Journal of Computers Vol. 32 No. 6, 2021, pp. 134-143
doi:10.53106/199115992021123206011

Research on M-sequence Estimation Based on CNN

Fang-Fang Qiang[1*], Zhi-Jin Zhao[2], Hai-Wei Zhong[1]

[1] School of Information and Electronic Engineering, Zhejiang University of Science & Technology, Hangzhou, China
qiangff_my@163.com

[2] School of Communication Engineering, Hangzhou Dianzi University, Hangzhou, China

Received 1 May 2021; Revised 7 June 2021; Accepted 8 June 2021

Abstract. A method based on convolutional neural networks (CNN) for estimation of sequences is proposed. First of all, a sequence estimation model based on CNN is established, the sequences to be estimated are directly used as the inputs of the model. The method effectiveness is verified by estimating m-sequences. Second, by combining FastICA algorithm and the proposed CNN estimation method, a new spreading sequences estimation method named FastICA-CNN for long code direct sequence code division multiple access (LC-DS-CDMA) signals is studied. The FastICA algorithm is used for blind separation of each user's spreading sequences, the amplitude fuzziness is eliminated by delay multiplication, and the CNN estimation method is used for spreading sequences estimation. The results show that, when the number of users is 2 and signal-to-noise ratio (SNR) is higher than −16 dB, the estimation accuracy of m-sequences in LC-DS-CDMA by FastICA-CNN method tends to be 1; the time required for model training and sequences estimation is 84.71s and 375.4s, respectively. Compared to the methods in references, FastICA-CNN method requires shorter simulation time and has higher estimation accuracy.

Keywords: sequence estimation, convolutional neural networks, code division multiple access, blind separation

1 Introduction

Binary pseudo-random sequences are widely used in spread spectrum communication due to their good pseudo-random property. The m-sequence is the most representative pseudo-random sequence. Its estimation and recognition are the basis of information decryption in the spread-spectrum system. Therefore, it is of great theoretical significance and value to the research on m-sequence recognition [1].

Studies on m-sequence recognition show that, Massey and Euclidean algorithms can achieve the purpose of recognizing sequence generator polynomials. However, their recognition performances are affected greatly by error codes [2-3]. The triple correlation function (TCF) method based on high-order statistical analysis has received extensive attention as it is simple and easy to understand [4]. The FastICA and segmentation optimization algorithms are proposed for spreading sequences estimation in the received LC-DS-CDMA signals [5]. Based on [5], the sequence estimation method based on FastICA algorithm and TCF properties is proposed in [6]. In [7], the m-sequences are estimated by the method of combining goodness-of-fit test with TCF properties. However, the accuracy of peak points detection needs further improvement at a low SNRs. Based on the TCF properties and sparse autoencoder, a m-sequence estimation method named TCF-SAE is studied, but it is only suitable for m-sequences [8]. On this basis, improved estimation method by directly used sequences as the input samples of SAE is proposed, and this method is also applied for spreading sequences estimation in LC-DS-CDMA signals [9].

* Corresponding Author

CNN can automatically extract features of different levels from input samples without prior knowledge, and the extracted features have superior generalization and abstract expression ability [10]. CNN is widely used in computer vision and natural language processing [11], however, there are few studies about spreading sequence estimation in LC-DS-CDMA signals by CNN methods, which needs further research. In this paper, we study the sequences estimation method based on CNN, and the m-sequences are selected to verify its effectiveness.

The remainder of this study is presented as follows. The CNN is outlined in Section 2. The sequence estimation method by CNN is explained and verified by estimating m-sequences in Section 3. The separation and estimation of spreading sequences in LC-DS-CDMA signals, and the estimation performances analysis by FastICA-CNN method is introduced in Section 4. The key points and future researches are outlined in Section 5.

2 CNN Estimation Model

Feature extraction in CNN throughs convolution layers, pooling layers and full connection layers, and finally the extracted features are used for recognition or classification. The operation process of convolution layer is as follows [12-13]:

$$\mathbf{X}^{(L)} = f(\mathbf{W}^{(L)} \otimes \mathbf{X}^{(L-1)} + \mathbf{b}^{(L)}), \quad (1)$$

where, L is the number of layers; $\boldsymbol{X}^{(L)}$ is a feature graph representing the output of L layer; $\boldsymbol{W}^{(L)}$ and $\boldsymbol{b}^{(L)}$ are the convolution kernel and bias, respectively; $f(\cdot)$ is the activation function. In this study, the activation function is rectified linear unit (ReLU) [9]:

$$f(x) = \mathrm{ReLU}(x) = \begin{cases} x & x \geq 0 \\ 0 & else \end{cases}. \quad (2)$$

The pooling layer usually appears after the convolution layer. Its function is to reduce the dimension of feature graph and training parameters, as well as alleviate the over fitting and improve the robustness of features extraction [14]. In this paper, average pooling is selected.

