localization. It is worth noting that the discussion about NLoS and LoS is a research focus in many indoor localization papers [15]–[17].

Most researches focuses on improving the accuracy of positioning rather than the real-time performance, however, the latter is also worth researching in the construction process of an actual system. Some related works [18], [19] have considered the model of cooperation in localization to improve the efficiency. But the preparation of fingerprints still takes a large amount of time because elements in the fingerprint usually come from different anchors and need to be sorted. Therefore, it is more practical to accelerate the sorting process or optimize the length of fingerprints to improve the system efficiency.

Motivated by that, in this paper, a mechanism is built to sort features from different anchors separately by GPUs according to the MAC address of anchors. Because the length of fingerprints affects the accuracy of localization, a utility function is proposed to keep balance between time consumption and localization accuracy.

In this paper, LSTM [20] is used as the basic network structure for localization. When the target is moving, LSTM performs better than many other neural networks because there exists temporal correlation of fingerprints. Features we choose from BLE5.0 are RSSI, which is a common feature in many researches [21], [22]. Our contributions can be summarized as follows:

1) We model an indoor localization system and generate fingerprints in different positions, then use LSTM to track a moving target.

2) We make decentralized sorting of RSSI by using a GPU based accelerated algorithm to overcome the time bottleneck caused by limit computation capabilities of the CPU.

3) We optimize the length of fingerprints in indoor localization.

The rest of paper is organized as follows. BLE5.0 based localization system is introduced in Section II. Then we propose a tracking-based LSTM algorithm to make localization in Section III. Next, we use a CPU-Accelerated sorting algorithm to prepare fingerprints and keep balance between computation speed and accuracy in Section IV. Finally, we give numerical results in SectionV and conclusion in Section VI.

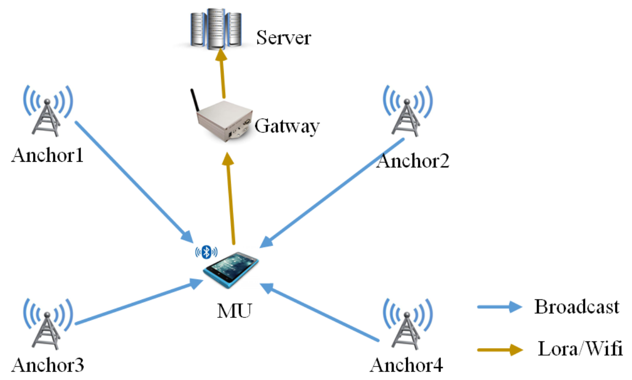


FIGURE 1. The architecture of BLE5.0 based indoor localization system.

II. INDOOR LOCALIZATION SYSTEM BASED ON BLE5.0

This section outlines the structure and the communication mechanism of the BLE5.0 based indoor localization system which uses RSSI as the element of fingerprints.

A. RSSI-BASED BROADCASTING-SCANNING SYSTEM

Similar to the RSSI-based indoor localization system mentioned in [23], in a BLE5.0 based indoor system, equipment can be split into two categories on the base of their functions. The first category includes nodes which assist the localization, and these nodes are usually called anchors, because their positions are fixed. The second category includes nodes need to be localized when the anchors are deployed, and these nodes are named as target nodes or MUs. Usually, anchors are homogenous while MUs can be various, but for ease of expression, in this paper, anchors and target nodes are all BLE5.0 based equipment.

According to the characteristics of the BLE5.0 based devices, the communication process can be divided into two phases, which are depicted in Fig. 1a. In the first phase, all anchors send their broadcast packages to MUs at a defined period. In the second phase, MUs scan these package messages and exact MAC addresses with corresponding RSSI values, then they will transmit these messages to the server by wireless communication technology such as WIFI or Lora. Usually, anchors and MUs are computationally restricted devices [14], so the processes of fingerprints preparation and localization computation are operated by the server. Centralized and decentralized preparations of fingerprints are shown in Fig. 2.

B. STRUCTURE OF RSSI-BASED FINGERPRINTS

In a specific indoor localization scenario, the number of anchors is N, and in a 3D environment, localization could be realized only just $N \in \{4, 5, 6, \ldots\}$ [14]. Assume that there is only one MU in this localization environment, the RSSI received from anchor i, $i \in \{1, 2, 3, \ldots, N\}$, can be written as $[RSSI_{i-1}, RSSI_{i-2}, \ldots, RSSI_{i-M_i}]$, where M_i is the number of RSSI from anchor i.

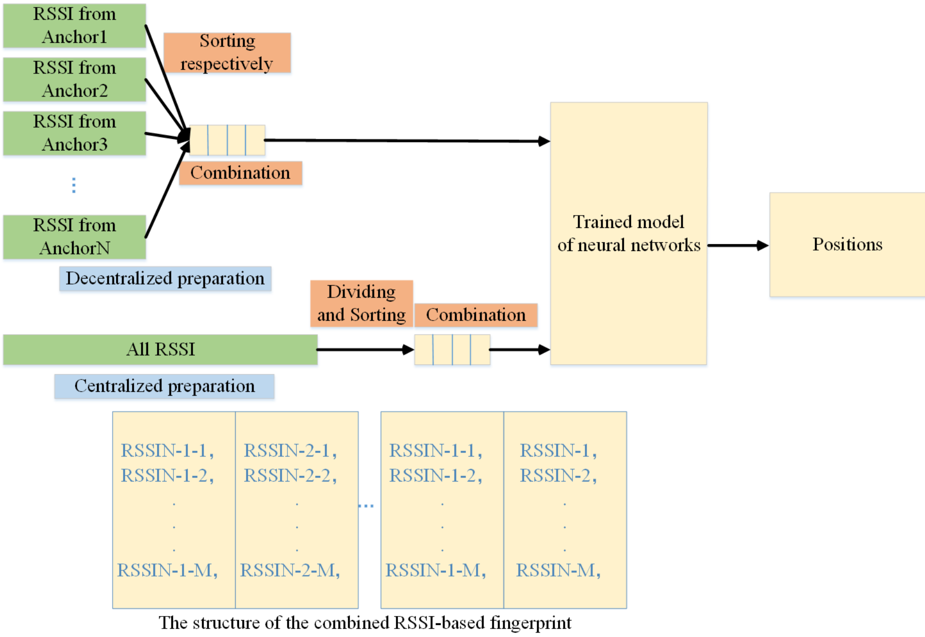


FIGURE 2. Algorithms under different fingerprint preparation methods.

