Model-Driven Enhanced Analytic Learned Iterative Shrinkage Threshold Algorithm

JUN LI, LIYANG SHANG, BO HE, WENJING ZHENG, AND FEI LIN

1School of Information and Automation, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250000, China
2School of Information Science and Engineering, Shandong University, Qingdao 266000, China

Corresponding author: Jun Li (rogerjunli@sdu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 12005108, and in part by the Natural Science Foundation of Shandong Province under Grant ZR2020QF016.

ABSTRACT The application of deep learning in compressed sensing reconstruction has achieved some excellent results. The deep neural network based on iterative algorithm can not only reflect the excellent performance of deep learning, but also reflect the interpretability of traditional compressed sensing reconstruction algorithm. The existing deep neural networks based on iterative algorithm mainly include learned iterative shrinkage threshold algorithm (LISTA), analytic learned iterative shrinkage threshold algorithm (ALISTA), etc., but each of them has its own shortcomings. We improved the network structure on the basis of predecessors, and proposed a new custom loss function to effectively improve the reconstruction performance of compressed sensing. Experiments show that our proposed neural network reduces the normalized mean square error (NMSE) by 10 dB compared with ALISTA and 15 dB compared with LISTA, and the support set accuracy of the recovered sparse signal can be optimized by our proposed custom loss function by at least 5%, especially LISTA, which has been improved by at least 80%.

INDEX TERMS Deep learning, compressed sensing, signal reconstruction.

I. INTRODUCTION

With the continuous progress and development of intelligent devices, the Internet of Thing (IoT) technology has become one of the main application scenarios of the 5th generation mobile communication system. It can be divided into three categories according to the required services: large-scale IoT, high reliability and low delay IoT and hybrid IoT. The vigorous development of the IoT not only brings infinite possibilities, but also brings new challenges.

The most direct impact of the development of the IoT is that the access of a large number of networking devices makes the signaling overhead become a great burden, and will take up a lot of channel resources. Therefore, appropriate user access technology is imminent. Because of the two characteristics of extremely small packets transmitted by IoT devices and sparsity of the time it sends packets, compression sensing technology has become an effective means to solve the access problem.

Compressed sensing technology is a promising technology, which has a wide range of application scenarios. Since it was proposed by Emmanuel candès, David Donoho and Terence Tao in 2006 [1], [2], it has been applied in various scenarios of communication and industrial internet of things, and the sparse signal reconstruction is the technical focus of compressed sensing technology. Since the compressed sensing technology was proposed, scholars have proposed many compressed sensing reconstruction algorithms with complete mathematical foundation. However, traditional compressed sensing reconstruction methods all based on sparse prior knowledge. By solving an optimization problem, the original signal is iteratively reconstructed, which limits the real-time processing capability of traditional compressed sensing reconstruction algorithms and greatly limits the application breadth and depth of compressed sensing. On this basis, the traditional compressed sensing reconstruction algorithm is combined with the popular deep learning, and the neural network is used to realize the compressed sensing reconstruction algorithm, so as to realize the real-time reconstruction, which creates a new research direction.
neural network in image learning, many image processing technologies based on compression sensing and deep learning have been created. At the beginning, people used data-driven neural networks to reconstruct compressed sensing, specifically include based on stacked denoising autoencoder(SDA) [3], DeepInverre [4], deep residual network reconstruction(DR²-Net) [5], ReconNet [6] and generative model [7], etc. However, data-driven neural networks only rely on data sets for learning, without the support of models, and cannot learn the prior knowledge of compressed sensing principle, so they do not have sufficient interpretability. Therefore, we focus on model-driven compressed sensing neural networks. Model-driven methods are divided into two types: black box neural network and iterative unfolding neural network. Black box neural network embed neural network into traditional compressed sensing reconstruction algorithms to serve as a function. Specifically: restricted Boltzmann machine based on approximate message passing algorithm(RBM-AMP) [8], recurrent image density estimator-compressed sensing(RIDE-CS) [9], recurrent image density estimator(RIDE) [10] and OneNet [11], etc. Iterative unfolding neural network is our main research direction, which will be introduced in detail in the following.

