Wavelet Feature Outdoor Fingerprint Localization Based on ResNet and Deep Convolution GAN

Yingke Lei 1,2,†, Da Li 1,2,*,†, Haichuan Zhang 1 and Xin Li 1

1 College of Electronic Engineering, National University of Defense Technology, Hefei 230037, China; ryxar@163.com (Y.L.); zhanghail258@163.com (H.Z.); gfkdlxin@163.com (X.L.)
2 Science and Technology on Communication Information Security Control Laboratory, Jiaxing 314000, China
* Correspondence: lida@nudt.edu.cn
† The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

Received: 4 August 2020; Accepted: 17 September 2020; Published: 22 September 2020

Abstract: Due to the explosive development of location-based services (LBS), localization has attracted significant research attention over the past decade. Among the associated techniques, wireless fingerprint positioning has garnered much interest due to its compatibility with existing hardware. At present, with the widespread deployment of long-term evolution (LTE) networks and the uniqueness of wireless information fingerprints, fingerprint positioning based on LTE networks is the mainstream method for outdoor positioning. However, in order to improve its accuracy, this method needs to collect enough data at a large number of reference points, which is a labor-intensive task. In this paper, experimental data are collected at different reference points and then converted into wavelet feature maps. Then, a Deep Convolutional Generative Adversarial Network (DCGAN) is leveraged to generate a symmetric fingerprint database. Localization is then carried out by the proposed Deep Residual Network (Resnet), which is capable of learning reliable features from a fingerprint image database. To further increase the robustness of the positioning system, a variety of data enhancement methods are used. Finally, we experimentally demonstrate that the generated symmetric fingerprint database and proposed Resnet reduce the manpower required for fingerprint database collection and improve the accuracy of the outdoor positioning system.

Keywords: fingerprint positioning; deep learning; outdoor positioning; wavelet transform

1. Introduction

The application of location-based services (LBS) has stimulated extensive research into wireless positioning in recent years [1]. Among positioning technologies, wireless fingerprint technology has proven to be effective, due to its simplicity and the practicality of its deployment [2]. Wireless fingerprint location relies on an existing network infrastructure, such as Wi-Fi (e.g., IEEE 802.11) or a cellular network (e.g., LTE), thereby avoiding hardware deployment costs and effort [3]. However, in order to establish a fingerprint database, the signal strength (RSS) or channel state information (CSI) received by surrounding access points must be measured at each reference point. Fingerprint-based positioning consists of two basic stages: (1) the offline stage of data collection (signals at different reference points are collected to establish a fingerprint database for the online training classification model stage) and (2) the online stage (receive the data online, where the received signals are compared to the fingerprint database and localized) [4]. In the training phase, the fingerprint database is constructed by collecting and pre-processing survey data related to the location of each reference point. In the online phase, the mobile device records real-time data and compares the received data with the database. The reference point closest to the received data is regarded as the location of the device.
However, there are three main difficulties that hinder its large-scale implementation. The first difficulty is how to extract reliable features, as wireless signals are susceptible to environmental effects, such as multipath, temperature and obstacles. In addition, the features of the signal may change over time, even in the same place. Due to the instability of wireless signals, many difficulties are involved in high-precision outdoor positioning. Due to simplicity and low hardware requirements, many existing indoor positioning systems use received signal strength (RSS) for fingerprint positioning; for example, the Horus system uses probabilistic methods to estimate the position of received RSS data [5]. However, due to signal instability in indoor environments, RSS data have high variability at a fixed location. This high variability can lead to serious positioning errors. In addition, RSS values are composed of relatively crude information, which cannot make full use of the collected signals [6].

