electronics

MDPI

Article

Digital Self-Interference Cancellation for Full-Duplex Systems Based on CNN and GRU

Jun Liu [1] and Tian Ding [2,*]

[1] School of Computer Science and Technology, Shandong University of Technology, Zibo 255049, China; 22505020007@stumail.sdut.edu.cn
[2] School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo 255049, China
* Correspondence: tian.ding@sdut.edu.cn

Abstract: Self-interference (SI) represents a bottleneck in the performance of full-duplex (FD) communication systems, necessitating robust offsetting techniques to unlock the potential of FD systems. Currently, deep learning has been leveraged within the communication domain to address specific challenges and enhance efficiency. Inspired by this, this paper reviews the self-interference cancellation (SIC) process in the digital domain focusing on SIC capability. The paper introduces a model architecture that integrates CNN and gated recurrent unit (GRU), while also incorporating residual networks and self-attention mechanisms to enhance the identification and elimination of SI. This model is named CGRSA-Net. Firstly, CNN is employed to capture local signal features in the time–frequency domain. Subsequently, a ResNet module is introduced to mitigate the gradient vanishing problem. Concurrently, GRU is utilized to dynamically capture and retain both long- and short-term dependencies during the communication process. Lastly, by integrating the self-attention mechanism, attention weights are flexibly assigned when processing sequence data, thereby focusing on the most important parts of the input sequence. Experimental results demonstrate that the proposed CGRSA-Net model achieves a minimum of 28% improvement in nonlinear SIC capability compared to polynomial and existing neural network-based eliminator. Additionally, through ablation experiments, we demonstrate that the various modules utilized in this paper effectively learn signal features and further enhance SIC performance.

Keywords: self-interference; deep learning; convolutional neural network; GRU; residual network; self-attention mechanism

check for updates

Citation: Liu, J.; Ding, T. Digital Self-Interference Cancellation for Full-Duplex Systems Based on CNN and GRU. *Electronics* **2024**, *13*, 3041. https://doi.org/10.3390/electronics13153041

Academic Editor: Chiman Kwan

Received: 10 July 2024
Revised: 25 July 2024
Accepted: 25 July 2024
Published: 1 August 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the continuous advancement of technologies such as the Internet of Things, cloud platforms, big data, and artificial intelligence, there is a growing demand for the high-speed transmission of massive data. This poses a significant challenge to the limited communication spectrum resources. Full-duplex (FD) communication technology allows terminals to simultaneously transmit and receive signals at the same frequency, theoretically allowing for exponential improvement in spectrum efficiency [1]. However, the existence of self-interference (SI) can notably diminish the sensitivity of the local receiver, and in extreme cases, it can saturate the receiver, thereby preventing correct demodulation of the desired signal [2]. This issue poses a serious challenge to communication systems. Therefore, efficiently eliminating the effect of SI is crucial for realizing FD technology. SI varies from the transmitted signal as a result of imperfect component characteristics and the multi-path propagation in the spatial channel. Therefore, to accurately demodulate the desired signal, the SI must be estimated with high precision and suppressed well below the noise floor [3]. Current self-interference cancellation (SIC) methods utilize analog domain cancellation methodologies through two distinct avenues: one, by harnessing the inherent physical separation between transmitting and receiving antennae passively; and two, by proactively introducing an offset signal into the reception path to counteract the

incoming wave [4]. Nonetheless, analog cancellation methodologies typically fail to fully eradicate the SI signal at the receiver. Consequently, further reduction in the remaining SI signal necessitates the application of digital domain cancellation techniques.

In recent years, as neural network technology has rapidly advanced, NN-based digital SIC techniques have gradually been replacing polynomial modeling methods. Ref. [5] proposes a two-part nonlinear SI canceller, in which one component is structured as a neural network (NN) to extract and handle nonlinear aspects, while the other component operates as a linear filter to capture and suppress linear components. This scheme achieves interference cancellation of more than 20 dB, significantly outperforming conventional polynomial and pure NN-based interference cancellation schemes. Ref. [6] proposes an RV-FFNN to model SI signals, where a small feed-forward neural network (FFNN) attains equivalent performance in nonlinear digital cancellation as polynomial-based nonlinear cancellers, yet accomplishes this with diminished computational complexity. Ref. [3] introduces a new deep learning-based digital SIC scheme that combines the strengths of sliding window strategies, gated recurrent unit (GRU) networks, and attentional mechanisms (AMs). This scheme's SIC performance is at least 32% better than existing polynomial and NN-based schemes. In [7], two new low-complexity NN structures are proposed to eliminate SI from nonlinearities and memory effects. The suggested LWGS and MWGS match the eradication capabilities of polynomial-based systems, achieving respective computational complexity reductions of 49.87% and 34.19%. Ref. [8] introduces a channel-robust, NN-supported adaptive SIC strategy, capable of diminishing self-interference power down to noise floor levels, even in dynamically changing SI channel conditions. Additionally, its computational complexity is significantly reduced compared to the polynomial-based adaptive enhancement of the SIC methodology.

A variety of deep learning approaches and affiliated technologies, notably Long and Short-Term Memory Neural Networks (LSTMs) [9], AM [10], and Residual Networks (ResNets) [11], among others, have seen broad adoption in the realms of natural language processing and traffic volume forecasting. Nowadays, a large number of LSTM variants are available, such as GRU, ConvLSTM, ConvGRU, and Quick Recurrent Neural Networks (QRNNs), which are widely used in wireless communication. Ref. [12] develops an accurate prediction method based on LSTM and Higher-Order Polynomial Linear Regression (HPLR) techniques to predict measurements for wireless train-to-ground communication networks. Additionally, the AM is incorporated to effectively discern and prioritize the key features' information. The efficacy of the proposed approach is validated by means of experimental evaluations. Ref. [13] proposes a GRU-assisted initial condition indexing CSK system, which utilizes the formidable learning and categorization prowess of deep learning to attain exceedingly dependable bit error rate (BER) metrics. Moreover, networks grounded in GRU exhibit diminished computational intensity alongside superior BER outcomes when contrasted with their LSTM counterparts. Ref. [14] introduces a novel methodology for aggregated decentralized downsampling based on Convolutional Neural Networks (CNNs) and ResNet. In this strategy, areas not initially partaken in the convolution process undergo re-convolution and are subsequently merged with depth data during forward propagation and within shortcut connections. This ensures that the feature maps gradually converge. The introduced residual network framework, tailored for classification assignments, achieves an overall accuracy enhancement averaging at 2.57%.

The GRU represents a class of neural networks designed for sequential data analysis, excelling in capturing intricate dependencies within data sequences. AMs empower neural networks to deliberately concentrate on specific segments of the input sequence, thereby augmenting the overall effectiveness of the network. ResNet overcomes the training challenges associated with increasing the depth of traditional DNN, particularly excelling at mitigating the problems associated with gradient explosion and gradient vanishing.