The fully connected layer expands the output feature graph into one-dimensional column vectors. The recognition and classification of input samples can be completed by connecting the full connection layer with softmax classifier [11].

In this study, sequences are directly used as the input samples of CNN to learn features from the original data, then this sequence estimation method is called CNN method. Because the samples analyzed are sequences, the network is a CNN with one-dimensional convolution. Let the 255-bits sequence estimation as an example, the constructed CNN model with two convolution layers is shown in Fig. 1. The dimension of the input samples is 255×1×1, the last 1 is the number of channels. There are K classes of sequences in total, C and S represent convolution layer and pooling layer, respectively.

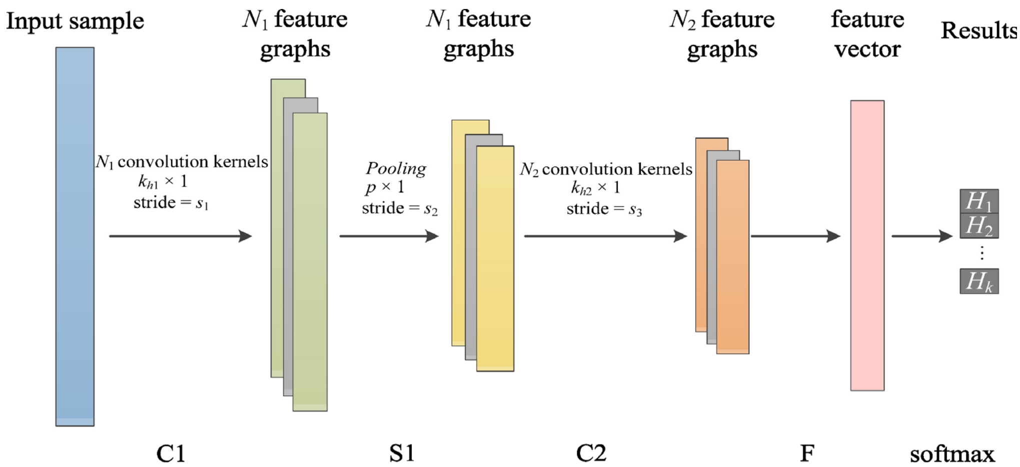


Fig. 1. Sequence estimation model based on CNN

C1 is the first convolution layer, the number and size of convolution kernel is N_1 and $k_{h1}\times1$, respectively, where k_{h1} is the height, and the stride is set to s_1. The output of this layer is N_1 feature graphs with size $\lfloor(255-k_{h1})/s_1+1\rfloor\times1$.

S1 is the pooling layer, in this layer, the output of C1 is pooled to reduce the dimension, pooling template with size $p\times1$ is used for average pooling, and the stride is set to s_2. The output of this layer is N_1 feature graphs with size $\lfloor(\mathrm{Height}_1-p)/s_2+1\rfloor\times1$, where $\mathrm{Height}_1=\lfloor(255-k_{h1})/s_1+1\rfloor$.

C2 is the second convolution layer, the number and the size of convolution kernel is N_2 and $k_{h1}\times1$, respectively, and the stride is set to s_2. The output is N_2 feature graphs with size $\lfloor(\mathrm{Height}_2-k_{h2})/s_3+1\rfloor\times1$, where $\mathrm{Height}_2=\lfloor(\mathrm{Height}_1-p)/s_2+1\rfloor$. The sequence estimation is simple and the dimension of input samples is small, the pooling layer is no longer setting after the second convolution layer.

F is the fully connected layer, its function is to integrate the extracted features and connect the softmax for sequence estimation.

The above layers constitute the CNN model. In addition, dropout is introduced to prevent over fitting and improve the generalization ability of the network [15].

3 Sequence Estimation Based on CNN Model

In this section, m-sequences are selected for simulation, and then the effectiveness of the CNN method can be verified. The production regularity of m-sequence is as follows [9]:

$$m_i=o_\delta m_{i-\delta}+\cdots+o_v m_{i-v}+\cdots+o_1 m_{i-1}, \tag{3}$$

where m_i means the i-th bit, o means the connection status, and δ means the order of m-sequence, then the period is $2^\delta-1$.

The CNN model is trained by constructing input samples from fifth- to eighth-order m-sequences. The corresponding primitive polynomials are 6, 6, 18, and 16, respectively. The m-sequences with length of 31, 63 and 127 are expanded into sequences with length of 255 by the way of complementing 0. SNR is randomly selected between −4 and 4 dB, then 46 noisy sequences can be obtained. After 200 repetitions, a total of 9200 samples are constructed as the training samples. In this section, two periodic sequences are identified, and their primitive polynomials are $f(x)=x^7+x+1$ and $f(x)=x^8+x^7+x^2+x+1$. 400 test samples are constructed when the SNR is −16, −12, −8, −4, and 0 dB, respectively. In the following experiments, the parameter of pooling layer $p=2$, the strides of convolution layer and pooling layer are 1.

