Hindawi
Journal of Electrical and Computer Engineering
Volume 2023, Article ID 6660715, 10 pages
https://doi.org/10.1155/2023/6660715

Research Article

BRD_ESRNet and SRS-Based Channel Estimation

Rifei Yang, Guohua Yao, and Zhuhua Hu

School of Information and Communication Engineering, Hainan University, Haikou, China

Correspondence should be addressed to Zhuhua Hu; eagler_hu@hainanu.edu.cn

Received 14 June 2023; Revised 27 August 2023; Accepted 25 September 2023; Published 7 October 2023

Academic Editor: Andrea Tani

In wireless communication, the channel function estimated commonly has errors due to the influence of noise, so traditional channel estimation methods cannot accurately estimate the real channel function. Aiming at this problem, we propose a channel estimation method that combines sounding reference signal (SRS) remapping with the deep-learning network BRD_ESRNet. BRD_ESRNet consists of image denoising using a deep convolutional neural network with batch renormalization (BRDNet) and an expanded superresolution convolutional neural network (ESRCNN). At the transmitter side, we first map the SRS into four-box structures, and then, the four-box structures are scattered distribution throughout the time-frequency resource block. At the receiver side, we first perform the modified least squares (LS) estimation based on the four-box structure and place the result into the top-left resource unit of the four box. Then, we perform linear interpolation for the whole resource block. Finally, we equate the estimated channel matrix to a low-resolution image containing noise and input it to BRD_ESRNet. Thus, we obtain data with high resolution and achieve the purpose of reducing the estimation error of the channel function. The experimental results show that the proposed method in this paper has a significant improvement in performance compared to the methods of Soltani et al. and Nithya et al. In this paper, the methods of Soltani et al. and Nithya et al. are referred to as methods 1 and 2, respectively.

1. Introduction

Mobile communication technology is one of the fastest-developing technologies in recent decades [1–4]. In order to meet the high demand for communication quality, the channel estimation is necessary at the receiver. In 3rd Generation Partnership Project (3GPP) LTE-Advanced (LTE-A), known SRS [5] is sent to the receiver coming with the information data, and at the receiving end, the channel estimation method, such as least squares (LS) and minimum mean-square error (MMSE), is performed on the basis of SRS, and then, the channel coefficients at the SRS location are estimated. After that, the channel coefficients of the whole LTE-A uplink resource block are obtained by time-frequency interpolation. Also, the signal-to-noise ratio (SNR) estimation can be made on the basis of SRS. The condition of the channel is judged according to the SNR. When the channel condition is good, the transmitter will use higher modulation such as 16QAM modulation or 64QAM modulation. When the channel condition is bad, the estimated channel function is not very accurate and the transmitter uses BPSK modulation.

With the booming development of deep learning, deep learning-based communication techniques have attracted great attention and are widely used in each module of the receiver, including channel estimation [6–9], channel state information feedback [10], signal detection [11], and channel equalization [12].

Compared with traditional channel estimation methods, the deep learning method can make use of the learned nonlinear features of the channel function to obtain better estimation performance [13–16]. In [14], the joint model and data-driven receiver scheme for data-dependent superimposed training (DDST) in the condition of imperfect hardware are proposed. In [15], the extreme learning machine (ELM) method is used to channel estimation for RIS-assisted OFDM systems with insufficient CP. In [16], a new idea for channel estimation based on deep learning is provided. In this idea, the whole time-frequency grid of the channel coefficients estimated by traditional channel estimation method such as LS or MMSE is considered as a twodimensional image, and then, we use the denoising network and image optimization network for further channel estimation.

From the literature research, we know that the LS method has been widely used because of its low complexity, but the performance of LS channel estimation is affected by the noise. This is due to the fact that the channel function estimated by the LS method contains the noise item. So the motivation of our study is to eliminate the noise from the estimated channel function as much as possible and improve the performance. To address this problem, this paper proposes a modified LS method based on remapping SRS on the four-box structure in the time-frequency grid. Through this process, the noise can be reduced from the estimated channel function.

Lately, in [17], a novel deep convolutional neural network (CNN) named BRDNet is proposed for image denoising, and image-denoising performance has been enhanced. In [18], an ESRCNN is presented for single image superresolution. If the estimated channel function matrix is regarded as a 2D image, we can use concatenated BRDNet and ESRCNN to denoise and enhance its resolution. We call this network model as BRD ESRNet.

The contributions of this paper are summarized as follows:

(1) Remapping the SRS by four-box structure in the time-frequency resource block

(2) Use modified LS method to reduce the noise from estimated channel function

(3) Model the estimated channel function as a 2D image and put it into the denoising network

(4) Use the concatenated BRDNet and ESRCNN named BRD ESRNet to denoise and enhance the resolution for channel function

The rest of this paper is organized as follows. In Section 2, we provide the related work of the proposed method. In Section 3, the system model, the SRS remapping structure, and the modified LS method are briefly described, and the proposed BRD ESRNet is presented. In Section 4, we present the experimental results of the proposed scheme. In Section 5, we give the conclusions.

2. Related Work

2.1. SRS Design in LTE-A. In the LTE-A [19] uplink, each wireless frame consists of 10 subframes in the time domain; each subframe contains two time slots; that is, one wireless frame of the LTE-A system contains 20 time slots. Each time slot contains seven SC-FDMA symbols (normal cyclic prefix) or six SC-FDMA symbols (extended cyclic prefix). The subframe structure of the LTE-A uplink and SRS position are shown in Figure 1 [20].

The SRS is obtained by the cyclic shifting of the ZC sequence, defined as

$$r^{\mathrm{SRS}}(n) = r^{\alpha}_{u,v}(n) = e^{j\alpha n}\overline{r}_{u,v}(n) \quad 0 \le \mathrm{n} \le M^{\mathrm{RS}}_{\mathrm{SC}}, \tag{1}$$

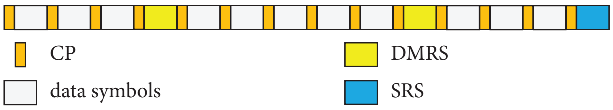


Figure 1: Subframe structure and SRS position.

where $r^{\alpha}_{u,v}(n)$ denotes the base sequence, u denotes the sequence-group number, v denotes the base sequence number within the base sequence group, j denotes the imaginary part unit, $M^{\mathrm{RS}}_{\mathrm{SC}}$ denotes the length of the SRS, and α is a circular shift defined as

$$\alpha = 2\pi \frac{n^{\mathrm{CS}}_{\mathrm{SRS}}}{8} \quad n^{\mathrm{CS}}_{\mathrm{SRS}} = 0, 1, 2, 3, 4, 5, 6, 7, \tag{2}$$

where $n^{\mathrm{CS}}_{\mathrm{SRS}}$ is provided by the upper level.

