Hu *et al. J Wireless Com Network* (2021) 2021:95
https://doi.org/10.1186/s13638-021-01900-y

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

IDSDL: a sensitive intrusion detection system based on deep learning

Yanjun Hu[1], Fan Bai[1], Xuemiao Yang[1] and Yafeng Liu[2*]

*Correspondence:
yafeng@cumt.edu.cn
[2] IoT Perception Mine Research Center, China University of Mining and Technology, Xuzhou 221008, China
Full list of author information is available at the end of the article

Abstract

Device-free passive (DfP) intrusion detection system is a system that can detect moving entities without attaching any device to the entities. To achieve good performance, the existing algorithms require proper access point (AP) deployment. It limits the applying scenario of those algorithms. We propose an intrusion detection system based on deep learning (IDSDL) with finer-grained channel state information (CSI) to free the AP position. A CSI phase propagation components decomposition algorithm is applied to obtain blurred components of CSI phase on several paths as a more sensitive detection signal. Convolutional neuron network (CNN) of deep learning is used to enable the computer to learn and detect intrusion without extracting numerical features. We prototype IDSDL to verify its performance and the experimental results indicate that IDSDL is effective and reliable.

Keywords: Passive intrusion detection, Channel state information (CSI), WiFi, Deep learning, Convolutional neural network (CNN)

1 Introduction

Passive intrusion detection technique has become one of the current research hotspots due to its characteristic of detecting entities without carrying any device. The characteristic means that the objects are not required to take any device during the detection process. There are existing methods that can achieve intrusion detection. Savkin and Huang [1] proposed a method to camera surveillance for full coverage over uneven areas. It is a camera-based method. However, camera-based method needs the object to move within the visible range of the camera. Furthermore, its main issues include the potential privacy leakage and high false alarm rate. Compared with camera-based detection approach, WiFi-based method can detect moving entities whether the entities are visible. This is the most significant advantage of WiFi-based method over camera-based method. Among the WiFi-based methods, CSI-based method has drawn increasing attention on motion detection and target localization [2–5].

WiFi-based passive intrusion detection is a system using signals which can be easily affected by human motion [6]: received signal strength (RSS) and channel state information (CSI). Received signal strength (RSS) [7] is widely used as the source signal in early passive detection due to its accessibility. Since RSS signal is susceptible to environmental interference and less sensitive to different motions of human body [8], passive detection of moving humans with dynamic speed (PADS) [9] and phase mode (PM) [10] use finer-grained CSI in passive intrusion detection. In contrast, CSI is stable to static environments and sensitive to dynamic environments, so it is a better detection signal.

Previous passive detection algorithms extract features such as mean and variance [11, 12] from RSS and CSI data to accomplish the function. Their core is to determine whether there is a moving human in the monitoring area or not. Although the algorithms work effectively and accurately [10], they usually detect human motion that occurs on line-of-sight (LOS). Therefore, those algorithms require the AP to be placed near the doorway or other places where human motion will directly influence CSI near LOS path. This makes the effectiveness of the whole system compromised when the intruder does not pass through the LOS path.

To decrease the high requirement for AP deployment, we use CSI phase as the source signal, extract propagation components on independent paths as the detection signal, and design a CNN to construct IDSDL. Then the performance of this system is verified on commodity WiFi devices. The result demonstrates that IDSDL is able to detect human motion on non-line-of-sight (NLOS) path effectively.

The rest of this paper is organized as follows. We review related work in Sect. 2. Preliminaries about CSI and convolutional neuron network (CNN) are mentioned in Sect. 3. We introduce the methodology in detail in Sect. 4. Experiment settings and performance evaluation are elaborated in Sect. 5. Conclusions are drawn in Sect. 6.

2 Related work

The idea of passive intrusion detection is to judge the monitoring area is intruded or not by analyzing human's impacts on wireless signal.

When the wireless signal passes through human body, received signal strength (RSS) undergoes certain changes and fluctuations so that it could be used in passive detection. Youssef and Moussa [6, 13] introduced the concept of device-free detection and proposed a maximum likelihood estimation-based algorithm (MLE) to improve the performance of device-free passive (DfP) system in real-world environments. The most well-known RSS-based motion detection, Radio Tomography Imaging (RTI) [11], which deployed a wireless sensor network in the target area to measure RSS to obtain an image of the moving people was proposed. On the foundation of RTI, variance-based radio tomography imaging (VRTI) [12] used the changes in RSS variance caused by people motion for motion detection. RASID [14] system used RSS standard deviation to improve detection accuracy. Since RSS is too sensitive to tiny changes in the environment, a device-free localization (DfL) [15] algorithm based on differential RSS, was proposed to overcome RSS's negative impact on environment.

Due to the coarse granularity and limited precision of RSS, people have cast their sights on fine-grained CSI. CSI was firstly applied in location technology. Pilot [16] tried device-free localization, using CSI and time correlation to detect anomalies in the channel and then locate the object. Apart from the location technology, CSI start to be utilized in motion detection. Fine-grained Device-free Motion Detection (FIMD) [17] achieved the accurate detection of sudden motion by using the time stability of CSI in static environment. Since most studies only make effective use of CSI amplitude, PADS

[9] used a method that sanitizes CSI phase, then uses both CSI amplitude and phase in detection. PM [10] utilized a method based on PADS to extract the phase difference between adjacent antenna pairs as the detection signal for passive intrusion detection.

Previous researches extract certain numerical features from CSI signals for passive detection and require the AP to be arranged in proper places. Therefore, when human motion occurs on NLOS paths, their effectiveness would be reduced. As a result, if there is a person intrude from the door and the APs are placed away from the door, detection effect is not satisfactory. To overcome this disadvantage, we propose IDSDL, an algorithm that is able to sensitively detect human motion on NLOS.

3 Preliminaries

In this section, we briefly introduce necessary information about CSI and analyze the impact of multi-path effect on CSI.

3.1 Channel state information

In the wireless communication system, CSI describes the channel properties of the communication link in the subcarrier level and represents what a signal undergoes while passing through the subcarriers. In the orthogonal frequency-division multiplexing (OFDM) system, after passing through the multi-path channel, the received signal can be represented as:

$$Y = HX + N. \tag{1}$$

X and Y are signal vectors of the transmitting terminal and the receiving terminal, respectively, N represents additive Gaussian white noise in the channel and H represents a channel matrix. H is shown as follows:

$$H = [\overrightarrow{H}(f_1), \overrightarrow{H}(f_2), \ldots, \overrightarrow{H}(f_N)]. \tag{2}$$

where $\overrightarrow{H}(f_n), n \in [1, N]$ is the channel frequency response (CFR) in the frequency domain on each subcarrier and N is the total number of subcarriers.

The channel frequency response can also be expressed as

$$H(f) = \sum_{i=0}^{N-1} a_i \mathrm{e}^{-j(\theta_i + 2\pi f \tau_i)}. \tag{3}$$

where a_i and $\theta_i + 2\pi f \tau_i$ are the CSI amplitude and phase on the ith subcarrier, respectively, f is frequency, N is the total number of subcarriers and τ is the delay.

3.2 Difference between human motion on LOS and NLOS

The received CSI is a multipath [18] synthesized signal, and the CSI will undergo different changes when the signals on different paths are interfered. Assuming that the wireless signal propagation path in space is shown in Fig. 1, the intrusion from the door will only affect a small number of CSI signals on NLOS paths, while the LOS path is not affected.