How to construct an efficient and accurate fingerprint database is another hot issue in current fingerprint database research. A common implement for constructing a fingerprint database is to manually collect fingerprints at multiple known reference points throughout the area of interest. Obviously, this task is labor-intensive, tedious, and time-consuming, especially when measuring an outdoor environment. In order to reduce human effort, self-guided robots equipped with signal collection sensors can automatically wander and explore an experimental area to collect geotagged fingerprint data [7]. However, the use of robots is not economical nor practical. Fingerprint databases can also be automatically constructed by leveraging crowdsourcing and machine learning methods [8]. Crowdsourcing depends on the willingness of people to take part in fingerprint data collection [2]. This method uses measured random traces (radio frequency measurement and inertial sensor measurement) which are collected by volunteers holding smartphones while walking in a certain area during their daily activities. Note that these methods do not decrease the number of samples to be collected and still demand a lot of time to collect the fingerprint data.

The last issue is how to build a reliable positioning model. The learning ability of the initial shallow model is limited, and the positioning accuracy is not ideal. With the development of neural networks and computing capabilities, deep learning has become a dazzling star in various fields. Deep learning methods have outstanding modeling ability and have beat other most advanced methods in the ImageNet competition [9]. At present, fingerprint positioning methods based on deep learning mainly focus on the development of indoor positioning technologies.

In the widely used LTE signal, the wavelet feature map uses a wavelet transform to obtain more accurate information than with RSS. Therefore, this paper considers fingerprint outdoor location based on a wavelet feature map. By using the one-dimensional continuous wavelet transform, the collected LTE signal can be converted into a wavelet feature map. In order to reduce the workload while expanding the fingerprint database, we propose a Deep Convolutional Generative Adversarial Network (DCGAN)-based method to generate symmetric fingerprints at each reference point. Specifically, after completing signal collection, we convert the signal of each reference area into wavelet feature maps. As the initial database is constructed using the wavelet feature map of the reference area, it contains the location information of the entire reference area. Then, we use the proposed DCGAN model and the corresponding training algorithm to generate images similar to the original wavelet map. Finally, the generated wavelet feature maps are incorporated into the original fingerprint database, in order to form an extended fingerprint database. To design a positioning model with strong learning ability, a deep residual network (Resnet) is designed in this paper, which can learn reliable features from the extensive fingerprint database. Besides, to improve the robustness of the fingerprint positioning system, various data enhancement approaches are adopted.

The main contributions of this paper are as follows:

1. A fingerprint database is constructed by converting the collected signal data into wavelet feature maps, which visualizes the collected signal features.
2. To enhance the richness of the fingerprint database, a DCGAN-based model is used to generate symmetric wavelet feature maps of the sampling areas, which can help to reduce the labor required to collect data.
Considering the large amount of data in the fingerprint database, Resnet, with satisfactory learning ability, is proposed for positioning. Furthermore, several data enhancement methods are proposed to improve the robustness of the positioning system. To verify the superiority of the positioning system, extensive experiments are conducted in a real outdoor environment. Our experimental results show that the proposed positioning system can achieve better performance than other schemes.

The remainder of the paper is organized as follows: Section 2 introduces the background and overview of the fingerprint location method. Section 3 introduces the proposed positioning system. The experimental area is described first, and results and performance evaluation are then reported in Section 4. Section 5 presents the summary of this paper.

2. Related Works

As positioning algorithms based on geometric figures have shown unstable performance in complex indoor environments, fingerprint-based positioning has received extensive attention in recent years. These technologies can be divided into two categories, based on the underlying infrastructure: The first category includes UWB [10], RFID [11], and WSN [12], which rely on specially designed hardware for positioning. The other category, containing Wi-Fi and LTE networks, makes use of existing wireless networks for positioning. The second category can also be divided into range-based positioning techniques and fingerprint-based positioning techniques. For range-based positioning, high synchronization between the equipment and the base station is required, often leading to unsatisfactory positioning performance in many application scenarios [13]. Fingerprint-based positioning techniques are essentially a pattern-matching approach. Therefore, it determines the user equipment (UE) by comparing the signal fingerprint generated on the device with a pre-constructed fingerprint database.