A. RELATED WORK
At present, there are many neural networks based on traditional compressed sensing reconstruction algorithms, Yan Yang et al. used generalized compressed sensing model with linear transformation and generalized sparse regularization, unfolded iterative alternating direction method of multipliers(ADMM) algorithm to discriminatively learn these transformations, sparse regularization and other hyper-parameters [12], they applied this method to magnetic resonance imaging(MRI) and achieved good results. Ma et al. unfolded the tensor nuclear norm minimization using ADMM (TNN-ADMM) algorithm into hierarchical deep neural network [13], and applied this method to the snapshot compressive imaging (SCI) system. They mapped each iteration to a phase of the neural network, and developed the deep network structure by connecting multiple phases in order. The output of each reconstruction layer can be regarded as a recovered signal. It is compared with the real signal to calculate the training loss to accelerate its convergence. Shipeng Zhang et al. integrated the learned tensor low-rank prior into the half quadratic splitting (HQS) based optimization algorithm, combined with the canonical-polyadic(CP) decomposition theory, to achieve end-to-end hyperspectral images(HSI) reconstruction [14].

The above image processing methods have achieved good results, while the IoT access we are concerned about does not require such a complex processing method, a simpler threshold shrinkage algorithm can achieve good results. Karol Gregor and Yann LeCun unfolded the iterative shrinkage threshold algorithm(ISTA) into a 16-layer quasi recurrent neural network and named it Learned ISTA(LISTA) neural network [15]. It learned the fixed matrix in the traditional structure, so as to achieve better recovery effect through truncated network than multiple iterations. LISTA contains two parameter matrices that need to be generated according to the measurement matrix and a parameter of the threshold function, both of them need to be trained. The dimensions of the matrices depend on the length of sparse signals and the length of sensing signals. Generally speaking, their dimensions will be large, which means that the task to train parameters will also be heavy. On this basis, Xiaohan Chen et al. reduced the number of parameter matrices to one by using coupling weights and proposed LISTA coupling(LISTA-CP) neural network [16], achieved the same interpretation ability of sparse signal recovery as LISTA with a lightweight structure. Jialin Liu et al. proved that the analytic weight matrix can be calculated in advance through the analytic method to replace the parameter matrix trained by LISTA-CP network with good effect, and established analytic LISTA(ALISTA) neural network [17]. ALISTA further lightens the network structure, and reduces the parameters to be trained to only two scalars by calculating the required inverse measurement matrix in advance. The LAMP algorithm, inspired by LASSO solution method, uses state evolution method to solve compressed sensing reconstruction. Its implementation method is quite different from the previous several networks, so it will not be interpreted too much in this paper. Not only can deep learning promote the progress of compressed sensing theory, but compressed sensing theory also affects the development of deep learning in turn. Huan Li et al. took inspiration from compressed sensing theory, and in turn used optimization algorithms to optimize the structure of deep neural network for image recognition tasks [18], which better stimulated the performance of deep neural network.

B. CONTRIBUTIONS
Although ALISTA has extremely lightweight structure of parameter to be trained, it also has the disadvantage of unstable performance due to the fact that each layer of network only trains scalar. Therefore, we propose an enhanced network structure to achieve better recovery performance and meet the requirements of stable performance at the same time. Meanwhile, we propose a new loss function. Using this loss function, we can further optimize the reconstruction performance of sparse signals, and the training time of this method is greatly reduced compared with the “support selection(SS)” method which has the same purpose.