Numerical features used in previous researches such as mean, variance and covariance would be large when human motion happens on LOS. But these values would not be that

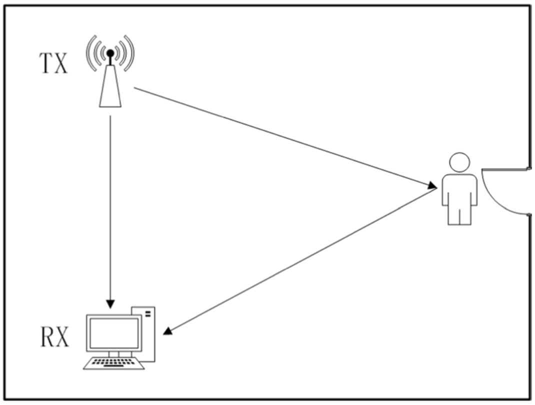


Fig. 1 Multipath effect of CSI

obvious when human motion happens on NLOS, especially on paths far away from LOS. Therefore, simple numeral features would not be effective. As shown in Fig. 2, we calculate the largest six eigenvalues of covariance matrix extracted from CSI phase, which is the feature utilized in detection in the literature [9]. It is obvious that the values obtained when human motion occurs on NLOS far from LOS are not much larger than those got when the monitoring area is clear, while the values of human motion on LOS is an order of magnitude larger. To solve this problem, we need a new kind of detection feature. We first use path decomposition to decompose the CSI signal to analyze the changes of the CSI signal on a certain independent path. The path-decomposed CSI signal can be used as a sensitive signal for intrusion detection because it can sensitively detect the change of the channel when the LOS signal does not change so much. Then we convert the signal into feature image instead of extracting numerical features because feature image retains not only the waveform characteristics of signals but also the timing characteristics of signal changes if we convert the signals of multiple consecutive moments into one feature image. This allows CNN to judge the occurrence of intrusion behavior based on signal changes at multiple times.

3.3 Convolutional neuron network

Deep learning is being used more and more in wireless communication networks, and it is commonly used to solve optimization problems in wireless communication networks. Chen et al. [19] formulates the joint learning, wireless resource allocation, and user selection problem as an optimization problem whose goal is to minimize an federated learning (FL) loss function that captures the performance of the FL algorithm. Yang et al. [20] investigates the problem of energy efficient transmission and computation resource allocation for FL over wireless communication networks. Wang et al. [21] studies the problem of optimizing the deployment of unmanned aerial vehicles (UAVs) equipped with visible light communication (VLC) capabilities.

In addition to optimization problem, deep learning is also used to solve classification problem. CNN is a neural network model that is useful for classifying images. CNN evolved from multi-layer perceptron (MLP) [22]. LeCun et al. [23] combined the idea

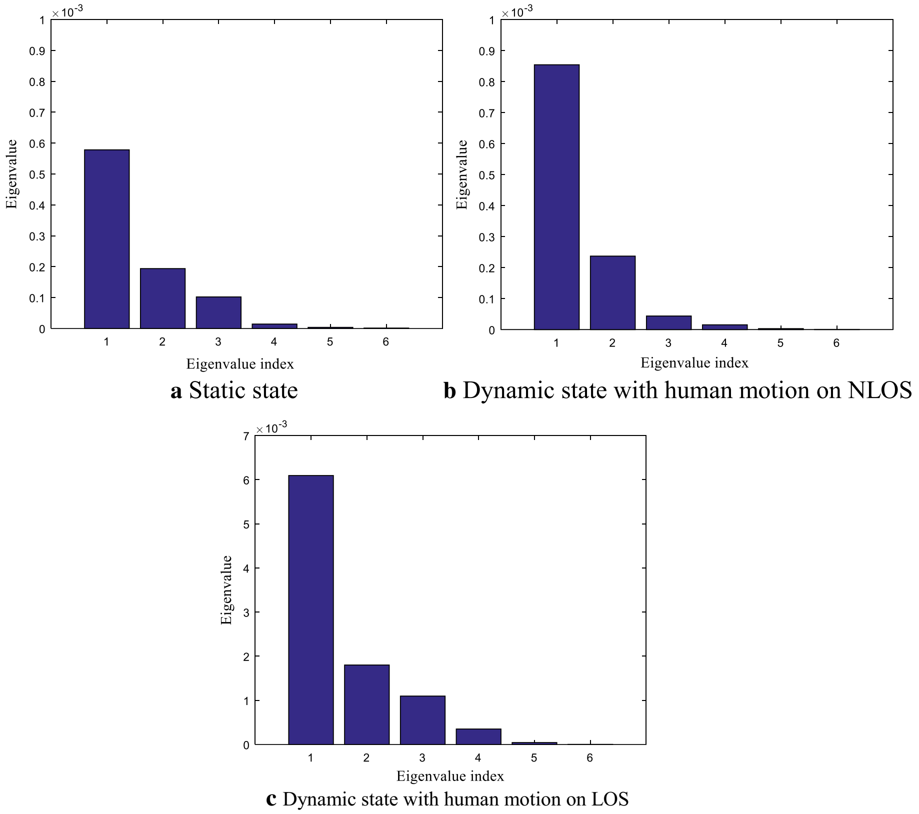


Fig. 2 Largest eigenvalues of covariance matrix from CSI phase in three situations

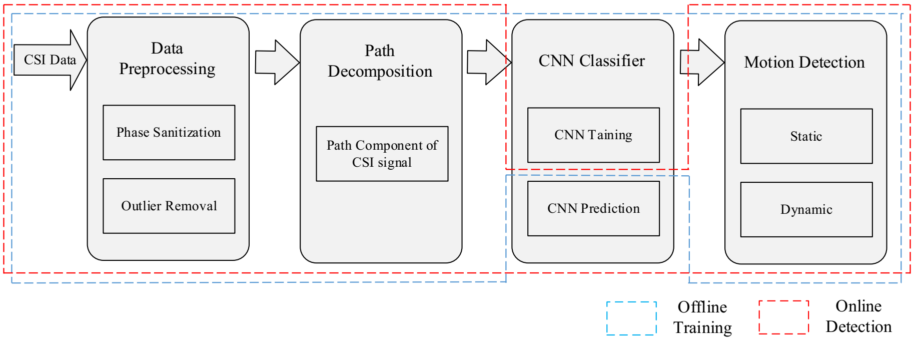


Fig. 3 System diagram

of back-propagation algorithm and weight sharing to invent the CNN network and successfully adapted it to the handwritten character detection system of the US Post Office for the first time.

The using process of CNN is as follows:

(1) Create a dataset consisting of annotated images, or use an existing dataset. Annotations can be image categories (for classification problems), bounding boxes and classes (for object detection problems), or pixel-level segmentation for each object

of interest in the image (for instance segmentation problems). In our manuscript, the annotation is image category because we consider the intrusion detection as a classification problem.

(2) Features related to the current task are extracted from each image. This is the key point of modeling. For example, the features used to identify human faces are significantly different from those used to identify tourist attractions. Fortunately, the feature selection and extraction are learned automatically during the training process.

(3) Train a CNN model. Training means providing many images to a CNN model, which will help CNN model learn how to solve the task at hand by using the features extracted from training images.

(4) Evaluate the model by utilizing images not used during the training phase. By doing so, we test the effectiveness of the trained model.