When a user sends an SRS in the LTE-A uplink, the SRS needs to be mapped to a resource block before it can be transmitted. An SRS sequence is mapped on every other subcarrier and on the last symbol per subframes. The entire SRS sequence consumes twice the bandwidth of the sequence length. Since there is no sufficient reference signal in the time domain, the channel estimation based on SRS is unsuitable for fast-fading channels.

2.2. Channel Estimation. As shown in Figure 2, it is assumed that in the mobile communication system, the base station has d antennas and the user side has 1 antenna, and the channel is a multipath fading channel in the system model. For simplicity, we only consider one of the d links, in other words, the single-input single-output (SISO) OFDM system. At the user side, let τ is a vector of SRS sequence in frequency domain, with length K and $|\tau_k|^2 = 1, k = 1, \ldots, K$. At the base station, after N-point FFT, the received vector can also be represented in the frequency domain by [21]

$$x = \tau h + n, \tag{3}$$

where the length of the received vector x is K and h denotes the channel function matrix of $K * K$ between the user and the base station. We consider the transmitter and receiver is perfect synchronization, and there is no intercarrier interference (ICI), so h is a diagonal matrix. n is a white Gaussian noise vector with the zero mean and element-wise variance σ_n^2.

Estimation of the channel function at the SRS position is performed using the LS algorithm, and the LS estimate of the channel function is expressed as [22]

$$h_{\mathrm{LS}} = \frac{1}{\tau} x = h + \frac{1}{\tau} n. \tag{4}$$

Correspondingly, the mean-square error (MSE) of the LS estimate is expressed as

$$J_{\mathrm{LS}} = E\left\{ \left\| h - h_{\mathrm{LS}} \right\|^2 \right\} = \sigma_n^2. \tag{5}$$

Note that the MSE is proportional to σ_n^2, which implies that the performance of LS depends on the noise intensity.

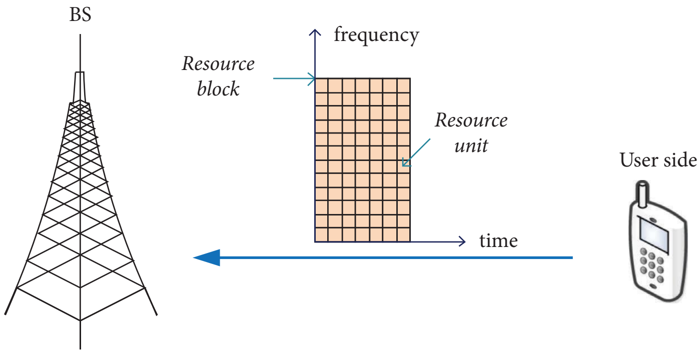


Figure illustration		
System Model		
Symbols	Counting	Description
🗼	1	Base station (with d antennas)
📱	1	User side (with 1 antenna)

Figure 2: Resource block in uplink transmission.

Better than the LS estimation is the MMSE estimation, which is obtained by multiplying the LS estimate at the SRS location with the filter matrix [23]:

$$h_{\text{MMSE}} = R_{h\hat{h}}\left(R_{\text{hh}} + \sigma_n^2\left(xx^H\right)^{-1}\right)^{-1} h_{\text{LS}}. \tag{6}$$

In the same way, the mean-square error of the MMSE estimate is expressed as

$$J_{\text{MMSE}} = \left\{\left\| h - R_{h\hat{h}}\left(R_{\text{hh}} + \sigma_n^2\left(xx^H\right)^{-1}\right)^{-1} h_{\text{LS}}\right\|^2\right\}, \tag{7}$$

where the channel matrix h has zero means, $\mathrm{R}_{h\hat{h}}$ denotes the mutual correlation matrix of the real channel matrix and the real channel matrix, and R_{hh} denotes the autocorrelation matrix of the real channel matrix.

2.3. Image Denoising and Superresolution. Deep learning neural networks, such as the denoising convolutional neural network (DnCNN) [24] and the image restoration convolutional neural network (IRCNN) [25], are very popular for image denoising. These networks utilize convolutional neural network (CNN) techniques as a noise prediction method. The recently proposed BRDNet is to achieve the denoising effect by performing the corresponding convolution, batch renormalization, and residual learning on the image. Image superresolution (SR) [18, 26, 27] is a class of algorithms for image resolution enhancement. Deep learning-based algorithms have made great progress in the problem of enhancing low-resolution images to high-resolution images, and ESRCNN provides a completely new idea for stable resolution enhancement, which leads to low-noise and high-resolution channel functions.

3. Proposed Channel Estimation Denoising Method Based on Remapping SRS and Then BRD_ESRNet

3.1. System Model. As shown in Figure 3, when the SRS sequence is mapped to physical resources, we use a new mapping method different from the SRS in the LTE-A. We map the SRS into a four-box structure, and the four-box structure is scattered throughout the resource block. We call this new mapping method as SRS remapping, it will be introduced in the next section. The transmitted SRS was transmitted through the channel and added with AWGN noise. At the receiver side, we proposed a modified LS channel estimation method based on SRS remapping, introduced in Section 3. Time-frequency interpolation is used to gain the whole estimated channel matrix, which is then put into BRD_ESRNet composed of BRDNet and ESRCNN to denoise and enhance resolution. Finally, the accurate estimated channel matrix can be obtained.

3.2. Remapping SRS. In order to eliminate the influence of noise in channel estimation, the remapping SRS is adopted in this paper, as shown in Figure 4. First, the four-box structure is consisted of four resource units that are adjacent OFDM symbols and subcarriers, then, the SRS signal is placed at the diagonal position of the four-box structure, represented by blue squares, and the remaining two resource units do not hold any data. Second, the four-box structures are scattered throughout the resource block, as shown in Figure 4(a), this is similar to the scattering distribution of pilot symbols in [28].

