

UWB NLOS Signal Recognition Based on Deep Learning

Guanhui Li, Guoliang Wei, Zhuang Xi

School of Optical-Electrical Computer Engineering, University of Shanghai for Science and Technology, Shanghai, China
Email: l475gh@163.com

How to cite this paper: Li, G.H., Wei, G.L. and Xi, Z. (2026) UWB NLOS Signal Recognition Based on Deep Learning. *Open Journal of Applied Sciences*, **16**, 779-797.

<https://doi.org/10.4236/ojapps.2026.163048>

Received: February 10, 2026

Accepted: March 9, 2026

Published: March 12, 2026

Copyright © 2026 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Ultra-wide band (UWB) positioning technology has attracted increasing attention due to its high ranging accuracy. However, in indoor environments, non-line-of-sight (NLOS) signals significantly degrade the ranging performance of UWB positioning systems compared with line-of-sight (LOS) signals. Therefore, accurate identification of LOS and NLOS signals is essential before adopting effective measures to improve UWB positioning accuracy. In this paper, a deep learning-based classification algorithm for UWB NLOS signal identification is proposed. Convolutional neural networks are employed to enhance the model's feature extraction capability, while long short-term memory networks are used to capture temporal dependencies in the time-series data. Furthermore, a self-attention mechanism is introduced to further strengthen feature representation, thereby improving the recognition accuracy of NLOS signals. Experimental results demonstrate that the proposed method achieves an overall accuracy of 89.95%, with a precision and specificity of 91.07% and 91.33%, respectively, indicating superior comprehensive performance.

Keywords

Ultra-Wide Band, Non-Line-of-Sight, Deep Learning, Convolutional Neural Network, Long Short-Term Memory, Self-Attention Mechanism

1. Introduction

In recent years, the demand for real-time and high-accuracy indoor positioning has been growing rapidly. However, in indoor environments, positioning systems often suffer from signal blockage caused by obstacles, leading to signal instability and, in some cases, positioning failure [1]. To address this challenge, extensive research efforts have been devoted to indoor localization, and various positioning

solutions have been proposed [2]. Technologies such as infrared, ultrasonic, Bluetooth, Wi-Fi, radio-frequency identification (RFID), and UWB have been applied to indoor positioning systems [3].

Among these technologies, UWB is considered particularly promising due to its high power efficiency, strong penetration capability, and robust performance in harsh multipath environments. UWB-based ranging systems can achieve an accuracy of up to ± 10 cm [4]. As a result, UWB positioning has become one of the most widely adopted indoor localization technologies, owing to its high positioning accuracy and strong anti-interference capability [5]. However, in practical applications, UWB-based indoor positioning still faces several challenges, among which the NLOS problem is one of the most critical issues. NLOS conditions generally occur when the direct propagation path between the transmitter and receiver is obstructed. Under NLOS conditions, signal propagation delays are introduced, resulting in biased distance estimation between the transmitter and receiver. Since most ranging algorithms assume LOS conditions, distance errors caused by NLOS propagation can lead to a sharp degradation in positioning accuracy. Compared with other positioning technologies, UWB-based ranging systems are more severely affected by NLOS conditions [6]. Therefore, effective handling of the NLOS problem is of great significance for improving indoor positioning accuracy.

With the rapid development of UWB positioning technology, extensive research has been conducted on NLOS state identification, providing strong support for enhancing indoor localization performance. In [7], a Kalman filter-based NLOS identification method was proposed, which determines the channel state by analyzing variations in the signal propagation cross-section. However, this method is sensitive to environmental changes and exhibits limited stability. In [8], a feature selection strategy was adopted to optimize NLOS identification, leading to performance improvement to some extent; nevertheless, its recognition performance varies significantly across different scenarios, indicating limited generalization capability. A prior-knowledge-free and environment-insensitive identification method was proposed in [9], which is able to extract partial NLOS channel features, but its recognition accuracy remains insufficient in complex environments. In [10], a weighted K-nearest neighbor (KNN) algorithm based on channel impulse response (CIR) features was proposed by incorporating feature saturation and feature selection optimization. Although this method improves recognition accuracy while maintaining low computational complexity, it relies heavily on manually designed features and may still overlook potentially informative characteristics.

However, methods based on handcrafted feature extraction typically involve complex processing pipelines and inevitably introduce subjective factors, which may result in the loss of valuable information. With the continuous advancement of deep learning, end-to-end NLOS/LOS identification methods have attracted increasing attention. These approaches enable automatic feature learning and clas-

sification by directly modeling UWB signals. In [11], reversible transform denoising was combined with a convolutional neural network (CNN), achieving a recognition accuracy of 81.68% on the public eWINE dataset. Furthermore, CNNs were integrated with long short-term memory (LSTM) networks in [12], improving the recognition accuracy to 82.14%, although the overall performance gain remained limited. A fully convolutional network (FCN)-based method was proposed in [13], further increasing the classification accuracy to 88.24%. In addition, studies in [14] and [15] explored transforming one-dimensional CIR data into two-dimensional images and applying deep learning models for classification. However, such methods are sensitive to the image transformation process and parameter settings and typically incur high computational complexity, making them less suitable for real-time applications.

To further improve NLOS/LOS classification performance in UWB environments, this paper proposes a deep learning-based NLOS identification method, referred to as the CL-Attention algorithm. The proposed approach organically integrates convolutional neural networks, long short-term memory networks, and a self-attention mechanism to construct an end-to-end feature learning and classification framework. In the proposed model, CNNs are employed to extract local features from UWB CIR signals, while LSTM networks model the temporal correlations of the signals. A self-attention mechanism is further introduced to adaptively weight and enhance temporal features, thereby improving the model's capability to capture critical information. Experimental validation is conducted on a public UWB CIR dataset, and the results demonstrate that, compared with several existing models, the proposed method achieves more stable and competitive improvements in classification performance.

2. Model Overall Architecture Design

This section presents the design and implementation of a deep learning-based UWB NLOS signal classification model. The proposed model focuses on modeling the feature differences of UWB signals under LOS and NLOS propagation environments to enable effective discrimination between LOS and NLOS states. The overall architecture of the proposed model is illustrated in **Figure 1**.