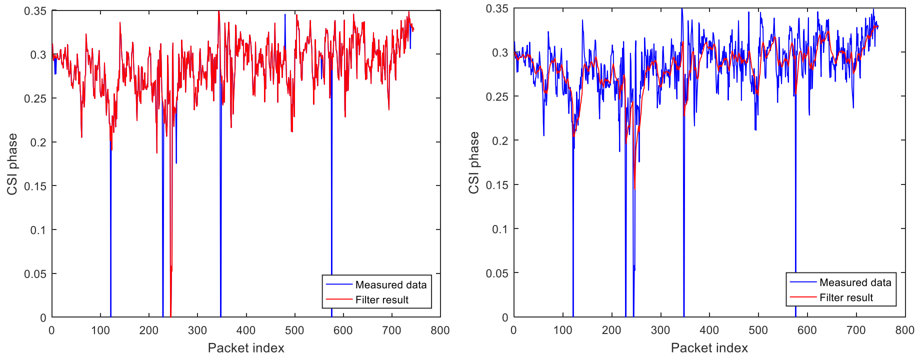
Due to its characteristics of local connection and weight sharing, CNN is more suitable for processing image data. Weight sharing enables information to be shared between neurons and reduces the number of parameters significantly. Local connection is different from full connection. Local connection is that the neurons in layer n-1 only have connections with some neurons in layer n, instead of all of them, which reduces the probability of overfitting.

CNN consists of convolutional layers, pooling layers, and fully connected (dense) layers. Each convolutional layer contains multiple feature maps. Each feature map is a "plane" composed of multiple neurons, which extracts features of the input through a convolution filter. The sampling layer is also called the pooling layer, and its role is to perform subsampling based on the principle of local correlation, so as to reduce the amount of data while retaining useful information. By applying the convolutional layers and the sampling layers, CNN maps the original image into a feature vector and finally completes the recognition task in fully connected layers and the output layer. CNN's construction method of multi-hidden layer stacking, that makes each layer of CNN processes the output of the previous layer, can be regarded as that CNN processes the input signal layer by layer. The initial input representation that is not closely related to the output target is changed into a representation that is more closely related to the output target, making it possible to perform tasks that were difficult to complete based only on the last layer of output mapping previously.

CNN is widely used for feature extraction of pictures due to its special network construction and excellent performance. It can not only automatically extract features, but also have strong generalization capabilities.

4 Methodology

IDSDL mainly includes four parts: data preprocessing, path decomposition of CSI signal, CNN classification and motion detection, shown as Fig. 3. The system monitors CSI in the WiFi wireless network space, uses path components to reflect CSI changes sensitively and classifies the data by utilizing CNN to carry out motion detection. The system



a Outlier removal with Hampel filter **b** Outlier removal with Kalman filter

Fig. 4 Outlier removal

contains offline training and online detection. The CNN classifier is trained in the offline phase and new CSI data is detected in the online phase.

4.1 Data preprocessing

(1) Phase sanitization

Due to the excessive offset error [24, 25] included in CSI phase, phase sanitization is required to effectively utilize CSI phase [9]. The measured phase P_{ci} can be linearly transformed to obtain raw phase P_{zi}:

$$P_{zi} = P_{ci} - ak_i - b = P_i - \frac{P_n - P_1}{k_n - k_1}k_i - \frac{1}{n}\sum_{j=1}^{n} P_j. \tag{4}$$

Compared with P_{ci}, P_{zi} eliminates the random phase offset and has effective features necessary for intrusion detection in true phase simultaneously.

(2) Outlier removal

Due to the influence of random noise, outliers appear in the measured CSI data. In order not to affect the performance of the system, filter is used to remove outliers.

Hampel filter [26]: All the data that do not belong to the interval $[\mu - \gamma\sigma, \mu + \gamma\sigma]$ are regarded as outliers supposed to be removed, where μ is the median, σ is the absolute deviation of the median data sequence, γ is a parameter usually set to 3.
Kalman filter [27]: Kalman filtering is a state equation of a linear system. Its basic idea is to use the input and output of the system to estimate the state of the entire linear system. Since the noise signal is doped in the measured data, the process of this optimal estimation is also regarded as a filtering process.

No matter which filtering method is adopted, outlier removal will improve system performance by reducing some unexpected errors. Figure 4 shows the filter results obtained by using Hampel filter and Kalman filter.

4.2 Path decomposition of CSI phase

According to the analysis in Preliminary, human motions on LOS and NLOS cause different changes in CSI. To extract a detection feature that is not overly affected by the LOS signal, we propose a path decomposition algorithm. The path decomposition is used to decompose the CSI signal for analyzing the changes of the CSI signal on a certain independent path. The path-decomposed CSI signal can sensitively detect the change of the channel when the LOS signal does not change so much. Therefore, it can be used as a sensitive signal for intrusion detection.

Let CSI_k denote the CSI on kth subcarrier, CSI_0 denote the CSI on the zeroth subcarrier, which is a virtual subcarrier whose frequency is f_0. For path l, we introduce two symbols S_0^l and Δ_k^l:

$$\begin{aligned} S_0^l &= \alpha_l \mathrm{e}^{-j2\pi f_0 \tau_l}. \\ \Delta_k^l &= \mathrm{e}^{-jk\Delta f \tau_l}. \end{aligned} \tag{5}$$

where α_l is the amplitude of CSI_k on path l, f_0 is the frequency of CSI_0, $k\Delta f$ is the frequency difference between f_0 and f_k, and τ_l is the delay of CSI on the path l. Then, CSI_k can be expressed as:

$$\mathrm{CSI}_k = \sum_{l=1}^{L} S_0^l \Delta_k^l. \tag{6}$$

In the 802.11n standard, when BW=40 MHz and Ng=4 (Grouping), the index of the sampled subcarriers ranges from −58 to 58 where the interval of subcarrier index is 4. For convenience of presentation, suppose that there are five paths, we can construct a Hankel matrix:

$$\mathbf{X} = \begin{bmatrix} \mathrm{CSI}_{-58} & \mathrm{CSI}_{-54} & \mathrm{CSI}_{-50} & \mathrm{CSI}_{-46} & \mathrm{CSI}_{-42} \\ \mathrm{CSI}_{-54} & \mathrm{CSI}_{-50} & \mathrm{CSI}_{-46} & \mathrm{CSI}_{-42} & \mathrm{CSI}_{-38} \\ \mathrm{CSI}_{-50} & \mathrm{CSI}_{-46} & \mathrm{CSI}_{-42} & \mathrm{CSI}_{-38} & \mathrm{CSI}_{-34} \\ \mathrm{CSI}_{-46} & \mathrm{CSI}_{-42} & \mathrm{CSI}_{-38} & \mathrm{CSI}_{-34} & \mathrm{CSI}_{-30} \\ \mathrm{CSI}_{-42} & \mathrm{CSI}_{-38} & \mathrm{CSI}_{-34} & \mathrm{CSI}_{-30} & \mathrm{CSI}_{-26} \end{bmatrix}. \tag{7}$$

