CIRNN: An Ultra-Wideband Non-Line-of-Sight Signal Classifier Based on Deep-Learning

Xiuli YU, Fenghao YANG*, Jie YUN, Shu WU

Abstract: Non-line-of-sight (NLOS) error is the main factor that reduces indoor positioning accuracy. Identifying NLOS signals and eliminating NLOS errors are the keys to improving indoor positioning accuracy. To better identify NLOS signals, a multi-stream model channel-impulse-response-neural-network (CIRNN) was proposed. The inputs of CIRNN include the channel impulse response (CIR) and a small number of channel parameters. To make a more obvious comparison between NLOS signals and line-of-sight (LOS) signals, a new energy normalization method is proposed. Fusing multi-dimensional features, the CIRNN network has a good convergence performance and shows stronger sensitivity to NLOS signals. Experimental results show that the CIRNN achieves the best accuracy on the open-source data set, the $F1$ score is 89.3%. At the same time, the working efficiency of CIRNN meets industry needs, CIRNN can refresh the target position at about 92.6 Hz per second.

Keywords: deep-learning; multi-flow neural networkindoor-localization; NLOS; signal processing; UWB

1 INTRODUCTION

With the rapid development of Internet of Things technology, positioning technology is maturing. The Global Satellite Navigation System (GNSS) has been widely used in vehicle navigation, which provides reliable positioning solutions outdoors. However, outdoor positioning methods (such as GPS, GNSS, etc.) cannot achieve the same effect in the indoor scene. Because of the large number of obstacles in the interior space, the GNSS signal cannot penetrate the wall effectively.

Ultra-wideband (UWB) technology has many advantages, such as strong anti-interference ability, high transmission rate, wide bandwidth, large system capacity, and strong penetration ability. It has been widely used in indoor positioning and achieved excellent positioning accuracy. The core of UWB indoor positioning technology is to calculate the linear distance between the target object and different base stations with the propagation time of the signal. However, when there are obstacles, the propagation time of the signal is seriously affected by NLOS error [1].

As shown in Fig. 1, the cause of NLOS error is that when there are obstacles between the base station and the target, the signal may bounce and scatter. At this time, the propagation time and direction of the signal received by the base station are not true values, which will cause certain positioning errors.

Figure 1 LOS (left) and NLOS (right)

The starting point of this paper is to design an NLOS/LOS signal classifier that can automatically distinguish NLOS signal (dashed line) from LOS signal (solid line) in the case of multi-base station co-location (Fig. 2). In the subsequent positioning process, scholars have provided various schemes [2-4].

The main ideas of these schemes can be divided into two categories: (1) only use the LOS signal to locate the target, but ignore the NLOS signal; (2) compensate for NLOS signal errors before positioning. This paper only focuses on the distinction between NLOS signal and LOS signal but does not include the subsequent positioning process.

So far, scholars all over the world have done a lot of research on NLOS signal recognition tasks [5], putting forward NLOS identification and error elimination algorithm in different ways. These methods can be applied to UWB signals and other wireless technologies. The main ideas can be summarized into the following categories:
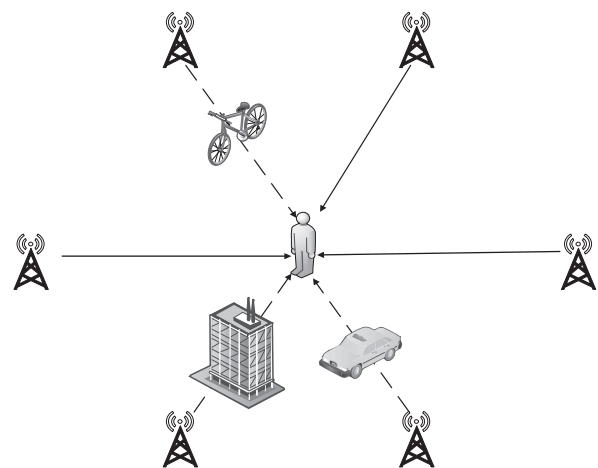

Figure 2 NLOS error compensation

(a) Using some characteristic parameters of the transmission channel to distinguish NLOS signals from LOS signals, such as rise time, overshoot, kurtosis, and slope distribution. This scheme is mostly found in the classical machine learning algorithm. For example, [6] used Support Vector Machine (SVM) to distinguish between LOS and NLOS signals, which extract the received power, maximum power, rise time, peak, and delay distribution as training features. Similarly, [7, 8] train SVM with these channel features, but combine with the fitness between the measured data and the slope distribution.

Instead of using SVM, [9] introduces an NLOS recognition algorithm based on Relevance Vector Machine (RVM) with similar training features. RVM usually takes a longer training time than SVM but works faster in actual implementation. [10] proposed a delayed angle domain method for LOS/NLOS identification, which considers the

diffusion of multipath networks in the angle domain and the spatial selective fading of signals. [11] considers characteristic parameters such as the Ricean K-factor to distinguish between LOS signal and NLOS signal. [12] trains ANN to recognize LOS/NLOS signal and use typical channel features such as the Ricean K-factor and peak value as training data. [13] trains ANN using denoised channel state information (CSI).

Class (a) methods distinguish NLOS signals from LOS signals by the signals' final state and mainly consider the signals' static features. Due to the small amount of input data and ignoring the energy loss of the signal in the propagation process, these methods have certain limitations. The little consideration of the intermediate process of signal propagation restricts the identification accuracy of the class (a) methods.