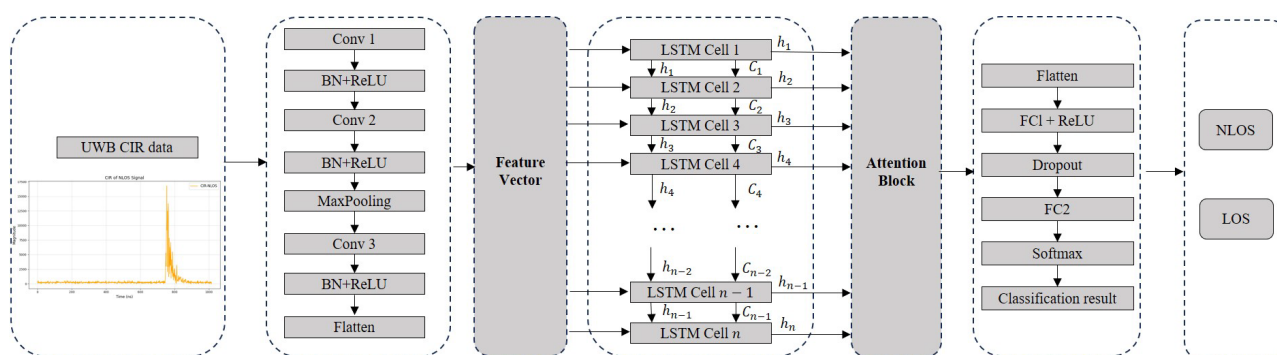


Figure 1. Schematic diagram of overall model architecture.

2.1. Convolutional Feature Extraction Module

To fully exploit the spatial structural characteristics embedded in UWB CIR data, a spatial feature extraction module based on one-dimensional convolutional neural network (1DCNN) is designed and implemented. This module takes normalized raw CIR sequences as input and performs hierarchical feature extraction through multiple one-dimensional convolutional operations, providing discriminative high-dimensional representations for subsequent temporal modeling.

One-dimensional convolution is capable of effectively capturing signal variation patterns within local temporal windows as well as multipath energy distribution characteristics, thereby extracting spatial structural information that is highly relevant to LOS/NLOS discrimination. After each convolutional layer, batch normalization (BN) and rectified linear unit (ReLU) activation functions are introduced to enhance training stability, accelerate convergence, and improve the non-linear representation capability of the network.

To further reduce feature dimensionality and suppress redundant information, a max pooling operation is incorporated into the convolutional module to downsample the convolutional features. This operation reduces computational complexity while preserving essential information. By stacking multiple basic units consisting of “Conv1D-BN-ReLU,” the model progressively extracts spatial feature representations from shallow to deep levels directly from the raw CIR data, thereby enhancing its ability to characterize differences in multipath structures under various propagation conditions. The architecture of the convolutional feature extraction module is shown in **Figure 2**.

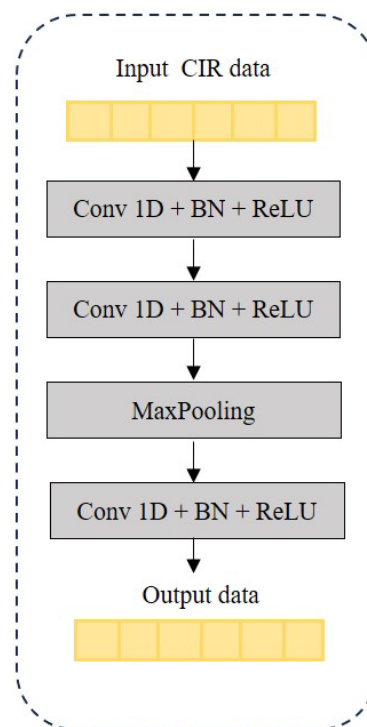


Figure 2. Convolutional feature extraction module.

Let X^l denote the input of the l -th basic block in the CNN. The local features extracted by the l -th block can be computed as follows:

$$Y_{Cov}^l = W^l \otimes X^l + b^l \quad (1)$$

$$Y_{BN}^l = BN_l(Y_{Cov}^l) \quad (2)$$

$$Y_{Cov}^l = W^l \otimes X^l + b^l \quad (3)$$

where W^l denotes the convolution kernel in the basic block, b^l represents the bias parameter of the convolutional layer, and \otimes denotes the convolution operation.

To reduce the dimensionality of the extracted features while preserving feature invariance, a max pooling operation is applied in the feature extraction stage. The adopted pooling function is a one-dimensional max pooling layer, which can be expressed as follows:

$$f_{\max\text{pool}}(x_i) = \max(x_{\text{stride} \times k + i}, x_{\text{stride} \times k + i + 1}, \dots, x_{\text{stride} \times k + i + m - 1}) \quad (4)$$

where m denotes the length of the pooling window, and stride represents the stride of the pooling operation along the input feature sequence. By controlling the sliding manner of the pooling window, this operation enables effective feature downsampling while retaining key information.

The input data X correspond to CIR sequences consisting of 1016 samples for each measurement. The detailed parameters of the convolutional layers are listed in **Table 1**.

Table 1. Detailed parameters of the convolutional layer.

Layer Type	Input channel	Output channel	Kernel size	Stride	Output shape
Input					1×1016
Convolution 1	1	64	11	1	64×1006
BN	64	64			64×1006
Activation	64	64			64×1006
Convolution 2	64	64	9	1	64×998
BN	64	64			64×998
Activation	64	64			64×998
MaxPool			2	2	64×499
Convolution 3	64	128	7	1	128×493
BN	128	128			128×493
Activation	128	128			128×493

2.2. Temporal Feature Modeling Module

After completing the extraction of spatial features from UWB signals, a temporal feature modeling module is further designed to model the temporal dependencies of the feature sequences output by the convolutional network. This

module is built upon LSTM networks and aims to characterize the dynamic evolution of UWB signals during the propagation process from a temporal perspective, thereby providing more discriminative feature representations for subsequent LOS/NLOS classification. The basic structure of an LSTM unit is illustrated in **Figure 3**.

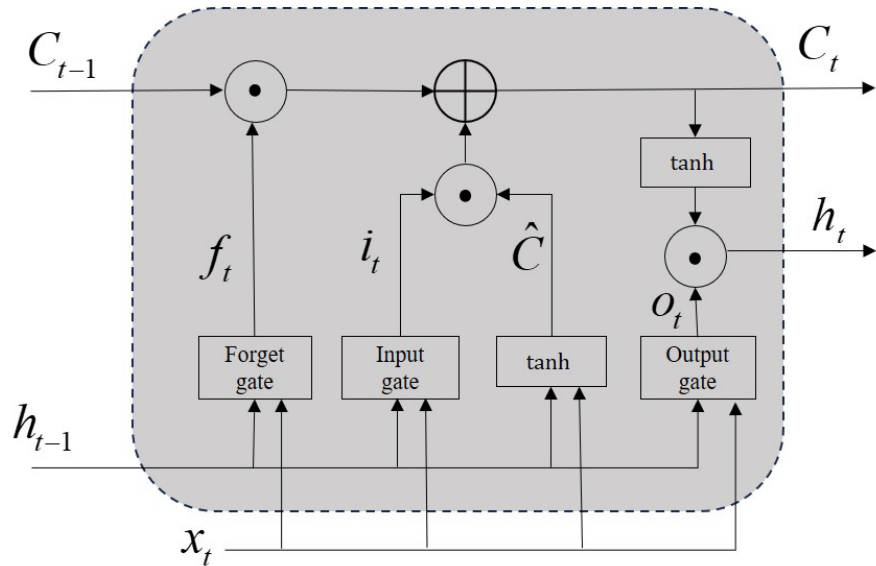


Figure 3. Long short-term memory network structure.