Combined with formula (5) and (6), $\mathbf{X}$ could be expressed as:

$$\mathbf{X} = \Delta \mathbf{S} \Delta^{\mathrm{T}} \tag{8}$$

where

$$\boldsymbol{\Delta} = \begin{bmatrix} \Delta_{-29}^1 & \Delta_{-29}^2 & \Delta_{-29}^3 & \Delta_{-29}^4 & \Delta_{-29}^5 \\ \Delta_{-25}^1 & \Delta_{-25}^2 & \Delta_{-25}^3 & \Delta_{-25}^4 & \Delta_{-25}^5 \\ \Delta_{-21}^1 & \Delta_{-21}^2 & \Delta_{-21}^3 & \Delta_{-21}^4 & \Delta_{-21}^5 \\ \Delta_{-21}^1 & \Delta_{-21}^2 & \Delta_{-21}^3 & \Delta_{-21}^4 & \Delta_{-21}^5 \\ \Delta_{-21}^1 & \Delta_{-21}^2 & \Delta_{-21}^3 & \Delta_{-21}^4 & \Delta_{-21}^5 \end{bmatrix}. \tag{9}$$

$$\mathbf{S} = \mathrm{diag}(S_0^1, S_0^2, S_0^3, S_0^4, S_0^5). \tag{10}$$

The problem of solving path components has turned into a mathematical problem. We define $\boldsymbol{\Delta} = \mathbf{VS}'$, where

$$\mathbf{V}=\begin{bmatrix}1&1&1&1&1\\ \Delta_4^1&\Delta_4^2&\Delta_4^3&\Delta_4^4&\Delta_4^5\\ \Delta_8^1&\Delta_8^2&\Delta_8^3&\Delta_8^4&\Delta_8^5\\ \Delta_{12}^1&\Delta_{12}^2&\Delta_{12}^3&\Delta_{12}^4&\Delta_{12}^5\\ \Delta_{16}^1&\Delta_{16}^2&\Delta_{16}^3&\Delta_{16}^4&\Delta_{16}^5\end{bmatrix}. \tag{11}$$

$$\mathbf{S}'=\mathrm{diag}(S_{-29}^1,S_{-29}^2,S_{-29}^3,S_{-29}^4,S_{-29}^5) \tag{12}$$

then $\mathbf{X}$ can be expressed as:

$$\mathbf{X}=\left(\mathbf{V}\mathbf{S}'\right)\mathbf{S}\left(\mathbf{V}\mathbf{S}'\right)^{\mathrm{T}}. \tag{13}$$

since $\mathbf{S}'=\left(\mathbf{S}'\right)^{\mathrm{T}}$, $\mathbf{X}$ is also presented as:

$$\mathbf{X}=\mathbf{V}\Sigma\mathbf{V}^{\mathrm{T}}. \tag{14}$$

where

$$\mathbf{\Sigma}=\mathbf{S}'\mathbf{S}\mathbf{S}'. \tag{15}$$

It is easy to prove:

$$\mathbf{V}=\begin{bmatrix}1&1&1&1&1\\ \Delta_4^1&\Delta_4^2&\Delta_4^3&\Delta_4^4&\Delta_4^5\\ \Delta_8^1&\Delta_8^2&\Delta_8^3&\Delta_8^4&\Delta_8^5\\ \Delta_{12}^1&\Delta_{12}^2&\Delta_{12}^3&\Delta_{12}^4&\Delta_{12}^5\\ \Delta_{16}^1&\Delta_{16}^2&\Delta_{16}^3&\Delta_{16}^4&\Delta_{16}^5\end{bmatrix}=\begin{bmatrix}1&1&1&1&1\\ \left(\Delta_4^1\right)^1&\left(\Delta_4^2\right)^1&\left(\Delta_4^3\right)^1&\left(\Delta_4^4\right)^1&\left(\Delta_4^5\right)^1\\ \left(\Delta_4^1\right)^2&\left(\Delta_4^2\right)^2&\left(\Delta_4^3\right)^2&\left(\Delta_4^4\right)^2&\left(\Delta_4^5\right)^2\\ \left(\Delta_4^1\right)^3&\left(\Delta_4^2\right)^3&\left(\Delta_4^3\right)^3&\left(\Delta_4^4\right)^3&\left(\Delta_4^5\right)^3\\ \left(\Delta_4^1\right)^4&\left(\Delta_4^2\right)^4&\left(\Delta_4^3\right)^4&\left(\Delta_4^4\right)^4&\left(\Delta_4^5\right)^4\end{bmatrix}. \tag{16}$$

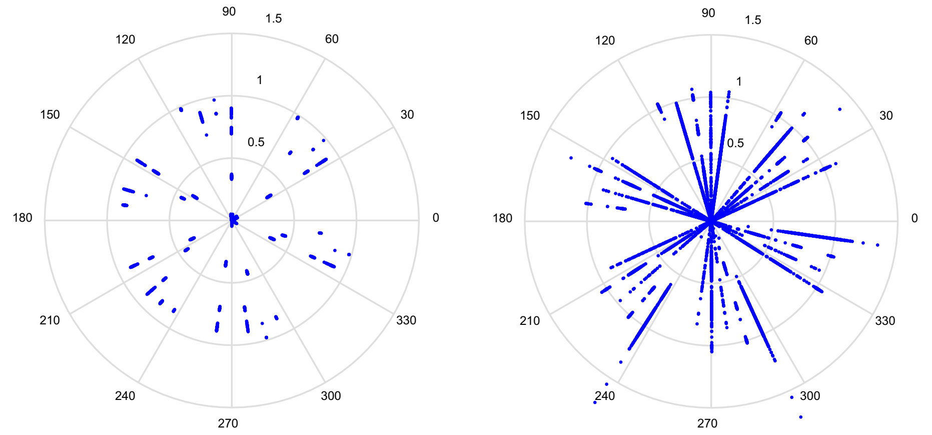
Obviously, $\mathbf{V}$ is a Vandermonde matrix. Therefore, the problem transforms into a problem of finding a Vandermonde decomposition of $\mathbf{X}$, of which the solution $\left[\left(\Delta_4^1\right)^1\ \left(\Delta_4^2\right)^1\ \left(\Delta_4^3\right)^1\ \left(\Delta_4^4\right)^1\ \left(\Delta_4^5\right)^1\right]$ is components of CSI phase on 5 paths. It is well known that the number of paths in multipath effects is actually very large. Here we only assume that there are five paths and solve the CSI phase components on each path. We summarize the overall in Algorithm 1.

Algorithm 1: IDSDL's path decomposition algorithm

```
Input: CSI phase on each subcarrier
Output: Phase component on each of the five paths
Construct a Hankel matrix $\mathbf{X}$ with formula (7);
Construct a vector $\mathbf{b}$ as $\mathbf{b}=\left[\ CSI_{-38}\ \ CSI_{-34}\ \ CSI_{-30}\ \ CSI_{-26}\ \ CSI_{-22}\ \right]^T$;
Obtain vector $\mathbf{A}=\left[\ a_0\ \ a_1\ \ a_2\ \ a_3\ \ a_4\ \right]^T$ by solving the equation $\mathbf{XA}=\mathbf{b}$;
Solve the equation $f(x)=0$, where $f(x)=x^m-a_{m-1}x^{m-1}-\cdots-a_0x^0, m=5$;
```

We apply the algorithm on P_{zi}, and the path-decomposed CSI phase is shown in Fig. 5.

We convert the path-decomposed CSI signal into feature image as the input of CNN classifier by converting the data of the CSI matrix into gray pixels. Feature image retains not only the waveform characteristics of signals but also the timing characteristics of signal changes if we convert the signals of multiple consecutive moments into one feature



a Path-decomposed CSI phase in static state **b** Path-decomposed CSI phase in dynamic state

Fig. 5 Path-decomposed CSI phase

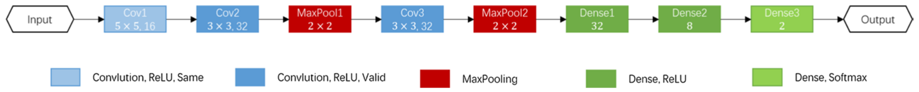


Fig. 6 CNN structure

image. Using these image inputs helps CNN detect the occurrence of intrusion behavior based on signal changes at multiple times.