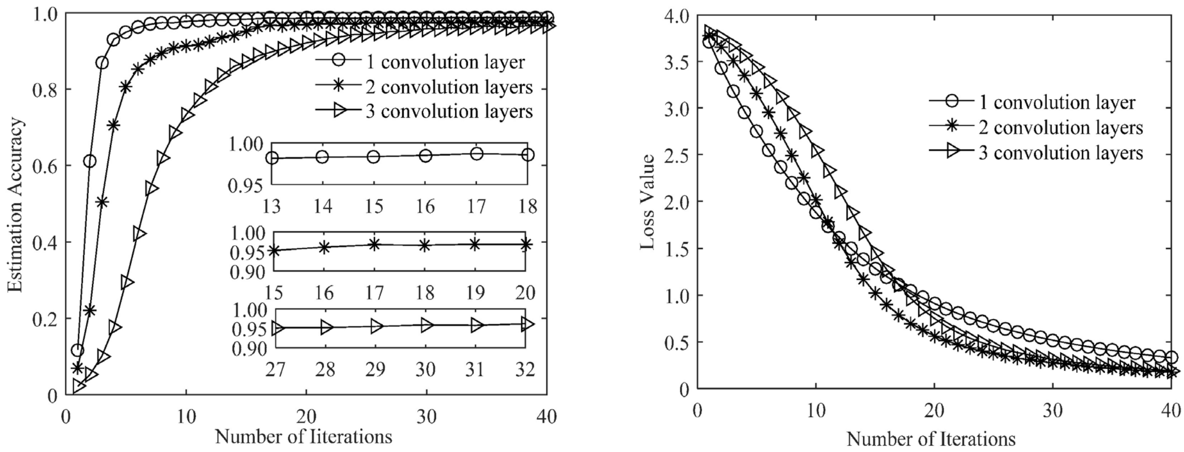
In this study, Matlab 2016a is used for samples construction, Python and keras framework are used for the construction, training and testing of CNN model, the computer CPU is “Intel Core i5-10210U”.

3.1 Model Training for Parameter Selection

In this section, the influence of the numbers of iterations and convolution layers, and the size of convolution kernel on the estimation accuracy is analyzed.

Experiment 1 Iterations number influence. Theoretically, the larger the number of iterations, the higher the estimation accuracy. However, large iteration number may lead to over-fit problem. In this experiment, we assume that the number of convolution layer is 1, 2, 3, respectively. When number is 1, $N_1=100$, $k_{h1}=2$; when number is 2, $N_1=100$, $k_{h1}=2$, $N_2=50$, $k_{h2}=2$; when number is 3, $N_1=100$, $k_{h1}=2$; $N_2=50$, $k_{h2}=2$; $N_3=20$, $k_{h3}=2$. The estimation accuracy and loss value at different number of iterations are show in Fig. 2(a) and Fig. 2(b), respectively.

Fig. 2(a) shows that for the CNN model with 1, 2 and 3 convolution layers, the accuracy of sequence estimation tends to be stable when the iterations are greater than 15, 17 and 30 respectively. Fig. 2(b) shows that as the iteration number increases, the loss value decreases, but the model is easy to over fitting. Therefore, in the following experiments, the iteration number of CNN model with 1, 2 and 3 convolution layers are set to 15, 20 and 30, respectively.



(a) estimation accuracy (b) loss value

Fig. 2. Iterations number influence on estimation performance

Experiment 2 Convolution layers number influence. In this experiment, we assume that the number of convolution layers is {1, 2, 3}. For the first layer, $N_1 = 100$, $k_{h1} = 2$; for the second layer, $N_2 = 50$, $k_{h2}= 2$; for the third layer, $N_3 = 20$, $k_{h3}= 2$. The iteration number of three models are 15, 20 and 30 respectively. The simulation results by the CNN model are listed in Table 1.

Table 1. Simulation results at different convolution layers numbers

Convolution layers number	Estimation accuracy at different SNR (dB)					Training time (s)	Loss value
	−16	−12	−8	−4	0		
1	**0.428**	**0.688**	**0.925**	**0.996**	**1.000**	**78.06**	**1.37**
2	0.300	0.522	0.811	0.978	1.000	132.92	0.57
3	0.297	0.497	0.779	0.973	1.000	204.43	0.31

The results in Table 1 show that when the number of convolution layers is one, the estimation performance is the best, the training time is the shortest, but the loss value is the biggest. With the increase in layer number, the loss value decreases, but the training time gets extend and the model is easy to cause over fitting.