The server converts all RSSI messages to fingerprints on the base of the MAC address i and the value of RSSI. The fingerprint in a certain location (x, y) is expressed as

$$\begin{aligned} RSSI_{x,y} = [&RSSI_{1-1}, RSSI_{1-2}, \ldots, RSSI_{1-M_1}, \\ &RSSI_{2-1}, RSSI_{2-2}, \ldots, RSSI_{2-M_2}, \\ &\ldots, \\ &RSSI_{N-1}, RSSI_{N-2}, \ldots, RSSI_{N-M_N}] \end{aligned} \quad (1)$$

where we assume that MU is able to scan every anchor in the positioning process because the number of anchors used in this system is lower than the load limit of the target node [24]. We also set that $M_1 = M_2 = \ldots = M_N = M$, which means the number of RSSI used in a fingerprint is equal for all of the anchors. This assumption is reasonable, because in a BLE5.0 based system, the broadcast intervals of anchors are set to be the same and the number of RSSI been perceived are nearly equal.

III. PROPOSED A LSTM BASED LOCALIZATION ALGORITHM FOR A MOVING OBJECT

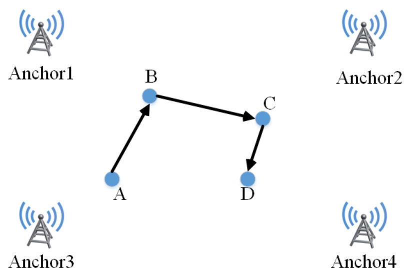
Usually, collecting fingerprints will take a lot of time for researchers, because a partition criterion which traverses almost every grid of the room is needed. But in the method we proposed, the process of collecting fingerprints doesn't need lots of grids. Here a brief introduction of this process is shown below and the sketch map is indicated in Fig. 3.

In proposed method, MU walks along a pre-designed trajectory which includes specific positions for training and when MU is moving, messages of RSSI and time-line are also received simultaneously by the server. RSSI will be transformed into fingerprints by the method in section III.

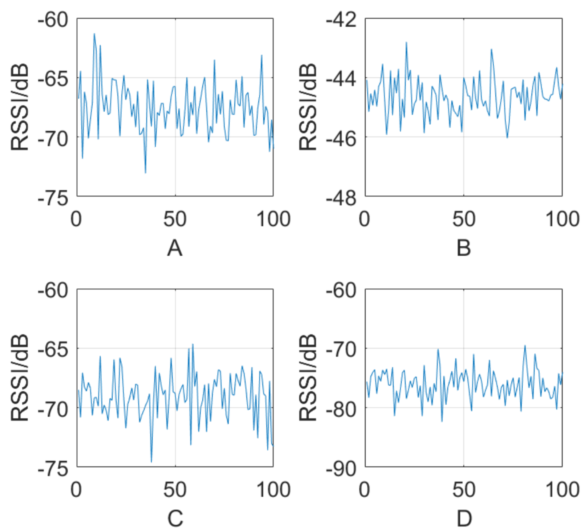
As shown in Fig. 3b, 100 RSSI data from Anchor 1 are listed in each position. In a real scenario, RSSI in position A, B, C, D couldn't represent the exact value corresponding positions, because when MU is moving, RSSI is fluctuant, and the measurement result of RSSI is an average value for a certain interval. To overcome this shortcoming, in this paper, we introduce variable τ which means the time interval between collection of fingerprint data. We can infer that the value of τ affects the length of the fingerprints when the system is working in a normal state. According to the relationship between position and time-line, the traverse route of MU is determined, so the passing position within the interval τ can be easily got. Usually, in a BLE5.0 based indoor localization, upload speed of RSSI is fast enough, so the position error will not deviate so much. Finally, the position mapping the RSSI collected in the interval τ can simply determine by choosing the midpoint of paths in the interval τ.

A. TEMPORAL CORRELATION OF RSSI-BASED FINGERPRINTS

Obviously, once taking the time interval τ into consideration, there maybe some temporal correction of fingerprints in this method. There are many ways to prove a sequence is temporally correlative [25]. Here we give an example to illustrate temporal correlation of RSSI in this method. When the MU is moving from B to C as shown in Fig. 3(a), we choose $\tau = 0.1s$ and record 10 group of RSSI from anchor1 in the moving process. For comparison and algorithm implementation, we sort these data and the result is displayed in Fig. 4. We see that there have intersection of fingerprints in some adjacent positions and the change of RSSI is not mutated. If we take a smaller τ, the difference between two adjacent



(a) MU moving from A to D



(b) RSSI perceived from Anchor1 in position A, B, C, D

FIGURE 3. Collected RSSI from anchor 1.

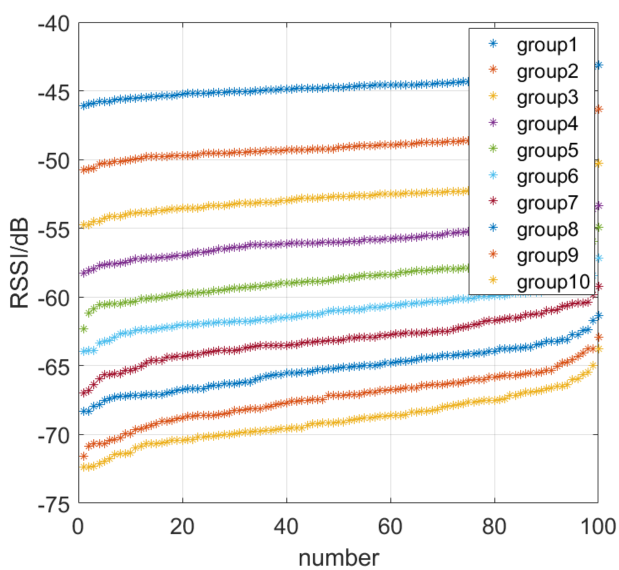


FIGURE 4. Sorted RSSI sequence in 10 successive groups.

sequence will be smaller. Under the analysis above, we can assume that the sequence of RSSI is temporally correlative, and based on this assumption, using LSTM may be an ideal method to make the fingerprint identification.