3.3. Modified LS Algorithm. In Figure 4, we represent the four adjacent channel coefficients as $h_{i,i}$, $h_{i,i+1}$, $h_{i+1,i}$, and $h_{i+1,i+1}$, respectively. Where $h_{i,i}$ denote the channel coefficients at the i th subcarrier and the i th OFDM symbol. τ_{i*2+1} and τ_{i*2+2} denote SRS symbols that are placed at the top left and bottom right, respectively. According to (3), the received signal is represented as

$$\begin{cases} x_{i,i} = \tau_{i*2+1}h_{i,i} + n_{i,i}, \\ x_{i,i+1} = 0 \cdot h_{i,i+1} + n_{i,i+1}, \\ x_{i+1,i} = 0 \cdot h_{i,i+1} + n_{i+1,i}, \\ x_{i+1,i+1} = \tau_{i*2+2}h_{i+1,i+1} + n_{i+1,i+1}. \end{cases} \tag{8}$$

Since these $n_{i,i}$, $n_{i,i+1}$, $n_{i+1,i}$, and $n_{i+1,i+1}$ are samples of white Gaussian noise random variable, we can assume that the noises on adjacent subcarriers are equal $n_{i,i} \approx n_{i+1,i}$; $n_{i,i+1} \approx n_{i+1,i+1}$. So, we can calculate the following expression:

$$x_{i,i} + x_{i+1,i+1} - x_{i,i+1} - x_{i+1,i} \approx \tau_{i*2+1}h_{i,i} + \tau_{i*2+2}h_{i+1,i+1}. \tag{9}$$

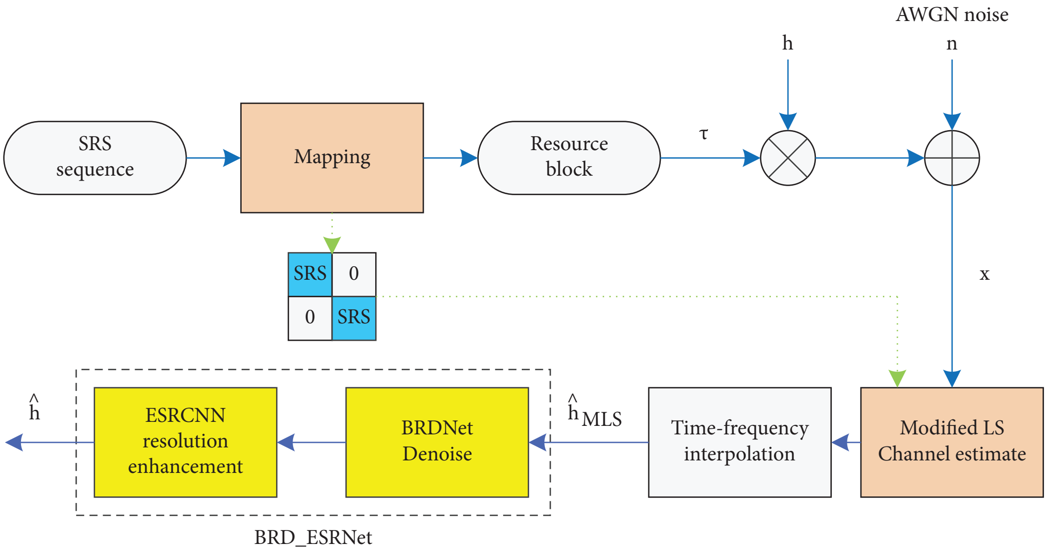
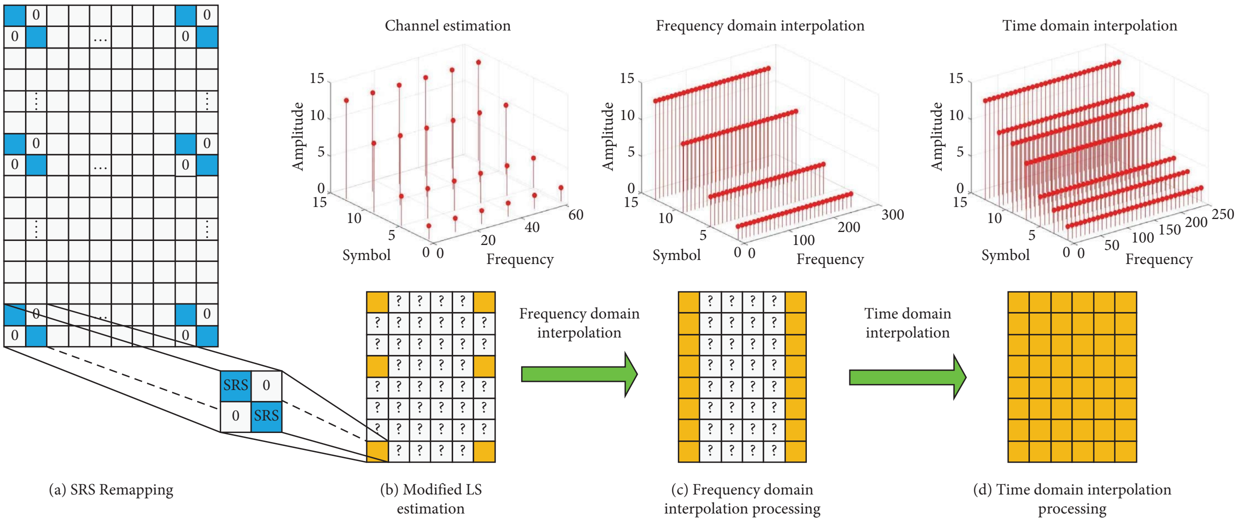

AWGN noise
h
n
SRS sequence
Mapping
Resource block
τ
x
SRS
0
0
SRS
ĥ
ESRCNN resolution enhancement
BRDNet Denoise
ĥMLS
Time-frequency interpolation
Modified LS Channel estimate
BRD_ESRNet

Figure 3: System model.


Channel estimation
Frequency domain interpolation
Time domain interpolation
Amplitude
Symbol
Frequency
Frequency domain interpolation
Time domain interpolation
SRS
0
?

