Journal of Computer and Communications, 2023, 11, 161-176
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

End-to-End Auto-Encoder System for Deep Residual Shrinkage Network for AWGN Channels

Wenhao Zhao[1], Shengbo Hu[2]

[1]School of Mathematical Sciences, Guizhou Normal University, Guiyang, China
[2]School of Big Data and Computer Science, Guizhou Normal University, Guiyang, China
Email: hsb@nssc.ac.cn

How to cite this paper: Zhao, W.H. and Hu, S.B. (2023) End-to-End Auto-Encoder System for Deep Residual Shrinkage Network for AWGN Channels. *Journal of Computer and Communications*, **11**, 161-176.
https://doi.org/10.4236/jcc.2023.115012

Received: May 3, 2023
Accepted: May 27, 2023
Published: May 30, 2023

Abstract

With the rapid development of deep learning methods, the data-driven approach has shown powerful advantages over the model-driven one. In this paper, we propose an end-to-end autoencoder communication system based on Deep Residual Shrinkage Networks (DRSNs), where neural networks (DNNs) are used to implement the coding, decoding, modulation and demodulation functions of the communication system. Our proposed autoencoder communication system can better re-duce the signal noise by adding an "attention mechanism" and "soft thresholding" modules and has better performance at various signal-to-noise ratios (SNR). Also, we have shown through comparative experiments that the system can operate at moderate block lengths and support different throughputs. It has been shown to work efficiently in the AWGN channel. Simulation results show that our model has a higher Bit-Error-Rate (BER) gain and greatly improved decoding performance compared to conventional modulation and classical autoencoder systems at various signal-to-noise ratios.

Keywords

Deep Residual Shrinkage Network, Autoencoder, End-To-End Learning, Communication Systems

1. Introduction

At present, signal processing in modern communication systems is well-established and can be proven to be optimal. Nevertheless, these processing methods are often linear [1] [2] [3]. In **Figure 1(a)** conventional communication systems, signal processing modules are composed of most sub-modules, e.g., source encoder, channel encoder. These sub-modules are non-linear and can only be approximated as linear modules. And these sub-modules are designed by individual optimization, the optimal submodules thus designed do not guarantee an optimal end-to-end communication system.

In recent years, deep learning has profoundly impacted the development of modern communications technology. The application of deep learning methods to traditional communication submodules has been extensively studied, such as channel coding [4] [5] [6], channel estimation [7]-[12], digital demodulation [10] [11] [12], etc. In addition, deep learning-based methods are used to optimize traditional communication blocks by jointly including joint channel coding [13] [14] [15] and joint channel estimation [16] [17].

However, in contrast to conventional communication systems, the design of end-to-end communication systems based on deep learning networks adopts a holistic approach where both the transmitter and receiver are represented by deep neural networks (DNNs), as in **Figure 1(b)** [18]; The model comprises an auto-encoder, auto-decoder, and a differentiated channel. The auto-encoder learns to encode the symbols to be transmitted as coded data and then sends the

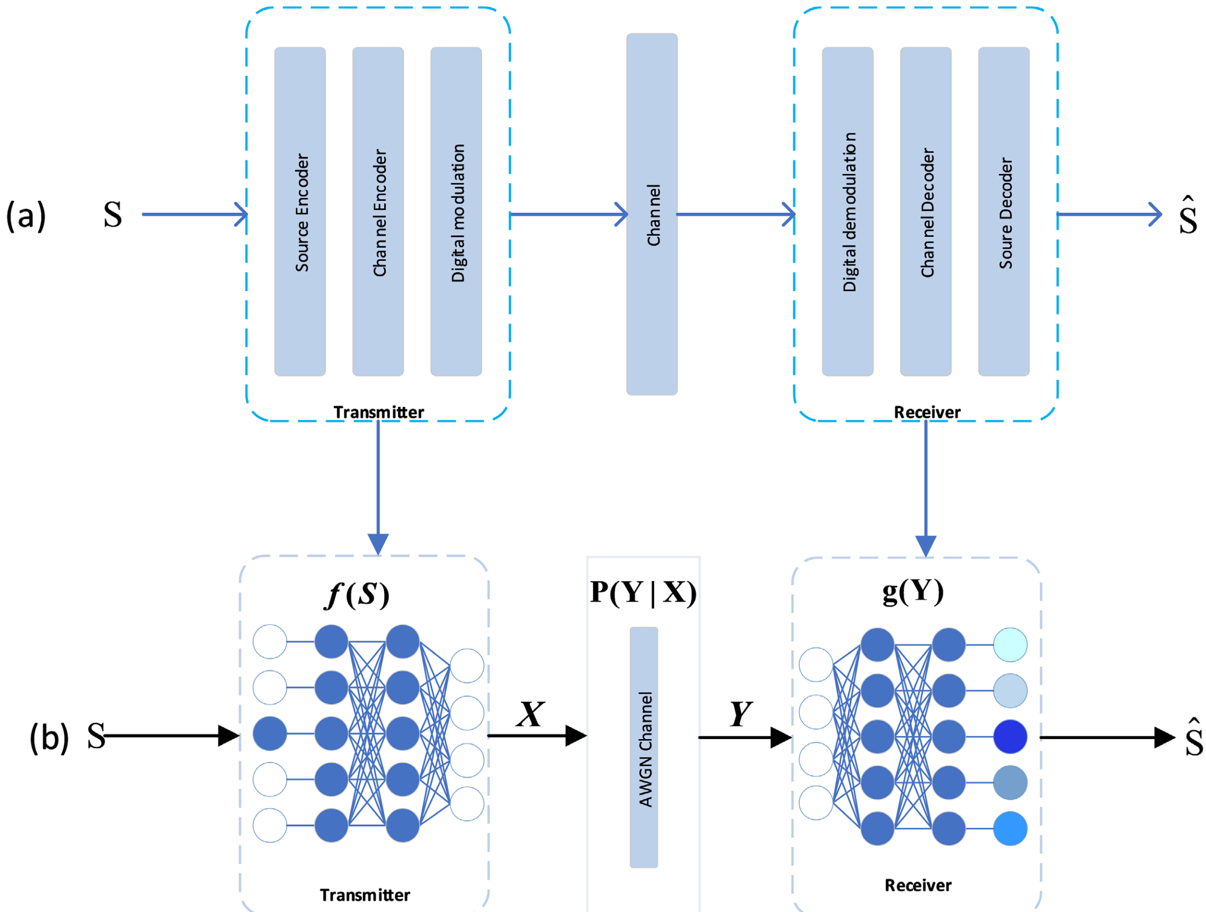

Figure 1. Conventional wireless communication system architecture and autoencoder-based end-to-end communication system architecture: **Figure 1(a)** conventional wireless communication system; **Figure 1(b)** application of neural networks to build autoencoder-based end-to-end communication system, in which neural networks replace the original transmitter and receiver.