Experiment 3 Convolution kernels number and size influence. The size of convolution kernels in CNN model determines the recognition granularity of input samples. In this experiment, different numbers and sizes of convolution kernels are set for comparison. The simulation results are listed in Table 2.

Table 2. Simulation results for different convolution kernels numbers and sizes

Convolution kernels number	Convolution kernels size	Estimation accuracy at different SNR (dB)					Training time (s)	Loss value
		−16	−12	−8	−4	0		
50	2×1	0.381	0.639	0.905	0.996	1.000	43.39	2.03
50	3×1	0.367	0.610	0.875	0.991	1.000	36.41	1.73
50	5×1	0.328	0.575	0.872	0.993	1.000	40.81	1.66
100	**2×1**	**0.428**	**0.688**	**0.925**	**0.996**	**1.000**	**78.06**	**1.37**
100	3×1	0.410	0.671	0.927	0.996	1.000	83.06	1.19
100	5×1	0.387	0.638	0.894	0.994	1.000	85.84	1.02
150	2×1	0.382	0.641	0.904	0.995	1.000	136.94	1.01
150	3×1	0.371	0.630	0.905	0.995	1.000	127.36	0.93
150	5×1	0.372	0.617	0.878	0.992	1.000	129.84	0.78

The results in Table 2 show that when the convolution kernel size is fixed, with the increase in convolution kernel number, the model training time gets extend, the loss value decreases, and the estimation accuracy increases and then decreases. When the convolution kernel number is fixed, the size of convolution kernels has little effect on the estimation accuracy and training time.

The results of experiments 1–3 show that it is optimal when the number of iterations, convolution layers, convolution kernels and the size of convolution kernels are 15, 1, 100 (N_1 = 100), and 2×1 (k_{h1} = 2), respectively.

3.2 Performance Analysis

Experiment 4 Performance comparison for different methods. The proposed estimation method based on CNN is compared with the TCF-SAE method proposed in [8] and Direct-SAE method proposed in [9]. A 255-bits m-sequence with the primitive polynomial $f(x) = x^8 + x^7 + x^2 + x + 1$ is selected for analysis. The comparison results are shown in Fig. 3.

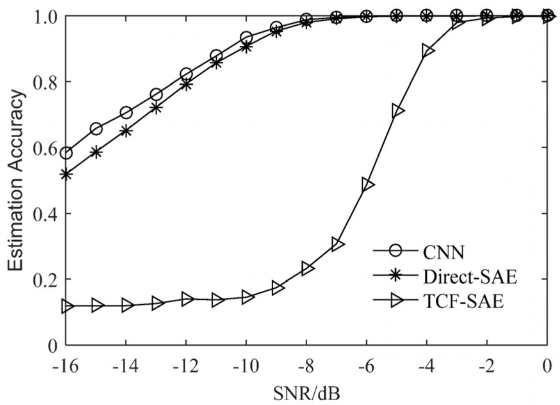


Fig. 3. Performance comparison

The results in Fig. 3 show that, the performance of CNN method is the best when the SNR is less than −7 dB, otherwise, the accuracy tends to be 1 for both CNN method and Direct-SAE method; for TCF-SAE method, the accuracy tends to be 1 when SNR is greater than −2 dB. This is because CNN has better deep feature extraction capability than SAE. In TCF-SAE method, the calculation of TCF is difficult and easily affected by noise, and ultimately affects the estimation performance of m-sequences. Therefore, the performance is the worst.

4 CNN Model Applied in LC-DS-CDMA Signals

According to the results in Section 3, the sequences estimation method based on CNN model is superior. In this section, the CNN method is applied in the spreading sequences estimation of LC-DS-CDMA signals.

4.1 FastICA-CNN Algorithm Theory

The LC-DS-CDMA signal can be expressed as [9]:

$$y(n) = \sum_{u=1}^{U} A_u d_u(n) c_u(n) + w(n), 1 \le n \le N , \tag{4}$$

where, U is the number of users, A_u, $d_u(n)$ and $c_u(n)$ are the signal amplitude, symbol sequence, and spreading sequence, respectively, $w(n)$ is white Gaussian noise, N is the length of the received signal.