Considering this, the paper introduces a novel digital SIC approach leveraging deep learning technology to enhance the performance of SICs. The paper's novel contributions are comprehensively detailed as follows:

1. Combining CNN and GRU (conv_GRU) leverages the strengths of both architectures. CNN excels at capturing local features and spatial correlations, enabling it to extract detailed patterns and structures from sequential data. On the other hand, GRU is proficient at capturing long-range dependencies within sequences, allowing it to discern trends in data evolution over time or spatial locations. Compared to traditional LSTM, GRU features a more streamlined architecture with fewer parameters, thereby diminishing model complexity and computational costs while facilitating easier training. By combining both architectures, the temporal relationships between these features can be further analyzed. This integration builds upon local feature extraction capabilities, enabling a more comprehensive understanding of complex data patterns;
2. The introduction of residual networks involves the implementation of shortcut connections, allowing information to flow directly from shallower network layers to deeper ones. This approach reduces gradient attenuation during backpropagation in deeper networks, addressing issues such as gradient vanishing and explosion. In the presence of complex signals, including self-interference, this method ensures effective gradient propagation, thereby stabilizing model training and accelerating convergence.
3. Adding the self-attention mechanism enables the model to directly compare each element within a sequence and weight the information based on their correlations. Simultaneously, these weights dynamically adjust according to the importance of different input segments, automatically focusing on the most relevant parts while disregarding less crucial information. This capability enhances the model's capacity to comprehend and capture long-range dependencies, reducing information loss and alleviating the challenges associated with long-distance information transfer in traditional RNNs or CNNs.

The structure of this paper is shown in Figure 1. Section 2 details the full-duplex system's model. Section 3 delivers an exhaustive explanation of the proposed SIC scheme, along with the structures of the CNN module, the residual module, the GRU, and the self-attention mechanism. Section 4 presents the experimental simulation results, including comparisons and results from ablation experiments.

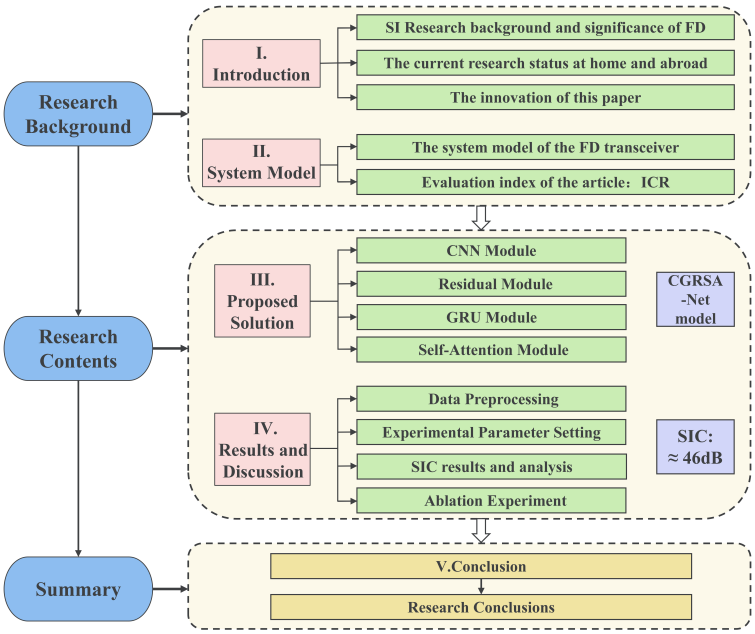


Figure 1. Overall flow chart of the article.

2. System Model

The FD transceiver system's architecture is shown in Figure 2.

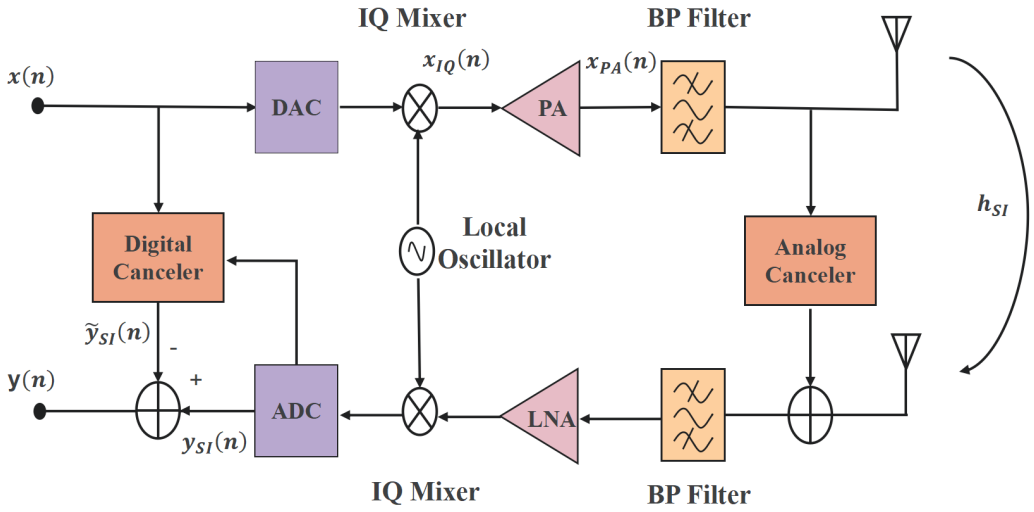


Figure 2. Full-duplex system model.

At moment n, the digital baseband transmission signal, represented by $x(n)$, undergoes conversion to an analog signal through a digital-to-analog converter (DAC). Following this, the analog signal is combined with the carrier wave at the IQ mixer and subsequently amplified by a power amplifier (PA). Ultimately, the amplified signal undergoes filtration through a band-pass filter (BPF) before being emitted by the transmitting antenna, where, after the introduction of IQ imbalance at the IQ mixer and assuming an ideal operation of the DAC, the digital baseband representation of the signal can be modeled as:

$$x_{IQ}(n) = K_1 x(n) + K_2 x^*(n), \tag{1}$$

where $K_1 = 1/2(1 + \psi e^{j\theta})$, $K_2 = 1/2(1 - \psi e^{j\theta})$.

The $x_{IQ}(n)$ is then amplified by the PA, and additional nonlinearities are introduced, which are characterized by utilizing the Parallel Hammerstein (PH) model [6]. Thus, the digital baseband representation of the signal following the PA output may be expressed as:

$$x_{PA}(n) = \sum_{p=1,odd}^{P} h_{m,p} x_{IQ}(n-m)|x_{IQ}(n-m)|^{p-1}. \tag{2}$$

Finally, the amplified signal traverses the SI channel, characterized by its impulse response $h_{SI}(l), l = 0, 1, \ldots (L-1)$, to reach the receiver. Supposing that both the ADC and any baseband amplifier operate ideally, the received downconverted, and digitized SI signal $y_{SI}(n)$ is given by:

$$y_{SI}(n) = \sum_{l=0}^{L-1} h_{SI}(l) x_{PA}(n-l). \tag{3}$$

By substituting (1) and (2) in (3), $y_{SI}(n)$ can be rewritten as:

$$y_{SI}(n) = \sum_{\substack{p=1 \\ p \text{ odd}}}^{P} \sum_{q=0}^{p} \sum_{m=0}^{M+L-1} h_{p,q}(m) x(n-m)^q x^*(n-m)^{p-q}. \tag{4}$$

The task of the nonlinear digital pair eliminator is to compute all the estimates of $h_{p,q}$, which we represent with $\hat{h}_{p,q}$, subsequently constructing the estimates of the SI signals, which we represent with $\hat{y}_{SI}(n)$. According to the evaluation metrics in [15], the capability of SIC is quantified by the Interference Cancellation Ratio (ICR), expressible in decibels (dB), as:

$$ICR_{dB} = 10log_{10}\left(\frac{\sum_{n=0}^{N-1}|y_{SI}(n)|^2}{\sum_{n=0}^{N-1}|y_{SI}(n) - \hat{y}_{SI}(n)|^2}\right), \tag{5}$$

where N denotes the length of the signal sequence. In real-world scenarios, $y_{SI}(n)$ comprises not only the SI but also the signal of interest. From (5), the assessment of SIC effectiveness is isolated from the characteristics of the signal of interest and relies solely on the precision of the SI signal reconstruction. Consequently, we disregard the signal of interest and treat the SI as the predominant received signal.