Fingerprint positioning technology first needs to determine what kind of fingerprint to use. Many scholars have used RSS as fingerprint features for positioning. However, signal features like RSS are rougher and more susceptible to environmental noise, which can lead to a decrease in positioning performance.

After determining the signal fingerprint features, a signal fingerprint database needs to be constructed. In order to reduce the cost of data collection, several methods have been proposed [14,15]. In particular, in [14], a fingerprint restoration method based on compressed sensing has been proposed. This approach demonstrated the hidden architecture and redundancy features of fingerprints. In [15], the authors put forward a semi-supervised manifold learning method to construct a fingerprint database based on partially labeled data, in which a small part of the signal strength measurements needed to be labeled with the corresponding locations. The time and effort required to establish the fingerprint database in the offline phase has encouraged the research of synchronous positioning and mapping [16,17]. However, although the labor to establish the fingerprint database is reduced, the performance is usually unsatisfactory for most practical indoor positioning systems. By modeling the received signal strength as a Gaussian process, the workload of establishing a fingerprint database can be reduced [6]. In this paper, we first divide the area of interest into multiple grids. Then, we randomly select some reference points (RPs) in the grid and perform signal sampling at these RPs. Finally, the DCGAN algorithm is used to generate additional wavelet feature maps to expand the fingerprint database. Therefore, a more robust fingerprint database can be obtained.

By using the pattern matching algorithm, the position of the UE can be figured out by comparing the query fingerprint to the pre-constructed fingerprints in the database. A commonly used algorithm is the nearest neighbor (NN) algorithm. In order to improve the positioning accuracy, the authors in [18] proposed a KNN algorithm, combining KNNs. The WKNN algorithm proposed in [19] has been used to improve the classic KNN algorithm. One of the main challenges facing the WKNN algorithm, at present, is determining an appropriate k value. Some studies have solved this problem by using a fixed K value, while other studies have solved this problem by determining the value of K based
on experience. In [20], a positioning system based on support vector machines (SVM) was proposed, which transforms the positioning problem into a classification problem. Ye et al. [21] proposed a neural network assisted positioning approach, in order to improve the positioning accuracy. A method for indoor hybrid wireless fingerprint location based on convolutional neural network (CNN) was proposed in [22]. Some deep learning models are affected by gradient dispersion and, consequently, their positioning performance is not ideal. This paper proposes a Resnet-based positioning model, which can effectively avoid the vanishing gradient problem and improve the positioning performance.

Differing from the above positioning methods, our proposed positioning system has three main merits: First, wavelet features are extracted from LTE signal, which provides more refined features, compared to those of RSS. Second, a DCGAN model is used to generate additional fingerprints, which can cut the workload of fingerprint collection and increase the robustness of the fingerprint database. Additionally, a positioning model with strong learning ability is developed.

3. Proposed Positioning System Architecture

In this work, a typical outdoor positioning environment is considered. First, we divide the location area into multiple grids and, then, use signal collection equipment to receive LTE signals from each grid. It is worth mentioning that the LTE signal collected in this paper takes time domain-power as its amplitude. As shown in Figure 1, the purpose of positioning is to find the location of the UE from the collected LTE signals. The proposed positioning system containing four steps: LTE signal collection, LTE signal wavelet transform, fingerprint database construction, and Resnet training and matching.

![Figure 1. Flow chart of the proposed fingerprint-based positioning technique.](image)

### 3.1. LTE Signal Wavelet Transform

Wavelet transforms have great advantages in obtaining time–frequency knowledge of a signal. Wavelets can obtain reliable features from a signal and, so, they have also been called “signal microscopes”. Considering the great advantage of wavelet transforms in signal analysis, we use the continuous wavelet transform to obtain the signal wavelet feature in this paper.