The received signal $y(n)$ is a mixture of u users, and cannot be directly estimated by CNN model. Therefore, the single channel signal $y(n)$ can be modeled as multi-channel signal received by multiple array units, and each user can be separated by FastICA algorithm; then the separated users can be constructed as the input samples of CNN model, and finally the spreading sequences can be estimated by the trained CNN model. The processing steps of blind separation based on FastICA algorithm refers to

[9]. The received signal is divided by the length of spreading sequence L to obtain Z ($Z = N/L$) segments:

$$\boldsymbol{y}(n) = \boldsymbol{A}(n)\boldsymbol{C}(n) + \boldsymbol{W}(n),\ \ 1 \le n \le L, \tag{5}$$

where,

$$\begin{aligned}
&\boldsymbol{y}(n) = [y_1(n), \cdots y_z(n), \cdots y_Z(n)]^T,\ \ y_z(n) = y((z-1)L+n) \\
&\boldsymbol{C}(n) = [c_1(n), \cdots c_u(n), \cdots c_U(n)]^T,\ \ c_u(n) = c((z-1)L+n) \\
&\boldsymbol{W}(n) = [w_1(n), \cdots w_z(n), \cdots w_Z(n)]^T,\ \ w_z(n) = w((z-1)L+n) \\
&\boldsymbol{A}(n) = \begin{bmatrix} A_1 d_1(e_1(n)) \cdots A_u d_u(e_1(n)) \cdots\ A_U d_U(e_1(n)) \\ A_1 d_1(e_2(n)) \cdots A_u d_u(e_2(n)) \cdots A_U d_U(e_2(n)) \\ \vdots \qquad\qquad \ddots \qquad\qquad \vdots \\ A_1 d_1(e_Z(n)) \cdots A_u d_u(e_Z(n)) \cdots A_U d_U(e_Z(n)) \end{bmatrix},\ \ e_z(n) = \lceil [(z-1)L+n]/G \rceil
\end{aligned}, \tag{6}$$

G is the spreading factor. $\boldsymbol{A}(n)$ is time-varying, so the FastICA algorithm cannot be directly used. For this problem, the signal segment is further divided according to the spreading factor G, then K fragments are obtained ($K = \lceil L/G \rceil$), the k-th segment is expressed as:

$$\boldsymbol{y}^k(n) = \boldsymbol{A}^k(n)\boldsymbol{C}^k(n) + \boldsymbol{W}^k(n),\ \ 1 \le n \le G, \tag{7}$$

where,

$$\begin{aligned}
&\boldsymbol{y}^k(n) = [y_1^k(n), \cdots y_z^k(n), \cdots y_Z^k(n)]^T, y_z^k(n) = y_z((k-1)G+n) \\
&\boldsymbol{C}^k(n) = [c_1^k(n), \cdots c_u^k(n), \cdots c_U^k(n)]^T, c_u^k(n) = [c_1^k(n), \cdots c_u^k(n), \cdots c_U^k(n)]^T \\
&\boldsymbol{W}^k(n) = [w_1^k(n), \cdots w_z^k(n), \cdots w_Z^k(n)]^T, w_z^k(n) = w_z((k-1)G+n) \\
&\boldsymbol{A}^k(n) = \begin{bmatrix} A_1 d_1(e_1^k(n)) & \cdots & A_u d_u(e_1^k(n)) & \cdots & A_U d_U(e_1^k(n)) \\ A_1 d_1(e_2^k(n)) & \cdots & A_u d_u(e_2^k(n)) & \cdots & A_U d_U(e_2^k(n)) \\ \vdots & & \ddots & & \vdots \\ A_1 d_1(e_Z^k(n)) & \cdots & A_u d_u(e_Z^k(n)) & \cdots & A_U d_U(e_Z^k(n)) \end{bmatrix}, e_z^k(n) = e_z((k-1) \cdot G + n)
\end{aligned}, \tag{8}$$

$\boldsymbol{A}^k(n)$ is time-invariant and then FastICA algorithm can be used for blind separation. U^K sequences can be obtained as follows:

$$\boldsymbol{\alpha}_t = [\hat{\mathbf{c}}_{u\_1}^1 \cdots \hat{\mathbf{c}}_{u\_k}^k \cdots \hat{\mathbf{c}}_{u\_K}^K], \tag{9}$$

where, $\hat{\mathbf{c}}_{u\_k}^k = [c_{u\_k}^k(1) \ldots c_{u\_k}^k(G)]$, $1 \le t \le U^K$, $1 \le u\_1, \ldots, u\_K \le U$. The amplitude fuzziness is eliminated by delay multiplication method:

$$\boldsymbol{\gamma}_t(n) = \boldsymbol{\alpha}_t(n) \cdot \boldsymbol{\alpha}_t((n+1) \bmod L), 1 \le n \le L, \tag{10}$$

according to these U^K sequences $\boldsymbol{\gamma}_t(n)$, the spreading sequences of U users can be estimated.

The specific steps of FastICA-CNN method for sequences estimation of LC-DS-CDMA signal are as follows:

(1) The training set of CNN model is consisted of N noisy spreading sequences, which are obtained by the delay multiplication method at different SNRs.