The rest of the article is organized as follows: Section II presents a detailed background and network structure of LISTA and ALISTA. Section III, introduces the network structure of enhanced analytic learned iterative shrinkage threshold algorithm(EALISTA) in detail, and gives the components of our custom loss function. Section IV presents the numerical results form experiments. Section V concludes the paper.
II. NETWORK STRUCTURE

Here is a brief introduction to the mathematical model of compressed sensing technology. We consider the model containing noise:

\[ y = Dx + n \]  

\( y \in \mathbb{R}^M \) is called sensing signal. Our purpose is to recover the sparse signal \( x \in \mathbb{R}^N \) in (1), this task is also called sparse coding in some studies. In (1), since \( M \) is much smaller than \( N \), the sensing matrix \( D \in \mathbb{R}^{M \times N} \) is an underdetermined matrix. In this case, it is impossible to directly solve the inverse problem by linear method. Therefore, it is necessary to find a suitable inverse solution process for this mathematical model, which is called compressed sensing reconstruction.

A. ITERATIVE SHRINKAGE THRESHOLD ALGORITHM

Iterative shrinkage threshold algorithm (ISTA) [19] is a classical traditional compressed sensing reconstruction method. ISTA is implemented based on compressed sensing reconstruction target after convex relaxation, which is as follows:

\[
\begin{align*}
\text{minimize} & \quad \|x\|_1 \\
\text{subject to} & \quad \|Dx - y\|_2 \leq \varepsilon 
\end{align*}
\]

Equation (2) and (3) can be transformed into the following optimization problem:

\[
\begin{align*}
\text{minimize} & \quad \|Dx - y\|_2^2 + \lambda \|x\|_1 \\
\end{align*}
\]

(4)

In (4), the 2-norm on the left has a continuous Lipschitz derivative, and \( \lambda \) is a weight variable used to weigh the accuracy and sparsity of the recovered signal. Based on Lipschitz property and combined with random gradient descent method, the final ISTA algorithm can be obtained:

\[
x^{k+1} = h_\theta(x^k - \frac{1}{L}D^T(Dx^k - y)) 
\]

where, \( L \) is Lipschitz constant, which is usually set to a constant slightly larger than the maximum eigenvalue of matrix \( D^TD \), and the threshold function \( h_\theta(\cdot) \) is:

\[
h_\theta(V) = \max((|V| - \theta), 0)\text{sign}(V)
\]

Generally, \( \theta \) is chosen as \( \lambda/L \). The above is the traditional ISTA algorithm of compressed sensing, and the structure of deep learning network to be explained in the following is derived on this basis, which can also be said to be the model-driven neural network created with this method as the model.

B. LISTA AND ALISTA

Based on the traditional ISTA compressed sensing reconstruction algorithm, Gregor and LeCun proposed a deep learning based sparse encoder named learned iterative shrinkage threshold algorithm (LISTA) [15]. Although this method was called by the author to achieve the best sparse coding, but its essence is also a process of recovering the sensing signal which sampled by sensing matrix to its original sparse signal.

LISTA adopts a quasi recurrent neural network structure, one layer represents an iteration in ISTA. Among the set number of layers, most of the parameters used in each layer are shared, and a few are independent. In this method, equation (5) is transformed into the following form:

\[
x^{k+1} = h_\theta(x^k - W_t y + Sx^k)
\]

Among them, \( W_t \in \mathbb{R}^{N \times M} \) is \( D^T/L, S \in \mathbb{R}^{N \times N} \) is \( I - (D^TD)^{-1} \), and \( I \) is the identity matrix. For (7), \( W_t \), \( S \) and \( \theta^i \) are parameters to be trained and their initial values are using the values in the traditional compressed sensing reconstruction algorithm. In this way, the LISTA algorithm is obtained, and its structure can be represented by Fig. 1.

We can see from Fig. 1 that \( W_t \) and \( S \) are the shared parameters just mentioned, and \( \theta^i (i = 1, \ldots, k + 1) \) is the independent parameter of each iteration.