If \( \phi(t) \in L^2(\mathbb{R}) \) and its Fourier transform \( \hat{\phi}(\omega) \) meet the following condition:

\[
C_\phi = \int_{-\infty}^{+\infty} |\hat{\phi}(\omega)|^2 |\omega| d\omega < \infty
\]

we call \( \phi(t) \) the mother wavelet or wavelet function; where \( L^2(\mathbb{R}) \) is the Square integrable complex function space. The corresponding wavelet family contains a set of sub-wavelets, which are generated by the expansion and translation of the wavelet function \( \phi(t) \), as shown below:

\[
\phi_{a,b}(t) = |a|^{-1/2} \phi \left( \frac{t - b}{a} \right)
\]
where \( a \) is a scale factor and \( b \) is the time location, and \( |a|^{-1/2} \) is used to ensure energy preservation.

The continuous wavelet transform of the signal \( x(t) \) is defined as the inner product of the \( L^2 \) norm Hilbert space, as shown below:

\[
W_b(a) = \langle \varphi_{a,b}(t), x(t) \rangle = |a|^{-1/2} \int x(t) \varphi_{a,b}^*(t) dt
\]  

(3)

The asterisk here represents the conjugate of a complex number. The scale factor \( a \) and the time position \( b \) continuously change.

For a discrete sequence \( x_m \), let \( t = m\delta t \) and \( b = n\delta t \), where \( m,n = 0, 1, 2, \ldots, N-1 \), \( N \) is the sampling point number and \( \delta t \) is the sampling interval. The CWT of \( x_m \) is defined as follows:

\[
W_n(a_j) = \sum_{m=0}^{N-1} x_m \varphi^* \left( \frac{(m-n)\delta t}{a_j} \right) 
\]  

(4)

By changing the indices \( j \) and \( n \) corresponding to the scale factor \( a \) and the time position \( b \), we can construct an image which shows the relationship between the amplitude of any feature and the scale, as well as how the amplitude changes over time. The details of implementing a continuous wavelet transform can be found in [23].

There exist two types of wavelet functions: orthogonal and non-orthogonal. Among them, the most widely used orthogonal wavelet functions include Daubechies, Haar, Symlets, Coiflets, and Meyer; while non-orthogonal wavelet functions include DOG, Morlet, and Mexican hat. For dyadic discrete wavelet transforms and wavelet packet transforms, orthogonal wavelet functions must be selected; while, for continuous wavelet transforms, orthogonal or non-orthogonal wavelet functions can be selected, thus offering greater freedom of choice. In fact, the wavelet coefficients measure the similarity between a signal and its sub-wavelets. The more similar the sub-wavelet and the characteristic component, the larger the corresponding wavelet coefficient. In this paper, the wavelet transform scale was manually set to 50. Figure 2 shows the wavelet feature map generated in our experiment.

![Figure 2](image)

**Figure 2.** LTE signal wavelet feature image. The wavelet feature transform scale is \( a = 50 \).

### 3.2. Fingerprint Database Construction

The fingerprint database collected in this paper consists of two parts: the collected signal fingerprints and the symmetric fingerprints generated by DCGAN. During the initial signal acquisition, the area of interest was divided into multiple grids. In each grid, we randomly selected several reference points and held a signal collector to collect the corresponding LTE signal. Then, we converted the LTE signal into wavelet feature images, in order to construct the initial fingerprint database. To enhance the richness of the fingerprint database, DCGAN was used to generate fingerprints. In [24], the same principle was used to expand a fingerprint database, in which the validity of the theory was proven.

The basic idea of a GAN is derived from the Nash equilibrium theory in game theory. The generator network model (Generator) and the discriminator network model (Discriminator) in the GAN network can be treated as the two parties participating in the game. The generator studies the distribution
features of real data to generate data which is similar to the real data. Its ultimate purpose is to form generated data to cheat the discriminator; while the ultimate purpose of the discriminator is to correctly determine whether the input data is real or generated. In order to win, the two sides must continue to learn and optimize, improve their own generation and discrimination ability, respectively, and finally achieve Nash equilibrium between the two [25].