(a) SRS Remapping

(b) Modified LS estimation

(c) Frequency domain interpolation processing

(d) Time domain interpolation processing

Figure 4: Schematic of channel estimation and interpolation process.

If the channel is not varying very fast in the frequency domain, we can assume that the channel coefficients on adjacent subcarriers are approximately equal $h_{i,i} \approx h_{i+1,i+1}$. Therefore, using (7), the estimated channel coefficients on the top left of the four-box structure can be expressed as

$$\widehat{h}_{i,i} = \frac{\left(x_{i,i} + x_{i+1,i+1} - x_{i,i+1} - x_{i+1,i}\right)}{\tau_{i*2+1} + \tau_{i*2+2}}, \qquad (10)$$

$$\widehat{h}_{\mathrm{MLS}} = \left\{\widehat{h}_{i,i} \middle| i = 0, 1, \ldots\right\},$$

$\widehat{h}_{\mathrm{MLS}}$ is much more accurate than h_{LS} in (4) because of no noise item. But because the noise on adjacent subcarriers is not exactly equal, the noise cannot be completely eliminated in the estimated channel coefficients.

As shown in Figure 4(b), after the modified LS channel estimation is performed for each four-box structure, the estimated channel coefficient is placed at the top-left resource unit in each four-box structure. In Figure 4(c), frequency domain interpolation is performed for the resource block. In Figure 4(d), the resource blocks are interpolated in the time domain. Finally, the channel function of the whole resource block has been obtained. Although the effect of MMSE is still better than that of our modified LS algorithm, but its complexity is higher. So our modified LS algorithm is chosen as the input for the deep learning network in this paper.

3.4. Network Structure. The network model BRD ESRNet in this paper is proposed based on deep learning, which consists of two network models, BRDNet and ESRCNN, to further improve the accuracy of our modified LS estimation. BRD_ESRNet processes the guide frequency information in two ways, which are denoising and resolution improvement. The network structure diagram of BRD_ESRNet is shown in Figure 5.

BRDNet is divided into two networks, mainly composed of convolution (Conv), batch renormalization (BRN), and rectified linear units (ReLU). In the upper network, layers 116 are Conv + BRN + ReLU and layer 17 is Conv. In the lower network, layers 1, 9, and 16 are Conv + BRN + ReLU, layers 2–8, and 10–15 are dilated convolution, and layer 17 is Conv.

ESRCNN is a more common network model for enhancing image resolution, which mainly consists of Conv and ReLU. After feeding low-resolution images to ESRCNN, it uses a three-layer convolutional network to improve the image resolution. The first convolutional layer uses 64 filters of size 9×9. The second layer consists of 3 convolutions, each using 32 filters of size 1×1, 3×3, and 5×5. The second layer is activated with ReLU, and finally, the outputs of the 3 convolutions of the second layer are averaged. The third layer uses a filter of size 5×5 to reconstruct the image.

As shown in Figure 5, the channel function estimated by modified LS is the first input to BRDNet for the denoising process. The denoised data are then fed into ESRCNN for resolution enhancement, and a more accurate estimate of the channel function is obtained. In particular, BRDNet and ESRCNN are connected using a cascade structure.

3.5. Datasets and Loss Function. In this paper, we focus on LTE-A uplink communication, where the channel function between the transmitter and receiver is a complex matrix of size 72×14, denoted as 72 subcarriers in the frequency domain and 14 symbols in a subframe, respectively. It can be represented as two 2D images, one 2D image of the real part of the channel function and the other 2D image of the imaginary part of the channel function, respectively.

We use the Rayleigh multipath channel model and pseudorandom Gaussian noise generator to produce the real channel function h, and it is efficient to simulate the real wireless communication scenarios. The noisy channel function $\widehat{h}_{\mathrm{MLS}}$ is produced by a modified LS estimation algorithm. We can easily generate a lot of the experimental sample, which can act as training data, for example, 4,000 channel matrices of size 72×14, which are complex matrices.

The input of BRD ESRNet is the modified LS-estimated channel function $\widehat{h}_{\mathrm{MLS}}$, and the output is the denoised channel function $\widehat{h}$, defined as

$$\widehat{h} = f_E\left(\Xi_E; f_B\left(\Xi_B; \widehat{h}_{\mathrm{MLS}}\right)\right), \tag{11}$$

where f_B and f_E denote the BRDNet and ESRCNN functions, respectively, and Ξ_B and Ξ_E denote the set of parameter values for BRDNet and ESRCNN, respectively.

As shown in Figure 5, BRD ESRNet consists of two network structures, and for the loss function of the first training algorithm, i.e., the loss function of BRDNet, we can express it as follows:

$$l_1 = \frac{1}{\|N\|} \sum_{h \in N} \left\| f_B\left(\Xi_B; \widehat{h}_{\mathrm{MLS}}\right) - h \right\|. \tag{12}$$

The best modified LS estimates are predicted by the best weights of BRDNet, and finally, by inputting into ESRCNN, the overall minimum loss function l_2 can be obtained:

$$l_2 = \frac{1}{\|N\|} \sum_{h \in N} \left\| f_{\mathrm{E}}\left(\Xi_E; f_B\left(\Xi_B; \widehat{h}_{LS}\right)\right) - h \right\|. \tag{13}$$

Summarizing the abovementioned equation, the loss function between the channel function estimated by BRD_ESRNet and the actual channel function can be defined as

$$l = \frac{1}{\|N\|} \sum_{h \in N} \|\widehat{h} - h\|, \tag{14}$$

where N denotes all the datasets, h denotes the real channel function, and $\widehat{h}$ denotes the estimated channel function.

4. Experimental Results

In this experiment, we apply the Keras package and the TensorFlow package to train the proposed BRD_ESRNet model; all experiments were performed in a Python 3.6 environment. For BRDNet, the learning rate is set to 1×10^{-3}, the minimum batch size is 50, and the maximum number of iterations is 50. For ESRCNN, we set the learning rate to 1×10^{-3}, the minimum batch size to 128, and the maximum number of iterations to 100.

The total number of samples for BRD_ESRNet is 4000, and the size of each uplink channel matrix is 72×14. The total number of samples for BRDNet is 4,000, consisted of 2,000 training samples and 2,000 testing samples. The test output of BRDNet is used as the input of ESRCNN, so the total number of samples of ESRCNN is 2,000, consisted of 1,950 training samples and 50 test samples.