B. LSTM BASED LOCALIZATION

LSTM can be seen as an improved form of RNN (Recurrent Neural Networks) [26], whose main characteristic is used to connect priors information to the current task. However, traditional RNN has the problem of long-term dependence, which is not practical in many application scenarios. LSTM uses long-term information as the default behavior in practice so as to avoid this disadvantage. In this paper, our target is to implement dynamic positioning based on BLE5.0, so the output model of the LSTM in this paper is regression rather than classification [27].

The LSTM is organized in a layered structure. Different from other neural networks, LSTM has a state c (cell state) in each hidden layer. Let τ_t denote the time interval of the current fingerprint, and $RSSI_{x_{\tau_t},y_{\tau_t}}$, $h_{\tau_{t-1}}$ and c_{τ_t} are the current network input value. The outputs are h_{τ_t} and c_{τ_t}, and h_{τ_t} can be expressed as:

$$h_{\tau_t} = [\widetilde{x_{\tau_t}}, \widetilde{y_{\tau_t}}] \tag{2}$$

where $\widetilde{x_{\tau_t}}$ and $\widetilde{y_{\tau_t}}$ are coordinate outputs at the current time. The value of c_{τ_t} is determined by forget gate(FG) and input gate(IG) in LSTM, where FG save the information of $c_{\tau_{t-1}}$ to c_{τ_t}, and IG save the information of c_{τ_t} to h_{τ_t}. The calculation process in FG can be expressed as:

$$f_{\tau_t} = \sigma(W_f \cdot [h_{\tau_{t-1}}, RSSI_{x_{\tau_t},y_{\tau_t}}] + b_f) \tag{3}$$

where W_f is the weight matrix of FG, and b_f is the offset item. Function σ is the rectified linear unit(ReLU) [28].

The calculation process in IG can be expressed as:

$$g_{\tau_t} = \sigma(W_g \cdot [h_{\tau_{t-1}}, RSSI_{x_{\tau_t},y_{\tau_t}}] + b_g) \tag{4}$$

where W_g is the weight matrix of FG, and b_g is the offset item.

Then the cell state in forward process can be expressed as:

$$\tilde{c}_{\tau_t} = \tanh(W_c \cdot [h_{\tau_{t-1}}, RSSI_{x_{\tau_t},y_{\tau_t}}] + b_c) \tag{5}$$

$$c_{\tau_t} = f_{\tau_t} \cdot c_{\tau_{t-1}} + g_{\tau_t} \cdot \tilde{c}_{\tau_t} \tag{6}$$

where $\tilde{c}_{\tau_t}$ and c_{τ_t} are respectively the input and output of cell state in τ_t. By the iteration process, the long-term message in this network is built. Similar to other neural networks, there has a output function of FG and IG, which is shown in (7):

$$o_{\tau_t} = \sigma(W_o \cdot [h_{\tau_{t-1}}, RSSI_{x_{\tau_t},y_{\tau_t}}] + b_o) \tag{7}$$

where W_o and b_o are respectively the the weight matrix and the offset item which will be calculated in training process. The last step of forward process is calculating h_{τ_t} based on the cell state and network output:

$$h_{\tau_t} = o_t \cdot \tanh(c_t) \tag{8}$$

Fig. 5 shows the process of forward communication, which makes a rough evaluation of $[\widetilde{x_{\tau_t}}, \widetilde{y_{\tau_t}}]$.

Similar to RNN, the back propagation of LSTM also needs to calculate error term in time dimension and layer structure. The variables to be calculated include W_f, W_g, W_c, W_o, b_f, b_g, b_c and b_o. The error function is shown in (9):

$$E_{\tau_t} = \sqrt{(x_{\tau_t} - \widetilde{x_{\tau_t}})^2 + (y_{\tau_t} - \widetilde{y_{\tau_t}})^2} \tag{9}$$

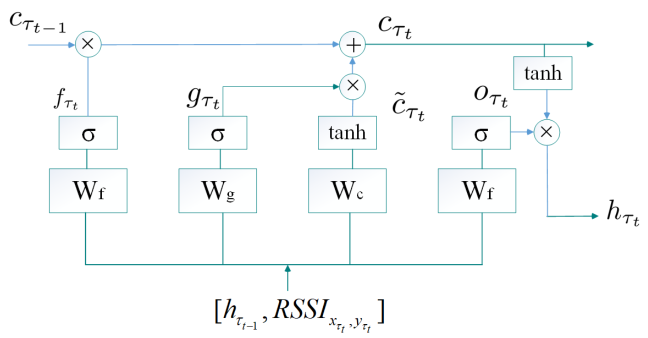


FIGURE 5. Froward propagation of LSTM.

Because the input of network is a combination of h_{τ_t} and $RSSI_{x_{\tau_t},y_{\tau_t}}$, W_f, W_g, W_c and W_o can be divided into W_{fx}, W_{fh}, W_{gx}, W_{gh}, W_{cx}, W_{ch}, W_{ox} and W_{oh}. Here we use the derivation of W_{oh} as an example to show the back propagation of these variables.

$$\begin{aligned}\frac{\partial E_{\tau_t}}{\partial W_{oh,\tau_t}} &= \frac{\partial E_{\tau_t}}{\partial net_{o,\tau_t}} \cdot \frac{\partial net_{o,\tau_t}}{\partial W_{oh,\tau_t}} \\ &= \delta^T_{o,\tau_t} \cdot h^T_{\tau_{t-1}} \\ &= \delta^T_{\tau_t} \cdot \tanh(c_{\tau_t}) \cdot o_{\tau_t} \cdot (1 - o_{\tau_t}) \cdot h^T_{\tau_{t-1}} \end{aligned} \quad (10)$$

where δ^T_{o,τ_t} is the error term in time dimension of layer o under the error function (9). $\delta^T_{\tau_t}$ is expressed in(11), which is the combination of weight matrix in(3), (4), (5), (7).