(2) CNN model is trained, and then the parameters of the optimal model can be obtained.

(3) The signal $\boldsymbol{y}(n)$ is blindly separated according to Eqs.(4)-(10).

(4) The {+1, −1} sequences $\gamma_h(n)$ are transformed into to {0, 1} sequences, which constitute the input set $\boldsymbol{X}_D$, and the number of sequences in $\boldsymbol{X}_D$ is U^K.

(5) Sequences in $\boldsymbol{X}_D$ are used as the input samples of the model trained in step (2), the outputs of the model are the estimation results then.

(6) The occurrences numbers of different classes are counted, and the most frequently appeared U classes are the estimation results.

The flowchart of the FastICA-CNN method is shown in Fig. 4.

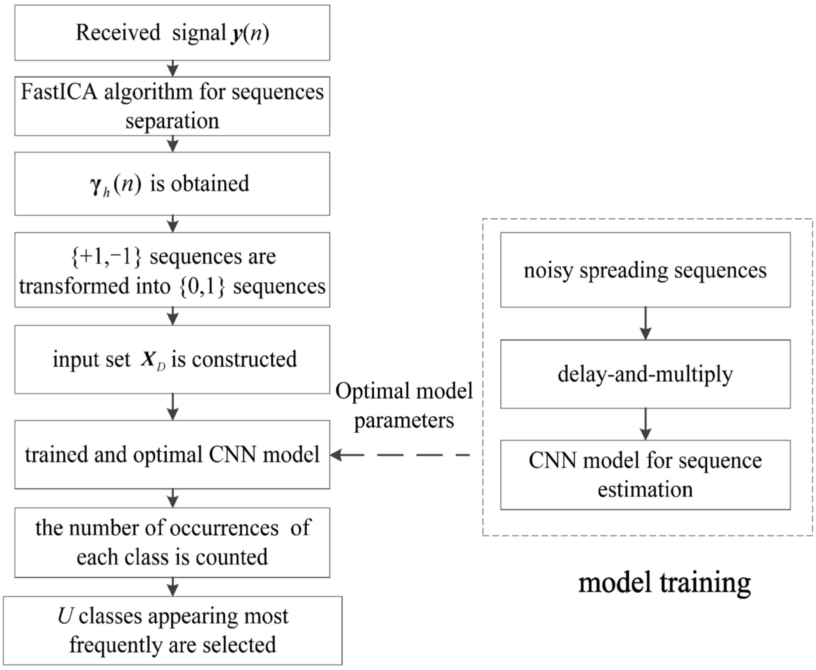


Fig. 4. The flow of FastICA-CNN algorithm

4.2 Simulation and Performance Analysis

In this section, we assume that the spreading sequences of LC-DS-CDMA signal are m-sequences, the LC-DS-CDMA signal is synchronized, the amplitude of each user with the same length of spreading sequences is the same, the symbol sequences are randomly generated, and the spreading sequence length has been estimated in advance. The SNR is defined as $10\log(A^2/\sigma^2)$, 1000 Monte Carlo simulation are carried out, and the estimation accuracy is defined as the ratio of the correct estimation number to the total simulation number.

The amplitude fuzziness caused by FastICA algorithm in blind separation is eliminated by delay multiplication method. Therefore, in the construction of training samples for CNN model, the m-sequences need to be preprocessed by the delay multiplication method. According to the network training process in Section 3.1, the optimal CNN model parameters of m-sequence estimation are obtained as follows: the number of convolution layers, convolution kernels are 1 and 100, respectively; the size of convolution kernels is 2×1.

Experiment 5 Users number influence. We assume that L = 255, G = 64, Z = 100, when U is2, 3, 4, respectively, the estimation accuracy of spreading sequences by the FastICA-CNN method is shown in Fig. 5.

Fig. 5 shows that FastICA-CNN algorithm can effectively estimate the spreading sequences of LC-DS-CDMA signal. The larger the number of users, the worse the estimation accuracy. This is because more users lead to more complex signal construction. It is more difficult for the separation of users, especially at low SNRs. As a result, the separation performance by FastICA becomes worse.

Experiment 6 Spreading sequence period influence. We assume that G = 64, Z = 100, L is 127 and 255, respectively, the signal length is 255×Z. When U = {2, 4}, the estimation accuracy is shown in Fig. 6.

Fig. 6 shows that when the signal length is fixed, the shorter the period of spreading sequence, the better the estimation performance. This is because the shorter the spreading sequence, the more the number of periods, then the separation performance by FastICA algorithm becomes better. As a result, the sequences estimation performance becomes better.