coded data to the channel. The decoder will receive the data passing through the channel to recover the transmitted symbols through learning the neural network. The auto-encoder in the model replaces the coding and modulation part of the traditional communication system, while the auto-decoder replaces the traditional communication system decoding and demodulation. The transmitter DNNs and receiver DNNs are trained in a supervised learning manner using actual or simulated data sets. The entire learning process is no longer chunked for the optimization of submodules. But rather, it is left entirely to the deep learning model, thus transforming the problem of data transmission in communication into an end-to-end machine learning optimization problem.

Hence, end-to-end autoencoder is one of the hot topics in communication systems. For example, O'Shea in [18] first proposed and used a deep neural network approach for end-to-end communication system design, demonstrating the feasibility of applying neural networks to end-to-end communication systems; Wu [19] *et al.* used a one-dimensional convolutional neural network to improve on the pioneering work and obtained better results, demonstrating the substantial gain of implementing end-to-end autoencoder systems with complex neural networks. The autoencoder designed by He in [20] using the coding structure of Turbo codes can take full advantage of the advantages of Turbo codes and the flexibility of the encoder. On the other hand, in the unknown channel model [21] [22] [23], for example, in [21], Ye proposed a conditional GAN-based network for solving the data transmission problem in the case of an unknown channel model. The application of meta-learning in [22] also enables end-to-end communication systems under unknown channel models. [23] also apply GAN neural network to solve the unknown channel overfitting problem, so as to improve part of the system performance. Both progressively verify the feasibility and effectiveness of the autoencoder communication system.

Besides, for the autoencoder mentioned above communication system using a conventional network architecture, although the designed method can achieve BER improvement, its gain is small, and the end-to-end autoencoder using a conventional neural network suffers from a reduced learning capability when the signal is in a low signal-to-noise situation. Therefore, it is necessary to propose a new autoencoder communication system for the low SNR situation.

Deep Residual Shrinkage Networks (DRSN) [24] introduce a filtering noise reduction technique based on a deep residual network, which ensures that the model dynamically removes redundant noise from the feature map during training and focuses on learning informative features. This noise reduction process is mainly achieved due to the soft thresholding function in the network structure. Using the unique soft thresholding and attention mechanism, the noise immunity of the network is significantly enhanced, and the recognition accuracy of the system at low SNR is improved. We can see that DRSN has shown an excellent denoising effect in [25] [26] [27].

Traditional signal-denoising algorithms require a lot of expertise and experience in signal analysis, relying mainly on the construction of wavelet filter functions or the selection of modal decomposition methods [28]. At the same time, for different signals, it is necessary to redesign the signal transformation method to re-select the threshold, which is very troublesome and costly in terms of workforce [29]. Therefore, the traditional idea of "signal noise reduction + feature extraction" is problematic. To address these issues, we design a DRSN-AE using an integrated approach to signal noise reduction, feature extraction and diagnosis. Our proposed approach is a novel end-to-end learning-based autoencoder for wireless communication using DRSN.

The unique structure of the DRSN network presents a new idea for signal noise reduction, which consists of a particular residual shrinkage building unit (RSBU), as shown in **Figure 2**, which is a network structure integrating deep residual network, attention mechanism and soft thresholding function. In layperson's terms, with this network, we can catch the points we deservedly need to pay attention to, retain them, and discard the irrelevant points. Thus, it improves the ability of deep neural networks to extract useful features from noisy signals. Recent research on residual blocks has focused on enhancing feature mapping.

Soft thresholding is used as the primary step in signal-noise reduction methods. Its use is to set features with absolute values below a certain threshold to zero and adjust other features towards zero, *i.e.* to shrink them. Here we will set a set of thresholds in advance, and the magnitude of their values may affect the

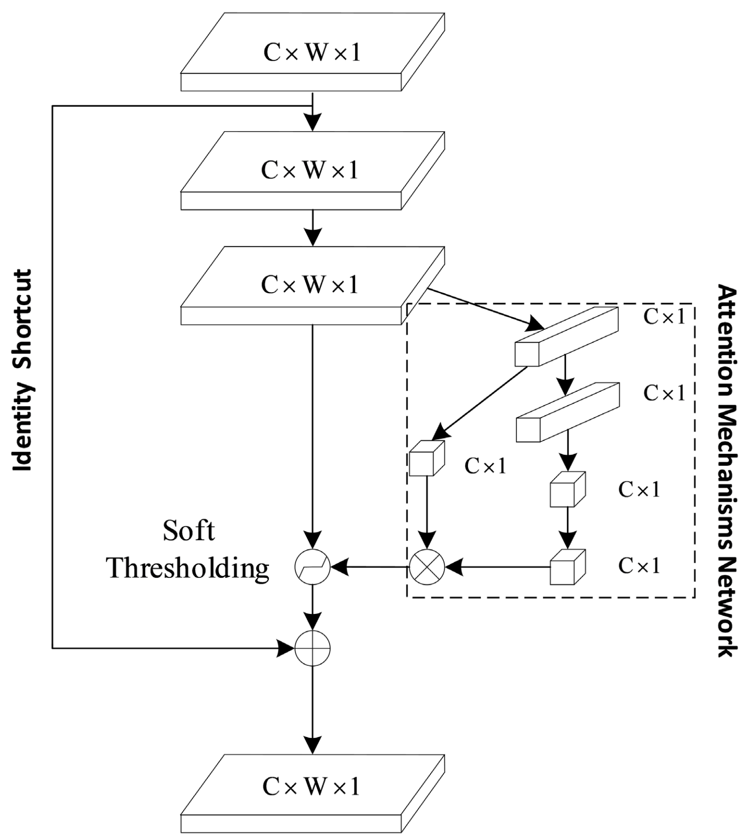

Figure 2. Residual shrinkage block structure.