$$\delta^T_{\tau_t} = \prod_{\tau_0}^{\tau_{t-1}} (\delta^T_{o,\tau_t} W_{oh} + \delta^T_{f,\tau_t} W_{fh} + \delta^T_{g,\tau_t} W_{gh} + \delta^T_{\tilde{c},\tau_t} W_{ch}) \quad (11)$$

By setting iterations and initial weights, LSTM will build a regression model for fingerprints in each time interval. Since ReLU function avoids the problem of disappearing gradients, the computational process mentioned above is convergent. The pseudocode of LSTM based localization algorithm is shown in Algorithm 1. Tracking-based LSTM is efficient when MU is moving under a certain trajectory that is because RSSI collected in this process shows a more temporal correlation. Moreover, since many positions in the room have been covered by these trajectories in the training process, LSTM is feasible in indoor localization.

IV. IMPROVE THE EFFICIENCY OF LOCALIZATION

LSTM based localization algorithm succeeds in making more accurate localization because of the temporal correlation of RSSI sequences. However, the real-time performance in the process of localization is failed to be taken into account, which is a key factor in real applications. Two main factors which affect the operational efficiency are described as follows:

1) The structure of fingerprints shown in (1) is a sequencer procedure which requests the server in Fig. 2 to spend much time in dividing and sorting RSSI according to the MAC address. CPU in the server usually prepares fingerprints and makes localization in a centralized model, which means if too much time is cost in making fingerprints, the system will fail to make a real-time localization.

2) Length of fingerprints or number of RSSI exacted from each anchor is another factor which influences computation time. In section IV, if the number of RSSI is increased, the number of positions mapping the RSSI will be decreased. Using a longer fingerprint sequence for positioning will achieve a higher accuracy at the trained points, but increase the risk of over-fitting [29]. On the contrary, using a shorter length will cause the overlapping like in Fig. 4. As a result, finding an optimal length is also significant.

Algorithm 1 LSTM Based Localization

Input:
$RSSI_{x_{\tau_t},y_{\tau_t}}$ in each time slot, M
Output:
regression model based on LSTM

1: collect RSSI when MU has finished a certain trajectory, then repeat this step until the amount of data in this trajectory is enough.
change the track, and take more fingerprints so as to cover more points in the room.
2: generate fingerprints in each time slot like in Fig. 2.
3: initialize the structure of LSTM
4: train LSTM, and compute parameters such as W_f, W_g, W_c, W_o, b_f, b_g, b_c and b_o
5: collect some fingerprints for testing, then verify the trained model in step4
6: increase or decrease the number and structure of hidden layers in the network, then repeat step 5,6 to achieve a more accurate localization.

A. ALGORITHM TO MAKE DISTRIBUTED PREPARATION OF FINGERPRINT

In this paper, we accelerate the sorting process of RSSI by using a splitter-based approach [30], which is powerful in dealing with multi-sorting problem. The main characteristic of this method is distributing tasks in GPUs and adjusting the number of process to make full use of computing resources.

Different from the CPU based method, the decentralized generation process of fingerprints consists four steps under the HykSort [30] algorithm:

1) Each GPU receives RSSI from the server according to MAC addresses and sorts them.

2) According to the number of elements in fingerprints, each GPU makes random cutting of sorted RSSI arrays.

3) Transfer data segment (this step will be abandoned because of the structure of the fingerprint is fixed which means sorting sequence in each GPU does not need to interweave).

4) Each process merges sorted RSSI arrays.

Based on these procedures, even the time cost of data transfer between CPU and GPU is taken in consideration, GPUs have greater computation potential to make the splitter-based sorting method be more efficient.

B. ALGORITHM TO FIND OPTIMAL FINGERPRINTS LENGTH

In LSTM based localization algorithm, number of elements in fingerprints determines the number of positions been located. Total time consumption under the elements number n is described as C_n, which is the time used from RSSI collected in the server to the position been determined. C_n includes two parts: fingerprint preparation time CS_n and calculation time used in neural network CN_n:

$$C_n = CS_n + CN_n \tag{12}$$

In each time slot τ_t, time for dividing and sorting is C_{n,τ_t}, and the total time used for fingerprint preparation is given by:

$$CS_n = \sum_{t=1}^{t} C_{n,\tau_t} \tag{13}$$

Number of time interval is also determined by element number of RSSI in fingerprints, which also affects the training locations in the trajectory:

$$T = \frac{\tilde{M}}{NM} \tag{14}$$

where $\tilde{M}$ is the total number of RSSI been received by server. In Fig. 4, to simplify the model, we preset $\tau = 0.1s$ rather than the number of element, but in real experiments, M will be preset and the number of sampling positions is inversely related to M.

The total utility function is:

$$\Phi(M) = \lambda_c \sum_{t=1}^{T} C_{n,\tau_t} + \lambda_D E_{test} \tag{15}$$

where λ_C and λ_D are scalar weights, which can be preset based on system application target. E_{test} is the test error in testing phase, and the optimal problem can be formulated as follows:

$$\min_{M} \Phi(M) \tag{16}$$

$$\text{s.t.} \sum_{t=1}^{T} \tau_t = T, \tag{16a}$$

$$\forall M \in N^+, \tag{16b}$$

$$p_v^t | 0 < p_v^t \leq p_{max}, \quad v \in \mathcal{V}, \tag{16c}$$

where T is the total time used in RSSI collecting. It is worth noting that the operation of optimal process must after MU has finished a complete trajectory. In this paper, we usually give a smaller initial value of M and gradually increase it so that we can go through almost all the cases of the utility function. obviously, making sorting in GPU is not contradictory to find an optimal length of fingerprints, and in the testing phase, this two methods can be operated simultaneously. Combined with the LSTM based localization algorithm, the algorithm to

Algorithm 2 Accuracy-Efficiency Tradeoff Algorithm

Input:
$RSSI_{x_{\tau_t}, y_{\tau_t}}$ in each time slot, N, M, T
Output:
value of utility function
initialize M
LSTM-based
training model
preparation for fingerprints under the distributed sorting of RSSI
while $\Phi(M) > \xi$ **do**
  for $t = 1; t < T; t++$ **do**
    calculate $E_{test\tau_t}$, C_{n,τ_t};
  end for
  $\Phi(M) = \lambda_c \sum_{t=1}^{T} C_{n,\tau_t} + \lambda_D E_{test}$
  increase M
end while return 2

optimize fingerprints length is depicted in Algorithm 2, where ξ is the threshold value of the system target.