4.3 CNN classifier

CNN is a classic deep learning model with good effect in the field of image processing and computer vision [28–30], which works pretty well on image classification tasks. An advantage of CNN lies in feature extraction [31]. Because the feature detection layer of CNN learns features from training data directly, it avoids explicit feature extraction, thus reducing the time cost on finding suitable numerical features. Using this CNN-based method and feature image improves the detection system. On the one hand, the feature image converted from the path-decomposed CSI signal retains the waveform characteristics of signals and the timing characteristics of signal changes. On the other hand, CNN effectively solves the problem that suitable numerical features for sensitively detecting human motion on NLOS are laborious to construct due to its ability to learn features from training data directly.

A CNN is designed as Fig. 6. The network uses three layers of convolution layer to extract the features of the input image, two max-pooling layers to scale down the size of feature maps and three dense layers to make classification.

The function of convolution operation is to determine a feature in the image, condense original image pixels of the convolution kernel size into one pixel, and extract features in the image after convolution operation. Large-sized convolution kernels can provide a larger receptive field, but meanwhile the amount of parameters is also larger, leading to more calculations. So multiple smaller convolution kernels are used in the second and third convolutional layers to ensure large receptive field while reducing the number of parameters. The max-pooling operation highlights the largest feature of 2 × 2 pixels, which is able to provide strong robustness and reduce the number of parameters. The softmax function is used in the third dense layer. The softmax function is a gradient logarithm normalization of a finite item discrete probability distribution which is suitable for multi-classification tasks.

In detail, the first convolutional layer has sixteen 5 × 5 convolution kernels with convolution step of 1, and due to zero padding, it outputs $L \times 30 \times 16$ feature map. The second convolutional layer owns 32 3 × 3 convolution kernels, and convolution step is also 1. The parameters of the third convolutional layer are consistent with those of the second one. Two max-pooling layers are after the second convolutional layer and the third convolutional layer, respectively. At this point, a flatten layer is performed to flatten 2-dimensional feature maps into 1-dimensional ones, connecting the convolutional layers and the dense layers. After three dense layers, the output is the prediction result we need.

Because CNN has a more complicated learning mechanism than conventional learning methods, our method does not have much advantage in terms of time complexity. For a single convolution layer, the time complexity is $O(M^2 \times K^2 \times \text{Cin} \times \text{Count})$. Where M is the size of output feature map, K is kernel size, Cin is number of input channels and Count is number of output channels. The time complexity of the entire neural network model is $O\left(\sum_{l=1}^{D} M_l^2 \times K_l^2 \times C_{l-1} \times C_l\right)$. D is the number of convolutional layers of the neural network. l represents the lth convolutional layer. C_l indicates the number of output channels of this layer. C_{l-1} represents the number of output channels of the previous layer, that is, the number of input channels of this layer.

We have to train the CNN in the offline phase to make the model learn to identify different samples of static state and dynamic state. After CNN training, the network obtains the ability to classify CSI data correctly. We use the trained CNN to predict new data in the online phase.

4.4 Motion detection

Since the input of CNN is generally images and feature extraction is not required, we directly convert path-decomposed CSI phase data into images, which are also known as feature maps. At this point, after preparing a classifier and its input, the structure of our system is complete. We use the CNN to classify the input into data of static state and data of dynamic state. In motion detection, sample selection would make an affect on the performance of the trained network. To eliminate the contingency of sample selection, fivefold cross-validation is necessarily implemented in motion detection. Firstly all the feature maps are divided into five parts; secondly, one of them is selected as the validation set each time, while the other four are used as training set; thirdly, we conduct experiments with different validation set five times. After doing so, the five results are averaged to obtain one final experimental result. Take scenario 1 as an example, the five results are shown in Fig. 7 and specific detection accuracies are shown in Table 1.

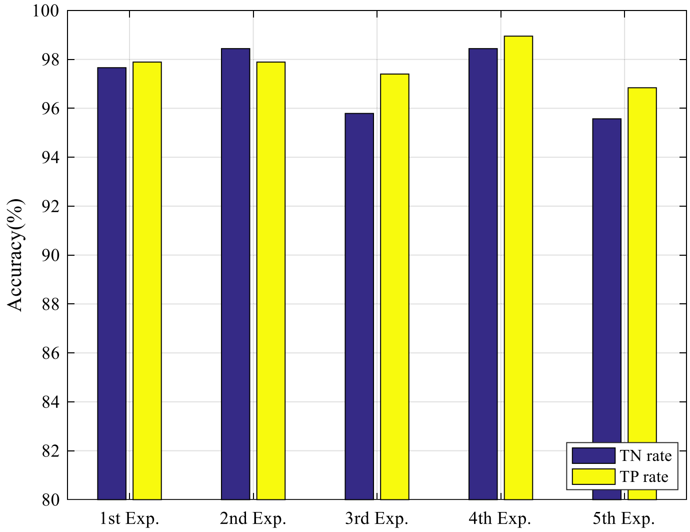


Fig. 7 Fivefold cross-validation detection accuracy

Table 1 Fivefold cross-validation and average detection accuracies

	TN rate (%)	TP rate (%)
#1	97.66	97.89
#2	98.44	97.89
#3	95.79	97.40
#4	98.44	98.95
#5	95.57	96.84
Average	97.18	97.79

The average TN rate with no human and TP rate with human walking are 97.18% and 97.79%, respectively.

5 Experiment and evaluation

5.1 Experimental scenario

To evaluate the performance of IDSDL, experiments are conducted on commodity WiFi devices. Specifically, we use a TP-LINK TL-WR886N router as the transmitting terminal, as well as a computer equipped with the Intel WiFi Link 5300 NIC [32, 33] and CSITool as the receiving terminal. Both the router and the receiving antenna are set at 1.2 m high. We collect CSI data in a laboratory with multiple propagation paths, in which a variety of electronic devices and multiple sets of tables and chairs are placed. In the experiments, CSI data of dynamic state is collected with an adult intruding from the door in the single participant scenario, and three adults intruding and moving in the area in the multiple participants scenario. The data of static state is collected with human being absent in the area.

To verify our system possesses good effect on detecting both human motion on LOS and NLOS, experiments of four different RX-TX position scenarios are implemented.

