Attention Model for Massive MIMO CSI Compression Feedback and Recovery

Qiuyu Cai, Chao Dong and Kai Niu
Key Laboratory of Universal Wireless Communications, Ministry of Education
Beijing University of Posts and Telecommunications
Beijing, China
{caiqiuyu, dongchao, niukai}@bupt.edu.cn

***Abstract*—In massive multiple-input multiple-output (MIMO) scenario, in order to adapt to channel features and ensure the high-reliability and high-rate of communication, the channel state information (CSI) should be sent back to the base station (BS). As the number of antennas increases, the amount of CSI feedback increases dramatically. Ensuring communication performance and reducing CSI feedback is a challenging research direction. Some deep learning (DL) based models have been proposed to solve this problem. This paper aims to improve recovery performance and decrease time complexity. First, in encoder network, long short-term memory (LSTM) network have been introduced; second, in decoder network, attention mechanism has been added; third, early stopping have been used in training step. The simulation results show that the structure proposed in this paper is not only better than the performance of the traditional compressed sensing algorithm, but also better than the existing DL-based feedback network and achieve state-of-the-art.**

***Index Terms*—deep learning, compressed sensing, CSI feedback, massive MIMO**

I. Introduction

Massive MIMO can greatly improve spectrum efficiency and is one of the key technologies of 5G communication systems. In time division duplex (TDD) mode, channel estimation can be executed directly on the BS to obtain the corresponding channel information. In the frequency division duplex (FDD) scenario, the user equipment (UE) need to estimate CSI and send it back to the BS. As the number of antennas increases, the amount of feedback information increases dramatically. In the LTE system, channel quality index (CQI), precoding matrix index (PMI) and rank index (RI) are used in feedback links, which can greatly reduce the amount of feedback, but introduce many inaccurate features.

CSI has attracted many people's research. [1], [2] mainly use the algorithm of compressed sensing (CS). By using the space-time correlation of channel response, the channel matrix is transformed into a sparse matrix for compression feedback and then recovered. Many compression sensing algorithms can be used for CSI compression feedback. Compressed sensing algorithms are mainly used in the field of image processing, such as LASSO [3], TVAL3 [4] and BM3D-AMP [5]. All of these methods can be used in CSI compression feedback and recovery. [3] can be easily implemented but the recovery performance is not very well. [4], [5] are hard used for practical deployment on real system because of time complexity caused by iteration of algorithm and hand-crafted parameters.

Due to the excellent performance of convolutional neural networks in computer vision, many studies have proposed deep learning algorithms for compressed sensing. These studies have achieved great performance compared with traditional algorithms [3], [4], [5] in image processing. [6] implement a fully connected neural network for CS called stacked denoising auto-encoder (SDA) and have improved signal recovery performance compared with the conventional CS approach. Convolutional neural network (CNN) has been successfully used in image compression and recovery in [7], [8], [9].

[10] proposed a neural network called CsiNet for CSI feedback, which is the first one introducing the deep learning algorithm into the CSI feedback and achieved the better performance than the conventional CS-based algorithm.

Based on CsiNet, we optimized the original structure in the following points:

- In encoder part, the LSTM [11] network have been introduced to be instead of the original fully connected network. The LSTM network can make full use of the correlation between the channel matrices and retain important information when the compression ratio (CR) is very high.
- In the decoder part, the original network is based on the simple CNN. The attention mechanism is widely used in recurrent neural network (RNN) and firstly introduced in [12]. Inspired by [13], we introduced attention mechanism in CNN. The feature maps from the CNN can be fully utilized by our model. This module improves the performance most, so we called our model Attention-CsiNet.
- We introduce early stopping to prevent overfitting, our network can convergence more quickly and save much time in training steps.

The remainder of the paper is organized as follows. Section II briefly describes the system model of Massive MIMO and the mechanism of CSI feedback. We present the proposed attention-CsiNet in Section III. The simulation results and conclusion are presented in Section IV and Section V.

II. System Model

A simple single-cell FDD downlink massive MIMO-orthogonal frequency division multiplexing (OFDM) system with N_c subcarriers is considered. There are $N_t \gg 1$ transmit

978-1-5386-7646-2/19/$31.00 ©2019 IEEE

antennas as uniform linear array (ULA) at a BS and a single receiver antenna at a UE. The received signal at i_{th} subcarrier for UE can be modeled as follows:

$$y_i = \mathbf{h}_i \mathbf{v}_i x_i + n_i, \tag{1}$$

where $\mathbf{h}_i, \mathbf{v}_i, x_i$ means the channel vector in frequency domain, precoding vector designed by the BS based on the received downlink CSI, transmit data symbol and additive white Gaussian noise respectively. We denote $\mathbf{H}_D = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_{N_c}]$ as the CSI matrix in the spatial frequency domain. In FDD links, the UE need to estimate $\mathbf{H}_D$ and send CSI back to the BS through feedback links to track the time-varied channel features. Once the BS receive the CSI feedback, it can design the precoding vectors, modulation mode and code rate. The numbers of feedback parameters are $N_c N_t$, which are not allowed to send through limited feedback links. Moreover, we assume that the perfect channel estimation in this paper and the perfect CSI can be acquired. So we only pay attention to the feedback scheme.