3. Proposed Solution

The $y_{SI}(n)$ signal of (4) can be decomposed as:

$$y_{SI}(n) = y_{lin}(n) + y_{nl}(n), \tag{6}$$

where $y_{lin}(n)$ represents the linear component of (4) (specifically, the summation term when $p = 1, q = 1$), while $y_{nl}(n)$ encompasses all the other nonlinear terms. We employ conventional linear pairwise elimination techniques to derive an estimation of $y_{lin}(n)$, labeled as $\hat{y}_{lin}(n)$, wherein the significantly less prominent $y_{nl}(n)$ component is treated as noise. Subsequently, we utilize a neural network to reconstruct $y_{nl}(n)$. More precisely, the process begins with the linear eliminator performing standard least squares channel estimation to compute $\hat{h}_{1,1}$, and then constructs $\hat{y}_{lin}(n)$ using $\hat{h}_{1,1}$ as shown below:

$$\hat{y}_{lin}(n) = \sum_{m=0}^{M+L-1} \hat{h}_{1,1}(m)x(n-m). \tag{7}$$

Then, the linear component of the signal is subtracted from the SI signal to yield:

$$y_{nl}(n) \approx y_{SI}(n) - \hat{y}_{lin}(n). \tag{8}$$

The objective of the neural network is to regenerate every instance of $y_{nl}(n)$, relying on the specific subset of x that directly influences that particular $y_{nl}(n)$ sample.

Based on this premise, this paper proposes a novel SIC approach leveraging deep learning technology, integrating CNN, GRU, ResNet, and a self-attention module. Initially, preprocessed data undergo convolution layers to extract feature structures. Subsequently, to enhance model accuracy and facilitate deeper data understanding, it passes through a residual module before inputting into the GRU module to establish relationships between transmitted signals and self-interference. Finally, a self-attention mechanism is introduced to assess varying levels of signal importance across different time intervals. Each module is detailed in subsequent subsections.

3.1. CNN Module

CNNs can progressively enhance the representation of inputs, advancing from shallow learning stages to more deep levels, thereby extracting increasingly precise features. In comparison to general DNN, the key benefit of CNNs is that they replace full connections with local connections, significantly reducing network complexity. This allows the network to become deeper and achieve a better field of view [16]. In our design, the input data are preprocessed and then fed into two layers of convolution modules (each consisting of a Convolution layer, a LayerNorm layer, and a ReLU). The structure of the CNN model used is shown in Figure 3.

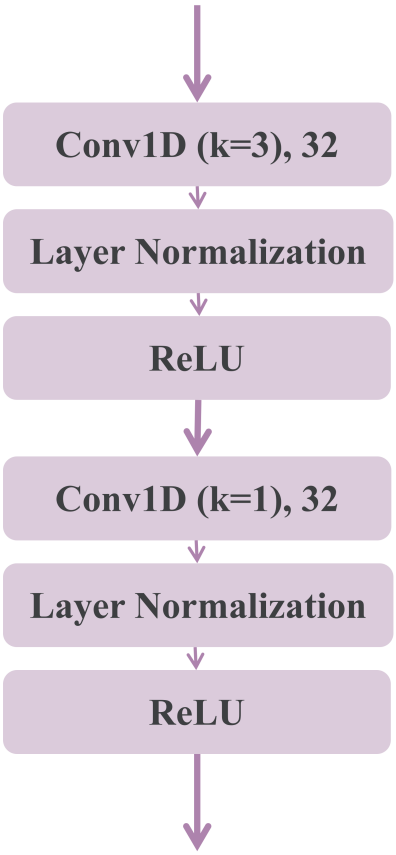


Figure 3. CNN model structure.

In this paper, we utilize a one-dimensional CNN module designed for feature extraction and representation learning on time-series input data. The specific process is below:

1. The preprocessed input data first pass through a one-dimensional convolution layer with convolution kernel length of 3, generating 32 different feature maps. The convolution layer, the core of the CNN, can locally sense the input data and extract local features by using a sliding window approach, making it highly effective for local pattern recognition in sequence data, which is crucial for our study. Additionally, the convolution layer is locally connected rather than fully connected, with the characteristic of parameter sharing. This ensures the sparsity of the network and helps prevent overfitting.
2. Furthermore, the data go into the normalization layer. We use Layer Normalization (LN) instead of Batch Normalization (BN). LN normalizes the input using the mean and variance of each sample independently within the feature dimension. This approach maintains the sequential information of the data along the temporal axis, ensuring that the temporal dependencies of the sequence are not disrupted by the normalization process. Additionally, compared to BN, LN has a shorter training time and is more suitable for the small batch data used in this paper, yielding better results.
3. Finally, the activation layer is used to introduce nonlinearity into the network to enhance its representation capability, with the activation function typically being a ReLU. Its implementation is very simple; the mathematical expression is ReLU(x) = max(0,x). In simple terms, the ReLU function is a blend of linear and nonlinear features. When the input value is negative, ReLU behaves as a nonlinear function and directly outputs 0, while when the input value is positive, ReLU behaves as a linear function. This form allows the ReLU function to mitigate the problem of gradient vanishing to some extent during the training process of deep learning [17]. At the same time, this property can lead to only a portion of the hidden layer neurons in the network

being activated, creating a sparse activation phenomenon, which can improve the expressiveness of the network and reduce the risk of overfitting, making the model more robust. Compared to the traditional sigmoid and tanh activation functions, the implementation of ReLU involves only threshold judgment and does not involve any exponential operations, which makes it computationally very efficient and the network converges faster.

4. After the data pass through the first convolution module, they enter the second convolution module whose convolution layer is a one-dimensional convolution layer with a convolution kernel of length 1. The rest of the parts remain the same.

3.2. Residual Module

The core idea of ResNet is to introduce the concept of "residual learning" to tackle the challenges of vanishing gradients and exploding gradients encountered during the training of DNNs. This allows the network to increase the number of layers to improve accuracy without encountering training difficulties [18]. The core principle of ResNet is to construct deep networks using residual modules. In traditional neural networks, each layer's output acts as the input to the subsequent layer. However, in residual networks, the input of each layer is passed not only to the next layer but also directly to deeper layers through shortcut connections. The structure of a residual block is shown in Figure 4.

A residual block can be represented as:

$$X_{i+1} = X_i + \mathcal{F}(X_i, W_i), \tag{9}$$

where X_i and $X(i+1)$ are the inputs and outputs of the module, respectively, $\mathcal{F}(X_i, W_i)$ denotes the operation performed by the stacking of the convolution layers, W_i is the weight of these layers, and X_i denotes the inputs that are passed directly through the shortcut connections.

The residual block is divided into two parts: the direct mapping part and the residual part. X_i represents the direct mapping, depicted as the lower curve in Figure 4, while $\mathcal{F}(X_i, W_i)$ represents the residual part, typically consisting of two or three convolution operations. This is illustrated by the upper part containing convolutions in Figure 4. In this paper, the residual network uses only one residual block, consisting of two one-dimensional convolution operations, where both one-dimensional convolution layers have a convolution kernel length of 3 and both produce 32 different feature mappings. And by introducing a shortcut connection, which skips several layers to directly transfer the input information to the output, the input is combined with the information transformed by multiple layers. This approach ensures that even if the gradient becomes very small after passing through multiple layers, it can be directly transferred via the shortcut. This ensures effective gradient propagation, effectively solving the problems of network degradation and gradient vanishing. Similarly, using LN instead of the BN layer can effectively enhance the performance of this experiment, addressing issues such as network degradation and gradient vanishing. Finally, after the ReLU activation function, the output results are sent to the next layer.