LISTA has opened the way of model-driven compressed sensing reconstruction neural network. Its network structure needs to train two parameter matrices related to the sparse signal dimension and the sensing signal dimension, and the dimensions of sparse signals are often huge, so there are often many parameters that need to be trained. On this basis, some scholars began to optimize LISTA network in order to reduce the training burden.

Chen et al. proposed learned iterative shrinkage threshold algorithm coupling (LISTA-CP) [16]. They further transformed (7) into the following form:

\[
x^{k+1} = h_\theta(x^k - W^T_t (Dx^k - y))
\]

In (8), \( W^T \in \mathbb{R}^{N \times M} \) is \( D^T / L \). Although the form has changed, its author has confirmed that this model has the same performance as (7). In this network, the idea of quasi recurrent neural network from LISTA is followed, and \( W^T \) is taken as the shared parameter of each layer, \( \theta^i (i = 1, \ldots, k + 1) \) continues to be an independent parameter of each layer.

Although the LISTA-CP network reduces a large dimension matrix need to be trained, the target \( W^T \) trained by its network can be replaced by a matrix with good recovery performance through pre-calculation, that is, a matrix that meets the “good” parameter requirements of the LISTA-CP network can be pre-calculated, as confirmed by JiaLin Liu et al., and the structure of the LISTA series model can be further simplified. This pre-calculation process is only related to the sensing matrix \( D \), based on this, JiaLin Liu et al. proposed the analytic learned iterative shrinkage threshold algorithm.

JiaLin Liu et al., through the method of generalized cross correlation, used the sensing matrix to pre-calculate the replacement matrix, plus some other settings, and the replacement matrix can directly obtain approximation reconstruction effect with the “good” trained parameter matrix \( W^T \). This replacement matrix can also be called the analytic matrix. Based on the above theory, they proposed the analytic learned iterative shrinkage threshold algorithm (ALISTA) model [17]:

\[
x^{k+1} = h_\theta(x^k - y^k W^T_t (Dx^k - y))
\]
The calculation of analytic matrix is as follows:

$$\tilde{W} \in \arg \min_{W \in \mathbb{R}^{M \times N}} \|W^TD\|_F^2 \text{ subject to } (W_{:,n})^TD_{:,n} = 1 \quad (\forall n = 1, 2, \cdots, N) \quad (10)$$

Thus, the ALISTA algorithm is obtained, and its structure is shown in Fig.2.

In the network structure of ALISTA, there is no parameter shared in each layer like that in LISTA structure, only independent parameters in each layer like $\theta^i$ and $\gamma^i$ ($i = 1, \cdots, k + 1$).

### III. ENHANCED ALISTA AND CUSTOM LOSS FUNCTION

#### A. ENHANCED ALISTA

Although ALISTA network extremely lightens the LISTA series network’s structure, it also has some disadvantages. Only considering a trainable scalar always has a certain impact on the back-propagation (BP) algorithm. For the random gradient descent algorithm, the gradient symbols between different bits of the vector are likely to be opposite. In this way, after the trainable scalar is changed in the previous bit, the next bit may have to be changed in the reverse direction. This will also affect the optimization results of the previous one.

To solve this problem, we propose a new LISTA series neural network model, which is called enhanced analytic learned iterative shrinkage threshold algorithm (EALISTA), as follows:

$$x^{k+1} = h_{\theta^k}(P\tilde{x}^k\cdot\tilde{W}^T(Dx^k - y)) \quad (11)$$

This algorithm follows the idea of ALISTA’s analytic matrix and improves its defect that only train scalars, before feeding the threshold function $h_{\theta^k}(\cdot)$, a trainable parameter matrix $P \in \mathbb{R}^{N \times N}$ is added, and it is initialized to the identity matrix during training. The purpose of this parameter matrix is to sensitively capture the accuracy of the support set when recovering sparse signals, and to make up for the defects of ALISTA mentioned above. Later experiments show that the network structure proposed in this paper can get better reconstruction results. The structure of EALISTA is shown in Fig.3.