Specifically, an LSTM network mainly consists of an input gate, a forget gate, and an output gate, whose core function is to selectively memorize and update sequential information. The input gate controls the extent to which the current input information influences the cell state, the forget gate regulates the retention of historical memory from the previous time step, and the output gate determines the contribution of the current cell state to the external output.

At time step t , the input gate first computes the gating coefficient based on the current input vector \mathbf{x}_t and the hidden state from the previous time step \mathbf{h}_{t-1} . The expression is given as follows:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \tag{5}$$

here σ denotes the sigmoid activation function, \mathbf{W}_f represents the weight matrix, and \mathbf{b}_f is the bias vector. The output of the forget gate determines the proportion of information retained from the previous cell state \mathbf{C}_{t-1} .

The input gate is responsible for incorporating new input features, and its computation can be expressed as:

$$\mathbf{i}_t = \sigma(\mathbf{W}_i [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \tag{6}$$

The candidate cell state is obtained through a nonlinear transformation, which can be expressed as:

$$\hat{\mathbf{C}} = \tanh(\mathbf{W}_c [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c) \tag{7}$$

where \mathbf{W}_i and \mathbf{W}_c denote the weight matrices of the input gate and the candidate state, respectively, and \mathbf{b}_i and \mathbf{b}_c are the corresponding bias terms. The candidate cell state $\hat{\mathbf{C}}$ is used to store newly extracted feature information at the current time step.

By jointly considering the regulatory effects of the forget gate and the input gate, the cell state at the current time step is updated as:

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{C}} \quad (8)$$

where \odot denotes the element-wise multiplication operation.

Finally, the output gate is employed to control the influence of the cell state on the hidden layer output, and its computation is given by:

$$\mathbf{o}_t = \sigma(\mathbf{W}_o [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o) \quad (9)$$

The corresponding hidden state output is given by:

$$\mathbf{h}_{t-1} = \mathbf{o}_t \odot \tanh(\mathbf{C}_t) \quad (10)$$

In the proposed model, the output of the convolutional feature extraction module is organized as a temporally ordered feature sequence and fed into the long short-term memory network. Through the selective information transmission and update mechanisms of the input, forget, and output gates, the LSTM units are able to suppress irrelevant disturbances while retaining effective historical information. As a result, the network can accurately capture the intrinsic temporal evolution patterns of UWB signals.

This architecture exhibits significant advantages in distinguishing the continuous and stable signal characteristics under line-of-sight conditions from the non-stationary temporal features induced by multipath effects in non-line-of-sight environments.

2.3. Self-Attention Feature Enhancement Module

In this study, a self-attention mechanism is introduced to model the temporal features of UWB channel impulse responses, thereby enhancing the ability to capture their global contextual relationships. In this module, the input is the feature sequence obtained after convolutional feature extraction and temporal modeling by the LSTM network. The feature representations at each time step are no longer processed independently; instead, they interact with other positions in the sequence through the self-attention mechanism, yielding feature representations that incorporate global contextual information.

Essentially, the self-attention mechanism performs a weighted mapping process, where the queries, keys, and values are all derived from the same input feature sequence [16]. In this way, the model can adaptively emphasize the most critical temporal segments for LOS/NLOS classification while suppressing redundant or irrelevant features, thereby effectively enhancing the discriminative power of the feature representations. The structure of the self-attention feature enhancement module is illustrated in **Figure 4**.

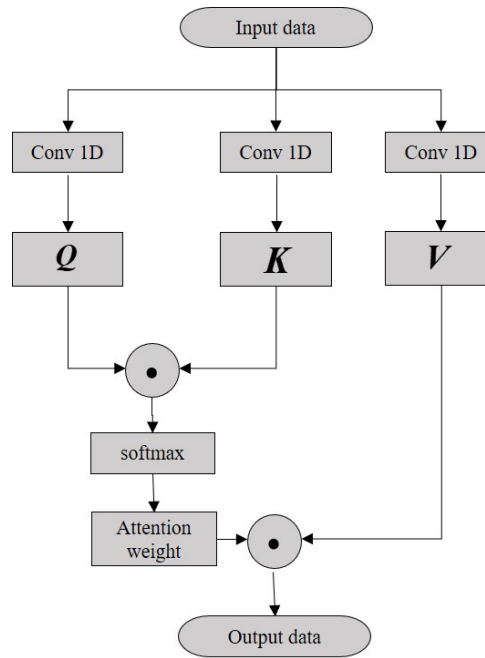


Figure 4. Schematic diagram of self-attention structure.

Assume the input data $\mathbf{X} = \{x_1, x_2, x_3, \dots, x_N\}$ is the temporal feature sequence output from LSTM network. The queries \mathbf{Q} , keys \mathbf{K} , and values \mathbf{V} are computed using independent convolutional functions $f_1(\mathbf{X})$, $f_2(\mathbf{X})$ and $f_3(\mathbf{X})$, respectively.

The computation process is as follows:

$$\begin{cases} \mathbf{Q} = f_1(\mathbf{X}) = \sum \mathbf{W}_1\mathbf{X} + \mathbf{B}_1 \\ \mathbf{K} = f_2(\mathbf{X}) = \sum \mathbf{W}_2\mathbf{X} + \mathbf{B}_2 \\ \mathbf{V} = f_3(\mathbf{X}) = \sum \mathbf{W}_3\mathbf{X} + \mathbf{B}_3 \end{cases} \quad (11)$$

where f_Q , f_K , and f_V are three convolutional functions, and their detailed parameters are listed in **Table 2**. The parameters of these three convolutional functions are independent and can be adaptively learned during training.

Table 2. Detailed parameters of the convolutional layer.

Layer Type	Input channel	Output channel	Kernel size	Stride	Output shape
Input data					128×493
Convolution function f1	128	16	1	1	16×493
Convolution function f2	128	16	1	1	16×493
Convolution function f3	128	16	1	1	16×493

The output of the attention mechanism is calculated as:

$$Score(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax} \left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_K}} \right) \mathbf{V} \quad (12)$$

where $d_{\mathbf{K}}$ is the dimensionality of each \mathbf{K} vector, and the softmax function is a generalization function that maps the elements of the input vector to corresponding values in the range of 0 to 1. Its computation is expressed as:

$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum e^{x_i}} \quad (13)$$

where x_i is an element of the input vector, and e is the base of the natural logarithm.

2.4. Fully Connected Classification Module

The fully connected (FC) layer is used to integrate and discriminate the feature vectors output by the attention mechanism, producing the final predicted labels corresponding to LOS/NLOS scenarios and outputting the classification results. The overall structure of this module is illustrated in **Figure 5**.