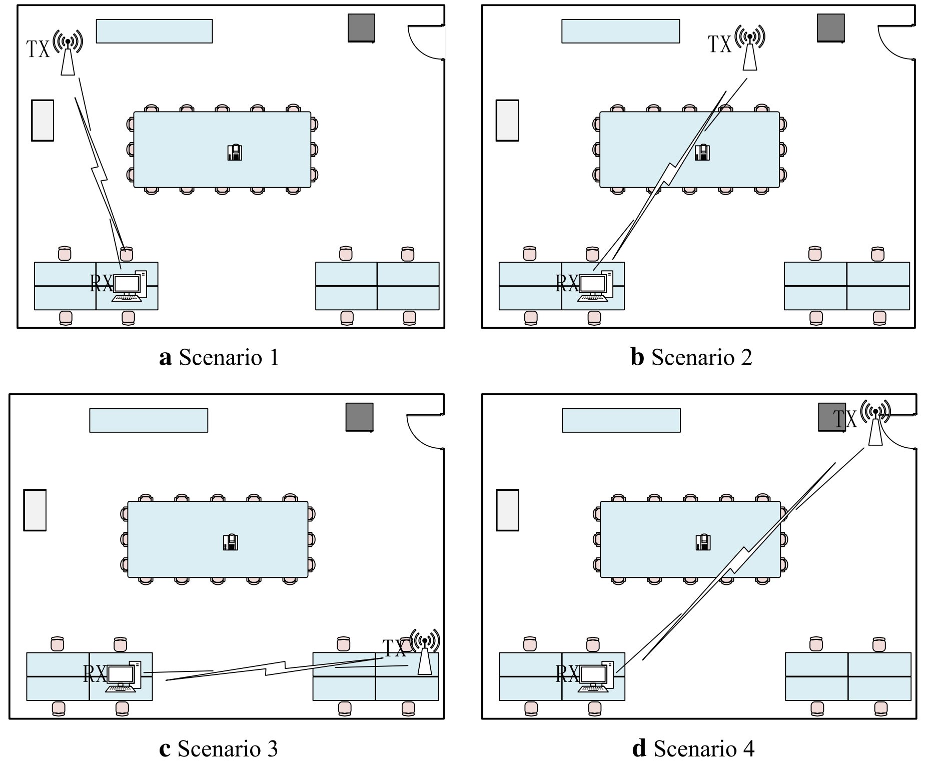


Fig. 8 Plan of experimental areas

Table 2 Parameters of experiment

Parameters	Laboratory
Size	7.2 m × 7.8 m
Person present	1 and 3 according to different scenes

Each scenario is shown in Fig. 8. The main moving area of experiment participants is around the conference table. Table 2 presents the parameters of experiment.

5.2 Performance evaluation

5.2.1 Evaluation metrics

We mainly use the two metrics for evaluating the performance of IDSDL.

① *True negative rate (TNR)* the probability that no human presence is correctly identified.

② *True positive rate (TPR)* the probability that a moving human presence is correctly classified.

③ *Detection accuracy* the probability of identifying the environmental changes in different scenarios.

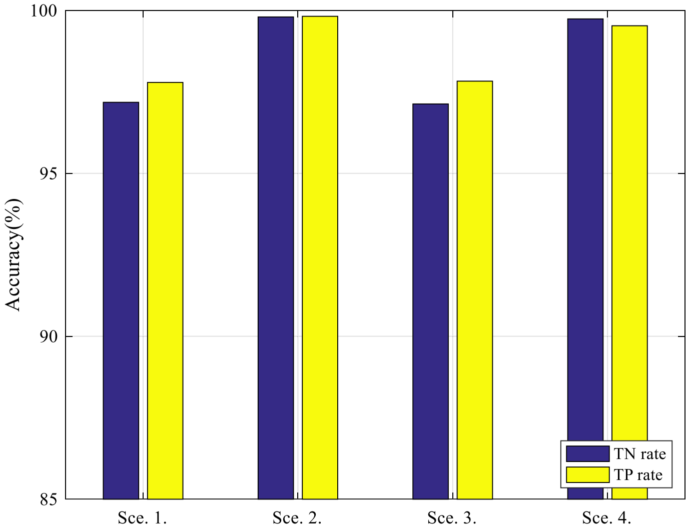


Fig. 9 TN rate and TP rate in four experimental scenarios with one participant

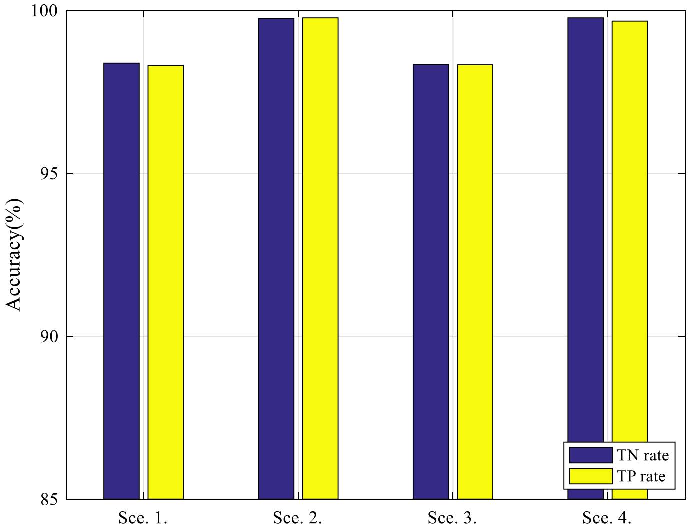


Fig. 10 TN rate and TP rate in four experimental scenarios with multiple participants

5.2.2 Overall performance

Firstly, we conduct our experiments in different scenarios to examine the overall performance of our system. Figures 9 and 10 show the performance of motion detection with four RX-TX positions, where Fig. 9 is the detection results of one participant participating in the experiment, as well as Fig. 10 shows the detection results of multiple participants participating in the experiment. From these two figures, we can see that the TN rate and TP rate of scenario 2 and scenario 4 are higher because their RX-TX positions

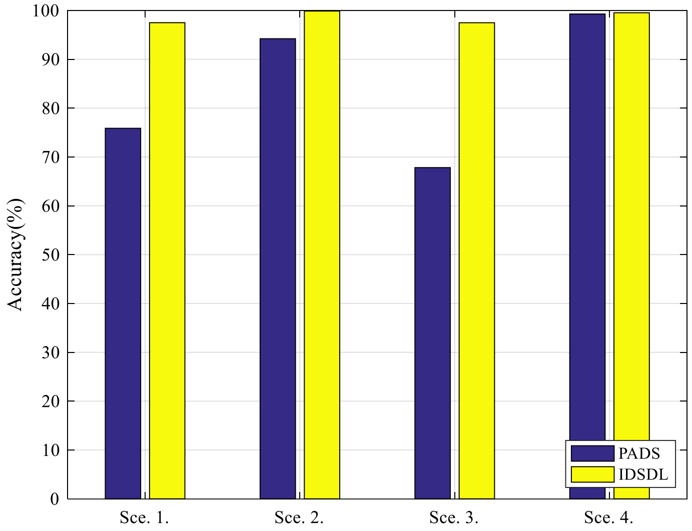


Fig. 11 Detection accuracies in four scenarios with PADS and IDSDL

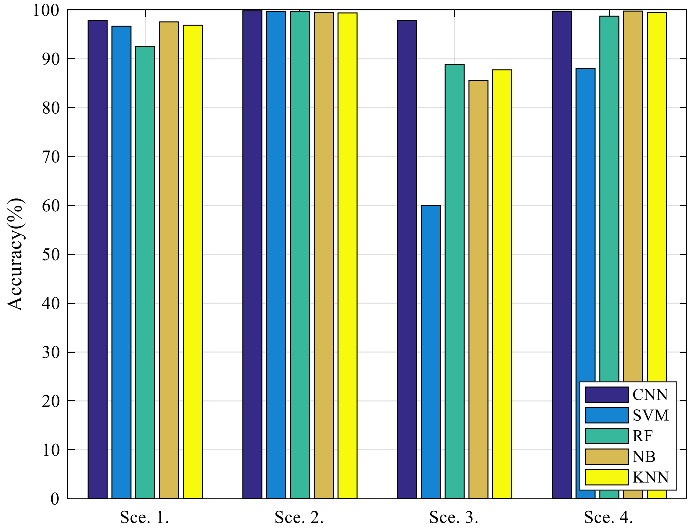


Fig. 12 Detection accuracies in four scenarios by using CNN, SVM, RF, NB and KNN

make the intrusion happen on paths that are close to LOS. TN rate and TP rate are a little lower because intrusion occurs on NLOS paths which are pretty far away from LOS in scenario 1 and scenario 3. Another conclusion we can draw from the detection results is that motion detection is more accurate when multiple participants appear in the experiments than when there is only one participant. The reason is that compared with single person participating, human motion of multiple participants will have a greater impact on CSI by affecting CSI on multiple paths simultaneously.