During the training process, one side is fixed and the network weights on the other side are updated and iterated alternately. During the training process, both parties try to optimize their network and form a competition until the two parties reach dynamic equilibrium (i.e., Nash Equilibrium). G improves the distribution of training data and creates samples which are more and more similar to real data, until D can no longer distinguish the results with an accuracy of more than 50%, at which time the discriminator and generator have reached Nash equilibrium. 

\[ \min_G \max_D V(D, G) = \mathbb{E}_{x \sim P_{data}(x)} \{ \log D(x) \} + \mathbb{E}_{z \sim P_z(z)} \{ \log (1 - D(G(z))) \} \]  

where \( P_{data}(x) \) is the real data, \( z \) is a uniformly distributed signal, \( P_z(z) \) is the fake data, and \( G(z) \) and \( D(x) \) are the output of generator and discriminator, respectively.

Figure 3 shows the structure of the proposed DCGAN. DCGAN introduces a convolutional network to GAN for the first time, using the powerful feature extraction ability of the convolutional layer to improve the effect of the GAN. In order to improve the convergence speed and sample quality, the following changes were made to the convolutional neural networks of G and D:

1. Cancel the pooling layer and use deconvolution for upsampling in G, while using strided convolutions in D to replace the pooling layer;
2. The data in G and D are batch normalized to solve the problem of poor initialization;
3. Remove the fully connected layer from G, thus making it a fully convolutional network;
4. The activation function in the G network uses the ReLU function, while the last layer uses the Tanh function;
5. The activation function of all layers in the D network uses the LeakyReLU function.

Figure 3. Structure diagram of the DCGAN generator. \( \varphi \) denotes the symmetric feature maps generated by the generator; \( \phi \) are feature maps from the training set.
3.3. Resnet Training and Matching

Resnet Model Introduction

The significance of residual module is that, when the network gradient is difficult to propagate to the next layer, the network directly skips these residual modules for training; that is, the network gradient bypasses the residual module for propagation.

Instead of letting every few stacked layers directly match a required underlying mapping, these layers are explicitly matched to a residual mapping. Formally, the required underlying mapping is \( H(x) \), while we make the stacked non-linear layer fit another mapping \( F(x) = H(x) - x \) and rewrite the directional mapping as \( F(x) + x \). We assume that optimizing the residual mapping is easier than optimizing the original (unreferenced) mapping. In extreme cases, if an identity map is optimal, it is much easier to push the residual to zero than to fit a bunch of non-linear layers to the identity map. The formula of \( F(x) + x \) can be realized by a feedforward neural network with shortcut connections [26]. Figure 4 shows the schematic diagram of a residual module.

![Figure 4. Schematic diagram of a residual module.](image)

As shown in Figure 5, the proposed Resnet consists of one basic residual block 1, four basic residual blocks 2, three basic residual blocks 3, an average pooling layer, and a fully connected layer.

![Figure 5. Schematic diagram of the proposed Resnet.](image)
The convolutional layer is used to extract the local area features of the neurons in the previous layer. A convolutional neural network is a deep neural network, with a huge number of parameters, which can lead to the problem of network overfitting. After passing through the convolutional layer, although the number of parameters can be reduced, to a large extent, the number of neurons in the network are not significantly reduced. If the features are classified at this time, there will be overfitting problems, leading to poor classification accuracy. Therefore, a pooling layer is usually added between the convolutional layers, in order to reduce the number of feature parameters needed to solve this problem. In addition, the pooling layer also has the function of removing redundant information. The activation function plays a very important role in enhancing the expression ability of the neural network. It converts the linear function of the upper layer input into a non-linear output (such that the neural network can approximate any function) and solves the problem of insufficient expression ability inherent to linear models. The calculation process of the specific parameters of each layer can be found in [27].