In this structure, $P$ is the shared parameter of each layer, $h_{\theta^k}(\cdot)$ is the independent parameter of each layer. The structure of each layer of this neural network is shown in Fig.4. $x^{k-1}$ as shown in the Fig.4 is the output of the upper layer network, with the same dimension as the original sparse signal and the recovered signal, and the dimension of the analytic matrix is $M \times N$. $y$ is the sensing signal, the dimension is $M$. The signal’s dimension after preprocessing is $N$, as the input layer of the neural network, it become $x^k$ after throughing the parameter matrix $P$ with dimension of $N \times N$ and throughing the threshold function, as the output layer of this single layer network.

#### B. CUSTOM LOSS FUNCTION

In addition to improving the structure of LISTA series network, we also optimizes the loss function in neural network. Using the optimized loss function can make the compressed sensing reconstruction effect of neural network better. The optimization of loss function is introduced in detail below.

In previous LISTA series networks, the mean square error (MSE) is often directly used as the loss function, which will ignore an important criterion in sparse signal reconstruction, that is, the support set of non-zero elements. The characteristic of sparse signal is that its non-zero elements account for a very small proportion, and zero elements are its main constituent elements. Therefore, the position of non-zero elements in the whole sparse signal is an extremely sensitive problem, and the position of non-zero elements is also known as the support set. Due to the characteristics of neural networks, the recovered signal output by the network often has non-zero values outside the support set, and because the deviation between these non-zero values and zero is very small, it has little impact on the MSE, which makes these neural networks show a downward trend in the loss function, but their performance in the reconstruction of non-zero element support set may be very poor.

Therefore, we use a custom error function as the loss function of the whole network, in order to continuously reduce the loss function and obtain better support set recovery performance at the same time. Our custom error function is
FIGURE 2. Structure diagram of ALISTA neural network.

FIGURE 3. Structure diagram of EALISTA neural network.

FIGURE 4. Single layer network structure of EALISTA.
as follows:

\[ L(\hat{x}, x) = L_1(\hat{x}, x) + \beta L_0(\hat{x}, x) \]  
(12)

Among them, \( L_1(\cdot) \) and \( L_0(\cdot) \) are respectively:

\[ L_1(\hat{x}, x) = \frac{1}{K} \sum_{i \in S} (\hat{x}_i - x_i)^2 \]  
(13)

\[ L_0(\hat{x}, x) = \frac{1}{N - K} \sum_{j \not\in S} (\hat{x}_j - x_j)^2 \]  
(14)

The parameters in \( L_1(\cdot) \) and \( L_0(\cdot) \) are network’s output and label data respectively, \( K \) is the number of elements contained in the support set, \( S \) is the support set of non-zero elements of the real sparse signal, \( \beta \) is the penalty coefficient when non-zero values appear in the non-support set, which is generally set as a constant greater than 1. But the penalty coefficient should not be set too large, otherwise it is easy to force all the elements of the neural network’s output to be zero directly. Later experiments show that this custom loss function can effectively improve the recovery performance of LISTA series neural networks, and the training time required is far less than the “support selection” (SS) method.

IV. NUMERICAL RESULTS

In this section, we will use experimental data to prove that:

1) EALISTA has better recovery performance than other LISTA series neural networks;

2) In terms of the accuracy of recovered sparse signal’s support set, our custom loss function can make each LISTA series neural network play a better recovery accuracy, and this method takes much less time than the existing SS method.