(b) Distinguishing based on signal propagation path loss model and channel impulse response (CIR). The main idea behind this scheme is that the energy of the first path is significantly greater than that of the subsequent path. Therefore, when there is a large energy difference between adjacent paths, it can be regarded as the occurrence of NLOS interference. This scheme is mostly found in the deep learning algorithm. For example, [14] uses random forest (RF) which takes channel impulse response as training data. [16] proposes a data set for the task of LOS and NLOS identification, in which the CIRs of the signals are the main data. Meanwhile, [16] proposes a CNN network classifier. [15] references the data set in [16], combining with CNN and long short-term memory network LSTM to identify LOS/ NLOS. Based on [16], [17] introduces a down sampling method (PDP) to reduce the influence of signal noise, and proposes a CNN dual-stream network, which takes the energy level features and CIR response of the signal as the input of the dual-stream network respectively.

Considering that most of the current algorithms only consider the time-domain features of the signal, [18] uses the Morlet wavelet transform to combine the time-domain features with frequency-domain features of CIR, and then used CNN for feature extraction. [19] develops a fast NLOS signal recognition method for V2V (vehicle to vehicle) communication, based on several static and time-varying features of CIR. This method fully considers the mobility of vehicles and the rich scattering environment in the street, thus can improve the mutual positioning accuracy between vehicles.

Different from the methods of class (a), the methods of class (b) mainly focus on the energy loss of the signal in the propagation process and focus on the dynamic characteristics of the signals. The defect of these methods is that the signals' final state after the propagation is not considered enough.

However, the final state of the signals can also reflect whether the signal encounters obstacles in the propagation process to a certain extent. Therefore, the class (b) methods lose the information about the signals' final state and the classification accuracy still can be improved.

(c) Statistical methods. The main idea behind these schemes is the statistical parameter analysis of ranging information. For example, [20-22] use a large number of heuristic and deterministic algorithms. [23] adopts the method of statistical analysis. Instead of directly using CIR

response. [24-26] consider the features derived from the signal, such as energy, signal detection time, and so on. These methods need a large number of databases as support, and the discrimination effect is not good.

Inspired by the methods of (a) and (b), a multi-stream neural network CIRNN is proposed in this paper, which can integrate the propagation channel parameter features and CIR features to complete the LOS/NLOS classification task from multiple scales. The CIR features reflect the signals' energy loss in the propagation process while the propagation channel parameter features reflect the signals' final state. Therefore, both the dynamic and static features can be used to classify NLOS signals and LOS signals.

The proposal of CIRNN combines the ideas of class (a) schemes and class (b) schemes and solves their shortcomings. Due to the integration of multi-dimensional features, the CIRNN is more sensitive to NLOS signals than the single stream neural network.

2 CONTRIBUTION

The related works show that the traditional machine learning methods mainly use some characteristic parameters of the propagation channel as training data, while the deep learning methods mostly consider the CIR features of the signal. The training data used in most related works is not comprehensive, only contains the features of one certain dimension of the signal, thus the recognition accuracy is not satisfactory.

To make full use of multi-dimensional information, this paper introduces a multi-stream neural network. The multi-stream network combines the characteristic parameters of the propagation channel and the CIR features of the signal. It has a strong sensitivity to UWB signals. Our job refreshes the best accuracy on the common data set [16].

The main contributions of this paper are summarized as follows:
1) Proposing an energy normalization method to preprocess the CIR of UWB signals. The visualization results show that the normalization method can reduce the noise interference in the input data and make the difference between the NLOS signal and LOS signal more obvious.
2) Constructing a multi-stream neural network, which contains three inputs: signal CIR, energy normalized CIR, and channel characteristic parameters.
Experiments show that the multi-stream neural network achieves a favourable $F1$ score (89.3%), refreshing the best recognition accuracy on the public data set [16]. At the same time with high precision, the running rate of CIRNN can meet the industrial requirements and the theoretical refresh rate of the positioning map reaches 90 Hz.
3) Introducing a new loss function for NLOS and Los identification tasks.
Experiments show that the loss function can improve the recognition effect of the neural network and improve the convergence performance of the model.

The effectiveness of these jobs is verified by comparative experiments.

3 ENERGY NORMALIZATION METHOD
3.1 Data Visualization

For each UWB signal, the CIRNN network takes 1016 CIR samples with 1 nanosecond resolution as input. The CIRNN takes the magnitude of CIR samples as the target feature. The magnitude of each CIR sample is defined as Eq. (1):

$$\left| h\left[n \right] \right|, \ \left\{ 0 \le n < 1016 \middle| n \in Z \right\} \tag{1}$$

Randomly picking an NLOS signal and a LOS signal, the visualization results of 1016 CIR samples are shown in Fig. 3. The figure shows that CIR samples have a large peak value, and the maximum and minimum values differ greatly. Secondly, the maximum CIR response of the LOS signal is larger than that of the NLOS signal, but it is not obvious.