result of noise reduction. The expression for the soft thresholding function is as follows [24]:

$$y=\begin{cases} x-\tau & x>\tau \\ 0 & -\tau\le x\le\tau \\ x+\tau & x<\tau \end{cases} \tag{1}$$

The derivative of the output of the soft thresholding concerning the input is:

$$\frac{\partial y}{\partial x}=\begin{cases} 1 & x>\tau \\ 0 & -\tau\le x\le\tau \\ 1 & x<\tau \end{cases} \tag{2}$$

In computer vision, the principle of the attention mechanism is to focus attention on regions of interest after a quick scan of the whole region, assigning different proportions of weight to regions of different levels of importance, with the more critical regions occupying more weight. The DRSN automatically learns a set of weights by training a small neural network to weight each channel of the feature map. The implication is that some feature channels are more important, while others are redundant in terms of information; In this way, we can enhance the valuable feature channels and weaken the redundant ones. This improves the accuracy of the prediction.

In the modified residual module, not only is there a soft thresholding function as a non-linear layer, but a sub-network is embedded for automatically setting the threshold required for soft thresholding.

In this paper, we verify the feasibility of our end-to-end paradigm using a differentiated channel model. In this AWGN channel, it is known that the update of weights in neural networks is done chiefly using stochastic gradient descent (SGD), which has minimized the loss function as the goal. Inspired by the excellent denoising effect of Deep Residual Shrinkage Networks (DRSN), we design the Deep Residual Shrinkage Network Auto-Encoder communication system (DRSN-AE)—a channel coding scheme with a data-driven encoder and decoder—to address the above challenges. DRSN-AE can achieve reliability in channel coding under AWGN channels with short to medium block lengths. Such problems are also explored in depth by Shannon in [30] and are summarized as Shannon's theorem, while we demonstrate once again that channel coding can be learned from the data itself in an end-to-end manner with good results. The main contributions are as follows:

- We use the DRSN to model. The design of the residual structure unique to the DRSN-AE autoencoder communication system can mitigate the vanishing gradient and exploding gradient arising from the deepening of the network layers. Our proposed DRSN-AE-based system outperforms the model proposed by N. Wu in [19] for different E_b/N_0 training, and the system still converges quickly and smoothly in a short epochs.
- The DRSN-AE autoencoder has good adaptability and generalization to short and medium block lengths ($L\le 100$) and different code rates. The experi-

mental results show that the DRSN-AE autoencoder with a self-attentive mechanism and a soft thresholding structure can support better coding and decoding in high-noise situations.

The rest of the paper is organized as follows. Section II describes the end-to-end learning of the communication system in AWGN channel. The simulation results and performance evaluation are detailed in Section III. Conclusions are given in Section IV.

2. End-to-End Communication Systems in AWGN Channel

As described in the introduction, end-to-end models designed by applying a data-driven approach are gradually being promoted and applied. We have adopted an integrated approach to design the autoencoder communication system in which signal noise reduction, feature extraction and diagnosis are performed. In this section, the system architecture of DRSN-AE is first introduced, as well as the system parameters.

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar.

2.1. System Architecture

Every communication system is composed of information sources, a channel, and an information host. The symbolic information s sent by the communication system contains k bits of information. After passing through the channel, it is subject to noise interference, which will cause the signal accepted by the host to differ from the original signal, *i.e.*, the communication system has a BER, and the criterion for judging the performance of the communication system is the BER. So we propose an end-to-end communication system aimed at reducing the BER, the DRSN self-coding communication system, whose architecture is shown in **Figure 1(b)**, where the encoder and decoder of the DRSN-AE system are jointly trained and optimized by applying the unique residual systolic block structure in **Figure 2**, the improved RSBU and the convolutional layer and built.

In order for the reader to more clearly understand this article in the text, we describe it here, where in **Figure 1(b)**, the encoder (*i.e.* transmitter) first maps k information bits of information into a message $s \in \{1, \cdots, M\}$, where $M = 2^k$. From the modulation point of view, the sequence of symbols s in **Figure 1(b)** is mapped into n natural numbers, which are transformed into a new signal $x = f(s) \in R^n$ occupies n channel time slots. Our system can simultaneously batch process $k \times L$ bits of complete messages, each message carries k bits, L is the number of symbols (block length) and the sent message sequence s occupies n channel time slots, *i.e.* the message's information rate of this system is calculated as $R = k/n$ (bits/channel used). In this system, the input sequence s is converted into a unique thermal vector, which means that the proposed system aims at minimizing the BER.

From the point of view of channel coding, the transmitter in **Figure 1(b)** achieves linear coding through linear weighting operations performed by the convolutional kernel, while the addition of rectified linear units (Relu) gives the network the capability of nonlinear coding. From a deep learning perspective, the convolution operation is a filtering process, and the DNNs in our transmitter and receiver are numerous filters that can extract information features better.

The channel layer between the transmitter and receiver can be considered a conditional probability density function $P(Y \mid X)$. The channel we use in this paper is the AWGN channel: $y = x + z, z \sim N(0, \sigma^2)$, which is the signal to be transmitted by the transmitter will be interfered with by Gaussian white noise with a fixed variance $\sigma = (2RE_b / N_0)^{-1}$.

The decoder (*i.e.* receiver) in **Figure 1(b)** is based on the data features learned by the system to classify the signal Y in the case of 2^k. The decoder produces an estimate $\hat{s}$ of the transmitted message *s*. Our system's transmitter and receiver are built by interleaving the RSBU units and the convolutional layers. The dilated convolution technique is used between the different convolutional layers for fast training without losing information features again. The whole learning and transformation process is represented by $g(Y)$ in **Figure 1(b)**.