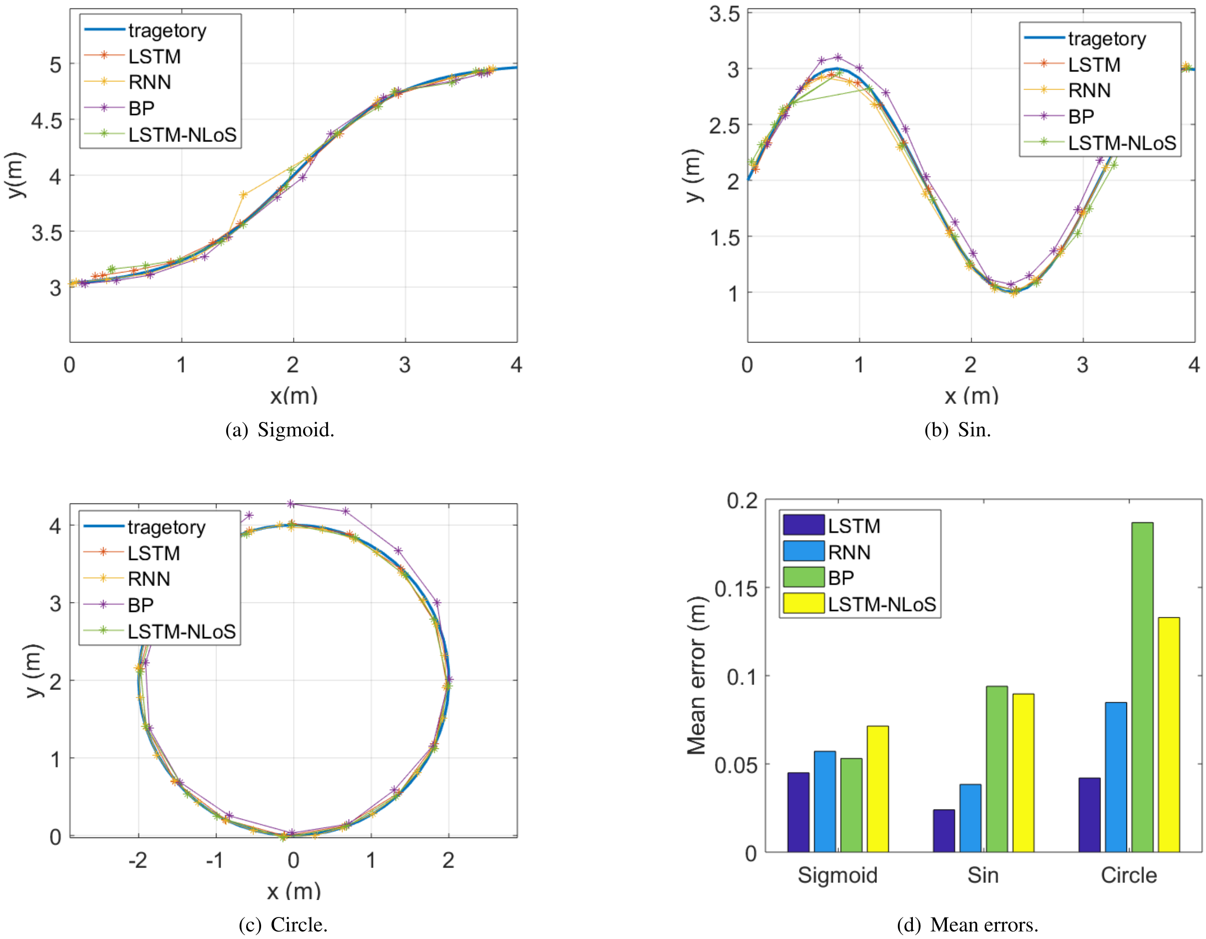
V. SIMULATED RESULTS

Our simulations run on MATLAB and Spyder [31] to make fingerprints preparation and build neural network respectively. We collect RSSI signals in each position of our four anchors based localization system so as to build the RSSI-fading model (an logarithm trend curve), and then we add gaussian noise to these signals which will be the data for our simulation. Since the capacity of RSSI detecting is only efficient in a small covered area, in this paper, localization area is limited in 4×4 m^2. To prove the reliability of the proposed algorithms, we also consider of the effect of non line-of-sight (NLoS) and line-of-sight (LoS) conditions in the RSSI fading model individually.

A. ACCURACY OF LOCALIZATION

We choose three common functions (Sigmoid, Sin, Circle) as the trajectory of the moving target in order to prove that the performance of LSTM in proposed method is better than other common neural networks. Each neural network has 1 hidden layers with 50 neural units. The input size of neural networks is the length of fingerprints and the output size is two-dimension. The length of fingerprints collected in 0.2s in each sample point is original 200. The rate of training data and testing data is 10:1, which are collected separately in each trajectory. Since the moving distance in each trajectory is different, the evaluation index to compare algorithms is the average distance error after MU has finished its work.

In Fig. 6, we first show that localization by using LSTM is more accurate than other neural networks in these trajectories. We see that RNN and BP has probability to departure the expected trajectory while LSTM is reliable under the hypothesis that most points in this trajectory have been well trained. In this paper, positions for testing



(a) Sigmoid.
(b) Sin.
(c) Circle.
(d) Mean errors.

FIGURE 6. **Comparison between LSTM, RNN, BP, and LSTM-NLoS in different trajectories.**

are random chosen, so there has no over-fitting in our method. We also compare the impacts of NLoS channel in indoor localization. Results show that localization accuracy is increased when the RSSI data are gathered from the case of NLoS.