This experiment was completed on the same computer to compare the performance and computational cost of different neural networks. In the experiment, we use some parameter settings used by the authors of LISTA-CP and ALISTA neural network, set the dimension \( N \) of the sparse signal to 500, the dimension \( M \) of the sensing signal compressed by the sensing matrix to 250, and the sensing matrix \( D \) to the Gaussian random matrix of \( M \) rows and \( N \) columns, \( D_{ij} \sim \mathcal{N}(0, 1/N) \), and standardize each column, so that each column of the sensing matrix has a standard \( l_2 \) norm. The number of non-zero elements of the sparse signal is set to account for about 0.1 of the total, and its position follows the Bernoulli distribution, while the value of the non-zero elements is randomly sampled from the standard Gaussian distribution. In this experiment, according to the above parameter settings, several sparse signals are randomly generated, and they are used to generate the same number of sensing signals through the sensing matrix. Each sparse signal corresponds to a sensing signal.

The analytic matrix of ALISTA network and EALISTA network was calculated by (10), the number of whole network layers also followed the 16 layers set by the LISTA-CP and ALISTA neural network during the experiment. The 16-layer network included the shared parameters and independent parameters mentioned above, which are set according to different networks. Since previous scholars have compared the performance of LISTA series neural network with traditional ISTA and fast ISTA algorithm, and verified that LISTA series neural network has better reconstruction performance than traditional algorithms, this paper does not add the performance of traditional algorithms, only the performance comparison with LISTA series neural network is listed. This part mainly compare the error performance by using the proposed EALISTA network with LISTA and ALISTA network, and compare the accuracy of support set recovered by using the proposed custom loss function with the MSE loss function and the SS method.

A. BASE ON OPTIMIZE ERROR ASPECT

This section mainly introduces the performance optimization brought by optimizing the neural network structure. In this section, the recovery performance of several LISTA series neural networks is compared when the MSE is directly used as the loss function, and the comparison mainly focuses on the loss value.

The following Fig.5 shows the comparison of the loss function values of several neural networks at different training steps.

From the left figure of Fig.5, we can see that the loss value of LISTA gradually decreases with the increase of training steps. In contrast, thanks to the pre-calculated analytic matrix, the loss value of ALISTA and EALISTA decreases rapidly. ALISTA reduced the loss value to the vicinity of the global optimal solution after 1000 training steps, but since each layer of network only train a scalar, the overall instability phenomenon mentioned above occurs. However, our proposed EALISTA can achieve the performance similar to that of ALISTA when 1000 training steps, and can reduce the loss value to lower than the minimum loss value of ALISTA within 10000 training steps, and maintain stable convergence. During numerical experiment, the lowest loss value of EALISTA can reach about 1/5 of the lowest loss value of ALISTA.

From the decibel form of normalized mean square error (NMSE), we can better see the performance comparison of several networks. The right figure of Fig.5 shows the NMSE comparison of the three networks. It is obvious that the performance of ALISTA is superior to LISTA, which is reduced by about 5 dB in NMSE, while our proposed EALISTA has better performance, the NMSE reduced by 10 dB compared with ALISTA and about 15 dB compared with LISTA.

In terms of the accuracy of support set detection, the comparison of several neural networks is shown in Fig.6. From Fig.6, we can see that the accuracy of ALISTA is 80% more accurate than LISTA, our proposed EALISTA is 6% more accurate than ALISTA at lower training steps, and it is basically the same in the later period. The reasons for the low accuracy of the LISTA support set has been discussed in detail above.
So far, we can see that when the MSE is used as the loss function, the EALISTA network has better performance in both the loss value and the accuracy of the support set.

**B. BASE ON OPTIMIZE SUPPORT SET ACCURACY ASPECT**

This section mainly introduces the optimization of sparse signal’s support set accuracy caused by the use of custom loss function. In this section, we compared the difference in the recovery performance of several LISTA series neural networks when using MSE loss function, custom loss function and SS method, and the comparison mainly focuses on the recovery accuracy of support set.

Since we define the penalty for non-zero elements of the non support set of the recovered signal in the custom loss function, when the neural network optimizes the parameters, it will give priority to make the network learn that not the more non-zero elements, the better. Therefore, the phenomenon of non-zero elements in the non support set of the recovered signal will be greatly reduced. The rest of the custom loss function is to perform the task of minimizing the MSE between the recovered signal and the original sparse signal.