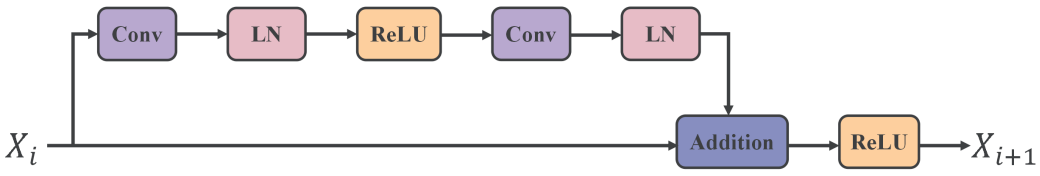


Figure 4. Residual block.

3.3. GRU Module

LSTM is a neural network designed for sequence modeling, characterized by an intricate architecture and a substantial quantity of parameters. As an enhanced version of

LSTM, GRU integrates forgetting gates and input gates into update gates. Consequently, GRU offers a simpler structure, and lower computational demands, while still matching the performance capabilities of LSTM [19].

The structure of GRU is illustrated in Figure 5, featuring two gating mechanisms: the update gate and the reset gate. The update gate controls the amount of relevant information to propagate downwards at the current step, while the reset gate controls how much of the past information should be discarded. The calculation of GRU is as follows:

$$\begin{aligned} z_t &= \sigma(W_z \times [h_{t-1}, x_t]), \\ r_t &= \sigma(W_r \times [h_{t-1}, x_t]), \\ \tilde{h}_t &= \tanh\left(W_{\tilde{h}} \times [r_t \times h_{t-1}, x_t]\right), \\ h_t &= (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t. \end{aligned} \tag{10}$$

The z_t and the r_t represent the update gate and the reset gate, respectively. At time t, x_t denotes the input data, while h_t represents the output data. The $\widetilde{h}_t$ is the candidate output data at the given time step. The W_z, W_r, and $W_{\tilde{h}}$ are the weight matrices of the update gate, the reset gate, and the candidate data, respectively. The $\times$ operator denotes element-by-element multiplication, and $+$ denotes element-by-element addition. $\sigma(\cdot)$ and $\tanh(\cdot)$ denote the activation functions for s-curves and hyperbolic tangent curves, respectively, with the following expressions:

$$\begin{aligned} \sigma(x) &= \frac{1}{1 + e^{-x}}, \\ \tanh(x) &= \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}. \end{aligned} \tag{11}$$

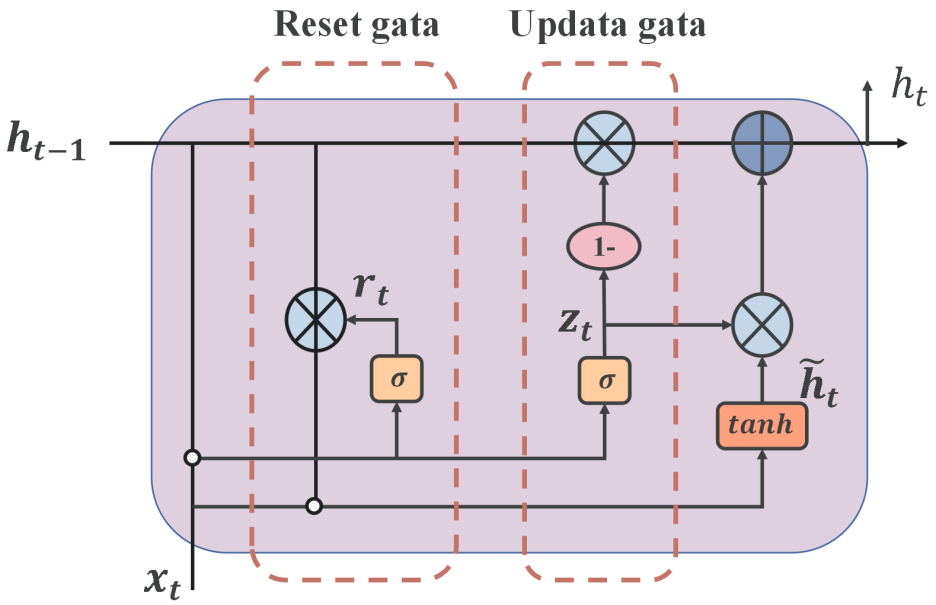


Figure 5. GRU structure.

In this paper, a two-layer GRU network is employed to effectively learn temporal features from the data, mitigating issues like gradient vanishing or explosion that can arise from excessively deep networks. For each layer of the GRU, the number of hidden neurons is 32, and a smaller number of neurons significantly reduces the number of parameters in the model, making it more concise and helping to mitigate the risk of overfitting. Additionally, integrating the CNN module allows for the aggregation and processing of feature information captured at different scales, enhancing the model's

capability to adapt to sequence data across various scales and enabling more accurate prediction and classification.

3.4. Self-Attention Module

The deep learning model receives the transmitted signal as its input and generates the SI signal as output. The output sequence exhibits memory effects due to power amplification (PA) and multi-path channels. This suggests that the present output in the sequence is influenced not just by the immediate input but also by a history of preceding input values spanning several time steps. Clearly, the information provided by the input sequences at different time steps varies in importance for predicting the output [20]. However, the standard GRU does not inherently discern which portions of the input sequence are crucial for predictions since it treats all signal sequences as being equally important at any given time. To address this issue, we introduce a self-attention module to automatically discover varying importance levels of signal sequences across different time steps. Unlike the traditional attention mechanism, the self-attention mechanism directly captures dependencies between any two positions in the input sequence and learns long-term dependencies. It dynamically assigns weights to each position based on the content of the input sequence, enabling flexible adaptation to changing self-interference patterns within the sequence [21]. Figure 6 depicts the self-AM structure, and the parameters are calculated as follows:

$$\begin{aligned} a^i &= Wx^i, \\ q^i &= W^q a^i, \\ k^i &= W^k a^i, \\ v^i &= W^v a^i, \\ a_{1,i} &= q^1 \cdot k^i, \end{aligned} \tag{12}$$

where W denotes the parameter matrix of embedding; $a_{1,i}$ is obtained after the dot product operation of q^1 and k^i; then, we apply the softmax operation to obtain $\hat{a}_{1,i}$:

$$\hat{a}_{1,i} = exp(a_{1,i}) / \sum_j exp(a_{1,j}). \tag{13}$$

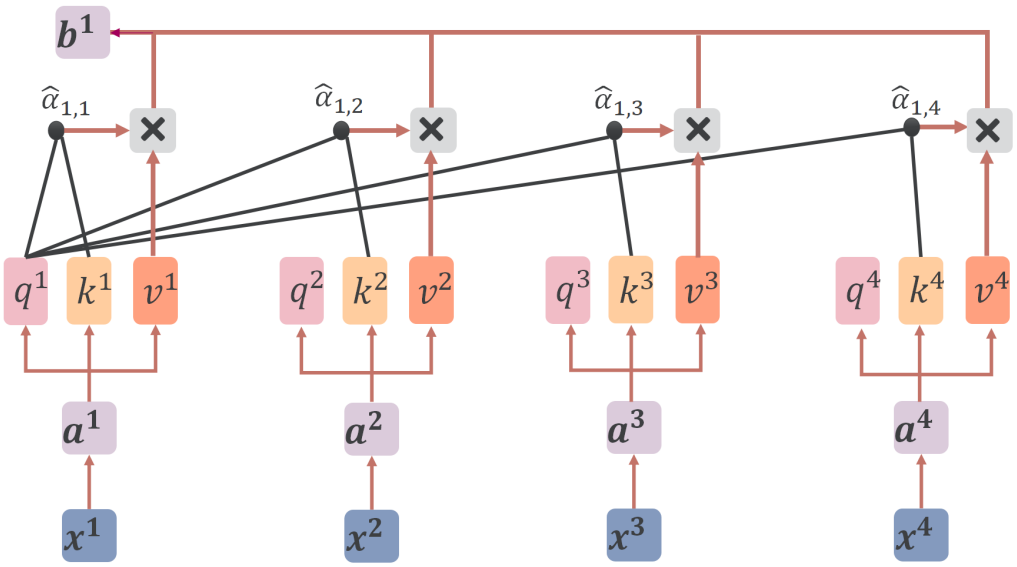


Figure 6. Self-attention structure.