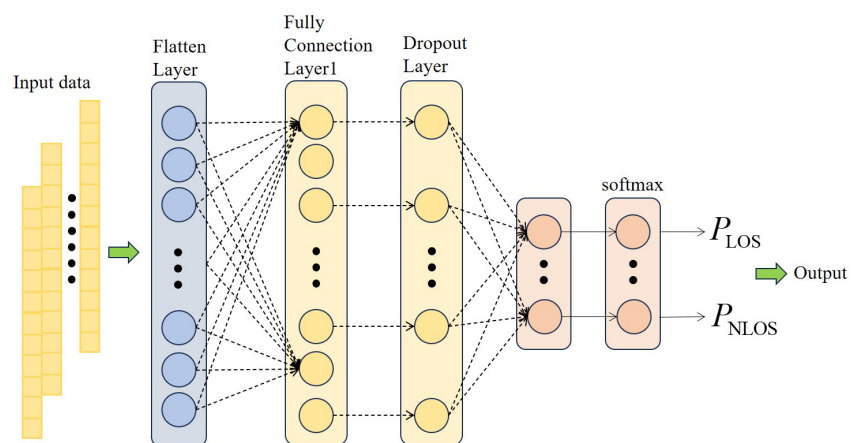


Figure 5. Schematic diagram of the fully connected layer structure.

In the implementation, a flattening layer first reshapes the features output from the self-attention feature enhancement module to prepare them for subsequent fully connected operations. Next, the first linear layer performs a linear mapping of the input vector through matrix operations, extracting higher-level feature representations. A dropout layer is then introduced, which randomly masks a portion of the neurons' outputs with a certain probability, effectively mitigating overfitting. Subsequently, a second linear layer further maps the features, projecting the high-dimensional representations into the target classification space.

The final layer contains two neurons corresponding to the NLOS and LOS classes, respectively, and employs the softmax activation function to output the class probability distribution. Let the input feature vector be \mathbf{x} ; then the classification output of the fully connected layer can be expressed as:

$$\text{label} = \text{argmax} \left(\text{softmax} \left(f_2 \left(\text{Dropout} \left(f_1(\mathbf{x}) \right) \right) \right) \right) \quad (14)$$

where f_1 and f_2 denote two linear transformation functions with learnable weights and biases. The Dropout operation randomly discards a portion of the

input features to enhance the model's generalization ability. The softmax function normalizes the network outputs to obtain the probability distribution of the sample belonging to the LOS and NLOS classes. Finally, the argmax function is used to select the class with the highest probability as the model's predicted result.

3. Experiments and Results

3.1. Data Description

UWB channel impulse response data collected in indoor environments are used for LOS/NLOS scenario classification. CIR data can effectively capture the multipath propagation characteristics of UWB signals between the transmitter and receiver. The energy distribution and arrival delays of different propagation paths contain rich information about the channel environment, making CIR suitable for distinguishing between LOS and NLOS conditions.

From a physical perspective, the CIR characterizes the impulse response of the wireless channel. Its baseband equivalent model can be represented as a superposition of multipath components [17]:

$$h(t) = \sum_{l=1}^L \sum_{k=1}^K \alpha_{k,l} e^{j\theta_{k,l}} \delta(t - T_l - \tau_{k,l}) \quad (15)$$

where L denotes the number of received UWB signal clusters, and K represents the number of multipath components within the l -th cluster; $\alpha_{k,l}$ and $\theta_{k,l}$ denote the path gain and phase of the k -th multipath component in the l -th cluster, respectively; T_l is the arrival time of the l -th cluster, and $\tau_{k,l}$ is the delay of the k -th multipath component relative to the cluster arrival time.

The CIR data of UWB essentially constitute a sequence of multipath responses sampled along the time-delay axis, which can describe the temporal distribution of signal energy. As shown in **Figure 6** and **Figure 7**, typical examples of UWB CIRs under NLOS and LOS scenarios are presented. It can be clearly observed that the two propagation environments exhibit significant differences in terms of the first-path energy, delay spread, and multipath structure.

The experimental data used in this study are derived from the eWINE project, which collected UWB channel impulse response datasets using the DW1000 UWB module [18]. Measurements were conducted under various transmitter-receiver distance conditions, and each CIR record contains 1016 time samples.

Due to variations in measurement conditions (e.g., transmission distance, obstacles, and environmental structures), the amplitude scale of the CIR signals differs across samples. Directly feeding the raw CIR amplitudes into the model may lead to unstable training. To eliminate amplitude scale discrepancies and enhance the model's sensitivity to structural features, the MAX_NOISE factor provided in the dataset was used to normalize the CIR signals. This normalization strategy reduces the influence of amplitude differences across measurement scenarios, improves the model's ability to capture multipath structure and first-path characteristics, and enhances both training stability and generalization performance.

The dataset includes a total of 42,000 valid CIR samples, encompassing both

NLOS and LOS channel measurements. To ensure fairness and generalization in model training and evaluation, the dataset was split into training, validation, and test sets in a 70%:15%:15% ratio, corresponding to 29,400, 6300, and 6300 CIR records, respectively.

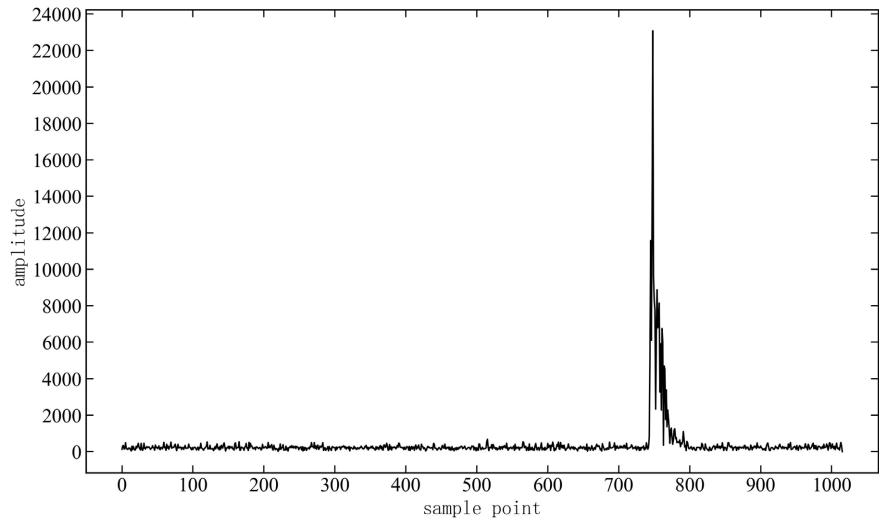


Figure 6. UWB CIR data in LOS scenarios.