The following Fig. 7 shows the comparison of the recovery accuracy of the support set between various LISTA series neural networks when the custom loss function and the MSE are used as the loss function respectively.

In Fig. 7, those with the suffix “mse” are the accuracy of using the MSE loss function, while those without the suffix “mse” are the accuracy of using the custom loss function. As we can see from Fig. 7, using the custom loss function, all LISTA series neural networks have a better recovery performance, the support set accuracy of both ALISTA and EALISTA increased by about 5%, and the support set accuracy of LISTA increased by about 80%. The increase in
FIGURE 7. Comparison of support set accuracy between custom loss function and MSE loss function.

FIGURE 8. Comparison of loss values between different loss function.

accuracy here is due to the “big” penalty in our custom loss function when the location of the non support set is recovered to a non-zero value. The bigger the penalty is, the higher the recovery accuracy is. However, there is the risk of making the recovered signal all force to zero, because the “loss” when forcing to zero is much smaller than the “loss” brought by the “big” penalty. It will make the network no longer updated, and this effect is undesirable.

Fig. 8 shows the comparison of the loss values between each neural network when using the custom loss function. Fig. 8 (a) shows the direct loss values, Fig. 8 (b), (c) and (d) respectively show the NMSE’s decibel form when using the MSE as the loss function, the NMSE’s decibel form with additional penalty value when using the custom loss function and the NMSE’s decibel form without additional penalty value when using the custom loss function. From Fig. 8(b) and (c), we can see that compared with directly use MSE as the loss function, the custom loss function increases the loss value of ALISTA and EALISTA, the NMSE of ALISTA and EALISTA increased by about 4 dB and 8 dB respectively. On the contrary, LISTA’s NMSE decreases by 2 dB instead of increasing because the error on its error support set is directly removed by the custom loss function. These changes both related to the “big” penalty just mentioned, but our proposed EALISTA still have the least error.

As mentioned above, for compressed sensing reconstruction, the accurate reconstruction of support set is an important criterion, so scholars have also studied this aspect. The proposer of LISTA-CP network proposed a method to
improve the reconstruction accuracy of support set at the same time. They called it “support selection”, which was used to optimize the network and obtained the LISTA-SS algorithm [16]. ALISTA’s author also compared this method in their experiments.

The following Fig.9 shows the comparison of various LISTA series neural networks’ support set recovery accuracy using the custom loss function as the optimization method and using the SS method as the optimization method under the MSE loss function.

In Fig.9, those without the suffix “ss” are the accuracy of using our custom loss function, and those with the suffix “ss” are the accuracy of using SS method. We can see from Fig.9 that LISTA-SS algorithm can indeed improve the accuracy of LISTA support set by about 63%. However, compared with the method we proposed, its performance is not good enough, and we find that the effect of this method to ALISTA and IALITA is not ideal, the custom loss function proposed by us can achieve good optimization results for all LISTA series neural networks. The data proves that replacing the loss function of MSE with the custom loss function proposed by us improves the accuracy and has better optimization performance than using SS method as the optimization method. On the other hand, we also found that when using SS method in this scenario, the loss value of each neural network is not as good as the method we proposed. The experimental results are shown in Fig.10.

From Fig.10, we find that the NMSE of LISTA and EALISTA under SS method is 5 dB higher than that under our custom loss function, and only ALISTA is about 2 dB lower.
than our loss function. Part of the reason why the effect of SS method is not good in this comparison is that the comparison is based on the scene of directly training multilayer neural networks. Because the layer by layer training method used by SS method’s author has a huge training time cost, and multilayer training can directly achieve good results. So, we will no longer compare the effect of layer by layer training scene, although SS method can do better in the layer by layer training scene.

We also compared the training time of the two methods, as shown in Fig. 11.