Finally, we extract information based on attention scores to identify and select the most critical part of the information to obtain b^i as:

$$b^i = \sum_j \hat{a}_{i,j} v^j. \tag{14}$$

The self-attention mechanism processes sequence data by allowing each element to relate to every other element in the sequence, rather than solely depending on neighboring elements. It adaptively identifies and retains long-distance relationships among sequence elements based on their assessed significance. In particular, the self-attention mechanism, for every element in the sequence, calculates its similarity with every other element, converts these similarities into normalized attention scores, and then constructs an output through a process of weighting and aggregating each element based on these scores.

3.5. Proposed SIC Scheme

The total deep learning-based SIC model proposed in this paper is shown in Figure 7.

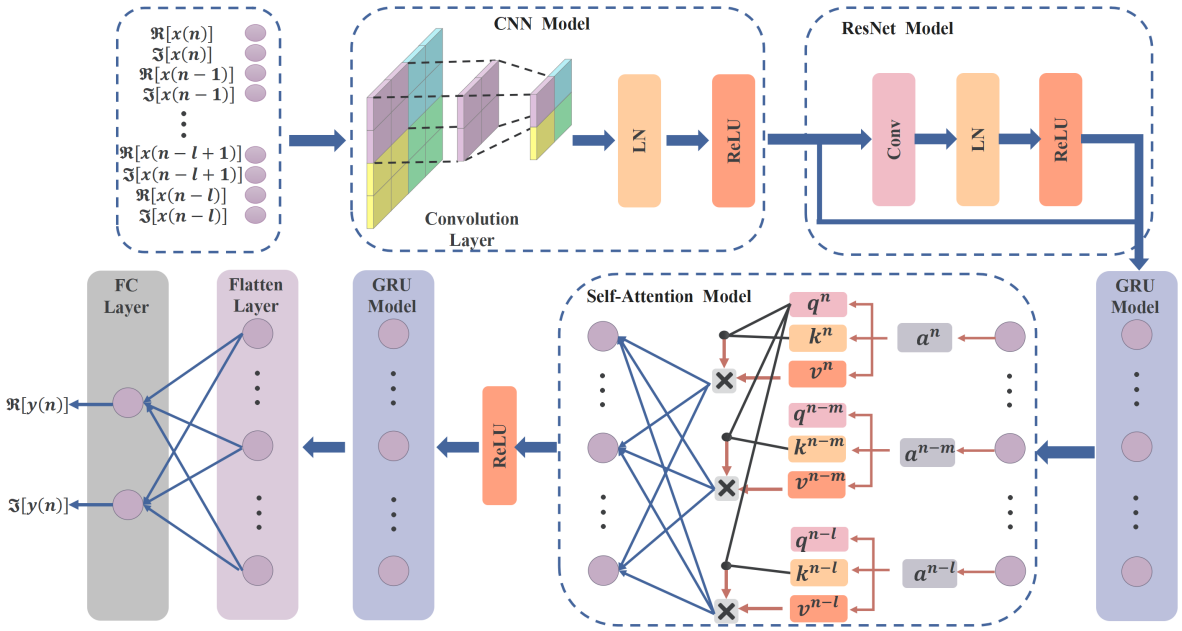


Figure 7. The proposed SIC model.

We start by doubling the length of the self-interference channel for the number of input channels and preprocessing the original sequence. Next, we feed it into the CNN module to facilitate a gradual transition from shallow to deep learning, extracting more precise local features. Following two one-dimensional convolution operations, the output enters a residual module to boost the network's expressive capabilities. Next, the processed information is channeled into both the GRU layer and a self-attention module for learning. To prepare for subsequent loss calculation operations, we include a compression layer and a fully connected layer at the network's end. These components facilitate target classification and prediction, with neurons possessing strong approximation capabilities. They produce two one-dimensional outputs (real and imaginary parts), culminating in the final prediction.

4. Results and Discussion

In this section, we begin by providing a concise overview of our experimental setup. We then proceed to compare the performance of our proposed scheme with traditional polynomial and neural network approaches. Additionally, to assess the effectiveness of each module, we conduct ablation experiments and present the simulation results after incorporating each module.

4.1. Data Preprocessing

In our work, the input signal sequences fed into deep learning networks are typically lengthy, imposing a significant computational burden and potentially diminishing network accuracy. To tackle these issues, prior to being introduced into the deep learning model, we subject both the input and output of the sample sequences to the preprocessing steps. Specifically, we segment the entire signal sequence into several fixed-length sub-sequences. Within this study, the length of the input sequence is set to $l = 16$, while the output sequence consistently has a length of 1. And the complex signal sequence is divided into real part ($\Re[x(n)]$) and imaginary part ($\Im[x(n)]$), which are processed separately. Following preprocessing, the input–output sequence within the deep learning framework undergoes a reduction in length and a dimensional transformation from 1 to 2. Simultaneously, the data are normalized to adhere to a standard normal distribution, characterized by a mean of 0 and variance of 1. This normalization process enhances the velocity of neural network training and its precision, while also mitigating issues related to gradient vanishing and explosion.

4.2. Experimental Parameter Setting

The experimental data for this project [6] were generated using a National Instruments FlexRIO device and two FlexRIO 5791R RF transceiver modules. The data involved OFDM signals using QPSK modulation with a passband width of 10 MHz and a carrier count of $N_c = 1024$. The signals were sampled at a frequency of 20 MHz. Every transmitted OFDM frame comprised approximately 20,000 baseband samples, encompassing the transmit signal, the corresponding self-interference signal, and the noise. For the training implementation of the neural network, 90% of these samples are used, while the remaining 10% is allocated for testing purposes.

The neural network is implemented using the PyTorch framework for Python 3.9.6. It is trained using the Adam optimization algorithm. The loss function is calculated according to the mean square error (*MSE*) as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}(n) - y(n))^2. \tag{15}$$

The learning rate is $\lambda = 0.001$, the batch size is $B = 32$, the training epoch is 20, the count of neurons within the neural network's hidden layer stands at 32, and the rest of the parameters are set as default values. In this work, we consider the polynomial eliminator as the baseline, for which the experimental parameters are set to be the same as the neural network parameters. The experimental parameters of this work are summarized in Table 1, and the parameter values of the polynomial model are shown in rows 9–11 of Table 1.

Table 1. Hyperparameters of the CGRSA-Net.

Parameters	CGRSA-Net
Optimizer	Adam
Loss	*MSE*
Self-interference channel length	16.000
trainingRatio	0.900
nEpochs	20.000
nHidden	32.000
learningRate	0.001
batchSize	32.000
pamaxordercanc	7.000
samplingFreqMHz	20.000
dataOffset	14.000

4.3. SIC Results and Analysis

The power spectral density (PSD) of the SI signal following the application of both linear and nonlinear phase cancellation techniques is depicted in Figure 8.