In addition, the CIR sample noise of the NLOS signal is larger and the fluctuation is more obvious than that of the LOS signal. If the original CIRs is directly input into the neural network for training, the neural network cannot distinguish NLOS signal from LOS signal effectively, for the two main reasons:
(a) Due to the large peak value, the neural network learns too much from the part with higher magnitude and becomes less sensitive to the part with lower magnitude.
(b) The difference between the maximum CIR magnitude of LOS and NLOS signals is not obvious, which is not enough to effectively distinguish NLOS and LOS data.
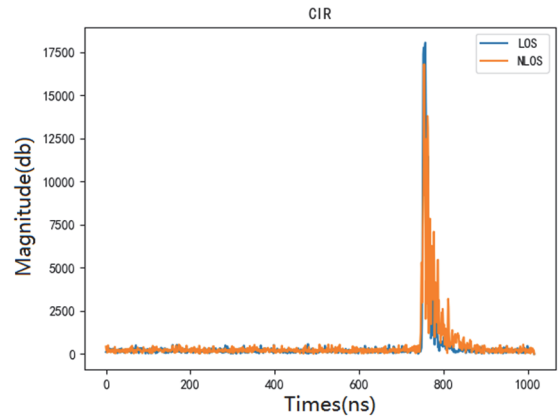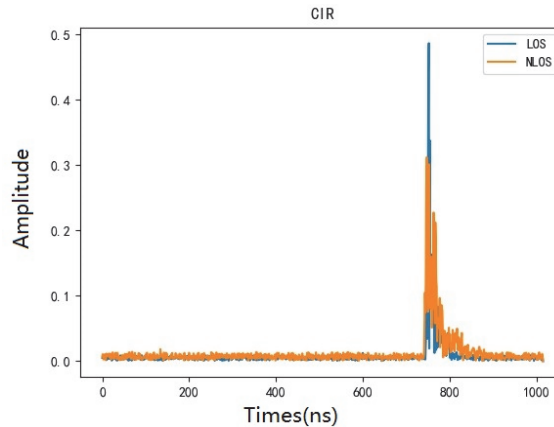

Figure 3 Comparision of NLOS and LOS signal CIR (before normalization)
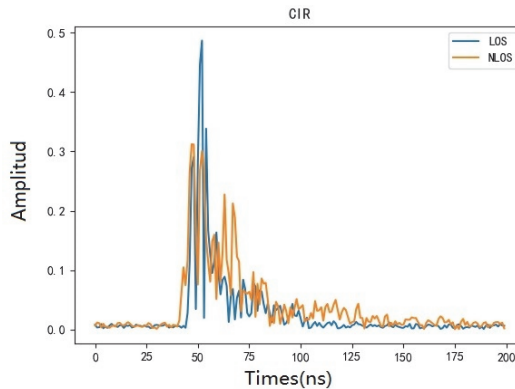
3.2 Energy Normalization

In order to solve the problems existing in the original input and improve the learning efficiency of the neural network, this section proposes a preprocessing method, which mainly includes three steps: noise calculation, total energy calculation, and energy normalization.

Noise calculation is the first step of preprocessing. Defining $h_{\max}$ as the maximum CIR magnitude of a signal and the part below 10% $h_{\max}$ as noise samples.

$$\begin{cases} \sigma_n^2 = \dfrac{1}{\alpha} \sum^{\alpha} \left| h\left(n \right) \right|^2 \\ \forall h\left(n \right) \le 0.1 h_{\max} \end{cases} \tag{2}$$

a)   Comparision of NLOS and LOS signal CIR(overall)

b)   Comparision of NLOS and LOS signal CIR (local amplification)
Figure 4 Comparision of NLOS and LOS signal CIR (after normalization)

In Eq. (2), α is the number of CIR noise samples. After obtaining the total noise of the signal, the second step of preprocessing is to calculate the total energy. Calculating the total energy ε_h of the signal according to Eq. (3):

$$\begin{cases} \varepsilon_h = \dfrac{1}{\beta} \sum_{}^{\beta} (\left| h(n) \right|^2 - \sigma_n^2) \\ \forall h(n) > 0.1 h_{max} \end{cases} \quad (3)$$

The third step of preprocessing is to normalize the signal using the total energy ε_h of the signal:

$$\left| H(n) \right| = \left| h(n) \right| / \sqrt{\varepsilon_h} \quad (4)$$

The preprocessing result of data in Fig. 3 is shown in Fig. 4. The overall result is shown in Fig. 4a, and the result at time 700 - 900 is zoomed in Fig. 4b. The results show that the preprocessing method can reduce the maximum magnitude of CIR to about 0.5. This preprocessing method can avoid the influence of accidental high-magnitude CIR samples on neural networks and enhance the sensitivity toward the CIRs with lower magnitude.

After energy normalization, the difference in peak value between LOS and NLOS CIRs increases significantly compared with the data before preprocessing. Therefore, the normalized data has a better distinguishing effect for LOS/NLOS signals.

4 CIRNN NETWORK

This paper proposes a multi-stream neural network, which is named channel impulse response neural network (CIRNN). The network fuses the CIR features with the signal channel features.

The multi-stream neural network contains three branches. Each branch can be trained separately and can complete the identification task of NLOS and LOS signals independently. The three branches also can be spliced together to form a whole multi-stream network. The structure of each branch and the whole network structure are introduced below.

4.1 CIRNN (a)

CIRNN (a) takes the pre-processed CIR samples as input.

Section 3 has proved that the energy normalization method in this paper can enhance the difference between the NLOS signal and the LOS signal. This part of the training data contains the most important information, thus why the layer of CIRNN (a) is deeper than CIRNN (b) and CIRNN (c). The structure of CIRNN (a) is shown in Fig. 5a, where N is the batch size, which is set during training. To improve the training accuracy and reduce the over-fitting of the model, the residual structure is introduced. This branch consists of 21 Network layers, including 17 convolutions, 2 pooling layers, 1 full connection layer, and 1 Softmax layer.

4.2 CIRNN (b)

The function of CIRNN (b) is to assist CIRNN (a) in training. It takes the original CIR samples as input.

Although the pre-processed data better reflect the difference between NLOS signal and LOS signal, some information may be lost during the pre-processed process. The existence of this branch compensates for this missing information and can guide the training of CIRNN (a) to a certain extent. With CIRNN (b)'s guide, CIRNN (a) performs more stable and converges faster in the process of training. The structure of CIRNN (b) is shown in Fig. 5b. CIRNN (b) contains 10 network layer structures, including 4 convolution layers, 4 pooling layers, 1 full connection layer, 1 Softmax layer, and no residual structure.