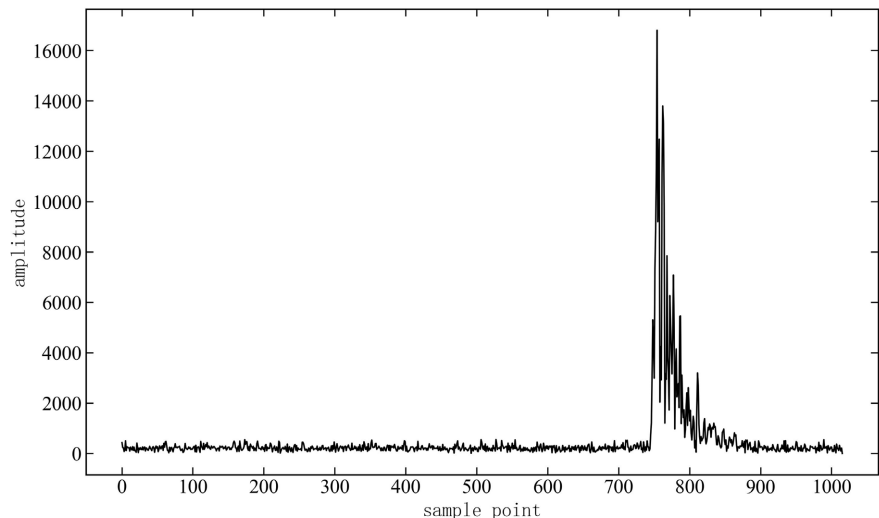


Figure 7. UWB CIR data in NLOS scenarios.

The training set is used for learning the model parameters, the validation set is employed for hyperparameter tuning and monitoring overfitting during training, and the test set is reserved for the final evaluation of the model's classification performance. In the test set, the number of NLOS and LOS samples is balanced, with 3150 records for each class, which facilitates an objective assessment of the model's performance under different channel propagation conditions. All experimental results are statistically analyzed based on the classification outputs of the test set.

3.2. Experimental Setup

To evaluate the effectiveness of the proposed model in the LOS/NLOS classification task of UWB CIR signals, several representative deep learning models were selected for comparison, including CNN [11], LSTM [6], CNN-LSTM [12], FCN [13], and LSTM-FCN [19].

During model training, the training data were divided into mini-batches, each containing 64 CIR records. The models were optimized using the Adam optimizer to improve training stability and convergence speed. The maximum number of training epochs was set to 100. During training, the validation set was used for hyperparameter tuning, and the model achieving the best validation performance was selected for final testing. All experiments were conducted under the same software and hardware environment to ensure fairness in comparison.

The experiments were conducted on a server running Windows 11 with 16 GB of system memory and equipped with an NVIDIA GeForce RTX 3050 GPU. The deep learning framework used in this study was PyTorch 2.5.0. The model architecture and training hyperparameters were configured as follows. At the model level, the LSTM module adopts a two-layer stacked structure, with 128 hidden units in the first layer and 64 hidden units in the second layer, enabling hierarchical extraction of temporal features while gradually reducing feature dimensionality. The attention mechanism employs Scaled Dot-Product Self-Attention tailored for sequential feature modeling. To mitigate overfitting, Dropout regularization is applied with a dropout rate of 0.5, and L2 regularization is incorporated through a weight decay coefficient of 1×10^{-4} .

During training, the Adam optimizer was utilized with an initial learning rate of 1×10^{-4} . A ReduceLROnPlateau learning rate scheduling strategy was adopted, using the validation F1 score as the monitoring metric. When the validation F1 score failed to improve for five consecutive epochs, the learning rate was reduced to 0.5 times its current value, with a minimum learning rate threshold set to 1×10^{-6} . To prevent overfitting and improve training efficiency, an early stopping mechanism based on the best validation F1 score was employed. Training was terminated if no performance improvement was observed for 15 consecutive epochs. The maximum number of training epochs was set to 100.

3.3. Evaluation Metrics

LOS/NLOS channel identification is a binary classification problem, and the classification results can be categorized into four cases:

True Positive (TP): LOS CIR data correctly classified as LOS.

True Negative (TN): NLOS CIR data correctly classified as NLOS.

False Positive (FP): NLOS CIR data incorrectly classified as LOS.

False Negative (FN): LOS CIR data incorrectly classified as NLOS.

In the experiments, five performance metrics were used to evaluate the classification algorithms: accuracy (ACC), precision, recall, specificity, and F1-score. The

formulas for computing these metrics are as follows:

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (16)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (17)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (18)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (19)$$

$$\text{F1-Score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \quad (20)$$

3.4. Ablation Study

To verify the effectiveness of each key module in the proposed model, an ablation study was conducted. Specifically, the CNN feature extraction module, the LSTM temporal modeling module, and the self-attention module were individually removed, and the resulting performance changes were compared under identical experimental conditions. The results of this study are presented in **Table 3**. The corresponding classification confusion matrices are shown in **Figure 8**.

Table 3. Performance metrics of the ablation study.

Model Variant	CNN Module	LSTM Module	Self-Attention Module	Accuracy	Precision	Recall	F1-Score
CNN Only	√	×	×	0.8233	0.8306	0.8124	0.8214
LSTM Only	×	√	×	0.8002	0.8053	0.7917	0.7985
CNN + LSTM	√	√	×	0.8775	0.8843	0.8686	0.8764
CNN + Attention	√	×	√	0.8541	0.8614	0.8441	0.8527
LSTM + Attention	×	√	√	0.8452	0.8509	0.8371	0.8440
Full Model	√	√	√	0.8995	0.9107	0.8857	0.8981

3.5. Comparative Experiments and Analysis

To further validate the effectiveness of the proposed model, comparative experiments were conducted with several other algorithms. **Table 4** presents the classification results of each method. From the data in **Table 4** and **Table 5**, it can be observed that the overall performance of different models in the NLOS/LOS classification task is comparable, but differences still exist in terms of misclassification distribution and class balance.

Based on the statistical results, the proposed method maintains high levels of both true positives and true negatives, while the number of false positives is