The detection results obtained by using PADS and IDSDL in four experimental scenarios are shown in Fig. 11. The results indicate that IDSDL is more sensitive than PADS in scenarios in which human motion occurs on NLOS paths.

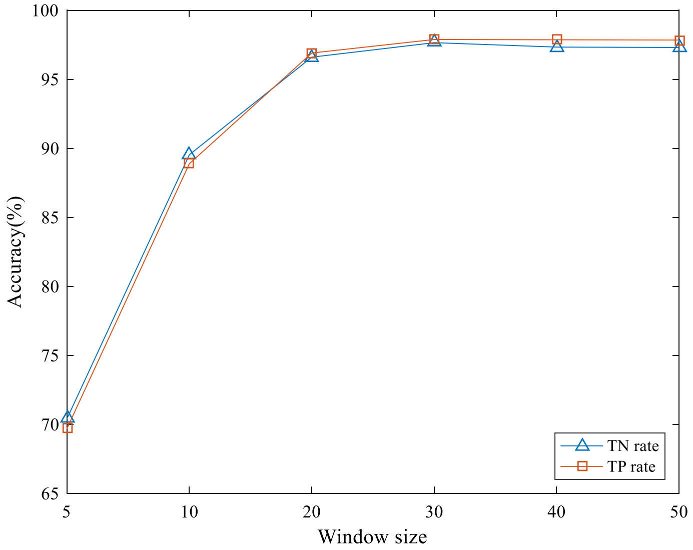


Fig. 13 Detection TN rate and TP rate with different window sizes

The detection accuracy results obtained by using CNN, support vector machine (SVM), random forest (RF), naive Bayes (NB), and k nearest neighbor (KNN) in four experimental scenarios are shown in Fig. 12. Detection with conventional learning methods works when human motion occurs near LOS but cannot achieve good detection effects when human motion occurs on NLOS paths. However, the results indicate that intrusion detection with the CNN-based method is more sensitive when human motion occurs on NLOS paths that are far away from LOS path.

In summary, IDSDL can not only obtain good detection effect when human motion occurs on LOS path, but also works well when human motion occurs on NLOS paths. Thus, IDSDL is a sensitive intrusion detection system.

5.2.3 Impact of sliding window size

Intuitively, in terms of normal walking speed, an intrusion process lasts 5–6 s from the time the door is opened. When the size of sliding window is large enough to contain a complete intrusion process, the detection accuracy is better. Experiments are carried out by setting the window size to 5, 10, 20, 30, 40, and 50. Figure 13 shows the change of TN rate and TP rate with different window sizes. The detection accuracy is improved with the sliding window size getting larger because the CSI data intercepted by larger sliding window contains more CSI changing features. However, it is not a panacea. When the size increases to a certain threshold, the rising detection accuracy tend is stalled.

5.2.4 Impact of window sliding step

Window sliding step affects the number of times the image features repeated on the image samples, and indirectly affects system performance. This impact is mainly

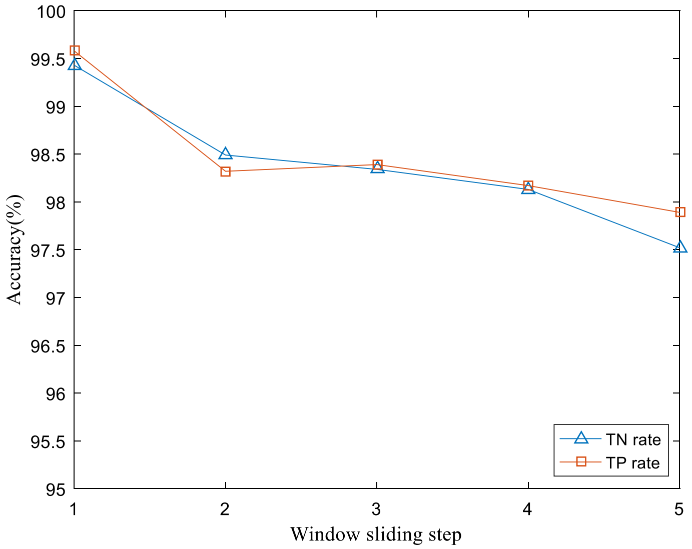


Fig. 14 Detection TN rate and TP rate with different window sliding steps

Table 3 Training time cost and detection accuracy with different window sliding steps

Step	Number of training rounds	Time cost per round (s)	Accuracy (%)	Total training time cost (s)
1	300	6.013	99.51	1803.9
2	600	3.013	98.44	1807.8
3	900	2.012	98.33	1810.8
4	1200	1.013	98.15	1215.6
5	1200	1.012	97.71	1214.4

reflected in the offline training process of the system. The result is shown in Fig. 14. Since the sliding step is short, the same CSI sequence data can generate more feature map samples, which leads to more comprehensive acceptable features, so that the highest TP rate and TN rate reach 99.51% and 99.43%, respectively. However, from Table 3 we realize that a larger number of samples leads to a larger training time consumption as well as increases the maximum detection accuracy. So we can be certain that if computer performance is good enough and time is not limited, short sliding steps are better choices, and if it is to consume shorter time while pursuing good detection performance, the sliding step of 5 is better. Taking the time consumption with the sliding step of 5 as an example, the total average time consumption during training phase is about 2.5 s per sample, while the processing time for a single test sample is about 10 ms, which is a short response time.

5.2.5 Impact of intrusion distance

We define some intrusion distances to verify the sensitivity of the system to human motion on the NLOS path. In the experiment, intrusion distance of 0.2, 0.4, 0.6, 0.8, and 1.0 m are set. An adult opens the door and enters for those preset distances. The

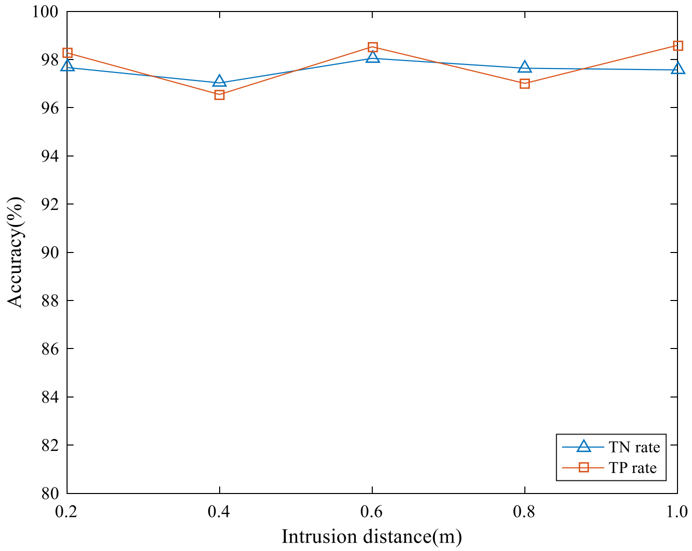


Fig. 15 TN rate and TP rate with different intrusion distances

experimental result is presented in Fig. 15. We can see from the figure that our system is sensitive to this kind of human motion far away from the LOS path. Since we use propagation components of CSI phase as detection signal and our algorithm can only obtain components on five paths, detection accuracy is not getting higher as the intrusion distance is larger.