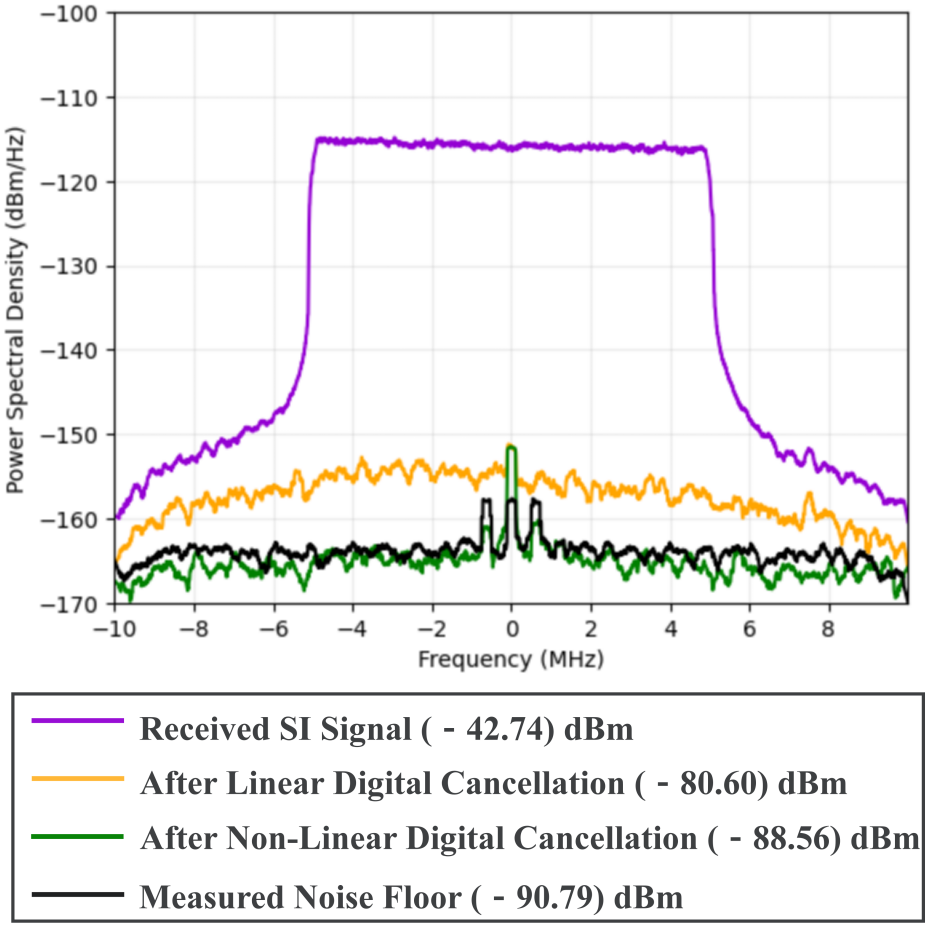


Figure 8. PSD of the SI after applying cancellation schemes.

It appears that the CGRSA-Net model architecture proposed in this paper achieves a 7.95 dB further reduction after the linear elimination, resulting in a total cancellation capability of 45.81 dB. Furthermore, as depicted in Figure 8, the spectrum of the residual SI signal closely resembles the receiver noise floor. This observation suggests that the structure proposed in this paper exhibits strong modeling capabilities in suppressing SI down to the level of receiver noise. Meanwhile, the trend of the loss values for the proposed model in this study at different iteration stages is depicted in Figure 9. The training and test loss curves clearly illustrate the model's learning process. Initially, both values decrease rapidly as training progresses, indicating the model's capability to swiftly assimilate pattern information from the training dataset and effectively minimize prediction errors. This underscores the effectiveness of the model architecture and its capacity to adapt quickly to the task.

In order to verify the advantages of the model proposed in this paper and to facilitate the comparison, the experimental parameters of the model architectures from other works in the literature are set to be the same as those in this paper, and the experiments include RVNN [22], LWGS, MWGS, DN-2HLNN (2-7), DN-3HLNN (2-4-5) [23] and feed-forward NN. All neural network-based eliminators are used to model the nonlinear part of the SI signal. The results are shown in Figure 10 and Table 2, where it can be seen that LWGS, MWGS, DN-2HLNN (2-7), and DN-3HLNN (2-4-5) achieve similar elimination performance to polynomial-based models with lower complexity. In comparison, Figure 10 and Table 2 clearly show that the model proposed in this paper can provide the highest

SI cancellation at the cost of higher complexity. The ICRs of each scheme are calculated from Equation (5) as 44.09 dB, 44.55 dB, 44.39 dB, 44.58 dB, 44.44 dB, 44.47 dB, and 45.81 dB, respectively, as shown in the fourth column of Table 2. Among them, the linear part of each scheme is eliminated by the least squares method, whose ICRs are all 37.86 dB. The nonlinear part is obtained by $NC(dB) = TC - LC$ (where, NC denotes Nonlinear Canc; TC denotes Total Canc; LC denotes Linear Canc), and the results are shown in the third column of Table 2. In order to clearly demonstrate the advantages of the models in this paper, we will express them in terms of percentage performance improvement compared to other models:

$$P = \frac{NC_{(CGRSA-Net)} - NC_{(Remaining\ model)}}{NC_{(Remaining\ model)}} \times 100\%. \tag{16}$$

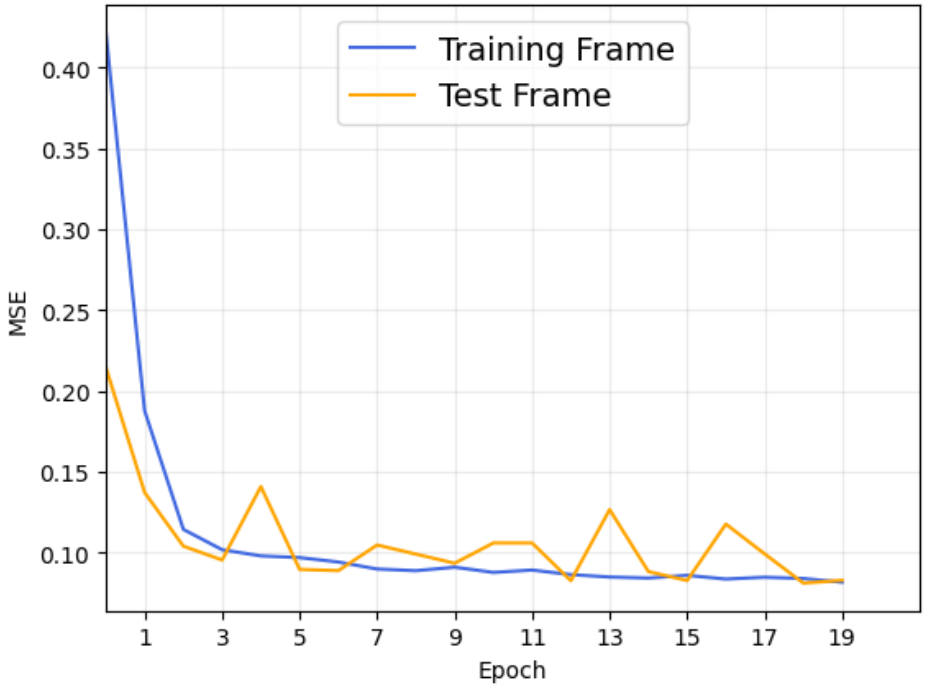


Figure 9. Loss value curves.

Table 2. Cancellation results for various NN structures.