4.3 CIRNN (c)

The function of CIRNN(c) is to combine multidimensional features. It takes the 10 signal channel parameters as input, including 3 intermediate parameters σ_n^2, ε_h, h_{max} which are calculated during data preprocessing. And 7 additional parameters are collected in the "DW1000" module, including RANGE (time of flight), MAX_ NOISE (maximum value of noise), FP_ AMP1 (first path amplitude - part1), FP_ AMP2 (first path amplitude - part2), FP_ AMP3 (first path amplitude - part3), STDEV_ NOISE (standard deviation of noise), CIR_ PWR (total channel impulse response power).

This branch fuses multidimensional features to guide the training process of CIRNN (a) and CIRNN (b). The structure of CIRNN (c) is shown in Fig. 5c. Due to the small amount of input data, this branch only contains five layers, including two convolution layers, two full connection layers, and one Softmax layer.

4.4 CIRNN32

CIRNN32 is composed of CIRNN (a), CIRNN (b), and CIRNN (c) (sharing the FC + Softmax layer). Its structure is shown in Fig. 6. CIRNN32 consists of 32 network layer structures, including 23 convolution layers, 6 pooling layers, 2 full connection layers, and 1 Softmax layer. CIRNN32 has three branches that can fuse features of multiple dimensions. Compared with the single branch network, it has stronger stability and higher classification accuracy.
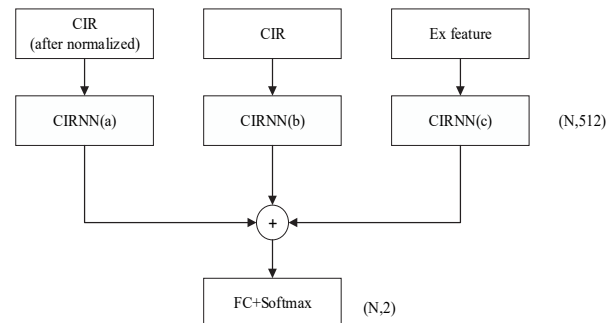

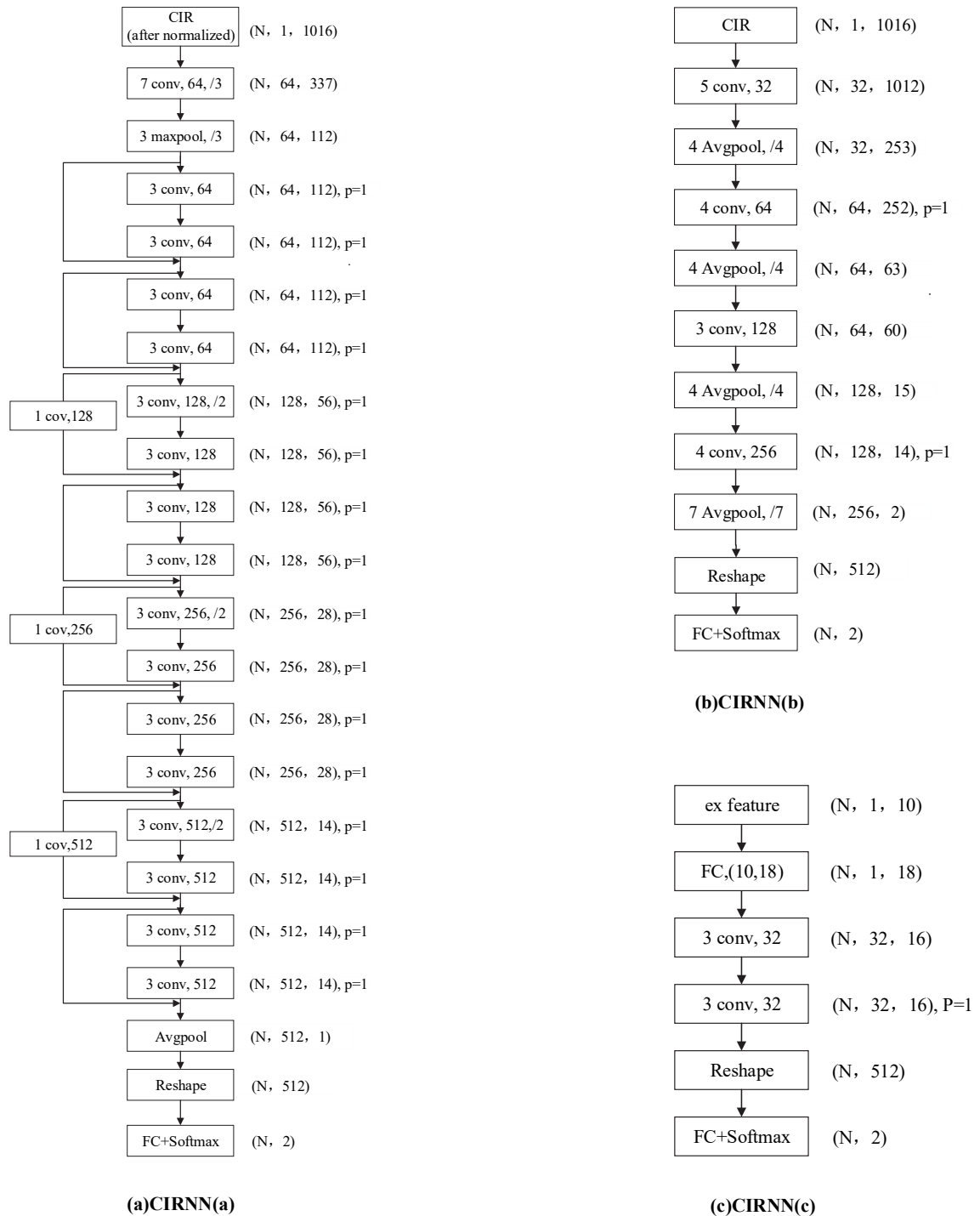**Figure 6** CIRNN32 overall network structure