3.4. Resnet Training and Matching

In this paper, an early stop training method is adopted to achieve superior training results. First, the expended fingerprint database is divided into a training set, a validation set, and a test set according to the ratio 60%, 20%, and 20%. When using Resnet for training, as the number of training epochs increases, the training set accuracy continues to increase, which indicates that Resnet is continuously learning features. To the contrary, owing to the fact that the network gradient cannot be transmitted very well and due to the overfitting problem, the validation set accuracy first increases and then decreases. This problem limits the performance of the test set. Therefore, the Resnet is considered fully trained, and we record the validation accuracy after each training epoch. During the Resnet training, if the accuracy of the validation set still does not improve after the model is trained for 15 epochs, we stop training, retain the model parameters, and perform accuracy testing on the test set. The accuracy of the test set is regarded as the final positioning accuracy.

In addition to using DCGAN for fingerprint generation, we used several other techniques to enhance the robustness of the fingerprint database. First, the wavelet feature images were enlarged by 1.25×. Second, the wavelet feature images were randomly rotated by 15°. These two techniques are called data enhancement methods, which is a general approach to enhance the robustness of a fingerprint database in image classification and recognition problems. Another technique to improve positioning accuracy was dynamically adjusting the learning rate. The learning rate is an important hyperparameter in a neural network, guiding us to revise the weight of the network through the gradient of the loss function. In the process of neural network training, a too large or too small learning rate can cause the network to fail to converge to a global minimum, which causes the network to fail to achieve the best possible performance. By dynamically adjusting the learning rate, the performance of the network model can be further improved.

In this article, we need to choose an appropriate criterion to judge the model and, so, the positioning accuracy needed to be calculated. When calculating the positioning accuracy, the positioning Resnet model first estimates the position of each sample in the test set and, then, compares the estimated position with the correct position. Therefore, the positioning accuracy can be obtained, as follows:

\[ P_{\text{acc}} = \frac{n}{N} \]  

where \( n \) is the number of correctly predicted samples in the test set and \( N \) is the total number of samples in the test set.

4. Experiments and Results

In order to evaluate the positioning system proposed in this paper, we conducted experiments based on a real outdoor environment. The schematic diagram of the experimental environment is
shown in Figure 6, where 30 m × 30 m was chosen as the grid size. To collect the LTE signal, we held a signal collector device and walked around each pre-divided grid. The signal collector, connected to the antenna, was controlled by a laptop computer to output signal data. All measurements were used to collect downlink LTE signals. Two to Five sampling points were randomly selected in each grid and 30 s of LTE signal was collected at each grid. Finally, the wavelet transform was used to convert the signals collected from each grid into wavelet feature images. The number of grids divided in this experiment was 23. DCGAN was then used to generate a further symmetry wavelet feature images. The positioning system was implemented on a Dell PC equipped with a high-performance graphics card.

4.1. Influence of Different Learning Rate on Positioning Performance

In this experiment, we evaluated the impact of learning rate on positioning performance. When the initial learning rate was large, the positioning accuracy dropped sharply. This is because a high learning rate can cause the network to skip the minimum point during the training process. A high learning rate may lead to a continuous increase in the loss function, which conflicts with the goal of improving the learning ability of the model. A small learning rate may cause the network to converge to a local minimum, thereby reducing the learning ability of the network. Therefore, the learning rate cannot be too large or too small. As shown in Figure 7, the red bar represents using the expanded fingerprint database for positioning, while the blue bar represents using the original fingerprint database for positioning. When using the expanded fingerprint database for positioning, the same data enhancement methods were used. When using the original fingerprint database and expanded fingerprint database for positioning, the best positioning accuracy was 87.4% and 94.7%, respectively. The performance when using the expanded fingerprint database for positioning, therefore, showed
greater performance than using original fingerprint database for positioning, which demonstrates the effectiveness of expanding the fingerprint database.

Figure 7. Positioning performance with different learning rate.