Network	Linear Canc (LC)/dB	Nonlinear Canc (NC)/dB	Total Canc (TC)/dB	P (%)
RVNN	37.86	6.23	44.09	27.61%
LWGS	37.86	6.69	44.55	18.83%
MWGS	37.86	6.53	44.39	21.75%
DN-2HLNN (2-7)	37.86	6.72	44.58	18.30%
DN-3HLNN (2-4-5)	37.86	6.80	44.66	16.92%
Feed-forward NN	37.86	6.61	44.47	20.27%
CGRSA-Net	37.86	7.95	45.81	——

As can be seen in Table 2, compared with the traditional NN model, the model in this paper is able to improve the self-interference cancellation capability of the nonlinear part by up to 27.61%.

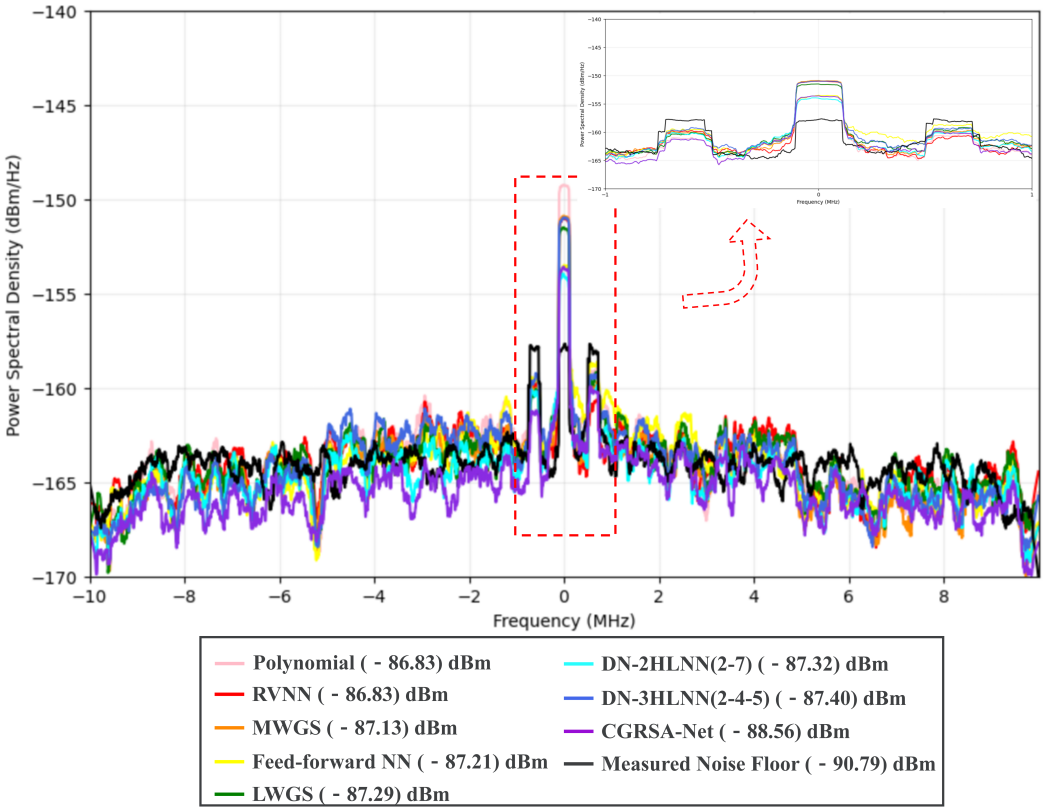


Figure 10. PSD of the SI after applying various cancellation schemes.

4.4. Ablation Experiment

To thoroughly investigate the contributions of individual components to overall model performance and their interactions, this paper initially constructs a comprehensive model incorporating all functional modules to simulate the nonlinear part of the SI signal. Specific components are then systematically removed, or key parameter settings are modified. After each adjustment, experiments are rerun, and detailed records are maintained of the resulting changes in performance metrics. Specifically, the first ablation experiment is to increase the training time to see the effect on the model effect. As shown in rows 1–2 of Table 3, when the training epoch is increased from 20 rounds to 30 rounds, the SIC is changed from 7.95 dB to 7.76 dB, which is a decrease of 2.38%, indicating that increasing the training epoch cannot improve the model performance. Therefore, with the proposed CGRSA model as the benchmark, all subsequent experiments are trained for 20 rounds.

The second experiment adopts GRU as the core component and conducts a comprehensive analysis by systematically eliminating other modules to deeply explore their independent contributions to the overall model effectiveness. First, the performance of the GRU is verified by training the network using only the GRU module within the entire model architecture; after 20 rounds of training, the SIC is 6.70 dB as shown in row 3 of Table 3. Next, based on the GRU, the CNN, ResNet, and self-attention modules are added separately to verify performance, resulting in SIC of 6.92 dB, 6.84 dB, and 6.97 dB, respectively, as shown in rows 4–6 of Table 3. Finally, within the entire CGRSA-Net model, the CNN, ResNet, and self-attention modules are removed separately, resulting in SIC of 7.09 dB, 7.48 dB, and 7.06 dB, respectively, as shown in rows 7–9 of Table 3. As can be observed from the table, the self-attention module exhibits the most prominent effect in enhancing the overall performance of the model.

Table 3. Results of ablation experiments.

	Non-SIC /dB	PSD /dBm	Loss
CGRSA-Net (epoch = 20)	7.95	−88.56	0.080650
CGRSA-Net (epoch = 30)	7.76	−88.37	0.077114
GRU	6.70	−87.31	0.091677
GRU-CNN	6.92	−87.53	0.090060
GRU-ResNet	6.84	−87.46	0.091901
GRU-SA	6.97	−87.59	0.085331
GRU-Res-SA	7.09	−87.70	0.088003
GRU-CNN-SA	7.48	−88.09	0.087609
GRU-CNN-Res	7.06	−87.67	0.088769
CGRSA-Net (BN)	7.53	−88.14	0.079407
CGRSA-Net (GN)	7.70	−88.31	0.083285
CGRSA-Net (Attention)	7.61	−88.22	0.083818
CGRSA-Net (multi-head attention)	7.68	−88.29	0.089990

The third experiment evaluates the impact of different components or modules on the model by replacing them. Firstly, the normalization layers in the CNN and ResNet modules are replaced to assess their effects. This paper examines the impact of the Layer Normalization (LN), Batch Normalization (BN) and Group Normalization (GN) layers on the entire model, resulting in SIC of 7.95 dB, 7.53 dB, and 7.70 dB, respectively, as shown in rows 1, 10, and 11 of Table 3. It is evident that the LN used in this paper offers the highest SIC. Next, the model's performance is verified using different attention mechanisms. In addition to the self-attention mechanism used in this paper, experiments were also conducted with the attention mechanism and multi-head attention [24]. As shown by the comparison of rows 1 and 12–13 of Table 3, after incorporating self-attention, attention, and multi-head attention, the SIC is 7.95 dB, 7.61 dB, and 7.68 dB, respectively. Therefore, it can be seen that self-attention provides the highest SIC and is the most suitable technique for this model.