In fig.7, we calculate the time costs in the testing processes. When the length of the fingerprint is the same, time cost for testing makes no difference in these neural networks. Testing process is based on the the weight matrix and offset item from training, so the Spyder can calculate them quickly.

Then we research the time costs in training processes as seen in Fig. 8. Although there has only 1 hidden layer for these networks, LSTM and RNN convergence faster than that in BP. We can infer that using LSTM can also save the waiting time in the training process. From what we have analysed, using LSTM as the instrument is an ideal choice for localization. It is worth noting that time cost we have researched in this section is only in neural network without fingerprint preparation, and in next section, we will investigate the influence of fingerprint length and time cost in preparation of fingerprints, which makes a difference in a real system.

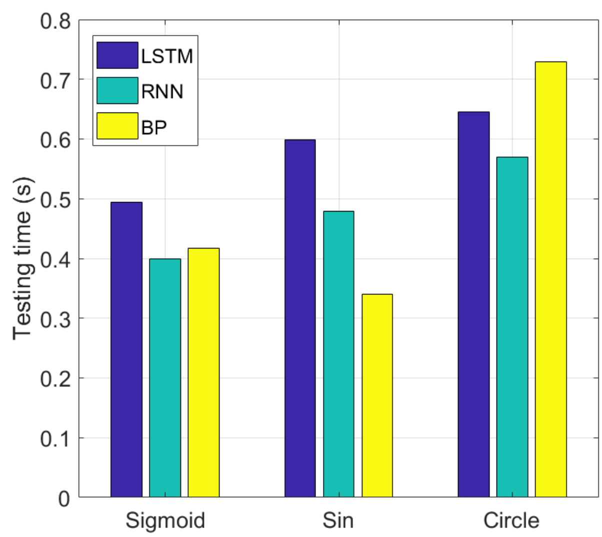


FIGURE 7. **Testing time used by LSTM, RNN, and BP.**

B. IMPROVE THE EFFICIENCY OF LOCALIZATION

In this section we change the length of the fingerprint into 50, 100, 200, 300 and record the time used for the preparation

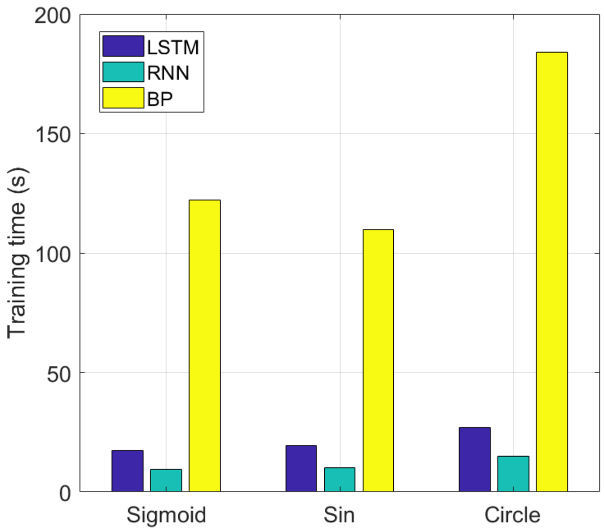


FIGURE 8. Training time used by LSTM, RNN, and BP.

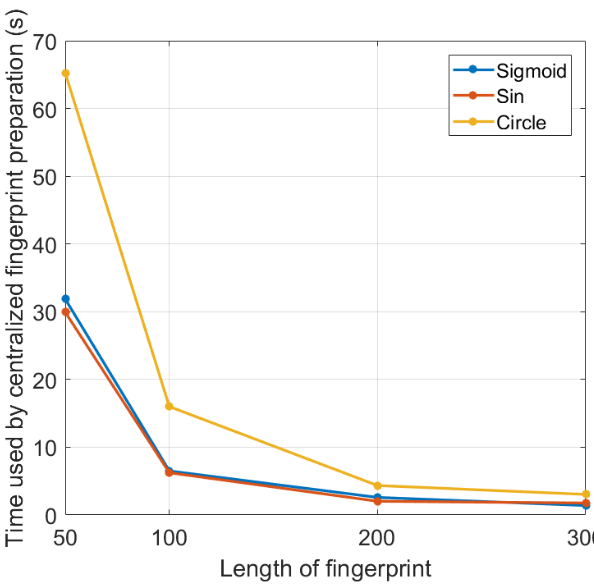


FIGURE 9. Time used by centralized fingerprint preparation.

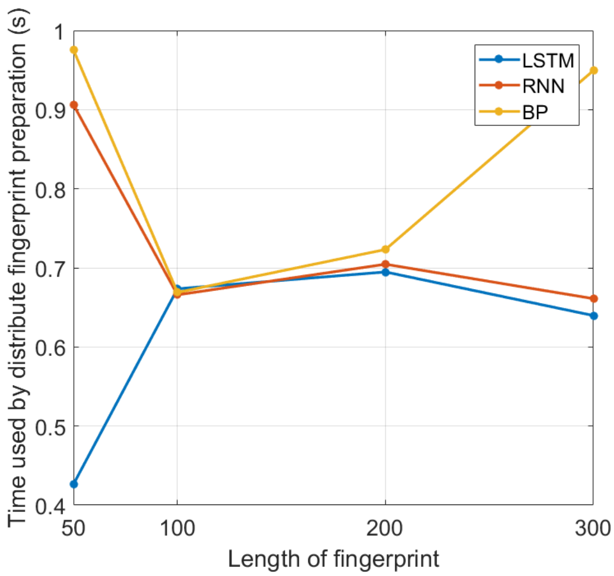


FIGURE 10. Time used by distributed fingerprint preparation.