To reduce the feedback overload, we can observe $\mathbf{H}_D$ in the angular-delay domain via 2D discrete Fourier transform (2D-DFT). $\mathbf{H}_D$ can be transformed into an approximately sparsified matrix $\mathbf{H}_S$:

$$\mathbf{H}_S = \mathbf{F}_d \mathbf{H}_D \mathbf{F}_a, \tag{2}$$

where $\mathbf{F}_d$ and $\mathbf{F}_a$ are $N_c \times N_c$ and $N_t \times N_t$ DFT matrices respectively. Due to the limited multipath delay, performing DFT on frequency domain channel vectors can transform $\mathbf{H}_D$ into a sparsed matrix $\mathbf{H}_S$ in the delay domain. Only the first $\widetilde{N}_c (\ll N_c)$ rows of $\mathbf{H}_S$ contain distinct no-zero elements and the other elements are close to zero. So we remain the first $\widetilde{N}_c$ rows of $\mathbf{H}_S$ and remove the other $(N_c - \widetilde{N}_c)$ rows. So the total number of feedback elements of $\mathbf{H}_S$ can be reduced to $N = \widetilde{N}_c \times N_t$ which is remarked as $\mathbf{H}$. This number is still very large in Massive MIMO senerio.

In this paper, we designed an encoder network,

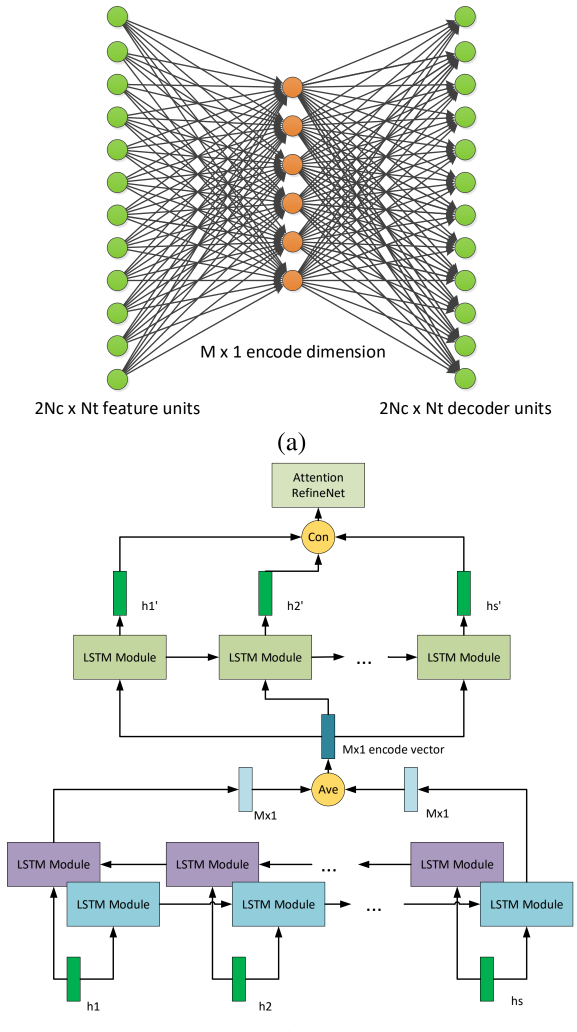
$$\mathbf{H}_{en} = f_{LSTM}(\mathbf{H}), \tag{3}$$

with LSTM neural network which can transform the CSI matrix into an M-dimensional vector, where the compression ratio (CR) is M/N.

Moreover, we designed a decoder network with attention CNN to recover $\mathbf{H}$ from $\mathbf{H}_{en}$, which is

$$\widehat{\mathbf{H}} = f_{CNN}(\mathbf{H}_{en}). \tag{4}$$

Above all, the CSI feedback process is as follows. First, the UE acquire the channel matrix $\mathbf{H}_D$ and we perform 2D DFT in (2) to transform $\mathbf{H}_D$ into sprased matrix $\mathbf{H}_S$. Second, we retain the first $\widetilde{N}_c$ rows with distinct no-zero elements as the CSI matrix $\mathbf{H}$. Third, we use LSTM network to compress $\mathbf{H}$ into M-dimensional vector and send this vector to the BS. Finally, BS use the CNN decoder network to recover $\mathbf{H}$ and use inverse DFT to get the final CSI matrix.