CIR (after normalized)	(N, 1, 1016)
7 conv, 64, /3	(N, 64, 337)
3 maxpool, /3	(N, 64, 112)
3 conv, 64	(N, 64, 112), p=1
3 conv, 64	(N, 64, 112), p=1
3 conv, 64	(N, 64, 112), p=1
3 conv, 64	(N, 64, 112), p=1
1 cov,128	
3 conv, 128, /2	(N, 128, 56), p=1
3 conv, 128	(N, 128, 56), p=1
3 conv, 128	(N, 128, 56), p=1
3 conv, 128	(N, 128, 56), p=1
1 cov,256	
3 conv, 256, /2	(N, 256, 28), p=1
3 conv, 256	(N, 256, 28), p=1
3 conv, 256	(N, 256, 28), p=1
3 conv, 256	(N, 256, 28), p=1
1 cov,512	
3 conv, 512,/2	(N, 512, 14), p=1
3 conv, 512	(N, 512, 14), p=1
3 conv, 512	(N, 512, 14), p=1
3 conv, 512	(N, 512, 14), p=1
Avgpool	(N, 512, 1)
Reshape	(N, 512)
FC+Softmax	(N, 2)

(a)CIRNN(a)

CIR	(N, 1, 1016)
5 conv, 32	(N, 32, 1012)
4 Avgpool, /4	(N, 32, 253)
4 conv, 64	(N, 64, 252), p=1
4 Avgpool, /4	(N, 64, 63)
3 conv, 128	(N, 64, 60)
4 Avgpool, /4	(N, 128, 15)
4 conv, 256	(N, 128, 14), p=1
7 Avgpool, /7	(N, 256, 2)
Reshape	(N, 512)
FC+Softmax	(N, 2)

(b)CIRNN(b)

ex feature	(N, 1, 10)
FC,(10,18)	(N, 1, 18)
3 conv, 32	(N, 32, 16)
3 conv, 32	(N, 32, 16), P=1
Reshape	(N, 512)
FC+Softmax	(N, 2)

(c)CIRNN(c)

Figure 5 CIRNN branch networks

5 LOSS FUNCTION

To train the CIRNN network better, a new loss function is proposed.

The training data set used in this paper consists of 42000 signals, and NLOS signals and LOS signals have the same amount. For all the NLOS signals, we average the magnitude of CIRs after preprocessing and obtain the standard NLOS CIRs curve. Similarly, the standard LOS CIRs curve can be obtained. The visualization results of the two standard curves are shown in Fig. 7.

The loss function proposed in this paper is calculated as follows:

$$Loss = |y' - y| \left[y' \times \log \left(\sum_{n=1}^{1016} \left(|h(n) - h_0(n)| \right) \right) + \\ + (1 - y') \times \log \left(\sum_{n=1}^{1016} \left(|h(n) - h_1(n)| \right) \right) \right] \quad (5)$$

In the formula, y is the ground-truth (0 or 1), y' is the predicted value, h is the input of the neural network, h_1 is the standard NLOS CIRs, h_0 is the standard LOS CIRs. The principle of the loss function is:

1). If the prediction is right, then $y' - y = 0$, the loss value is 0.

2). If the prediction is wrong and y' is 1, the loss value is calculated by comparing the input h with the standard LOS CIRs h_0.

3). If the prediction is wrong and y' is 0, the loss value is calculated by comparing the input h with the standard NLOS CIRs h_1.

The loss function can collect different information from different training data. For the wrong-predicted samples, the more different it compares with standard CIRs, the more information the loss function collects. The loss function can improve the classification effect and the convergence performance of the network.
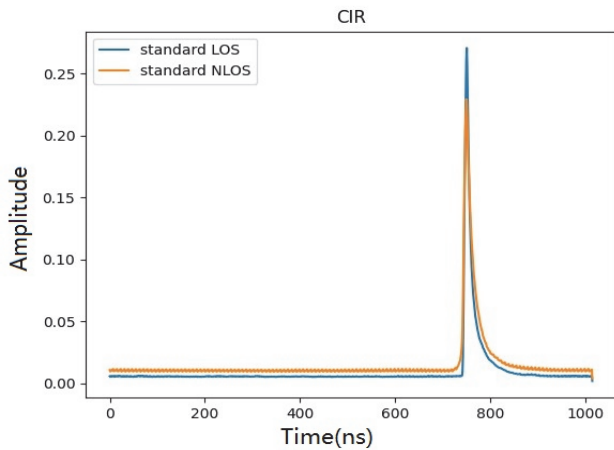

Figure 7 NLOS/LOS benchmark curve

6 EXPERIMENTAL ANALYSIS

Data set is the key to comparing the performance of machine learning algorithms. [16-18] all use the same dataset [17], which contains 42000 samples collected from different indoor locations in the DW1000 module. These locations include two offices, a small apartment, a kitchen with a living room, a bedroom, a small workshop, and a boiler room. LOS/NLOS signals in the dataset are symmetrically distributed, and each class accounts for 50% of the samples. In addition, in order to avoid the deviation caused by different acquisition locations, the data set is randomized.

To compare with the previous works, our job is also carried out on this data set. We divide the training set, test set, and verification set in a 7:2:1 ratio. Among them, the training set contains 29600 samples, the test set contains 8400 samples, and the verification set contains 4200 samples. Since the data set has been randomized, there will be no uneven distribution of NLOS/LOS samples.