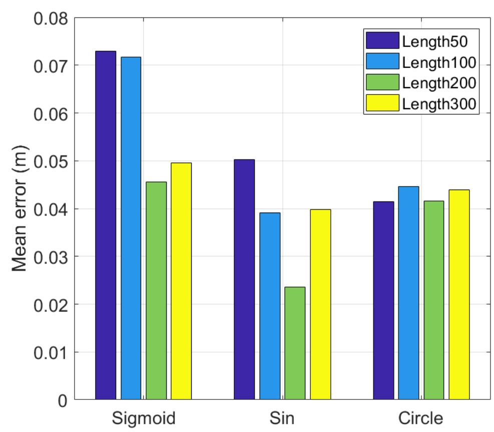


FIGURE 11. Mean errors in length of 50, 100, 200, 300.

of fingerprints. In the centralized preparation, when the server receives the RSSI from MU, it will choose to divide the RSSI based on the MAC address of anchors and sort each parts. However, testing fingerprints after they have been prepared is not realistic in a centralized model because of the limited computation capabilities and the loss of high RSSI stream. In a centralized model, we record the time used for fingerprint preparation after MU has finished its work. As shown in Fig. 9, if we choose a small length such as 50 and 100, there will be more sampled points in the trace and more circulation for the server to find these interval of points. We also find that when the length is increased to 200 and 300, the time for sorting is not a key factor for time cost, that is because many efficient sorting algorithms have been integrated in the server.

We also investigate the performance of decentralized preparation of fingerprints. Similar to the centralized method, we compare the time of fingerprints preparation by different length in Fig. 10.

In decentralized calculation, GPU sorts RSSI in different threads, and there has another thread to finish the testing process. We repeated the experiment many times and recorded the time cost. Simulation results depict that although there hardly have a specific rules to compare the time cost with increased length in distributed method, using decentralized algorithm is more efficient than centralized preparation.

The accuracy of localization is truly influenced by the length as shown in Fig. 11. In the training process of LSTM, if use a short length of fingerprints such as 50, there will have the possibility to cover more overlapping areas between adjacent fingerprints like in Fig. 4, which has negative impact on the LSTM. When the length increases to 200 or 300, although the training model is perfect, trained locations is so sparse, which means once the tested position is not trained, the system will not give an accurate localization.

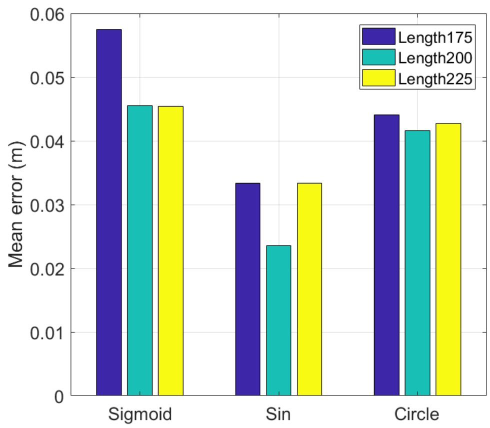


FIGURE 12. Mean errors in length of 175, 200, 225.

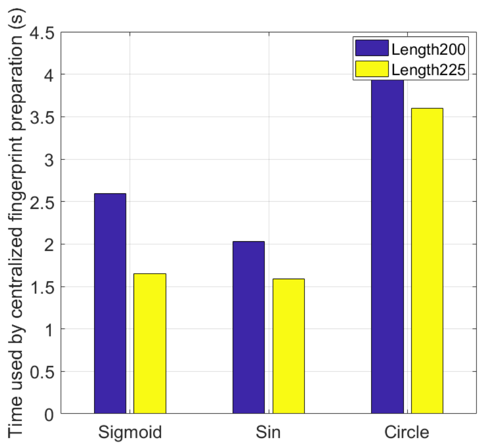


FIGURE 13. Centralized time used in length of 200 and 225.

As mentioned in algorithm2, the length of fingerprints we choose is increased. Since we find 200 is a better length in current schemes, we also investigate adjacent length such as 225 and 185 in Fig. 12.

C. JOINT OPTIMIZATION OF ACCURACY AND EFFICIENCY

Scalar weight λ_c and λ_D in (15) reflect the preference between accuracy and efficiency of localization. When using decentralized preparation of fingerprint, because there is no difference in time cost, length nearly 200 is an ideal strategy for us and $\lambda_c = 1$, $\lambda_D = 0$. When it comes to centralized localization, the curve of time cost will tend to flatten out with the increase of length. Since the irregularity of time in decentralized preparation, in this section, joint optimization is in centralized computation. In Fig. 13, we find that length of 255 will be optimal if add the value of λ_D because of its advantages in time cost. As a result, in a BLE5.0 based system, length of the fingerprints should be controlled according to the personal preference.

VI. CONCLUSION

This paper promotes a learning model which is practical in indoor localization. When the target is moving, there will have highly temporal correlation of RSSI, which means LSTM is a better method to make regression between locations and fingerprint. We consider time cost in fingerprints preparation is a main loss which will affect the real-time of the system, then we use GPUs to make fingerprints preparation according to the MAC address of anchors, and combine them into a whole fingerprint. Specially, our results show that the trend of localization accuracy is convex with the length of fingerprints and show that a decentralized GPUs based acceleration algorithm will improve the efficiency of a indoor localization system.