(a)

(b)

Fig. 1. (a) The structure of encoder network in CsiNet with fully connected neural network; (b)The structure of encoder network in Attention-CsiNet with bidirectional LSTM neural network.

III. THE PROPOSED ATTENTION MODEL

Although the CsiNet in [10] shows great performance in CSI sensing and reconstruction, we still find there is room for improvements. In this Section, we introduced our improvements in two section. On the one hand, we introduced bidirectional LSTM neural network in encoder network. On the other hand, we add attention mechanism in CNN decoder network.

A. LSTM encoder

The CsiNet uses fully connected neural network to extract features and compress $\mathbf{H}$ to M-dimensional vector. Fig. 1(a) shows the encoder network structure in CsiNet. The CsiNet ignored the correlation between subcarriers. Inspired by the recurrent neural network (RNN) [14] that has good performance in Natural Language Processing (NLP) and extracting correction in sequence data. We use LSTM neural network to replace fully connected neural network to improve recovery quality. The proposed structure showed in Fig. 1(b). And we use bidirectional LSTM (bi-LSTM) to get two M-dimensional vector and find the average of two vectors as encode vector for feedback. Moreover, LSTM network share the same parameter. Because of this, it can dramatically reduce parameter overload compared with the encoder network in CsiNet. As show in Fig. 1(b), we reshape $\mathbf{H}$ to $[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_s]$ and send s vectors into bi-LSTM network. s is also the time step of LSTM network. Finally, we get a M-dimensional vector as final encode CSI vector for feedback.

B. Attention Mechanism

Attention mechanism have been used in machine translation and other NLP tasks widely. Inspired by the squeeze-and-excitation (SE) network in [13], we add attention mechanism in CNN to let the decoder network focus on different convolution feature maps. The structure of attention CNN showed in Fig. 2.

When we get convolution $L \times H \times W$ feature maps, we first use global average pooling to get a $L \times 1 \times 1$ vector, then we use fully connected neural network to transform this vector to C-dimensional vector. Next, we still use a fully connected network to reconstruct this C-dimensional network to L-dimensional vector with sigmoid activation function. We use this L-dimensional vector to multiply with the convolution feature maps to get the final feature maps. Different CSI matrix will get different attention weights on different feature map channels. Because of this scheme, our network can extract more useful information so that it can recover CSI matrix better.

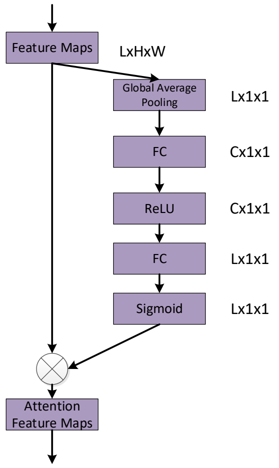


Fig. 2. The structure of attention mechanism in Attention-CsiNet.

C. The Structure of Attention Csi-Net

The whole structure of our model and feedback mechanism are showed in Fig. 3. Firstly, the real and imaginary parts of CSI matrix $\mathbf{H}$ are treated as two channels of the input of networks. The first layer is a convolutional layer with dimensions of 3×3 kernels and batch normalization [15] layer which can accelerate training and prevent overfitting. This layer will generate two feature maps. Then we reshape the feature maps into s vectors and send it into bi-LSTM neural network to generate code $\mathbf{H}_{en}$ which is a M-dimensional vector. Finally, UE will send the code back to the BS.

Once we acquire the code word $\mathbf{H}_{en}$ in the BS, we use LSTM decoder network to reconstruct the information initially. We repeat $\mathbf{H}_{en}$ s times and send these vectors into LSTM neural network to recover $\mathbf{H}$ basically. Then the output of LSTM will be reshaped into $2 \times N_t \times N_t$ matrix. This matrix will be send to two attention RefineNet module for reconstructing the CSI matrix fully. One Refine net are consisted of three convolutional layer with $3{\times}3$ convolution kernels. Three layers generate eight, sixteen and two feature maps respectively and we used Leaky ReLU [16] as activation function. Moreover, We add attention mechanism in first and second convolution layer which can help reconstruct CSI matrix. Because the residual network [17] performs well in image classification and other computer vision task, we remained the residual structure in refineNet which can avoid gradient vanish problem and remain more information in deep networks. The final output layer is activated with sigmoid function which can scale values into $[0, 1]$.