5. Conclusions

In this paper, deep learning techniques are employed for designing a full duplex self-interference cancellation system. To further enhance self-interference cancellation capabilities, approach the noise floor, and improve communication efficiency, a deep learning-based model architecture is proposed. A combination of CNN, GRU, ResNet, and the self-attention model is adopted to replace the traditional polynomial model and the basic feed-forward neural network so as to realize the efficient nonlinear reconstruction of self-interference signals in the digital domain. Extensive experimental tests are conducted to verify the performance of this model. We first use the CNN model to capture local features and spatial correlations to extract local patterns and structures in the sequence data. Then, through the residual network, shortcut connections are introduced so that information can be directly passed from the shallower to the deeper layers of the network, addressing the challenges of vanishing gradients and exploding gradients. Secondly, the SI is reconstructed by utilizing GRUs that are well suited for forecasting sequences with long-term dependencies. What is more, in order to improve the accuracy of the SI prediction, we introduce self-attention to weight the information of the input sequences, which dynamically adjusts the weights according to the importance of different parts of the

inputs and automatically focuses on the most relevant parts of the sequences. Experiments show that the CGRSA-Net model can eliminate 7.95 dB in the nonlinear aspect of SI, and the total SI can eliminate 45.81 dB. At the same time, the power spectral density of the remaining SI signal can reach −88.56 dBm, which can suppress the SI to the noise floor level. Through comparative experiments, the nonlinear SIC capability of the CGRSA-Net model can be improved by up to 28% compared to polynomial eliminators and existing neural network-based eliminators. In addition, through systematic ablation experiments, we verify the key role of the model's constituent modules in self-interference signal suppression, which further confirms the rationality and effectiveness of the overall architecture design.

Author Contributions: Conceptualization, J.L.; methodology, J.L. and T.D.; software, J.L.; validation, J.L. and T.D.; formal analysis, J.L.; investigation, T.D.; writing—original draft preparation, J.L.; writing—review and editing, T.D.; visualization, J.L.; supervision, T.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shammaa, M.; Mashaly, M.; El-mahdy, A. A deep learning-based adaptive receiver for full-duplex systems. *AEU-Int. J. Electron. Commun.* **2023**, *170*, 154822. [CrossRef]
2. Li, H.; Van Kerrebrouck, J.; Caytan, O.; Rogier, H.; Bauwelinck, J.; Demeester, P.; Torfs, G. Self-Interference Cancellation Enabling High-Throughput Short-Reach Wireless Full-Duplex Communication. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 6475–6486. [CrossRef]
3. Hu, C.; Chen, Y.; Wang, Y.; Li, Y.; Wang, S.; Yu, J.; Lu, F.; Fan, Z.; Du, H.; Ma, C. Digital self-interference cancellation for full-duplex systems based on deep learning. *AEU-Int. J. Electron. Commun.* **2023**, *168*, 154707. [CrossRef]
4. Nguyen, H.V.; Nguyen, V.-D.; Dobre, O.A.; Sharma, S.K.; Chatzinotas, S.; Ottersten, B.; Shin, O.-S. On the spectral and energy efficiencies of full-duplex cell-free massive mimo. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 1698–1718. [CrossRef]
5. Wang, Z.; Ma, M.; Qin, F. Neural-network-based nonlinear self-interference cancelation scheme for mobile stations with dual-connectivity. *IEEE Access* **2021**, *9*, 53566–53575. [CrossRef]
6. Balatsoukas-Stimming, A. Non-linear digital self-interference cancellation for in-band full-duplex radios using neural networks. In Proceedings of the 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 25–28 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.
7. Elsayed, M.; El-Banna, A.A.A.; Dobre, O.A.; Shiu, W.; Wang, P. Low complexity neural network structures for self-interference cancellation in full-duplex radio. *IEEE Commun. Lett.* **2021**, *25*, 181–185. [CrossRef]
8. Kong, D.H.; Kil, Y.-S.; Kim, S.-H. Neural network aided digital self-interference cancellation for full-duplex communication over time-varying channels. *IEEE Trans. Veh. Technol.* **2022**, *71*, 6201–6213. [CrossRef]
9. Zheng, W.; Chen, G. An accurate gru-based power time-series prediction approach with selective state updating and stochastic optimization. *IEEE Trans. Cybern.* **2022**, *52*, 13902–13914. [CrossRef] [PubMed]
10. Liu, J.; Jin, B.; Wang, L.; Xu, L. Sea surface height prediction with deep learning based on attention mechanism. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5.
11. Ren, X.; Mosavat-Jahromi, H.; Cai, L.; Kidston, D. Spatio-temporal spectrum load prediction using convolutional neural network and resnet. *IEEE Trans. Cogn. Commun. Netw.* **2022**, *8*, 502–513. [CrossRef]
12. Duan, J.; Wei, X.; Zhou, J.; Wang, T.; Ge, X.; Wang, Z. Latency compensation and prediction for wireless train to ground communication network based on hybrid lstm model. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 1637–1645. [CrossRef]
13. Zou, J.; Tao, Y.; Fang, Y.; Ma, H.; Yang, Z. Receiver design for ici-csk system: A new perspective based on gru neural network. *IEEE Commun. Lett.* **2023**, *27*, 2983–2987. [CrossRef]
14. Jiang, Z.; Ma, Z.; Wang, Y.; Shao, X.; Yu, K.; Jolfaei, A. Aggregated decentralized down-sampling-based resnet for smart healthcare systems. *Neural Comput. Appl.* **2023**, *35*, 14653–14665. [CrossRef]
15. Wang, Q.; He, F.; Meng, J. Performance comparison of real and complex valued neural networks for digital self-interference cancellation. In Proceedings of the 2019 IEEE 19th International Conference on Communication Technology (ICCT), Xi'an, China, 16–19 October 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1193–1199.
16. Wang, B.; Xu, K.; Zheng, S.; Zhou, H.; Liu, Y. A deep learning based intelligent receiver for improving the reliability of the mimo wireless communication system. *IEEE Trans. Reliab.* **2022**, *71*, 1104–1115. [CrossRef]
17. Dittmer, S.; King, E.J.; Maass, P. Singular Values for ReLU Layers. *IEEE Trans. Neural Netw. Learn. Systems.* **2019**, *31*, 3594–3605. [CrossRef] [PubMed]
18. Ding, Z.; Chen, S.; Li, Q.; Wright, S.J. Overparameterization of deep resnet: Zero loss and mean-field analysis. *J. Mach. Learn. Res.* **2022**, *23*, 1–65.

19. Zeng, C.; Ma, C.; Wang, K.; Cui, Z. Parking occupancy prediction method based on multi factors and stacked gru-lstm. *IEEE Access* **2022**, *10*, 47361–47370. [CrossRef]
20. Wei, S.; Qu, Q.; Zeng, X.; Liang, J.; Shi, J.; Zhang, X. Self-attention bilstm networks for radar signal modulation recognition. *IEEE Trans. Microw. Theory Tech.* **2021**, *69*, 5160–5172. [CrossRef]
21. Albarakati, H.M.; Khan, M.A.; Hamza, A.; Khan, F.; Kraiem, N.; Jamel, L.; Almuqren, L.; Alroobaea, R. A novel deep learning architecture for agriculture land cover and land use classification from remote sensing images based on network-level fusion of self-attention architecture. *IEEE J. Sel. Top. Appl. Earth Obs. Remote* **2024**, *17*, 6338–6353. [CrossRef]
22. Korpi, D.; Anttila, L.; Valkama, M. Nonlinear self-interference cancellation in mimo full-duplex transceivers under crosstalk. *EURASIP J. Wirel. Commun. Netw.* **2017**, *2017*, 24. [CrossRef]
23. Elsayed, M.; Aziz El-Banna, A.A.; Dobre, O.A.; Shiu, W.; Wang, P. Full-duplex self-interference cancellation using dual-neurons neural networks. *IEEE Commun. Lett.* **2022**, *26*, 557–561. [CrossRef]
24. Pourdaryaei, A.; Mohammadi, M.; Mubarak, H.; Abdellatif, A.; Karimi, M.; Gryazina, E.; Terzija, V. A new framework for electricity price forecasting via multi-head self-attention and cnn-based techniques in the competitive electricity market. *Expert Syst. Appl.* **2024**, *235*, 121207. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.