REFERENCES

[1] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Pers. Commun.*, vol. 7, no. 5, pp. 28–34, Oct. 2000.
[2] F. Lo Piccolo, "A new cooperative localization method for UMTS cellular networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov./Dec. 2008, pp. 1–5.
[3] H. Godrich, A. P. Petropulu, and H. V. Poor, "Power allocation strategies for target localization in distributed multiple-radar architectures," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3226–3240, Jul. 2011.
[4] O. Cheikhrouhou, G. M. Bhatti, and R. Alroobaea, "A hybrid DV-hop algorithm using RSSI for localization in large-scale wireless sensor networks," *Sensors*, vol. 18, no. 5, p. 1469, 2018.
[5] K. Batstone, M. Oskarsson, and K. Åström, "Robust time-of-arrival self calibration and indoor localization using Wi-Fi round-trip time measurements," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, May 2016, pp. 26–31.
[6] J. Tiemann and C. Wietfeld, "Scalable and precise multi-UAV indoor navigation using TDOA-based UWB localization," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2017, pp. 1–7.
[7] Y. Ma, B. Wang, S. Pei, Y. Zhang, S. Zhang, and J. Yu, "An indoor localization method based on AoA and PDOA using virtual stations in multipath and NLOS environments for passive UHF RFID," *IEEE Access*, vol. 6, pp. 31772–31782, 2018.
[8] K. Liu, H. Zhang, J. K.-Y. Ng, Y. Xia, L. Feng, V. C. S. Lee, and S. H. Son, "Toward low-overhead fingerprint-based indoor localization via transfer learning: Design, implementation, and evaluation," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 898–908, Mar. 2018.
[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] A. Poulose, O. S. Eyobu, and D. S. Han, "An indoor position-estimation algorithm using smartphone IMU sensor data," *IEEE Access*, vol. 7, pp. 11165–11177, 2019.
[11] F. Seco and A. R. Jiménez, "Smartphone-based cooperative indoor localization with RFID technology," *Sensors*, vol. 18, no. 1, p. 266, 2018.
[12] Z. Chen, H. Zou, H. Jiang, Q. Zhu, Y. C. Soh, and L. Xie, "Fusion of WiFi, smartphone sensors and landmarks using the Kalman filter for indoor localization," *Sensors*, vol. 15, no. 1, pp. 715–732, Jan. 2015.
[13] X. Guo and N. Ansari, "Localization by fusing a group of fingerprints via multiple antennas in indoor environment," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 9904–9915, Nov. 2017.
[14] K. Bregar and M. Mohorčič, "Improving indoor localization using convolutional neural networks on computationally restricted devices," *IEEE Access*, vol. 6, pp. 17429–17441, 2018.
[15] I. Guvenc and C.-C. Chong, "A survey on TOA based wireless localization and NLOS mitigation techniques," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 3, pp. 107–124, 3rd Quart., 2009.
[16] C. Morelli, M. Nicoli, V. Rampa, and U. Spagnolini, "Hidden Markov models for radio localization in mixed LOS/NLOS conditions," *IEEE Trans. Signal Process.*, vol. 55, no. 4, pp. 1525–1542, Apr. 2007.
[17] I. Guvenc, C.-C. Chong, and F. Watanabe, "Analysis of a linear least-squares localization technique in LOS and NLOS environments," in *Proc. IEEE 65th Veh. Technol. Conf.-VTC-Spring*, Apr. 2007, pp. 1886–1890.
[18] S. Chen, J. Zhang, and C. Xu, "Robust distributed cooperative localization with NLOS mitigation based on multiplicative convex model," *IEEE Access*, vol. 7, pp. 112907–112920, 2019.

[19] K. Guo, X. Li, and L. Xie, "Ultra-wideband and odometry-based cooperative relative localization with application to multi-UAV formation control," *IEEE Trans. Cybern.*, to be published.

[20] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[21] S. Yiu, M. Dashti, H. Claussen, and F. Perez-Cruz, "Wireless RSSI fingerprinting localization," *Signal Process.*, vol. 131, pp. 235–244, Feb. 2017.

[22] X. Zheng and J. Fu, "Study on PDR and RSSI based indoor localization algorithm," *Chin. J. Sci. Instrum.*, vol. 36, no. 5, pp. 1177–1185, 2015.

[23] K. D. Yüksel and B. U. Töreyin, "A deep learning and RSSI based approach for indoor positioning," in *Proc. 26th Signal Process. Commun. Appl. Conf. (SIU)*, May 2018, pp. 1–4.

[24] Y. Zhang, Z. Song, W. Wang, and Y. Tian, "An optimized algorithm for device discovery using BLE," in *Proc. 4th Int. Conf. Mach., Mater. Inf. Technol. Appl.* Paris, France: Atlantis Press, 2017, pp. 1–6.

[25] C.-Y. Ma, M.-H. Chen, Z. Kira, and G. AlRegib, "TS-LSTM and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition," *Signal Process., Image Commun.*, vol. 71, pp. 76–87, Feb. 2019.

[26] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2017.

[27] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. Int. Workshop Multiple Classifier Syst.* Berlin, Germany: Springer, 2000, pp. 1–15.

[28] Y. Li and Y. Yuan, "Convergence analysis of two-layer neural networks with ReLU activation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 597–607.

[29] J. Lever, M. Krzywinski, and N. Altman, "Points of significance: Model selection and overfitting," *Nature Methods*, vol. 13, no. 9, pp. 703–704, 2016.

[30] J. Liang, V. Makoviychuk, A. Handa, N. Chentanez, M. Macklin, and D. Fox, "GPU-accelerated robotic simulation for distributed reinforcement learning," 2018, *arXiv:1810.05762*. [Online]. Available: https://arxiv.org/abs/1810.05762

[31] H. P. Langtangen, *A Primer on Scientific Programming With Python*, vol. 6. Berlin, Germany: Springer, 2011, pp. 1–40.

BO XU received the B.S. degree from the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China, in 2018, where he is currently pursuing the Ph.D. degree (successive postgraduate and doctoral programs) in communication and information systems. His research interests include mobile edge computing, big data, and distributed learning. He received the First-Class Scholarship and the Special Freshman Scholarship from NJUPT, in 2018.

XIAORONG ZHU received the Ph.D. degree in wireless communications from Southeast University, Nanjing, China, in 2008. She held a Postdoctoral position at The Chinese University of Hong Kong, in 2008 and 2009, respectively. She is currently a Professor with the Nanjing University of Posts and Telecommunications. Her research interests include wireless networks, wireless access technology, and the Internet of Things.

HONGBO ZHU received the B.S. degree in communications engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 1982, and the Ph.D. degree in information and communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 1996. He is currently a Professor with the Nanjing University of Posts and Telecommunications. He is also the Head of the Coordination Innovative Center of IoT Technology and Application (Jiangsu), which is the first government authorized Coordination Innovative Center of IoT in China. He also serves as a referee or expert in multiple national organizations and committees. He has authored or coauthored over 200 technical papers published in various journals and conferences. He is currently leading a big group and multiple funds on the IoT and wireless communications with current focus on architecture and enabling technologies for the Internet of Things. His research interests include mobile communications, wireless communication theory, and electromagnetic compatibility.