We input the noisy channel functions with different signal-to-noise ratios (SNR) into BRD_ESRNet and then compare the noisy channel function, the denoised channel function, and the real channel function, which is clean and without noise. As shown in Figure 6, the error between the modified LS estimated channel function and the real channel function is large when the SNR is between 0 dB and 8 dB, respectively; however, the error between the output denoised channel function and the real channel function is significantly reduced.

The conventional channel estimation is shown in Figure 7. When conventional SRS mapping is used, the channel function error of LS estimation is the largest and the channel estimation error of MMSE estimation is the smallest. The channel function error of modified LS estimation and the channel estimation error of MMSE estimation are significantly reduced when using SRS remapping (four box), and the ideal MMSE estimation has the smallest channel function error and gives a lower bound on the achievable average error, which is impossible in engineering practice due to the fact that the whole of the channel statistical characteristic are unlikely to be obtained in practice. When the SNR is 0 dB ~

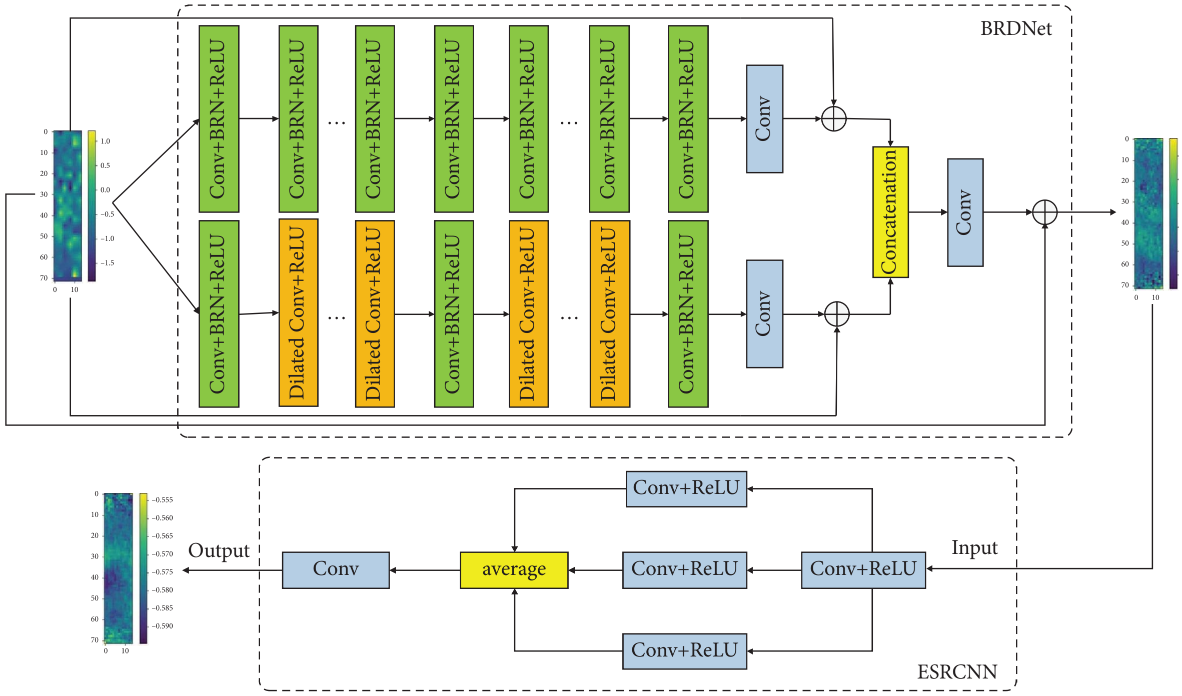


Figure 5: BRD_ESRNet network structure.

16 dB, the estimation performance of MMSE is significantly better than that of LS. At the SNR of 0 dB, the average error is reduced by 0.1572 dB using the four-box mapping approach compared with the SRS traditional mapping approach in the LS estimation method. When the SNR is 16 dB ~ 25 dB, the estimation performance of MMSE is almost the same as that of LS, regardless of how the SRS is mapped. This is because of the effect of noise has been reduced.

The channel function estimated by modified LS and the channel function estimated by MMSE are input to BRD_ESRNet for denoising, respectively, and the denoising effect is shown in Figure 8. Although the channel functions estimated using MMSE are input to BRD_ESRNet to obtain a smaller average error, but the MMSE algorithm is much more complex than the modified LS algorithm. So, in the subsequent comparison of the denoising performance of the network models, the channel function estimated by modified LS is used.

To prove that our method has better denoising effect, we compare the proposed method with method 1 [16], method 2 [29], ideal MMSE, and MMSE in different SNR cases and use the average error as the comparison criterion. The comparison results are shown in Figure 9, where 144 SRSs are inserted in each resource block.

In Figure 9, the proposed method in this paper has the best estimation performance, followed by method 2 and method 1, and finally by the ideal MMSE and MMSE. When the SNR is 0 dB~12 dB, the estimation performance of our method, method 1 and method 2, exceeds that of the ideal MMSE because the estimated channel function contains a large amount of noise, which is beneficial for the network model to extract the noise features. At an SNR of 0 dB, the average errors of our method, method 2 and method 1 are reduced by 0.072 dB, 0.0516 dB, and 0.0286 dB, respectively, compared with the ideal MMSE. Compared with MMSE at SNR 0 dB, the average errors of our method, method 2 and method 1 are reduced by 0.1572 dB, 0.1368 dB, and 0.1138 dB, respectively. When the SNR is 12 dB ~ 25 dB, it is still our method that has the best estimation performance. As the noise decreases and the network model extract fewer noisy features, the estimation performance of the ideal MMSE is better than that of methods 2 and 1.