The network architecture of the encoder and decoder of the DRSN-AE system is shown in **Table 1**, where the normalization layer can also be thought of as a batch norm without affine projection, which is essential for the training of encoder. The final layer of the encoder pairs x to further constrain the encoded symbols. The following are possible such constraints and are implemented using normalization layers:

$$\text{Energy constraints: } \|x\|_2^2 \le n \tag{3}$$

$$\text{Average power constraint: } \mathrm{E}\left[|x_i|^2\right] \le 1, \forall i \tag{4}$$

2.2. System Parameters

To conform to the sending of anonymous data in daily life and to facilitate comparison with the next pair [19]. We also use randomly generated binary sequences, and the training and validation sets conform to a 0 - 1 uniform distribution. The data is based on binary data generated by [19] to mimic the real-life data flow [19]. The autoencoder system of **Figure 1(b)** is trained using 12,800 data messages, where each message contains a block length of L symbols, each with k bits of information, the network is tested using 64,000 data messages, and the batch size is set to 64 [19]. The autoencoder is an unsupervised learning system because the data used has no external markers. Typically, the autoencoder aims to find a low-dimensional representation of the input at some intermediate layer, thus allowing reconstruction with minimal error at the output. This concept allows the autoencoder to learn without any prior knowledge.

Moreover, The DRSN network of the system decoder can learn how to discriminate the corresponding received signals, while we define the loss function between the sent and received symbol sequences in **Figure 1** as the binary cross entropy (BCE), converting both into a one-hot vector. Using Adam optimization to train the end-to-end system, we use a dynamic learning rate, with the initial learning rate set to 0.001 and the learning rate decaying by a factor of 10 when saturation occurs, resulting in fast and accurate training.

In our work, the network architecture shown in Table 1 is sufficient to achieve optimal BER performance without any loss of learning capability. In addition, the encoders and decoders in the system are batch processing the symbols instead of processing the symbol sequences with data inefficiently as in conventional communication systems. The hyperparameter information of our proposed DRSN-AE is shown in Table 2.

Table 1. Structure of the DRSN-AE autoencoder communication system (before activation, the Conv1D layer is followed by a batch return layer).

Type of layer	Activation	Output dimensions
One hot input	None	$L \times 2^k$
Conv1D	Relu	$L \times 32$
RSBU	Relu	$L \times 32$
Conv1D	Relu	$L \times 64$
RSBU	Relu	$L \times 64$
Conv1D	Relu	$L \times 256$
RSBU	Relu	$L \times 256$
Power Norm Layer	None	$L \times 256$
AWGN channel	None	$L \times 256$
Conv1D	Relu	$L \times 128$
RSBU	Relu	$L \times 128$
Conv1D	Relu	$L \times 64$
RSBU	Relu	$L \times 64$
Conv1D	Relu	$L \times 32$
RSBU	Relu	$L \times 32$
Conv1D	Relu	$L \times 2^k$
RSBU	Relu	$L \times 2^k$
Conv1D	Softmax	$L \times 2^k$

Table 2. Hyper-parameters of DRSN-AE.

Parameter Name	Setting
Loss	Binary Cross-Entropy (BCE)
Batch Size	100
Optimizer	Adam with initial learning rate 0.001
Block Length	10
Number of Epochs	200

Generally, the BER gain of an auto-encoder system trained at a high SNR is better than that of an auto-encoder system trained at a low SNR. This idea is also discussed in [19] to verify that, for example, the performance of CNN-AE trained at 27 dB achieves better gain results. The DRSN network we use has good generalization capability, allowing our system to perform well at different E_b/N_0 and short and medium block lengths. In this study, we use an E_b/N_0 of 9 dB to train the encoder and decoder to verify that our designed auto-encoder system can adapt to low SNR.

3. Simulation Results and Performance Evaluation

In this section, we not only compare the BER of conventional communication at different code rates but also verify the CNN-AE auto-encoder communication system described in [19]. Then we compare the CNN-AE auto-encoder, and the auto-encoder communication system with a residual network (Resnet-AE) proposed in this paper. Unless otherwise stated, our parameters are set at the same code rate for a fair comparison, with block length L being to set 10 and the E_b/N_0 used for training sequence being set to 9 dB.

We provide here extensive simulation results to demonstrate the generalization capability of the proposed DRSN-based autoencoder (DRSN-AE) system in terms of block length, training convergence, and code rate R, when communicating over AWGN channels. In addition, the simulation platform we use is pycharm, the graphics device used is the Nvidia 3050, our source code is implemented in Keras.

3.1. Code Rates Gain of DRSN-AE

This subsection extends the proposed DRSN-AE system to AWGN channels at various code rates. It is also compared with classical modulation schemes such as BPSK and QAM as benchmarks. To facilitate the comparison of the experiments, we set n to 1 and changed the value of k to change the system's R. It is worth noting that the R here is the message's information rate, not the standard code rate.

Figure 3 plots the BER performance of DRSN-AE with different code rates R = 1, 2, 4 (bits/channel use) when transmitting over AWGN channels. In this, we compare the DRSN-AE system with the conventional uncoded modulation. In conventional communication, the transmitter in **Figure 1(b)** is replaced with conventional modulation, and the receiver is replaced with the corresponding demodulation and decoding. We compare here only the modulation model when uncoded, and as expected, it can be seen that our scheme significantly outperforms the conventional BPSK, 16QAM, and 64QAM schemes. The DRSN-AE system provides a gain improvement of at least 2 dB at BER of even a code rate of 1. The signal feature representation capability of the DRSN network facilitated by the deep network architecture can learn the appropriate symbol conversion even at high data rates. Also, we note that at low E_b/N_0, our BER gain