6 Results and discussion

In this paper, we design and implement a WiFi sensing system for detecting intrusion by using CSI in the physical layer of WiFi network as detection signal. In this paper, we utilize a path decomposition algorithm and CNN to improve the sensitivity of passive intrusion detection system. The path-decomposed CSI phase can more accurately reflect the human motion occurring on NLOS paths. The application of CNN improves the performance of the detection system as well as simplifies construction of feature engineering. The proposed system solves the problem that human motion happens on NLOS paths is not well detected. Multiple experiments are conducted to verify the performance of IDSDL. The average detection accuracy in four different scenarios is 98.69% with single participant, and 98.91% with multiple participants. Compared with previous algorithms, IDSDL can sensitively detect human motion on NLOS paths and improve system reliability.

Abbreviations

AP: Access point; CSI: Channel state information; CNN: Convolutional neural network; RSS: Received signal strength; LOS: Line of sight; NLOS: Non-line of sight; OFDM: Orthogonal frequency division multiplexing; MIMO: Multiple-input multiple-output.

Authors' contributions

FB established the model and finished the writing of this manuscript. YH proposed the idea and completed the theoretical derivation. XY carried out the simulation. YL proofread the manuscript. All the authors read and approved the final manuscript.

Funding

ShangDong Major Science and Technology Innovation Projects (Grant No. 2019JZZY020505); National Natural Science Foundation of China (Grant No. 61901475).

Availability of data and materials

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Author details

[1] School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221008, China.
[2] IoT Perception Mine Research Center, China University of Mining and Technology, Xuzhou 221008, China.

Received: 24 August 2019 Accepted: 12 January 2021
Published online: 16 April 2021

References

1. A.V. Savkin, H. Huang, Proactive deployment of aerial drones for coverage over very uneven terrains: a version of the 3D art gallery problem. Sensors **19**(6), 1438 (2019)
2. Q. Pu, S. Gupta, S. Gollakota, et al. Whole-home gesture recognition using wireless signals, in *Proceedings of the 19th Annual International Conference on Mobile Computing and Networking*, pp. 27–38 (2013)
3. F. Adib, Z. Kabelac, D. Katabi, et al. 3D tracking via body radio reflections, in *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*, pp. 317–329 (2014)
4. B. Kellogg, V. Talla, S. Gollakota. Bringing gesture recognition to all devices[C], in *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*, pp. 303–316 (2014)
5. M. Seifeldin, A. Saeed, A.E. Kosba et al., Nuzzer: a large-scale device-free passive localization system for wireless environments. IEEE Trans. Mob. Comput. **12**(7), 1321–1334 (2012)
6. M. Moussa, M. Youssef. Smart devices for smart environments: device-free passive detection in real environments, in *2009 IEEE International Conference on Pervasive Computing and Communications*. IEEE, pp. 1–6 (2009)
7. J. Yang, Y. Chen, S. Desai, et al. Passive intrusion detection in wireless networks by exploiting clustering-based learning. in *Wireless Sensing, Localization, and Processing V*. International Society for Optics and Photonics, vol. 7706, p. 770604 (2010)
8. X. Wang, L. Gao, S. Mao et al., CSI-based fingerprinting for indoor localization: a deep learning approach. IEEE Trans. Veh. Technol. **66**(1), 763–776 (2016)
9. K. Qian, C. Wu, Z. Yang, et al. PADS: passive detection of moving targets with dynamic speed using PHY layer information, in *2014 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, pp. 1–8 (2014)
10. E. Ding, X. Li, T. Zhao et al., A robust passive intrusion detection system with commodity WiFi devices. J. Sens. **2018**, 1–12 (2018)
11. J. Wilson, N. Patwari, Radio tomographic imaging with wireless networks. IEEE Trans. Mob. Comput. **9**(5), 621–632 (2010)
12. J. Wilson, N. Patwari, See-through walls: motion tracking using variance-based radio tomography networks. IEEE Trans. Mob. Comput. **10**(5), 612–621 (2010)
13. M. Youssef, M. Mah, A. Agrawala. Challenges: device-free passive localization for wireless environments, in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, pp. 222–229 (2007)
14. A.E. Kosba, A. Saeed, M.R. Youssef. A robust wlan device-free passive motion detection system, in *2012 IEEE International Conference on Pervasive Computing and Communications*. IEEE, pp. 180–189 (2012)
15. J. Wang, Q. Gao, Y. Yu et al., Robust device-free wireless localization based on differential RSS measurements. IEEE Trans. Industr. Electron. **60**(12), 5943–5952 (2012)
16. J. Xiao, K. Wu, Y. Yi, et al. Pilot: passive device-free indoor localization using channel state information, in *2013 IEEE 33rd International Conference on Distributed Computing Systems*. IEEE, pp. 236–245 (2013)
17. J. Xiao, K. Wu, Y. Yi, et al. Fimd: fine-grained device-free motion detection, in *2012 IEEE 18th International Conference on Parallel and Distributed Systems*. IEEE, pp. 229–235 (2012)
18. A. Bhartia, Y.C. Chen, S. Rallapalli, et al. Harnessing frequency diversity in wi-fi networks, in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, pp. 253–264 (2011)
19. M. Chen, Z. Yang, W. Saad, et al. A joint learning and communications framework for federated learning over wireless networks. arXiv:1909.07972 (2019)
20. Z. Yang, M. Chen, W. Saad, et al. Energy efficient federated learning over wireless communication networks. arXiv: 1911.02417 (2019)
21. Y. Wang, M. Chen, Z. Yang, et al. Deep learning for optimal deployment of UAVs with visible light communications. arXiv:1912.00752 (2019)
22. F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain. Psychol. Rev. **65**(6), 386–408 (1958)
23. Y. LeCun, B.E. Boser, J.S. Denker et al., Handwritten digit recognition with a back-propagation network. Adv. Neural Inf. Process. Syst. **2**(2), 396–404 (1990)
24. S. Sen, B. Radunovic, R.R. Choudhury, et al. You are facing the Mona Lisa: spot localization using PHY layer information, in *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, pp. 183–196 (2012)
25. W. Yang, L. Gong, D. Man et al., Enhancing the performance of indoor device-free passive localization. Int. J. Distrib. Sens. Netw. **11**(11), 256162 (2015)
26. R.K. Pearson, Y. Neuvo, J. Astola et al., Generalized hampel filters. EURASIP J. Adv. Signal Process. **2016**(1), 1–18 (2016)
27. G. Bishop, G. Welch, An introduction to the Kalman filter. Proc. SIGGRAPH Course **8**(27599–23175), 41 (2001)

28. K. Fukushima, Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybern. **36**(4), 193–202 (1980)
29. Y. LeCun, L. Bottou, Y. Bengio et al., Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
30. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. **60**, 1097–1105 (2012)
31. L.O. Chua, T. Roska, The CNN paradigm. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **40**(3), 147–156 (1993)
32. D. Halperin, W. Hu, A. Sheth et al., 802.11 with multiple antennas for dummies. ACM SIGCOMM Comput. Commun. Rev. **40**(1), 19–25 (2010)
33. D. Halperin, W. Hu, A. Sheth et al., Predictable 802.11 packet delivery from wireless channel measurements. ACM SIGCOMM Comput. Commun. Rev. **40**(4), 159–170 (2011)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.