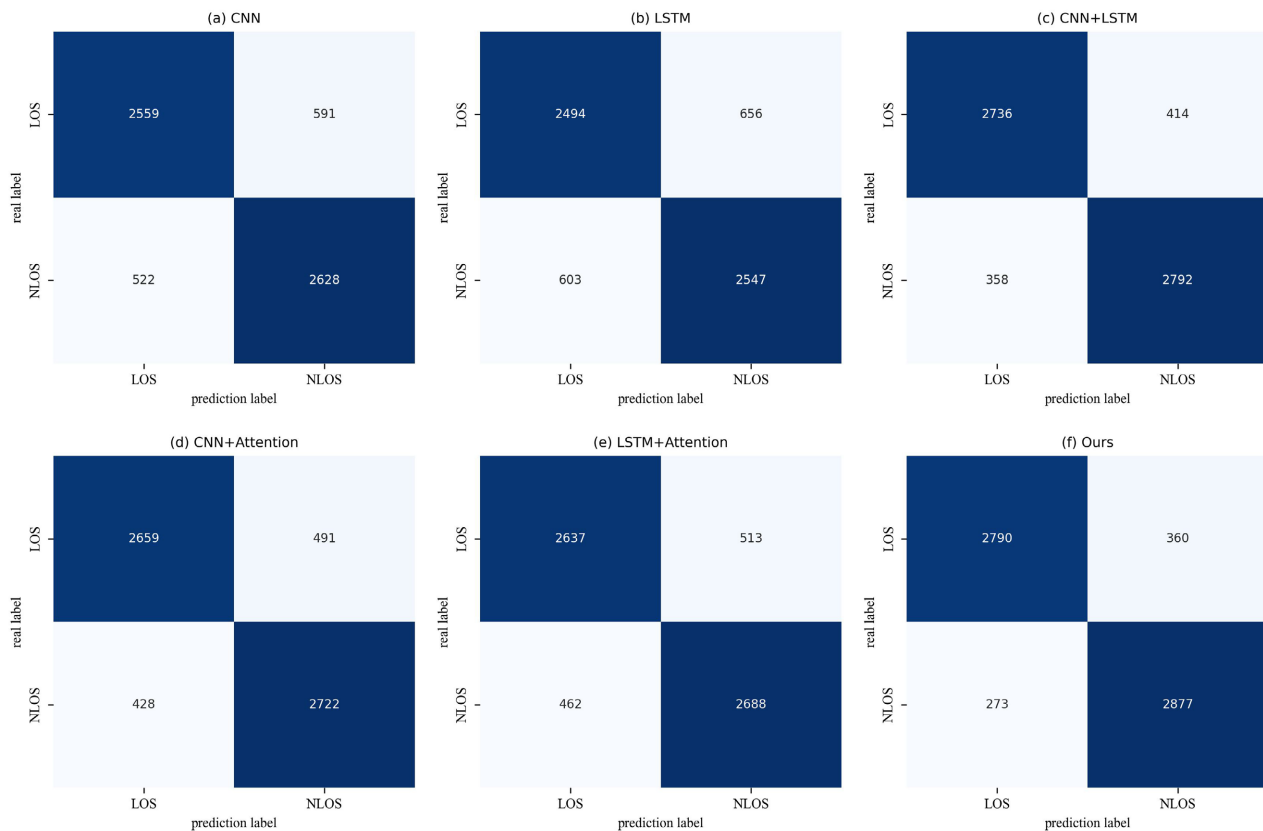


Figure 8. Confusion matrices of the ablation study.

Table 4. NLOS/LOS classification result.

algorithm	TP	TN	FP	FN
CNN	2819	2626	558	297
LSTM	2776	2437	747	340
CNN-LSTM	2753	2484	699	364
FCN	2821	2663	521	295
LSTM-FCN	2808	2664	520	308
Ours	2790	2877	273	360

significantly lower than most of the compared models. This indicates that the proposed approach has a clear advantage in reducing the risk of misclassifying NLOS samples as LOS. Although the proposed model does not show an exaggerated improvement in classification performance compared with conventional methods such as CNN and LSTM, its overall performance is more stable, demonstrating better robustness in distinguishing CIRs under complex propagation environments.

To further understand the classification performance of the proposed model, an error analysis was conducted on the FP and FN in the test set. It was observed that false positives primarily occur in scenarios where the first-path energy is weak or the signal is partially obstructed, causing NLOS signals to be misclassi-

fied as LOS. False negatives mainly arise in short-distance scenarios or under severe multipath interference, where LOS signals with significant multipath components are mistakenly classified as NLOS. These error modes may affect downstream ranging or positioning applications. For example, misclassifying NLOS as LOS could lead to underestimation of ranging errors, whereas misclassifying LOS as NLOS might trigger overly conservative interference mitigation strategies, reducing positioning efficiency. Understanding these FP and FN patterns can guide the design of more robust weighting or filtering strategies in positioning algorithms, thereby mitigating the impact of classification uncertainty on system performance.

As shown in **Table 5**, the proposed algorithm demonstrates balanced performance across multiple evaluation metrics. In particular, the precision and specificity reach 91.07% and 91.33%, respectively, the highest among all compared methods, indicating a clear advantage in reducing misclassification and improving discriminative reliability. Meanwhile, the accuracy and F1-score also remain at high levels, further confirming the overall effectiveness of the proposed approach.

Table 5. Evaluation indicators for each algorithm.

algorithm	Accuracy	Precision	Recall	specificity	F1-Score
CNN	0.8635	0.8347	0.9047	0.8248	0.8690
LSTM	0.8289	0.7881	0.8909	0.7653	0.8365
CNN-LSTM	0.8329	0.7975	0.8832	0.7802	0.8383
FCN	0.8673	0.8439	0.9053	0.8360	0.8735
LSTM-FCN	0.8654	0.8437	0.9012	0.8361	0.8715
Ours	0.8995	0.9107	0.8857	0.9133	0.8981

Based on the evaluation of discriminative capability, Gaussian noise with standard deviations of 0.01, 0.05, 0.10, and 0.20 was added to the test set to assess the model's robustness under noisy conditions. Comparisons were made with CNN, LSTM, CNN-LSTM, FCN, and LSTM-FCN models. The experimental results are shown in **Figure 9**.

The results indicate that when the noise standard deviation reaches 0.20 or 0.30, the performance of all models significantly decreases as the noise intensity increases. However, the proposed model maintains a relatively high accuracy even under the strongest noise, with only a 1.8% decrease. This performance surpasses that of other models, demonstrating its notable robustness and stability, as well as strong potential for practical applications.

In addition, **Figure 10** presents the receiver operating characteristic (ROC) curves for CNN, LSTM, CNN-LSTM, FCN, and LSTM-FCN on the dataset to further evaluate model effectiveness. It is evident that the area under the curve (AUC) of the proposed model is higher than that of the other five models.

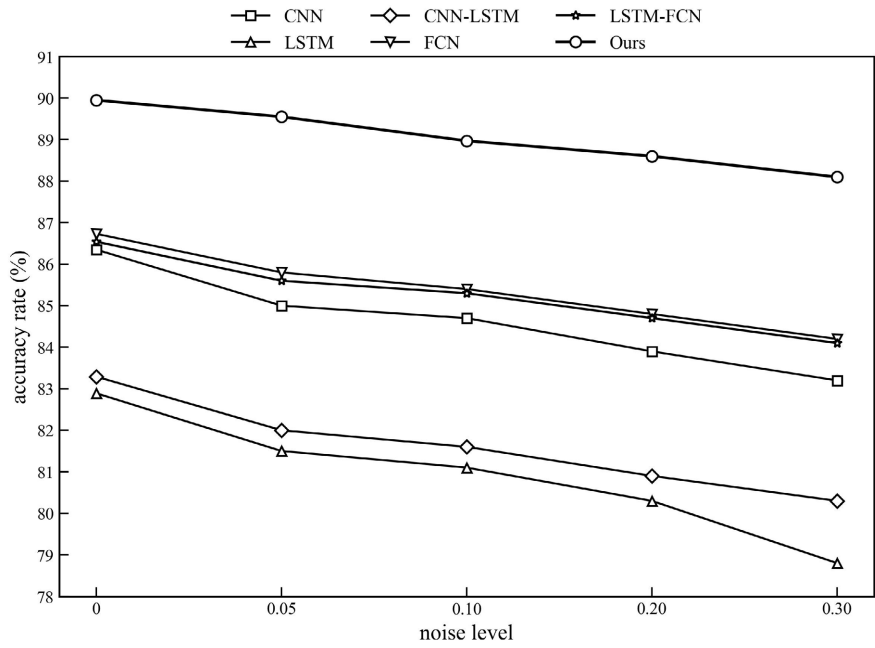


Figure 9. Comparison of model accuracy under different noise levels.