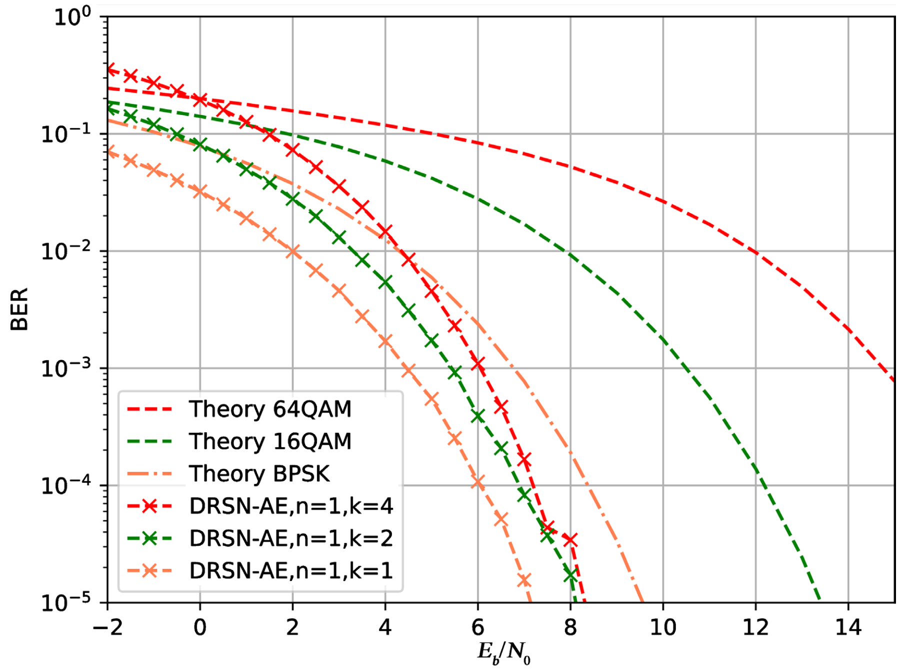

Figure 3. *BER* performance of the DRSN-AE system at different rates R = 1, 2, 4 (bits/channel use) compared to the corresponding BPSK, 16QAM, 64QAM modulations in AWGN channels.

is less significant than the high BER gain, mainly because the neural network-based auto-encoder communication system can extract the maximum number of features to optimize the training process when there is enough data. Also, the performance gain of the DRSN-AE communication system concerning conventional communication is more pronounced in the case of channel impairment. The reason is that the performance of conventional communication is degraded because the added channel impairment distorts the waveform, while our system is trained to learn and compensate for these effects. The performance gain of the DRSN-AE communication system improves with increasing k because the DRSN-AE communication system jointly optimizes the channel usage and can more flexibly adapt to various signal-to-noise ratio situations.

Figure 4 shows the performance of the DRSN-AE system at different rates compared with the corresponding CNN-AE in the AWGN channel. For example, DRSN-AE helps improve the BER performance (n = 1, k = 4) at 10^{-3} from approximately 11 dB to 6 dB providing a 5 dB improvement. Figure 5 depicts the BER performance of the DRSN-AE system at different rates compared to the corresponding Resnet-AE at AWGN channels. We observe that the BER only slightly improves for any SNR. For example, DRSN-AE helps improve the BER performance (n = 1, k = 4) from approximately 7 dB to 6 dB, only providing a 1 dB improvement. We can see that the BER performance of our system is better than the CNN-AE system and the Resnet-AE system. The DRSN-AE system in this paper outperforms the comparison model in terms of gain for BER in different signal-to-noise ratio ranges.

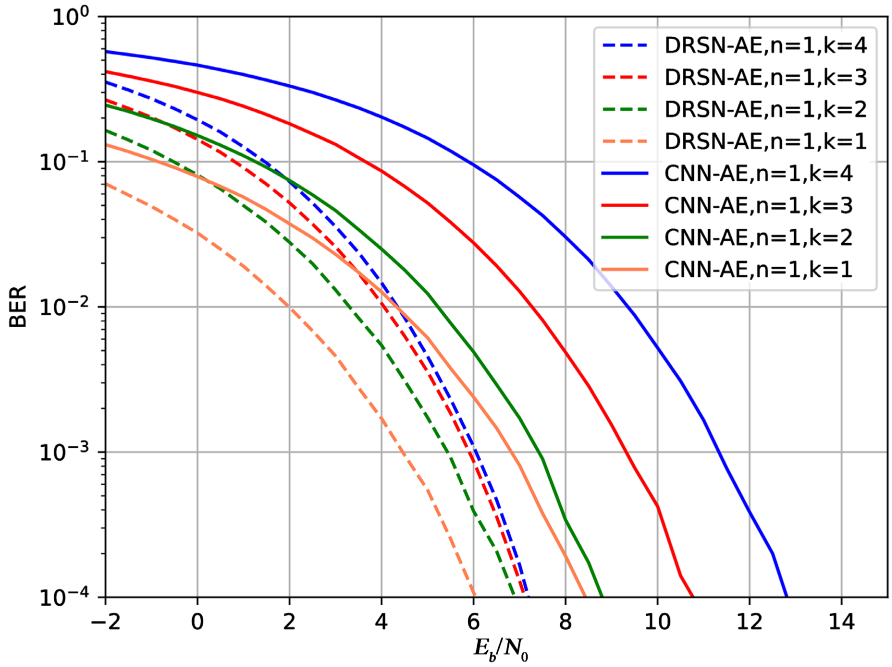

Figure 4. *BER* performance of the DRSN-AE system at different rates R = 1, 2, 3, 4 (bits/channel use) compared with the corresponding CNN-AE in AWGN channel.

We now explain the performance of the above different signal-to-noise ratios in CNN-AE and Resnet-AE. From [19], it is known that the encoder and decoder of the CNN-AE auto-encoder communication system have only three short convolutional layers, and although this design scheme can avoid problems such as gradient disappearance and gradient explosion, the encoder and decoder designed by the shallow network architecture are localized and limited for the acquisition of information features. While Resnet-AE, with its unique shortcut structure, is equivalent to taking the previously processed information directly to the present together again, which has the effect of impairment, there is still the problem of reduced learning ability because the Resnet network used by the encoder and decoder uses the convolutional kernel as the local feature extractor, which will ignore some feature information disturbed by noise. In turn, the higher-level features learned in the output layer have poor discriminative power and are insufficient for accurate signal classification. In deep learning, optimizing parameters is often time-consuming and essential, but the gradient of error loss must be passed back to the previous network layers, making the initial layer of trainable parameters less optimal.