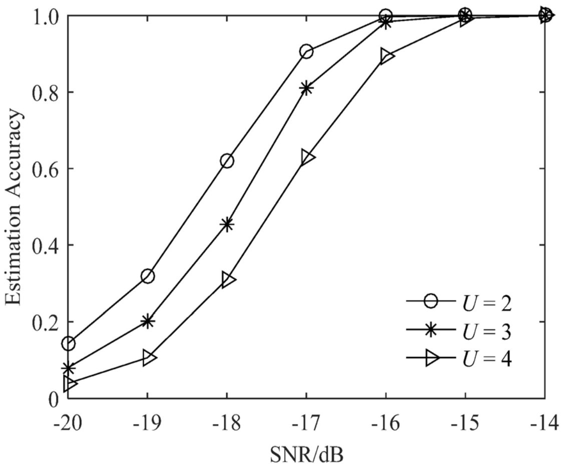


Fig. 5. Users number influence

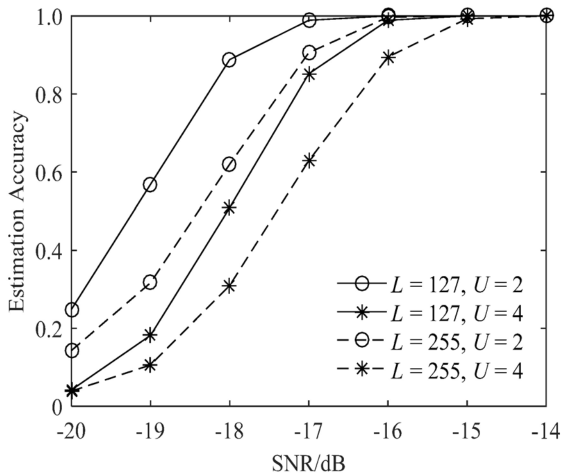


Fig. 6. Period influence

Experiment 7 Performance comparison. To further verify the effectiveness of the FastICA-CNN algorithm, we compared it with the FastICA method in [5], the FastICA-TCF method in [6] and the FastICA-SAE method in [9] from two aspects: estimation accuracy and required time. The spreading sequences are m-sequences. We assume that L = 255, Z = 100, G = 64, when U = {2, 4}, the simulation results are shown in Fig. 7.

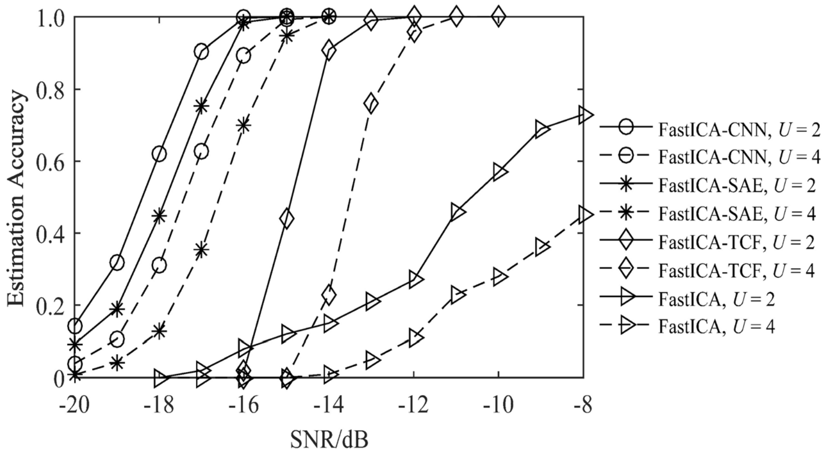


Fig. 7. Algorithm performance comparison

Fig. 7 shows that when the number of users is 2 and SNR is lower than −16 dB, the performance of FastICA-CNN is better than that of FastICA-SAE; when SNR is higher than −16 dB, the performance of the two methods is similar, and the accuracy tends to 1. In addition, the performance of FastICA-CNN and FastICA-SAE is better than that of FastICA-TCF and FastICA methods. From the results, it can be seen that the spreading sequence estimation methods based on deep learning have better estimation performance, especially at low SNRs.

For the time required for sequence estimation, the FastICA-CNN and FastICA-SAE methods mainly include model training and sequence estimation. The simulation time required by these four methods are listed in Table 3.

Table 3. Comparison of required simulation time

Algorithm	Model Training(s)	Sequence Estimation (s)	
		$U = 2$	$U = 4$
FastICA-CNN	84.71	375.4	758.2
FastICA-SAE	93.08	378.6	755.7
FastICA	–	1022.6	1859.6
FastICA-TCF	–	1938.6	16678.9

The results in Table 3 show that the training time of CNN and SAE model is about 84.71s and 93.08s. The trained models can be used all the time. The estimation time of spreading sequences is mainly reflected in the blind separation by FastICA algorithm and the construction of input samples. FastICA-CNN and FastICA-SAE methods share the same data set, and the time required for the calculation of accuracy is almost the same. Therefore, the total time required for spreading sequences estimation by these two methods is nearly the same. The FastICA and FastICA-TCF methods require more estimation time. In addition, when the FastICA-TCF method is used for estimation, the TCF values of m-sequences need to be calculated firstly, however, the time for the TCF values calculation rapidly increases as the number of users increases. The computational complexity of the FastICA-TCF method is the highest.