Pytorch is the framework of the deep-learning network, while batch_size is set to 256 and the learning rate is set to 0.0001. We select Adam as the optimizer and train CIRNN for 70 epochs on the data set [17] to evaluate the training results. In this paper, F1 score is used to evaluate the recognition effect. $F1$ score is a common measurement standard in the literature. $F1$ score considers both accuracy and recall, and is calculated according to Eq. (6):

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{6}$$

In order to verify the effectiveness of CIRNN, a comparative experiment is conducted from five aspects: 6.1 Data preprocessing verification; 6.2 Network structures verification; 6.3 Loss function verification; 6.4 Comparison with relevant works; 6.5 Efficiency verification.

6.1 Data Preprocessing Verification

CIRNN32 has three branches, and only CIRNN (a) takes the pre-processed CIRs as input. In order to verify the effectiveness of the preprocessing method in Section 3 we input the original data and pre-processed data to the neural network CIRNN (a) respectively, and compare the $F1$ scores. Fig. 8 shows the result. The horizontal axis represents the number of training epochs and the vertical axis represents the $F1$ score. The solid line is the training effect of CIRNN (a) with pre-processed data, and the dotted line is the training effect of CIRNN (a) with source data.
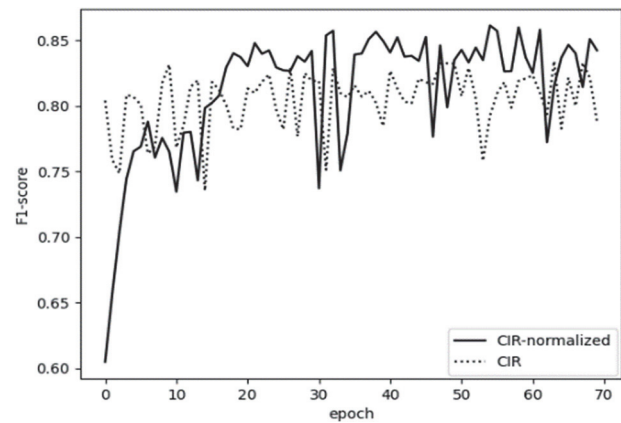

Figure 8 Validation of the normalization method

It can be seen from Fig. 8 that the $F1$ score increases when using pre-processed CIRs for training, which proves the preprocessing method proposed in this paper is effective.

However, the single-stream network CIRNN (a) in Fig. 8 shows great oscillation during the training process, thus we propose the multi-stream model CIRNN32.

6.2 Network Structure Verification

Each branch of CIRNN32 can classify NLOS and LOS signals independently. For comparison, we train each branch separately. The training process of each branch and CIRNN32 is shown in Fig. 9. The black line represents CIRNN32 and the color line represents each branch. Their $F1$ scores were evaluated separately.

It can be seen from Fig. 9 that there is an obvious oscillation phenomenon in the training process of the single-stream network, and the multi-stream network is more stable, which is the embodiment of mutual supervision and guidance between branches. In addition, the classification effect of multi-stream networks is superior to that of single-stream networks. We also try to combine the branches network in pairs, and the results are shown in Tab. 1. It can be seen from the table that CIRNN32 has the best $F1$score 89.3%.

The comparative experiment proves the effectiveness of the multi-stream model CIRNN. The result shows that CIRNN32 has a strong sensitivity to NLOS signals and the convergence performance of CIRNN32 is more stable than that of the single-stream network. CIRNN performs as we envisioned when we designed it, which proves the idea of integrating multidimensional features is feasible.

The size of the network model is also an important indicator to measure the performance of neural networks. Generally, the larger the trained network is, the more GPU resources it occupies. CIRNN32 consists of 32 layers, of which CIRNN (a) has 20 layers, CIRNN (b) has 9 layers, and CIRNN (c) has 4 layers. Therefore, the size of each branch network model varies greatly. Comparing the model size, CIRNN (c) is an ultra-lightweight network with a model size of only 56 KB, which can be used for mobile deployment. In the application, users can choose the network structure according to actual requirements.
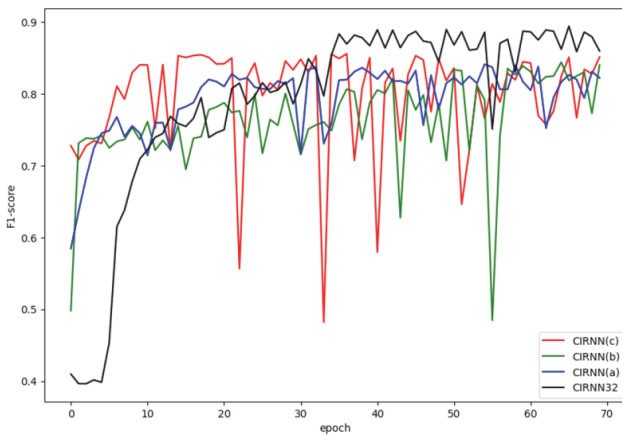

Figure 9 Comparison of CIRNN network structures

Table 1 Comparison of CIRNN network structures

structure	$F1$ score	Model size	GPU using (batch_size = 256)
CIRNN (a)	0.878	44.10 MB	1975 MB
CIRNN (b)	0.862	2.00 MB	1155 MB
CIRNN (c)	0.859	**56 KB**	**805 MB**
CIRNN (ab)	0.880	44.14 MB	2658 MB
CIRNN (ac)	0.882	46.10 MB	1977 MB
CIRNN (bc)	0.872	2.05 MB	1235 MB
CIRNN32	**0.893**	46.15 MB	3025 MB

6.3 Loss Function Verification

The cross-entropy function is often used as the loss function in classification tasks. In order to verify the improvement of the loss function in section 5, we use the cross-entropy function and our loss function to train CIRNN32 respectively. Fig. 10 shows the result. The solid line represents the effect of our loss function, and the dotted line represents the effect of the cross-entropy loss function.