Deep learning network has good estimation performance; however, it needs time for training and prediction. The complexity is evaluated by the number of floating point operations (FLOPs) metric. That is, the complexity of the network model is expressed as [30]

$$O\left(M_h \cdot M_w \cdot K_h \cdot K_w \cdot C_{\text{in}} \cdot C_{\text{out}}\right), \tag{15}$$

where M$_h$ and M$_w$ represent the height and width of the output feature map, respectively, K_h and K_w represent the height and width of the kernel, respectively, and C_{in} and C_{out} are the number of input channels and output channels, respectively. In this paper, the output feature map is a channel matrix, so $M_h = 72$, $M_w = 14$. The complexity of the different networks can be expressed as follows: BRD_ESRNet is denoted as $(72 \cdot 14 \cdot 1217730)$. In [16], Soltani et al. proposed the network model as ChannelNet, which has a complexity denoted as $O(72 \cdot 14 \cdot 673986)$. In [29], Nithya et al. proposed the network model as FFDNet, which has a complexity denoted as $(72 \cdot 14 \cdot 677878)$. The estimated channel function by the modified LS method is input into BRD_ESRNet, ChannelNet, and FFDNet, respectively, and then the FLOPs and actual running times of different deep learning network models are counted, as shown in Table 1.

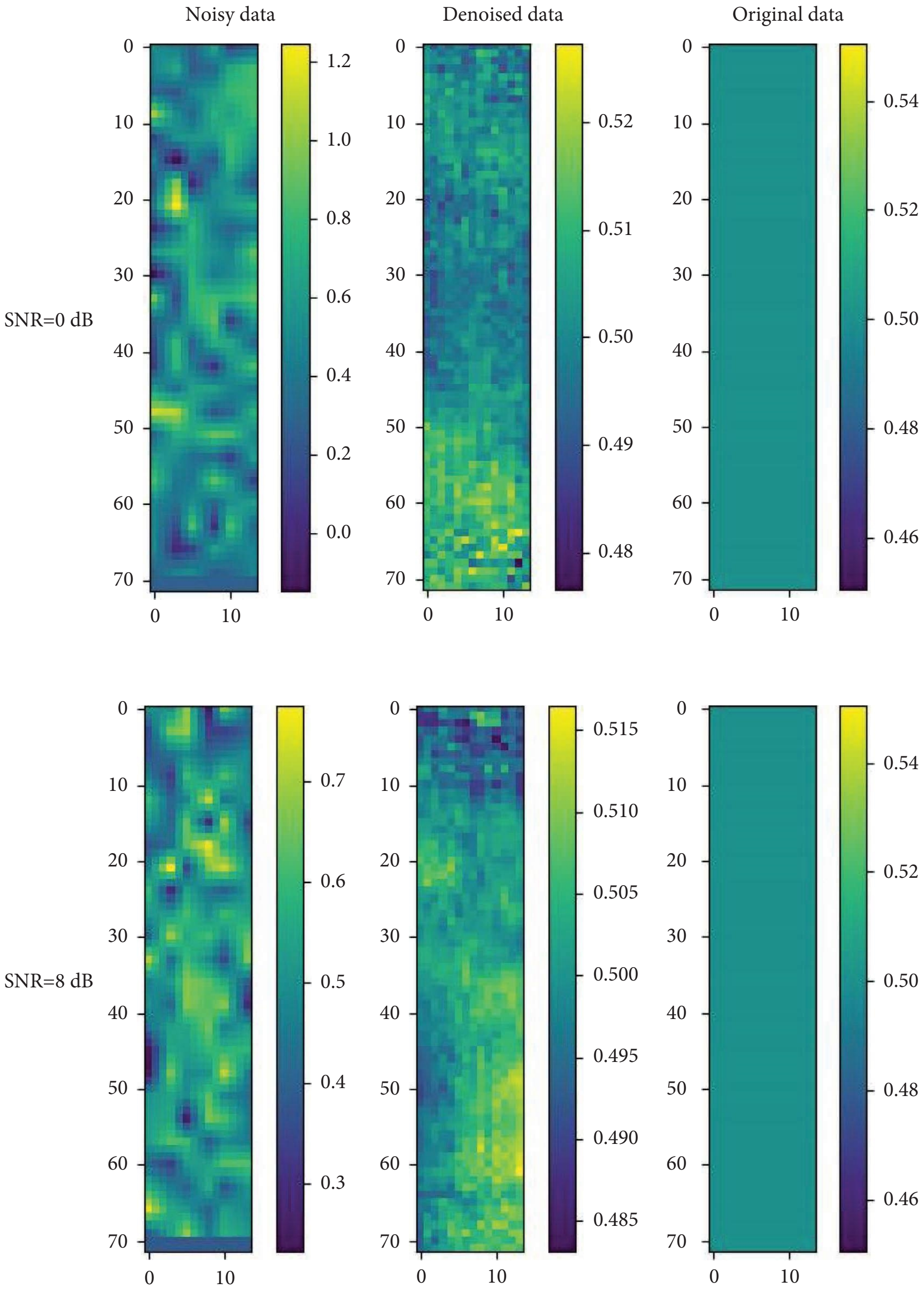


Figure 6: Comparison of the denoising effect of channel function with different SNR.

When the input is a channel function estimated by LS, the time overheads of BRD_SRNet, FFDNet, and ChannelNet are 149.59 milliseconds, 69.33 milliseconds, and 26.88 milliseconds, respectively.

Finally, this paper simulates the effect of different numbers of SRSs on the channel estimation. The average error between the estimated channel function and the real channel function for different numbers of SRSs with an SNR of 8 dB is shown in Figure 10. The average error curves estimated by the ideal MMSE, our method, method 1 and method 2 do not strictly decrease with the increase in the number of SRSs, but fluctuate up and down in a certain range. The reason is that the noise does not decrease with the increase of the SRS signal, so it is not possible to achieve a strict decrease in the average error as the number of SRS increases. Compared with other estimation methods, the average error curves of our method for different numbers of SRSs fluctuate less and are close to a straight line.