4.2. Influence of Different Batch Size on Positioning Accuracy

Generally speaking, within the appropriate range, the larger the batch size, the more accurate the network descending direction and the smaller the oscillation. If the batch size is too large, local optimization may occur and the direction of network decline basically no longer changes. If the batch size is set too small, the neural network may not converge. Therefore, we compared the influence of different batch sizes on the original fingerprint database and extended fingerprint database. Figure 8 shows that the best performance was observed when the batch size was 32. It also shows the expanded fingerprint database generated by DCGAN can achieve superior performance compared to original fingerprint database.

Figure 8. Positioning performance with different batch size.

4.3. Influence of Different Number of Fingerprints on Positioning Accuracy

The quality of the fingerprint database is one of the most important factors determining the positioning accuracy. An important factor that determines the quality of a fingerprint database is the number of fingerprints contained in the fingerprint database. In this experiment, we compared the impact of different numbers of fingerprints on positioning accuracy. We divided the fingerprint databases into five types: The first to third types of fingerprint databases contained 25%, 50%, and 75% of the original fingerprint database fingerprints. The latter two fingerprint databases were the original fingerprint database and the expanded fingerprint database. Figure 9 presents the influence of the number of fingerprints on the positioning accuracy. It can be seen that, as the number of fingerprints
increased, the positioning accuracy also improved. Additionally, the fingerprints generated by DCGAN had a positive impact on positioning performance, which verified the effectiveness of DCGAN.

4.4. Performance Comparison

In order to prove the superior performance of this algorithm, we compared its positioning performance with other deep learning algorithms. The commonly used deep learning positioning model are MLP and CNN. Therefore, we designed these two positioning models for the purpose of this paper. The specific settings of the models are shown in Tables 1 and 2, respectively. In order to ensure the fairness of comparison, we used the same fingerprint database as the input to the comparison algorithms. Figure 10 shows that the proposed Resnet achieved better performance, compared to the other algorithms. When using the original fingerprint database for positioning, the positioning accuracy of ResNet was 24.3% and 13.3% higher than that of MLP and CNN, respectively. When using the extended fingerprint database for positioning, the positioning accuracy of ResNet was 26.1% and 16.4% higher than that of MLP and CNN, respectively. This is because, compared with ResNet, the structures of MLP and CNN are simpler, making it difficult to learn reliable features and potentially leading to overfitting.

| Table 1. Configuration of MLP. |
|--------------------------------|
| Configuration | Settings and Values |
| Number of Hidden layers | 3 |
| Number of Neurons in Each Layer | 200 |
| Activation Function | Leaky ReLU |
| Loss | Cross-entropy |
| Learning Rate | 0.001 |

| Table 2. Configuration of CNN. |
|--------------------------------|
| Configuration | Settings and Values |
| Activation Function | Leaky ReLU |
| Loss | Cross-entropy |
| Learning Rate | 0.001 |
Figure 10. Positioning comparation with different algorithms: (a) Positioning accuracy with original fingerprint database; and (b) Positioning accuracy with expanded fingerprint database.

5. Conclusions

In this paper, an outdoor localization system based on a wavelet feature transform, DCGAN, and Resnet was proposed. By using a one-dimensional continuous wavelet transform, we could extract more refined features for neural network training. Considering the often labor-intensive data collection in fingerprint positioning techniques, DCGAN was used to generate symmetric wavelet fingerprints, in order to enhance the diversity of the fingerprint database. Then, the expanded fingerprint database is used as the input to Resnet. Besides, several data enhancement techniques were used to enhance the richness of the fingerprint database. Resnet has multiple residual blocks, which can serve to effectively avoid overfitting problem which can occur in neural networks. To obtain the best-performing model, the learning rate is dynamically adjusted. The results of extensive experiments demonstrate that the proposed localization system can achieve satisfactory positioning performance and performs better than other advanced localization techniques.

We anticipate that localization will play an important role in the future. To make the positioning more accurate, we will consider extracting more signal features and combining these signal features in future research. Finally, we also wish to achieve the purpose of engineering application.