The experimental results show that the cross-entropy loss function has a better recognition effect in the early stage of training. However, our loss function can better stimulate the best performance of the CIRNN network. When using the cross-entropy loss function, the best $F1$ score is 88.7%, while using our loss function, the best $F1$ score of the model is increased to 89.3%.
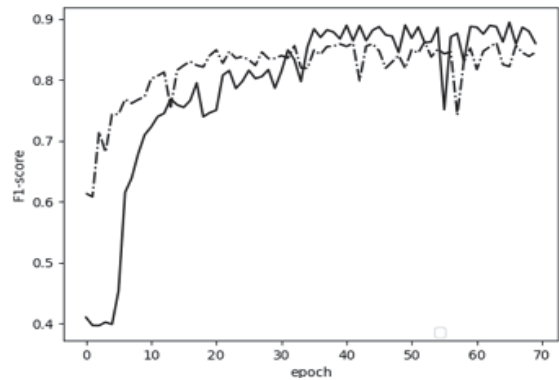

Figure 10 Validation of loss function

6.4 Comparison with Relevant Work

Many other scholars also use the common dataset to train their network models. To compare with earlier methods, we also reproduce the classical machine learning methods. The comparison results between our job and related works are shown in Tab. 2.

Table 2 Comparasion between CIRNN32 and related works

Networks	Precision	$F1$ score
SVM (linear)	0.825	0.844
SVM (RBF)	0.836	0.849
MLP	0.839	0.853
ResNet50	0.851	0.857
K. Bregar [16]	0.874	0.876
C. Jiang [15]	0.821	--
V. B. Vales [17]	--	0.889
CIRNN32	**0.891**	**0.893**

Experimental results show that CIRNN32 achieves the best recognition effect on the data set. Compared with the previous work [16] with the highest precision, our job improves the precision by 1.7%. And compared with the previous work [17] with the highest $F1$ score, our job improves the $F1$ score by 0.4%.

6.5 Efficiency Verification

Since CIRNN32 is a multi-stream network, it has more computation compared with single-stream networks. We use the trained network models to classify UWB signals and record their working efficiency. The result is shown in Tab. 3.

Compared with the single-stream network, the working efficiency of CIRNN32 is lower, but it still meets the practical needs. The working efficiency of CIRNN32 is 463 signals per second. That means when 5 base stations are used for cooperative positioning, CIRNN can refresh the target position at about 92.6 Hz per second.

The experimental results show that CIRNN32 can be used in practical engineering.

Table 3 Efficiency of CIRNN

Structure	Speed (signals per second)
CIRNN (a)	836
CIRNN (b)	1724
CIRNN (c)	3648
CIRNN 32	463

7 CONCLUSION

The classification of LOS and NLOS signals is of great significance to reduce the error of UWB indoor positioning technology. In this paper, a multi-stream neural network CIRNN is proposed, which combines the CIR features with the signal channel features. The CIRNN introduces a new preprocessing method and a new loss function to enhance the performance of the network. The experimental results show that the CIRNN refreshes the best accuracy on the common data set. The $F1$ score of CIRNN is 89.3% and the accuracy is 89.1%. At the same time, the working efficiency of CIRNN meets the practical needs, the refresh rate of the positioning map is about 92.6 Hz.

8 REFERENCES

[1] Abolfathi, A., Behnia, F., & Marvasti, F. (2018). Nlos mitigation using sparsity feature and iterative methods. http://arxiv.org/pdf/1803.06838

[2] Cui, W., Cheng-Dong, W. U., Zhang, Y. Z., Jia, Z. X., & Cheng, L. (2014). Gmm-based localization algorithm under nlos conditions. *Journal on Communications*.

[3] XIE Liangbo. (2021). Scatterer information based indoor NLOS multiple base station cooperative localization algorithm. *Journal on Communications*.

[4] Xiao, Z., Chen, J., Wang, D., Xiao-Hong, L. I., & Ren-Fa, L. I. (2015). Three-dimensional localization under severely obstructed nlos environment. *Journal on Communications*. http://en.cnki.com.cn/Article_en/CJFDTOTAL-TXXB201508009.htm

[5] Wymeersch, H. (2012). A machine learning approach to ranging error mitigation for uwb localization. *IEEE Transactions on Communications*, *60*(6), 1719-1728. https://doi.org/10.1109/TCOMM.2012.042712.110035

[6] Maranò, S., Gifford, W. M., Wymeersch, H., & Win, M. Z. (2017). Chalmers publication library copyright notice nlos identification and mitigation for localization based on uwb experimental data. *IEEE Journal on Selected Areas in Communications, 28*(7), 1026-1035. https://doi.org/10.1109/JSAC.2010.100907

[7] Xiao, Z., Wen, H., Markham, A., Niki, T., Phil, B., & Jeff, F. (2015). Non-line-of-sight identification and mitigation using received signal strength. *IEEE Transactions on Wireless Communications*, *14*(3), 1-1. https://doi.org/10.1109/TWC.2014.2372341

[8] Xiao, Z., Wen, H., Markham, A., Trigoni, N., & Frolik, J. (2013). Identification and mitigation of non-line-of-sight conditions using received signal strength. Wireless and Mobile Computing, Networking and Communications (WiMob). *2013 IEEE 9th International Conference on. IEEE*. https://doi.org/10.1109/WiMOB.2013.6673428

[9] Nguyen, T. V., Jeong, Y., Shin, H., & Win, M. Z. (2015). Machine learning for wideband localization. *IEEE Journal on Selected Areas in Communications*, *33*(7), 1-1. https://doi.org/10.1109/jsac.2015.2430191