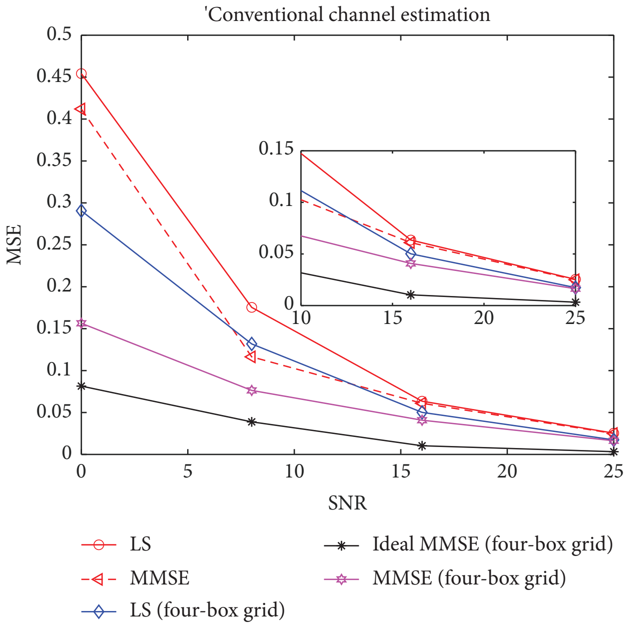


Figure 7: Conventional channel estimation.

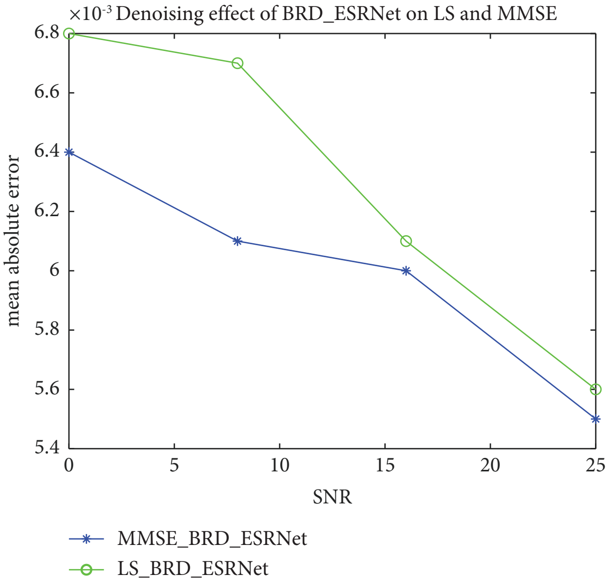


Figure 8: Denoising effect of BRD_ESRNet on LS and MMSE, respectively.

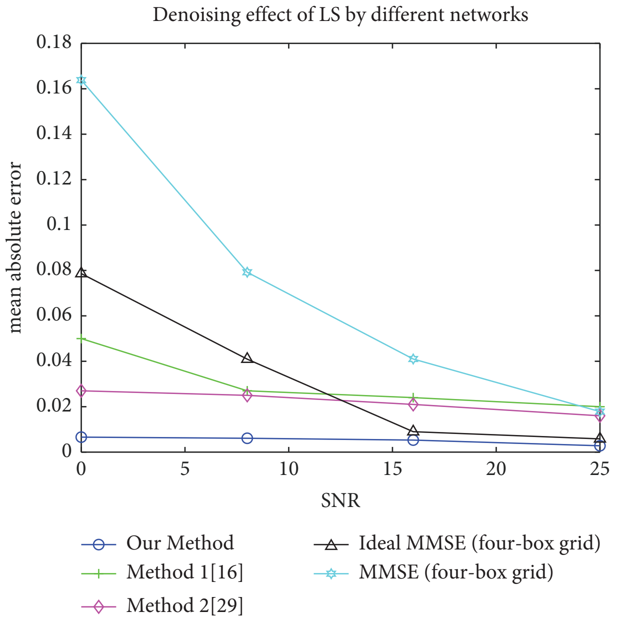


Figure 9: Training results of channel function with different SNR.

Table 1: Complexity analysis of different networks.

Methods	GFLOPs	Running time (ms)
ChannelNet [16]	1.36	26.88
FFDNet [29]	1.37	69.33
BRD_ESRNet (our)	2.45	149.59

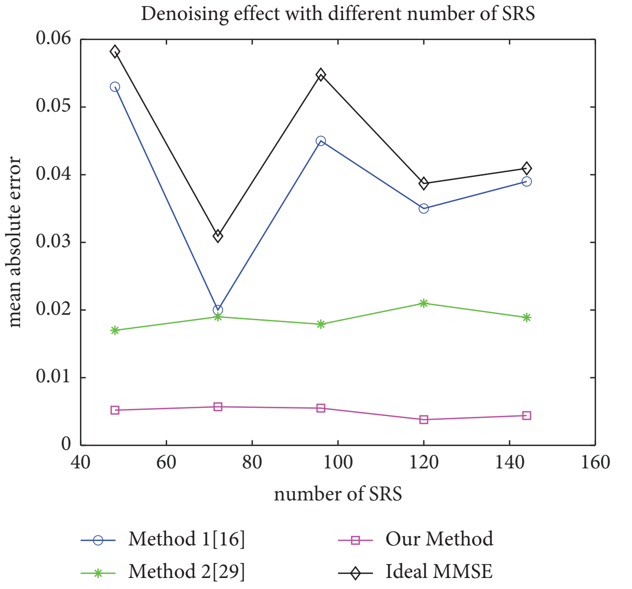


Figure 10: Average error for different numbers of SRS.

5. Conclusion

In this paper, we propose a novel channel estimation method based on the combination of remapping SRS, modified LS, and deep learning. We propose the BRD_ESRNet model. At the user side, we first remap the SRS into the four-box structure, and then at the receiver, the modified LS estimation method based on the four-box structure is applied to estimate the channel coefficient, then it is place at the top-left resource unit of the four-box structure. Following the time and frequency interpolation, the whole estimated time-frequency channel matrix is obtained; it can be represented as two 2D images and is put into the BRD_ESRNet network, which performs denoising and resolution enhancement processes on the channel matrix in turn. It is experimentally demonstrated that our proposed method is very competitive in performance with other advanced channel estimation algorithms; for example, at an SNR of 0 dB, the average error of our proposed method estimation is reduced compared to MMSE estimation by 0.1572 dB. But, compared to ChannelNet and FFDNet, the BRD_ESRNet model has higher complexity and higher time consumption. In the future, we plan to find some lighter network models that can debase the complexity and reduce the time consumption meanwhile guaranteeing the accuracy of channel estimation.

Data Availability

The data used to support the results of this study are available from the corresponding authors upon request.