According to the comparing results, we can draw the conclusion that the proposed FastICA-CNN method is superior for spreading sequence estimation.

5 Conclusions

In this paper, a novel method to build a CNN model to estimate the m-sequences is proposed, and it is applied in LC-DS-CDMA signal for its spreading sequences estimation. The original sequences are directly used as the input samples of CNN and the feasibility of the method is verified by m-sequences. The experimental results show that the estimation performance of the CNN method is better than that based on SAE. Then the CNN method is combined with the FastICA algorithm to obtained the FastICA-CNN method for spreading sequences estimation. The experimental results show that the total time required for model training and sequences estimation is shorter than the comparison methods, and the estimation accuracy is also higher than the comparison methods.

Only the estimation of m-sequences is analyzed in this study. Actually, this method is also suitable for other kinds of sequences. In the future work, we will try to use other sequences and networks to analyze the sequences estimation methods to achieve better performance. In addition, we only consider the situation when signal is synchronized, the study on asynchronous signals is also the next research focus.

References

[1] T. Q. Zhang, L. Zhao, T. Zhang, K. Yang, A blind recognition method of binary pseudo-random sequence, Journal of Electronics & Information Technology 40(2)(2018) 394-399.

[2] P. C. J. Hill, M. E. Ridley, Blind estimation of direct-sequence spread spectrum m-sequence chip codes, in: Proc. IEEE Sixth International Symposium on Spread Spectrum Techniques and Applications, 2000.

[3] E. Heydtmann, J. M. Jensen, On the equivalence of the Berlekamp-Massey and the Euclidean algorithms for decoding, IEEE Transactions on Information Theory, 46(7)(2000) 2614-2624.

[4] E. S. Warner, B. Mulgrew, P. M. Grant, Triple correlation analysis of m-sequences, Electronics Letters, 29(20) (1993) 1755-1756.

[5] X. T. Ren, H. Xu, Z. T. Huang, F. H. Wang, F. B. Lu, Fast-ICA based optimize blind estimation of spreading sequence of CDMA signals, Acta Electronica Sinica, 40(8)(2012) 1532-1538.

[6] Z. J. Zhao, X. W. Gu, F. F. Qiang, L. Shen. Blind estimation of PN codes in multi-user LSC-DSSS signals, Journal of Communications, 12(1)(2017) 55-61.

[7] Z. J. Zhao, F. F. Qiang, M. Li, L. Shen, H. Q. Wang, Blind estimation of pseudo-random noise codes in NPLSC-DSSS signals based on goodness of fit test, Journal of Electronics & Information Technology, 39(3)(2017) 749-753.

[8] F. F. Qiang, Z. J. Zhao, Y. Chen, L. Shen, X. Y. Jiang, TCF–SAE Algorithm for m-Sequence Recognition, in: Proc. The 4th International Conference on Fuzzy Systems and Data Mining, 2018.

[9] F. F. Qiang, Z. J. Zhao, J. N. Shang, L. Shen, Estimation of spreading sequences in LC-DS-CDMA signals based on sparse auto-encoder, Evolutionary Intelligence, 13(8)(2020) 235-246.

[10] A. Verma, P. Singh, J. S. Rani Alex, Modified convolutional neural network architecture analysis for facial emotion recognition, in: Proc. International Conference on Systems, Signals and Image Processing (IWSSIP), 2019.

[11] A. S. Shamsaldin, P. Fattah, T. A. Rashid, N. K. Al-Salihi, The study of the convolutional neural networks applications, UKH Journal of Science and Engineering, 3(2)(2019) 31-40.

[12] I. Bakkouri, K. Afdel. Convolutional neural-adaptive networks for melanoma recognition, in: Proc. International Conference on Image and Signal Processing, 2018.

[13] T. A. Rashid, Convolutional neural networks based method for improving facial expression recognition, in: Proc. The International Symposium on Intelligent Systems Technologies and Applications, 2016.

[14] L. W. Zhang, Z. Q. Shi, J. Q. Han. Pyramidal temporal pooling with discriminative mapping for audio classification, IEEE-ACM Transactions on Audio, Speech, and Language Processing, 28(2020) 770-784.

[15] S. Y. Li, Y. W. Chen, R. X. Jiang, X. Tian, Image denoising via multi-scale gated fusion network, IEEE Access, 7(2019) 49392-49402.