Although our DRSN-AE auto-encoder system is designed to operate under low SNR, it can be found in **Figures 3-5** that our system can operate at different code rates and has high BER gain not only at low SNR but also at high SNR to achieve efficient coding (decoding) performance.

3.2. Block Length Gain of DRSN-AE

According to Shannon's finite-length block code information theory, we know that to achieve the minor possible error, the channel coding needs to use the longest possible (even infinite) block code sequence for coding. As the block length increases, the channel coding can achieve better reliability. We compare DRSN-AE with CNN-AE and Resnet-AE, tested at L = 100, as shown in Figure 6. We compare the *BER* test experiments at block length (L = 100) with different

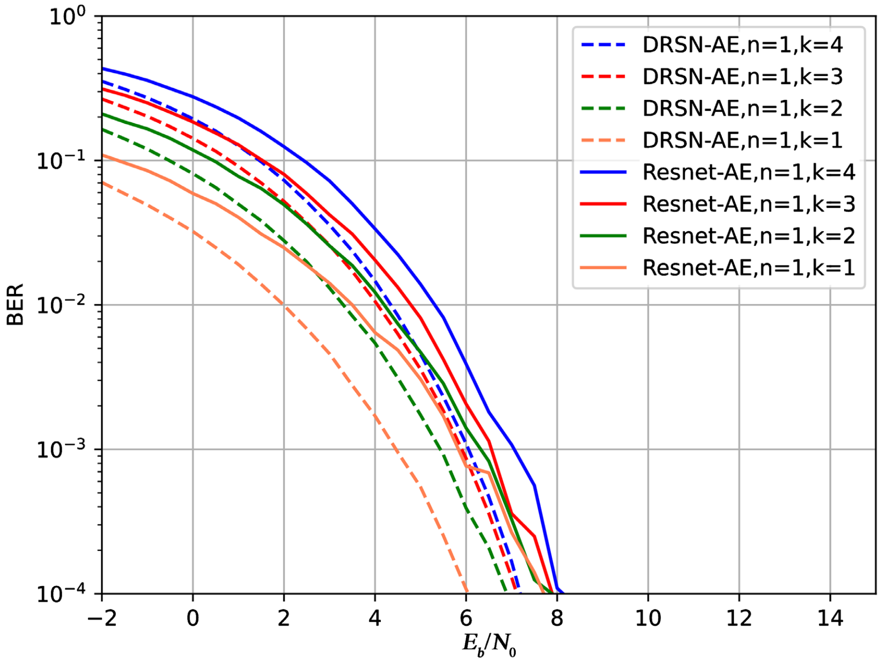

Figure 5. *BER* performance of the DRSN-AE system at different rates R = 1, 2, 3, 4 (bits/channel use) compared to the corresponding Resnet-AE at AWGN channels

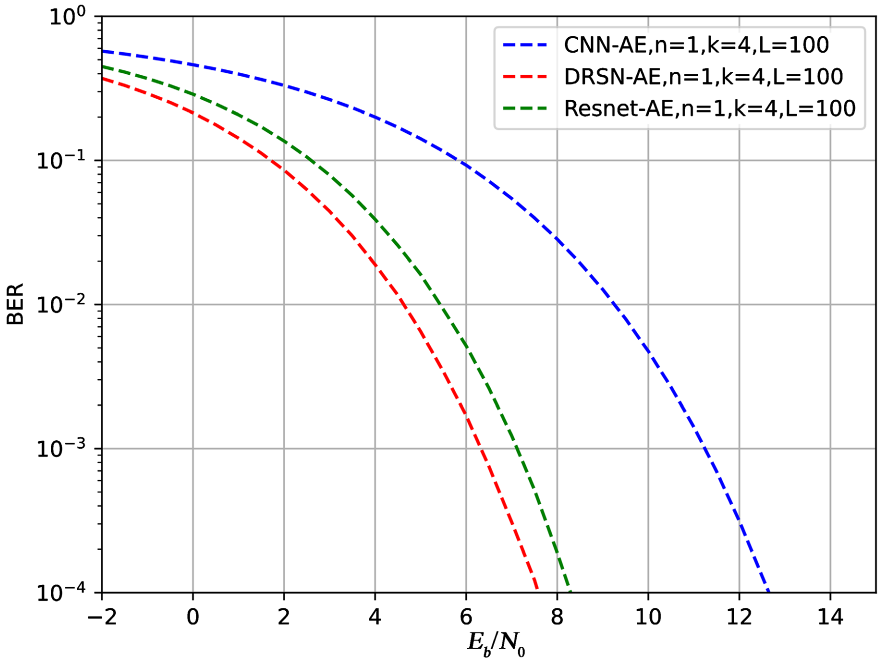

Figure 6. *BER* performance of the DRSN-AE under AWGN channel with block length L = 100 compared to different models.

signal-to-noise ratios.

In Section 4.1, we use the DRSN-AE system in **Figure 1(b)** train with a shorter block length ($L = 10$). Here we tested the *BER* on different block lengths ($L = 100$) using the same network parameters and verified that the DRSN-AE system still has good conception capability at $L = 10$. It can be seen from **Figure 6** that at $n = 1$, $k = 4$, our DRSN-AE autoencoder communication system still outperforms other autoencoder communication systems for block length $L = 100$. As the block length increases, large block lengths ($L = 1000$) require a large amount of training memory, and our system does not achieve better performance on large block lengths due to its own hardware memory. Also, we can expect the block length gain of DRSN-AE to be saturated when the block length is large, because long block lengths require the use of large memory stores and more complex structures to train. Furthermore, it is still a worthwhile direction for us to focus on how to train learning for larger block lengths.

3.3. Model Training Convergence

The DRSN-AE system proposed in this paper is in the AWGN channel, and the system's advantages with other methods at different code rates have been demonstrated in Section 4.1. In this section, we analyze the model adaptation and training convergence of the DRSN-AE system. **Figure 7** shows the verification loss of the DRSN-AE system and the CNN-AE system under the AWGN channel with $k = 4$, which is trained using 200 epochs for comparison.