Conflicts of Interest

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

Acknowledgments

This work was supported in part by the Major Science and Technology Projects of Hainan Province (ZDKJ2021022), the Key Research and Development Project of Hainan Province (ZDYF2022GXJS348), the National Natural Science Foundation of China (61963012 and 62361024), and the Teachers Basic Research Ability Improvement Project of Guangxi (2020KY27011).

References

[1] H. Yang, "Application and development of mobile communication technology," in *Proceedings of the 2021 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 893–896, IEEE, Harbin City, China, June 2021.

[2] X. Li, Z. Hu, C. Shen, H. Wu, and Y. Zhao, "TFF_aDCNN: a pre-trained base model for intelligent wideband spectrum sensing," *IEEE Transactions on Vehicular Technology*, pp. 1–16, 2023.

[3] Z. Hu, Y. Bai, L. Cao, M. Huang, and M. Xie, "A sequential compressed spectrum sensing algorithm against SSDH attack in cognitive radio networks," *Journal of Electrical and Computer Engineering*, vol. 2018, Article ID 4782718, 9 pages, 2018.

[4] Z. Hu, Y. Bai, M. Huang, M. Xie, and Y. Zhao, "A self-adaptive progressive support selection scheme for col- laborative wideband spectrum sensing," *Sensors*, vol. 18, no. 9, p. 3011, 2018.

[5] H. Tian, L. Yang, and S. Li, "SNR estimation based on sounding reference signal in LTE uplink," in *Proceedings of the 2013 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2013)*, pp. 1–5, IEEE, KunMing, China, August 2013.

[6] H. Ye, G. Y. Li, and B. H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, 2018.

[7] C. J. Chun, J. M. Kang, and I. M. Kim, "Deep learning-based channel estimation for massive MIMO systems," *IEEE Wireless Communications Letters*, vol. 8, no. 4, pp. 1228–1231, 2019.

[8] Q. Bai, J. Wang, Y. Zhang, and J. Song, "Deep learning based channel estimation algorithm over time selective fading channels," *IEEE Trans- actions on Cognitive Communications and Networking*, vol. 6, no. 1, pp. 125–134, 2020.

[9] A. Le Ha, T. Van Chien, T. H. Nguyen, W. Choi, and V. D. Nguyen, "Deep learning-aided 5G channel estimation," in *Proceedings of the 2021 15th international conference on ubiquitous information management and communication (IMCOM)*, pp. 1–7, IEEE, Seoul, South Korea, January 2021.

[10] J. Guo, C.-K. Wen, S. Jin, and G. Y. Li, "Overview of deep learning-based CSI feedback in massive MIMO systems," *IEEE Transactions on Communications*, vol. 70, no. 12, pp. 8017–8045, 2022.

[11] T. Diskin, N. Samuel, and A. Wiesel, "Deep MIMO detection," in *Proceedings of the 2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, Sapporo, Japan, July 2017.

[12] D. Erdogmus, D. Rende, J. C. Principe, and T. F. Wong, "Nonlinear channel equalization using multilayer perceptrons with information-theoretic criterion," in *Proceedings of the Neural Networks for Signal Processing XI: Proceedings of the 2001 IEEE Signal Processing Society Workshop (IEEE Cat. No. 01TH8584)*, pp. 443–451, IEEE, North Falmouth, MA, USA, September 2001.

[13] Y. Yang, F. Gao, X. Ma, and S. Zhang, "Deep learning-based channel estimation for doubly selective fading channels," *IEEE Access*, vol. 7, pp. 36579–36589, 2019.

[14] C. Qing, L. Dong, L. Wang, J. Wang, and C. Huang, "Joint model and data-driven receiver design for data-dependent superimposed training scheme with imperfect hardware," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 3779–3791, 2022.

[15] C. Qing, L. Wang, L. Dong, and J. Wang, "Enhanced ELM based channel estimation for RIS-assisted OFDM systems with insuf- ficient CP and imperfect hardware," *IEEE Communications Letters*, vol. 26, no. 1, pp. 153–157, 2022.

[16] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 652–655, 2019.

[17] C. Tian, Y. Xu, and W. Zuo, "Image denoising using deep CNN with batch renormalization," *Neural Networks*, vol. 121, pp. 461–473, 2020.

[18] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[19] 3rd Generation Partnership Project (3GPP), *Evolved Universal Terres-Trial Radio Access (E-UTRA) Physical Channels and Modulation*, 3rd Generation Partnership Project (3GPP), France, 2015.

[20] G. Yao and Z. Hu, "SNR estimation method based on SRS and DINet," in *Proceedings of the 2023 15th International Conference on Computer Modeling and Simulation (ICCMS '23)*, pp. 218–224, ACM, Dalian, China, June 2023.

[21] Q. Hu, F. Gao, H. Zhang, S. Jin, and G. Y. Li, "Deep learning for channel estimation: interpretation, performance, and comparison," *IEEE Transactions on Wireless Communications*, vol. 20, no. 4, pp. 2398–2412, 2021.

[22] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Transactions on Broadcasting*, vol. 48, no. 3, pp. 223–229, 2002.

[23] S. Colieri, M. Ergen, A. Puri, and A. Bahai, "A study of channel estimation in OFDM systems," in *Proceedings of the IEEE 56th vehicular technology conference*, vol. 2, pp. 894–898, IEEE, Vancouver, Canada, September 2002.

[24] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[25] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3929–3938, Honolulu, HI, USA, July 2017.

[26] Y. Wang, H. Lu, and H. Sun, "Channel estimation in IRS-enhanced mmWave system with super-resolution network," *IEEE Communications Letters*, vol. 25, no. 8, pp. 2599–2603, 2021.

[27] M. Soltani, V. Pourahmadi, and H. Sheikhzadeh, "Pilot pattern design for deep learning-based channel estimation in OFDM systems," *IEEE Wireless Communications Letters*, vol. 9, no. 12, pp. 2173–2176, 2020.

[28] M. Dong and L. Tong, "Optimal design and placement of pilot symbols for channel estimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 12, pp. 3055–3069, 2002.

[29] B. Nithya, D. Brijesh, S. K. Kumar, and J. Pathmakarthik, "Pilot based channel estimation of OFDM systems using deep learning techniques," *International Journal on Information Technology*, vol. 15, no. 2, pp. 819–831, 2023.

[30] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," 2016, https://arxiv.org/abs/1611.06440.