Figure 11 shows that the training time of each network under the custom loss function is about 12 seconds, while the SS method reaches 47-49 seconds. In the direct multilayer training scenario, SS method has already consumed so much training time. It is conceivable that the training time cost will be greater in the layer by layer training scenario. From the above results, we can draw a conclusion that the custom loss function we proposed can get a better recovery accuracy of support set within less training time cost than SS method.

Through the above comparison, we can see that our proposed custom loss function can effectively improve the recovery performance of compressed sensing reconstruction. All the above experimental results are under the condition that the signal-to-noise ratio (SNR) is set to infinity, that is, there is almost no noise. We also compared the recovery performance of various neural networks under different SNR scenarios, and the experimental results are shown in Fig. 12.

Those with the suffix “mse” are the accuracy of using the MSE loss function, while those without the suffix “mse” are the accuracy of using the custom loss function. From Fig. 12, we find that because LISTA neural network has more trainable parameters than ALISTA neural network, the loss value of LISTA neural network is smaller than that of ALISTA in the case of low SNR. From this we can see the disadvantages of ALISTA network mentioned before, but the support set accuracy of ALISTA’s recovered signal is much better than that of LISTA in the case of low SNR, meanwhile, our custom loss function can improve the accuracy of neural networks in all SNR cases.

V. CONCLUSION

Based on the recent studies of LISTA series neural networks, we propose a new network structure and a new custom loss function, they can improve the compressed sensing reconstruction performance. The experiment shows that the NMSE of EALISTA is 10 dB lower than ALISTA and 15 dB lower than LISTA in terms of error. In addition, the custom loss function proposed by us can improve the accuracy of support set of ALISTA and EALISTA by about 5%, and LISTA can increase by more than 80%. While the training time is only 1/4~1/5 of SS method.

REFERENCES

[1] D. L. Donoho, “Compressed sensing.” IEEE Trans. Inf. Theory, vol. 52, no. 4, pp. 1289–1306, Jan. 2006.

[2] E. J. Candes, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information.” IEEE Trans. Inf. Theory, vol. 52, no. 2, pp. 489–509, Feb. 2006.
JUN LI received the M.S. degree in communication and information systems from Shandong University, in 2005, and the Ph.D. degree in signal and information processing from the Beijing University of Posts and Telecommunications, in 2011. He is currently a Vice Professor at the School of Electronic Information Engineering, Qilu University of Technology. His research interests include deep learning, 5G technologies, MIMO-OFDM, cooperative communication, and cognitive radio.

LIYANG SHANG received the B.S. degree in communication engineering from the School of Electrical Engineering, Dalian University of Science and Technology, Dalian, China, in 2010, and the Ph.D. degree in electronic information engineering with the Qilu University of Technology (Shandong Academy of Sciences), Jinan, China, in 2018. He is currently pursuing the M.S. degree in electronic information engineering with the Qilu University of Technology (Shandong Academy of Sciences), Jinan, China. His research interests include wireless communication systems, compressed sensing, and deep learning.

BO HE received the M.S. degree in communication and information system from Shandong University, in 2002, and the Ph.D. degree in signal and information processing from the Beijing University of Posts and Communications, China, in 2006. In 2006, she joined the School of Information Science and Engineering, Shandong University. Her research interests include mobile communications and visible light communications.

WENJING ZHENG received the Ph.D. degree in particle physics and nuclear physics from Shandong University, in 2017. She is currently a Lecturer at the School of Electronic Information Engineering, Qilu University of Technology. Her research interests include deep learning, signal processing, and electric field sensors.

FEI LIN received the B.S. degree in applied electronic technology and the M.S. degree in power electronics and power devices from Shandong University, Jinan, China, in 1998 and 2002, respectively, and the Ph.D. degree in communication and information system from the Beijing University of Posts and Telecommunications, in 2008. She is currently a Professor with the School of Electronic Information Engineering, Qilu University of Technology, Jinan. Her current research interests include the areas of intelligent wireless communications and energy internet.