[10] Zheng, Q., He, R., Ai, B., Huang, C., & Zhang, H. (2020). Channel non-line-of-sight identification based on convolutional neural networks. *IEEE Wireless Communication Letters*, *PP*(99), 1-1. https://doi.org/10.1109/LWC.2020.2994945

[11] Tang, P., Zhang, J., Molisch, A. F., Smith, P. J., Shafi, M., & Tian, L. (2019). Estimation of the k-factor for temporal fading from single-snapshot wideband measurements. *IEEE Transactions on Vehicular Technology*, *68*(1), 49-63. https://doi.org/10.1109/TVT.2018.2878352

[12] Fu, X., Guo, Z., Hai, Z., Xie, X., & Wang, R. (2017). AmpN: Real-time LOS/NLOS identification with WiFi. *ICC 2017 - 2017 IEEE International Conference on Communications. IEEE*. https://doi.org/10.1109/ICC.2017.7997068

[13] Decurninge, A., Ordóez, L. G., Ferrand, P., Gaoning, H., Bojie, L., Wei, Z., & Maxime, G. (2018). Csi-based outdoor localization for massive mimo: experiments with a learning approach. *IEEE*. https://doi.org/10.1109/ISWCS.2018.8491210

[14] Kurniawan, E., Lin, Z., & Sun, S. (2018). Machine Learning-Based Channel Classification and Its Application to IEEE 802.11ad Communications. *GLOBECOM 2017 - 2017 IEEE Global Communications Conference. IEEE*.

[15] Jiang, C., Shen, J., Chen, S., Chen, Y., & Bo, Y. (2020). Uwbnlos/los classification using deep learning method. *IEEE Communications Letters*, *PP*(99), 1-1.

[16] Bregar, K. & Mohorcic, M. (2018). Improving indoor localization using convolutional neural networks on computationally restricted devices. *IEEE Access*, 1-1. https://doi.org/10.1109/ACCESS.2018.2817800

[17] Vales, V. B., Dominguez-Bolano, T., Escudero, C. J., & Garcia-Naya, J. A. (2020). Using the power delay profile to accelerate the training of neural network-based classifiers for the identification of los and nlosuwb propagation conditions. *IEEE Access*, *8*, 220205-220214. https://doi.org/10.1109/ACCESS.2020.3043503

[18] Cui, Z., Gao, Y., Hu, J., Tian, S., & Cheng, J. (2020). Los/nlos identification for indoor uwb positioning based on morlet wavelet transform and convolutional neural networks. *IEEE Communications Letters*, *PP*(99), 1-1. https://doi.org/10.1109/LCOMM.2020.3039251

[19] Huang, C., Molisch, A. F., He, R., Wang, R., & Zhong, Z. (2020). Machine learning-enabled los/nlos identification for mimo systems in dynamic environments. *IEEE Transactions on Wireless Communications*, *PP*(99), 1-1. https://doi.org/10.1109/TWC.2020.2967726

[20] Decawave. Dw1000 Metrics for Estimation of Non Line of Sight Operating Conditions. Accessed: Nov. 23, 2020. [Online].Available:https://www.decawave.com/sites/default/fifiles/aps006_part_3_dw1000_diagnostics_for_nlos_channel s.pdf

[21] Meghani, S. K., Asif, M., Awin, F., & Tepe, K. (2019). Empirical based ranging error mitigation in ir-uwb: a fuzzy approach. *IEEE Access*, 1-1. https://doi.org/10.1109/ACCESS.2019.2904201

[22] Yu, K., Wen, K., Li, Y., Zhang, S., & Zhang, K. (2018). A novel nlos mitigation algorithm for uwb localization in harsh indoor environments. *IEEE Transactions on Vehicular Technology*, 686-699. https://doi.org/10.1109/TVT.2018.2883810

[23] Smail, G., Chia-Chin, C., Watanabe, F., & Hiroshi, I. (2007). Nlos identification and weighted least-squares localization for uwb systems using multipath channel statistics. *EURASIP Journal on Advances in Signal Processing*, *2008*(1), 271984-271984. https://doi.org/10.1155/2008/271984

[24] Song, B., Li, S. L., Tan, M., & Ren, Q. H. (2018). A fast imbalanced binary classification approach to nlos identification in uwb positioning. *Mathematical Problems in Engineering*, *2018*, 1-8. https://doi.org/10.1155/2018/1580147

[25] Zhuoling, X., Hongkai, W., Andrew, M., Niki, T., Phil, B., Jeff, F. (2015). Non-Line-of-Sight Identification and Mitigation Using Received Signal Strength. *IEEE transactions on wireless communications*, 1689-1702. https://doi.org/10.1109/TWC.2014.2372341

[26] Zeng, Z., Liu, S., & Wang, L. (2019). UWB NLOS identification with feature combination selection based on genetic algorithm. *2019 IEEE International Conference on Consumer Electronics (ICCE). IEEE*. https://doi.org/10.1109/ICCE.2019.8662065

Contact information:

Xiuli YU, senior engineer
Beijing University of Posts and Telecommunications,
Xitucheng Road, Haidian District, Beijing, China
E-mail: yxl@bupt.edu.cn

Fenghao YANG, assistant research
(Corresponding author)
Beijing University of Posts and Telecommunications,
Xitucheng Road, Haidian District, Beijing, China
E-mail: bupt_yfh@bupt.edu.cn

Jie YUN, assistant research
Beijing University of Posts and Telecommunications,
Xitucheng Road, Haidian District, Beijing, China
E-mail: cloud716@outlook.com

Shu WU, assistant research
Beijing University of Posts and Telecommunications,
Xitucheng Road, Haidian District, Beijing, China
E-mail: wsq7s4y1@163.com