In [19] the CNN-AE system encoder and decoder are built with three convolutional layers, which have a simple structure and low network complexity so

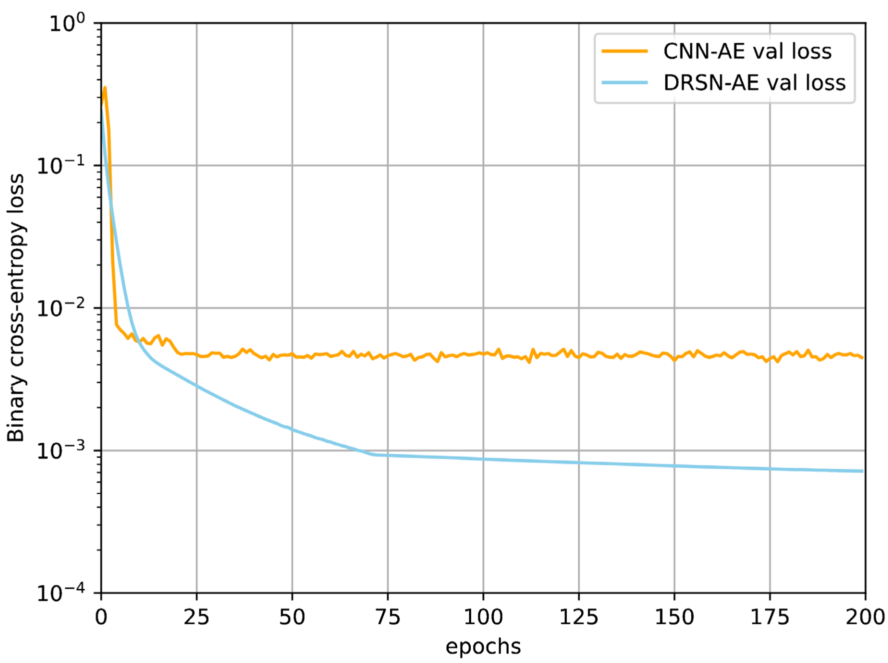

Figure 7. Verification loss of CNN-AE system and DRSN-AE system at epochs = 200 for AWGN channel with n = 1, k = 4.

that it can achieve faster convergence, but the CNN-AE system is not complete for signal feature extraction and will miss some feature information. Meanwhile, the DRSN-AE system use dilated convolution technology. We set a parameter to adjust the dilation rate, *i.e.*, to fill the convolution kernel with zeros of dilation rate −1, so we pick different dilation rates that will have different perceptual fields and thus obtain multi-scale information. Moreover DRSN-AE system can also achieve fast convergence within epochs = 100, which is not as fast as the CNN-AE system, but the DRSN-AE system converges more smoothly.

Compared with learning the original signal, the DRSN-AE auto-encoder system learns the difference of the signal, which simplifies the training process because of its unique residual unit. Also, in our work, the convergence of our model with different initialization parameters can reach as good convergence as **Figure 7**, which is not shown here.

From the comparison of the simulation results above, we can obtain that our model outperforms the conventional uncoded modulation and has a larger gain in intersection with CNN-AE [19].

4. Conclusions and Prospects

This paper presents a deep residual shrinkage network-based autoencoder communication system that can better extract signal features by using improved residual units through a data-driven approach without needing specialist knowledge in communication. Through the proper design of the residual unit and convolutional layers, different dilated convolution techniques are used between the different convolutional layers. This can reduce the computational cost of the network significantly without losing feature information and preventing the occurrence of overfitting. In comparison with other models, our proposed autoencoder communication system can achieve good performance under AWGN channel.

In future research, we apply this model to unknown channels by solving the gradient problem in channel backpropagation using methods such as the diffusion model or Generative Adversarial Networks (GAN); Also, we will design the autoencoder communication system from the perspective of Low-Density Parity-Check (LDPC) codes to solve the more significant block length coding gain problem.

Conflicts of Interest

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

References

[1] Annaby, M.H., Al-Abdi, I.A., Abou-Dina, M.S. and Ghaleb, A.F. (2022) Regularized Sampling Reconstruction of Signals in the Linear Canonical Transform Domain. *Signal Processing*, **198**, Article ID: 108569. https://doi.org/10.1016/j.sigpro.2022.108569

[2] Anbiyaei, M.R., Liu, W. and McLernon, D.C. (2018) White Noise Reduction for Wideband Linear Array Signal Processing. *IET Signal Processing*, **12**, 335-345. https://doi.org/10.1049/iet-spr.2016.0730

[3] Skovranek, T. and Despotovic, V. (2019) Audio Signal Processing Using Fractional Linear Prediction. *Mathematics*, **7**, Article No. 580. https://doi.org/10.3390/math7070580

[4] Wang, M., Li, Y., Liu, R., Wu, H., Hu, Y. and Lau, F.C.M. (2022) Decoding Quadratic Residue Codes Using Deep Neural Networks. *Electronics* (*Switzerland*), **11**, Article No. 2717. https://doi.org/10.3390/electronics11172717

[5] Nachmani, E., Bachar, Y., Marciano, E., Burshtein, D. and Be'ery, Y. (2018) Near Maximum Likelihood Decoding with Deep Learning. http://arxiv.org/abs/1801.02726

[6] Nachmani, E., Beery, Y. and Burshtein, D. (2016) Learning to Decode Linear Codes Using Deep Learning. 2016 54*th Annual Allerton Conference on Communication*, *Control*, *and Computing* (*Allerton*), Monticello, 27-30 September 2016, 341-346. https://doi.org/10.1109/ALLERTON.2016.7852251

[7] Abdallah, A., Celik, A., Mansour, M.M. and Eltawil, A.M. (2022) RIS-Aided mmWave MIMO Channel Estimation Using Deep Learning and Compressive Sensing. *IEEE Transactions on Wireless Communications*, **22**, 3503-3521.