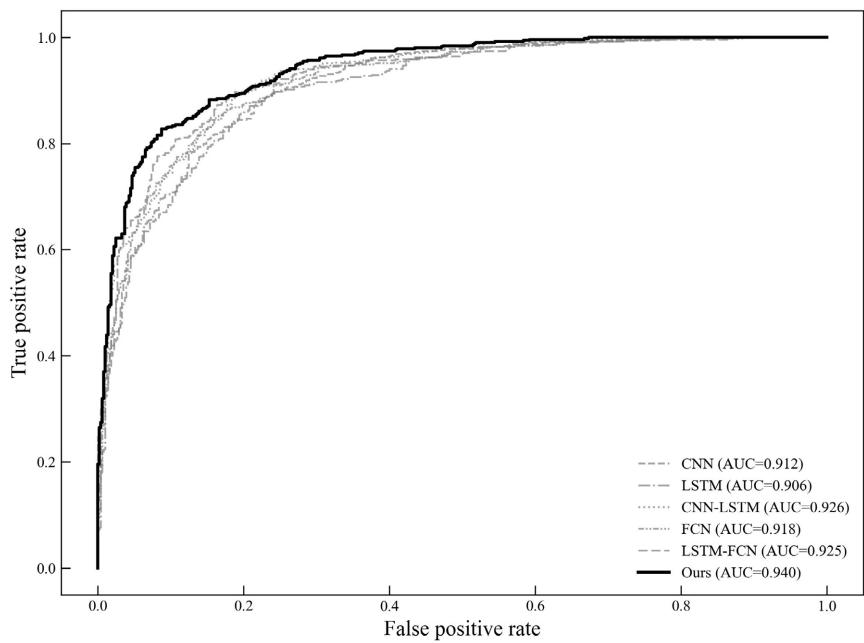


Figure 10. ROC curves of different models.

As shown in Figure 11, the experimental training loss curves, validation loss curves, and accuracy curves on the training and validation sets indicate that the proposed model achieves superior classification performance compared with the other models.

The above experiments and evaluation metrics indicate that, on the eWINE dataset, the proposed algorithm outperforms the other compared methods. Overall,

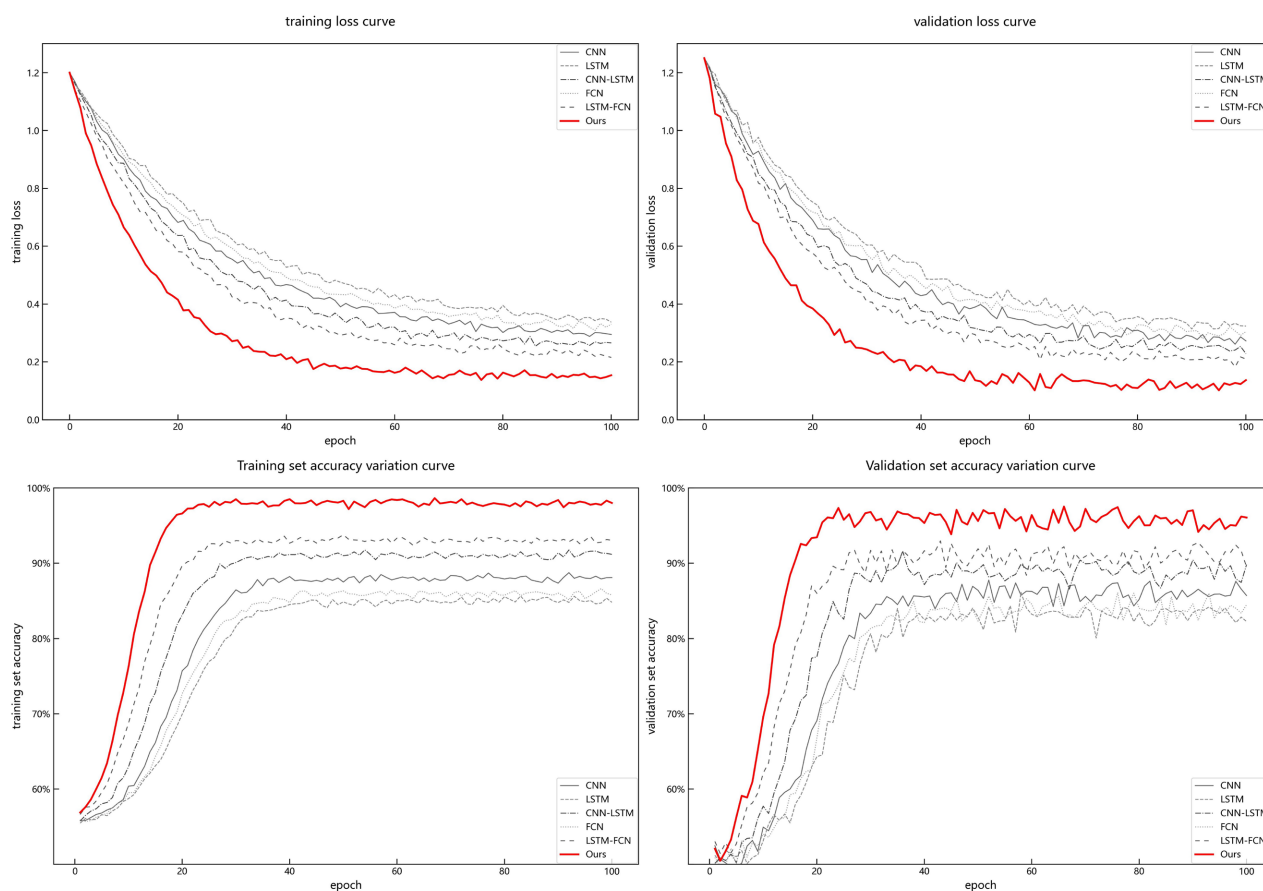


Figure 11. Experimental result curves.

the proposed algorithm not only maintains strong overall classification performance but also demonstrates superior control of misclassifications and result stability, thereby validating its effectiveness and practical value in UWB NLOS signal recognition tasks.