Author Contributions: Conceptualization, D.L. and Y.L.; methodology, D.L. and Y.L.; software, D.L.; validation, Y.L. and H.Z.; formal analysis, D.L. and X.L.; investigation, D.L. and Y.L.; data curation, Y.L. and X.L.; writing—original draft preparation, D.L.; supervision, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Huang, H.; Gartner, G.; Krisp, J.M.; Raubal, M.; Van De Weghe, N. Location based services: Ongoing evolution and research agenda. J. Locat. Based Serv. 2018, 12, 63–93. [CrossRef]
2. Ali, W.H.; Kareem, A.A.; Jasim, M. Survey on Wireless Indoor Positioning Systems. Cihan Univ. Erbil Sci. J. 2019, 3, 42–47. [CrossRef]
3. Vo, Q.D.; De, P. A survey of fingerprint-based outdoor localization. IEEE Commun. Surv. Tutor. 2015, 18, 491–506. [CrossRef]
4. Niu, J.; Lu, B.; Cheng, L.; Gu, Y.; Shu, L. Ziloc: Energy efficient wifi fingerprint-based localization with low-power radio. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Shanghai, China, 7–10 April 2013; pp. 4558–4563.
5. Youssef, M.; Agrawala, A. The Horus location determination system. Wirel. Netw. 2008, 14, 357–374. [CrossRef]
6. Yiu, S.; Dashti, M.; Claussen, H.; Perez-Cruz, F. Wireless RSSI fingerprinting localization. Signal Process. 2017, 131, 235–244. [CrossRef]
7. Yeh, L.W.; Hsu, M.H.; Huang, H.Y.; Tseng, W.T. Design and implementation of a self-guided indoor robot based on a two-tier localization architecture. *Pervasive Mob. Comput.* 2012, 8, 271–281. [CrossRef]

8. Kessel, M.; Werner, M. Automated WLAN calibration with a backtracking particle filter. In Proceedings of the International Conference on Indoor Positioning and Indoor Navigation, Sydney, Australia, 13–15 November 2012; pp. 1–10.

9. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.

10. Zhang, C.; Kuhn, M.J.; Merkl, B.C.; Fathy, A.E.; Mahfouz, M.R. Real-time noncoherent UWB positioning radar with millimeter range accuracy: Theory and experiment. *IEEE Trans. Microw. Theory Tech.* 2009, 58, 9–20. [CrossRef]

11. Ni, L.M.; Liu, Y.; Lau, Y.C.; Patil, P.A. LANDMARC: Indoor location sensing using active RFID. In Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, Fort Worth, TX, USA, 26 March 2003; pp. 407–415.

12. Xu, J.; Liu, W.; Zhang, Y.; Wag, C.L. RSSI-based anti-interference WSN positioning algorithm. *J. Northeast. Univ. Nat. Sci.* 2010, 31, 647–650.

13. Yan, J.; Tiberius CC, J.M.; Janssen GJ, M.; Teunissen, P.J.G.; Bellusci, G. Review of range-based positioning algorithms. *IEEE Aerosp. Electron. Syst. Mag.* 2013, 28, 2–27. [CrossRef]

14. Gu, Z.; Chen, Z.; Zhang, Y.; Zhu, Y.; Lu, M.M.; Chen, A. Reducing fingerprint collection for indoor localization. *Comput. Commun.* 2016, 83, 56–63. [CrossRef]

15. Pulkkinen, T.; Roos, T.; Myllymäki, P. Semi-supervised learning for wlan positioning. In Proceedings of the ICANN: International Conference on Artificial Neural Networks, Munich, Germany, 17–19 September 2011; Honkela, T., Duch, W., Girolami, M., Kaski, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 355–362.

16. Li, Q.; Qu, H.; Liu, Z.; Sun, W.; Shao, X.; Li, J. Wavelet Transform DC-GAN for Diversity Promoted Fingerprint Construction in Indoor Localization. In Proceedings of the 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, UAE, 9–13 December 2018; pp. 1–7.

17. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* 2015, arXiv:preprint/1511.06434.