[8] Abdallah, A., Celik, A., Mansour, M.M. and Eltawil, A.M. (2021) Deep Learning-Based Frequency-Selective Channel Estimation for Hybrid mmWave MIMO Systems. *IEEE Transactions on Wireless Communications*, **21**, 3804-3821. http://arxiv.org/abs/2102.10847

[9] Kang, J.M., Chun, C.J. and Kim, I.M. (2020) Deep Learning Based Channel Estimation for MIMO Systems with Received SNR Feedback. *IEEE Access*, **8**, 121162-121181. https://doi.org/10.1109/ACCESS.2020.3006518

[10] Daldal, N., Sengur, A.K., Polat, I.M. and Cömert, Z. (2020) A Novel Demodulation System for Base Band Digital Modulation Signals Based on the Deep Long Short-Term Memory Model. *Applied Acoustics*, **166**, Article ID: 107346. https://doi.org/10.1016/j.apacoust.2020.107346

[11] Wang, H., *et al.* (2019) Deep Learning for Signal Demodulation in Physical Layer Wireless Communications: Prototype Platform, Open Dataset, and Analytics. *IEEE Access*, **7**, 30792-30801. https://doi.org/10.1109/ACCESS.2019.2903130

[12] Institute of Electrical and Electronics Engineers (2020) 2020 IEEE 7th International Conference on Energy Smart Systems (2020 IEEE ESS): Conference Proceedings: May 12-14, 2020, Kyiv, Ukraine.

[13] IEEE (2020) 2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC).

[14] Ariyavisitakul, S.L. and Li, Y. (1998) Joint Coding and Decision Feedback Equalization for Broadband Wireless Channels. *IEEE Journal on Selected Areas in Communications*, **16**, 1670-1678. https://doi.org/10.1109/49.737636

[15] Institute of Electrical and Electronics Engineers and IEEE Signal Processing Society (2020) SPCOM: International Conference on Signal Processing and Communications: July 19-24, 2020.

[16] Xu, W., Alshamary, H.A.J., Al-Naffouri, T. and Zaib, A. (2020) Correction to "Optimal Joint Channel Estimation and Data Detection for Massive SIMO Wireless Systems: A Polynomial Complexity Solution". *IEEE Transactions on Information Theory*, **66**, 5316-5316. https://doi.org/10.1109/TIT.2019.2957084

[17] Song, H., You, X., Zhang, C. and Studer, C. (2021) Soft-Output Joint Channel Estimation and Data Detection Using Deep Unfolding. 2021 *IEEE Information Theory Workshop* (*ITW*), Kanazawa, 17-21 October 2021, 1-5. https://doi.org/10.1109/ITW48936.2021.9611404

[18] O'Shea, T.J. and Hoydis, J. (2017) An Introduction to Deep Learning for the Physical Layer. *IEEE Transactions on Cognitive Communications and Networking*, **3**, 563-575. http://arxiv.org/abs/1702.00832

[19] Wu, N., Wang, X., Lin, B. and Zhang, K. (2019) A CNN-Based End-to-End Learning Framework toward Intelligent Communication Systems. *IEEE Access*, **7**, 110197-110204. https://doi.org/10.1109/ACCESS.2019.2926843

[20] Jiang, Y., Kim, H., Asnani, H., Oh, S. and Viswanath, P. (2019) Turbo Autoencoder: Deep Learning Based Channel Codes for Point-to-Point Communication Channels.

[21] Ye, H., Liang, L., Li, G.Y. and Juang, B.H. (2020) Deep Learning-Based End-to-End Wireless Communication Systems with Conditional GANs as Unknown Channels. *IEEE Transactions on Wireless Communications*, **19**, 3133-3143. https://doi.org/10.1109/TWC.2020.2970707

[22] Wu, T., Peurifoy, J., Chuang, I.L. and Tegmark, M. (2018) Meta-Learning Autoencoders for Few-Shot Prediction. http://arxiv.org/abs/1807.09912

[23] Jiang, H., Bi, S., Dai, L., Wang, H. and Zhang, J. (2021) Residual-Aided End-to-End Learning of Communication System without Known Channel. http://arxiv.org/abs/2102.10786

[24] Zhao, M., Zhong, S., Fu, X., Tang, B. and Pecht, M. (2020) Deep Residual Shrinkage Networks for Fault Diagnosis. *IEEE Transactions on Industrial Informatics*, **16**, 4681-4690. https://doi.org/10.1109/TII.2019.2943898

[25] Jiang, W. and Liu, A. (2022) Image Motion Deblurring Based on Deep Residual Shrinkage and Generative Adversarial Networks. *Computational Intelligence and Neuroscience*, **2022**, Article ID: 5605846. https://doi.org/10.1155/2022/5605846

[26] Wang, H., Wang, J., Xu, H., Sun, Y. and Yu, Z. (2022) DRSNFuse: Deep Residual Shrinkage Network for Infrared and Visible Image Fusion. *Sensors*, **22**, Article No. 5149. https://doi.org/10.3390/s22145149

[27] Tang, P., Xu, Y., Wei, G., Yang, Y. and Yue, C. (2021) Wireless Security Challenges and Countermeasures for Dynamic Spectrum Sharing Specific Emitter Identification for IoT Devices Based on Deep Residual Shrinkage Networks.

[28] Hernandez-Matamoros, A., Fujita, H., Escamilla-Hernandez, E., Perez-Meana, H. and Nakano-Miyatake, M. (2020) Recognition of ECG Signals Using Wavelet Based on Atomic Functions. *Biocybernetics and Biomedical Engineering*, **40**, 803-814. https://doi.org/10.1016/j.bbe.2020.02.007

[29] Kanaan, T. (2021) Performance Analysis of Threshold Selection in Energy Detector Working over Noise Uncertainty Cahnnel for Cognitive Radio Networks. *IOSR Journal of Electrical and Electronics Engineering*, **16**, 7-21.

[30] Shannon, C.E. (1948) A Mathematical Theory of Communication. *The Bell System Technical Journal*, **27**, 623-656. https://doi.org/10.1002/j.1538-7305.1948.tb00917.x