4. Conclusions

In summary, the deep learning-based UWB signal LOS/NLOS recognition method proposed in this study demonstrates strong classification performance on the eWINE dataset. The proposed approach achieves high levels across multiple evaluation metrics, including accuracy, precision, specificity, and F1-score, and maintains stable and competitive performance compared with various mainstream neural network models. This method can effectively extract discriminative features from UWB CIR signals and reliably identify LOS and NLOS conditions in complex propagation environments, validating the effectiveness and practical applicability of the proposed model for UWB scenario perception tasks.

Although the proposed method achieves promising results in the LOS/NLOS classification task, there remain areas for further improvement. First, the experimental validation is primarily based on a publicly available dataset, which has limited scale and scene diversity. The model's generalization capability in more com-

plex or large-scale real-world environments requires further investigation. Future work could incorporate multi-scene, multi-device, and diverse indoor-structure datasets to enhance the model's robustness and adaptability. Second, this study focuses mainly on LOS/NLOS state recognition itself, without deeply integrating the results with specific UWB indoor positioning algorithms. Future research could incorporate the recognition results into positioning error modeling and weighted localization frameworks, further improving indoor positioning accuracy and stability. Moreover, there is still room for optimization in terms of computational complexity and real-time performance, highlighting the need to explore more efficient network architectures to meet practical system deployment requirements.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Xue, W.L., Zhao, D. and Zhang, Y.C. (2020) Review of Research on Indoor Positioning. *Computer and Modernization*, No. 5, 80-88.
- [2] Dai, Y.P. and Liu, T.Y. (2024) Review and Development Prospect of Indoor Positioning Technology. *Digital Communication World*, No. 8, 89-91, 100.
- [3] Moon, S. and Youn, W. (2022) A Novel Movable UWB Localization System Using UAVs. *IEEE Access*, **10**, 41303-41312. <https://doi.org/10.1109/access.2022.3164701>
- [4] Morón, P.T., Salimpour, S., Fu, L., Yu, X., Queraltá, J.P. and Westerlund, T. (2023) Benchmarking UWB-Based Infrastructure-Free Positioning and Multi-Robot Relative Localization: Dataset and Characterization. 2023 *IEEE Sensors Applications Symposium (SAS)*, Ottawa, 18-20 July 2023, 1-6. <https://doi.org/10.1109/sas58821.2023.10254018>
- [5] Cong, F.Y., Hong, Z.H., Tang, Z., *et al.* (2022) An Optimization Method of UWB Ranging Based on Pulse Response Disambiguation and Reconstruction. *Chinese Journal of Scientific Instrument*, **43**, 13-21.
- [6] Kim, D., Farhad, A. and Pyun, J. (2023) UWB Positioning System Based on LSTM Classification with Mitigated NLOS Effects. *IEEE Internet of Things Journal*, **10**, 1822-1835. <https://doi.org/10.1109/jiot.2022.3209735>
- [7] Lee, K., Oh, J. and You, K. (2016) Closed-Form Solution of TDOA-Based Geolocation and Tracking: A Recursive Weighted Least Square Approach. *Wireless Personal Communications*, **94**, 3451-3464. <https://doi.org/10.1007/s11277-016-3785-8>
- [8] Yang, X., Zhao, F. and Chen, T. (2018) NLOS Identification for UWB Localization Based on Import Vector Machine. *AEU—International Journal of Electronics and Communications*, **87**, 128-133. <https://doi.org/10.1016/j.aeue.2018.02.003>
- [9] Yu, K., Wen, K., Li, Y., Zhang, S. and Zhang, K. (2019) A Novel NLOS Mitigation Algorithm for UWB Localization in Harsh Indoor Environments. *IEEE Transactions on Vehicular Technology*, **68**, 686-699. <https://doi.org/10.1109/tvt.2018.2883810>
- [10] Wei, Z.H., Liao, G., Li, M.X., *et al.* (2024) NLOS Identification Technique Based on K-Means Clustering Algorithm Improved by ISODATA. *Electronic Measurement Technology*, **47**, 172-180.
- [11] Jiang, C., Chen, S., Chen, Y., Liu, D. and Bo, Y. (2020) An UWB Channel Impulse

- Response De-Noising Method for NLOS/LOS Classification Boosting. *IEEE Communications Letters*, **24**, 2513-2517. <https://doi.org/10.1109/lcomm.2020.3009659>
- [12] Jiang, C., Shen, J., Chen, S., Chen, Y., Liu, D. and Bo, Y. (2020) UWB NLOS/LOS Classification Using Deep Learning Method. *IEEE Communications Letters*, **24**, 2226-2230. <https://doi.org/10.1109/lcomm.2020.2999904>
- [13] Pei, Y., Chen, R., Li, D., Xiao, X. and Zheng, X. (2023) FCN-Attention: A Deep Learning UWB NLOS/LOS Classification Algorithm Using Fully Convolution Neural Network with Self-Attention Mechanism. *Geo-Spatial Information Science*, **27**, 1162-1181. <https://doi.org/10.1080/10095020.2023.2178334>
- [14] Cui, Z., Gao, Y., Hu, J., Tian, S. and Cheng, J. (2021) LOS/NLOS Identification for Indoor UWB Positioning Based on Morlet Wavelet Transform and Convolutional Neural Networks. *IEEE Communications Letters*, **25**, 879-882. <https://doi.org/10.1109/lcomm.2020.3039251>
- [15] Wang, J., Yu, K., Bu, J., Lin, Y. and Han, S. (2022) Multi-Classification of UWB Signal Propagation Channels Based on One-Dimensional Wavelet Packet Analysis and CNN. *IEEE Transactions on Vehicular Technology*, **71**, 8534-8547. <https://doi.org/10.1109/tvt.2022.3172863>
- [16] Hwang, H., Jeong, S. and Hwang, W. (2025) Ultrawideband Non-Line-Of-Sight Classification Using Transformer-Convolutional Neural Networks. *IEEE Access*, **13**, 86323-86332. <https://doi.org/10.1109/access.2025.3568830>
- [17] Al-Sammna, A.M., Hadri Azmi, M. and Abd Rahman, T. (2018) Time-Varying Ultra-Wideband Channel Modeling and Prediction. *Symmetry*, **10**, Article 631. <https://doi.org/10.3390/sym10110631>
- [18] Bregar, K. and Mohorcic, M. (2018) Improving Indoor Localization Using Convolutional Neural Networks on Computationally Restricted Devices. *IEEE Access*, **6**, 17429-17441. <https://doi.org/10.1109/access.2018.2817800>
- [19] Karim, F., Majumdar, S., Darabi, H. and Chen, S. (2018) LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access*, **6**, 1662-1669. <https://doi.org/10.1109/access.2017.2779939>