To get better performance, we use end-to-end learning to train all parameters for encoder and decoder network. The input data are normalized into the $[0, 1]$ range. So the whole network can be defined as:

$$\widehat{\mathbf{H}} = f(\mathbf{H}; \Theta) \triangleq f_{CNN}(f_{LSTM}(\mathbf{H}; \Theta_{en}); \Theta_{de}), \qquad (5)$$

where all the parameters of neural networks denote as $\Theta = \{\Theta_{en}, \Theta_{de}\}$ and the loss function of our network is the mean square error (MSE) , which can be defined as follows:

$$L(\Theta) = \frac{1}{T}\sum_{i=1}^{T}\left\|\widehat{\mathbf{H}} - \mathbf{H}\right\|_2^2, \qquad (6)$$

where the T is the batch size every time we train the neural network and norm $\|\cdot\|_2$ is the Euclidean norm. We choose ADAM [18] gradient descent optimizer to update all the parameters.

IV. Simulation results and Analysis

In order to compare with CsiNet, we use the same COST 2100 [19] model to simulate MIMO channels and generate training, validation and testing CSI matrix samples. Our system model work on a 20MHz bandwidth with $N_c = 1024$ subcarriers and use ULA with 32 antennas at the BS and 1 antenna at UE. We set the $\widetilde{N}_c = N_t = 32$ and the size of $\mathbf{H}$ is 32×32. There are two scenario in our results: the indoor scenario at 5.3 GHz with UE velocity $v = 0.0036km/h$ and the outdoor scenario at 300 MHz with UE velocity $v = 3.24km/h$. And we set $C = L/2$ in attention module. The training, validation, and testing sets contain 100,000, 30,000, and 20,000 samples, respectively. Our network will not be trained with validation samples and testing samples just be tested on these samples. We set the learning rate and batch size as 0.001 and 200 respectively. In order to avoid overfitting, we used early stopping so that the epochs varies from 700 to 1000 in every training step. All the parameter showed in the Table I.

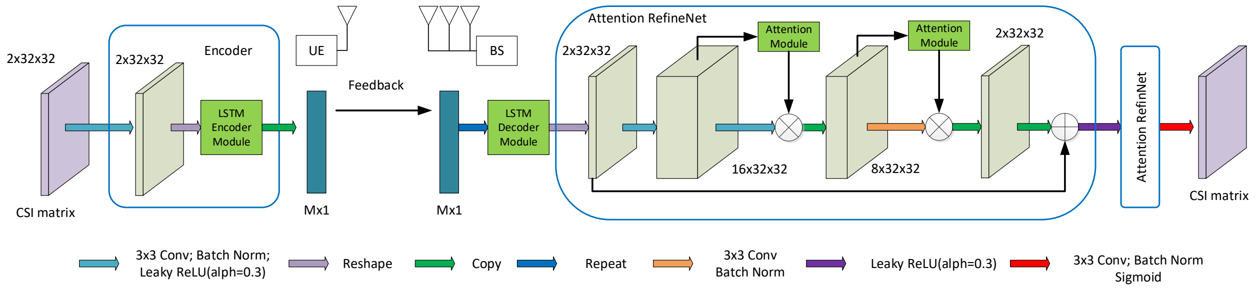


Fig. 3. The structure of Attention-CsiNet and Feedback Mechenism.

TABLE I
SIMULATION PARAMETERS

Parameter	Indoor	Outdoor
Bandwidth	20MHz	
Subcarriers	1024	
UE velocity	0.0036km/h	3.24km/h
Frequecny	5.3GHz	300MHz
Antenna	32 × 1	
Batch size	200	
Learning Rate	0.001	

We compared our model with CsiNet and other three classic compressed sensing based methods, LASSO [3], TVAL3 [4], and BM3D-AMP [5]. Among these methods, CsiNet get good performance which is the first deep learning based method in CSI feedback. LASSO uses simple sparsity priors but achieves good performance. TVAL3 provides great recovery quality but with high computing load. BM3D-AMP achieves worse recovery performance than other methods, but runs faster than other CS-based iterative methods.

We achieve our model via open source deep learning structure Keras. The results of conventional CS-based methods are come from the work in [10] which is completed on Intel CoreTM i7-6700 CPU. We simulated our model and CsiNet on Nvidia GeForce GTX 1080 Ti GPU.

In order to compare the performance of different methods, we use Normalized MSE (NMSE) to explicit the recovery performance which is defined at below:

$$NMSE = E\{\frac{1}{N}\sum_{n=1}^{1}\left\|\mathbf{H}_n - \widehat{\mathbf{H}}_n\right\|_2^2 / \|\mathbf{H}_n\|_2^2\}. \tag{7}$$

We also use the cosine similarity to compare different method which is depicted as follows:

$$\rho = E\left\{\frac{1}{N_c}\sum_{i=1}^{N_c}\frac{\left|\widehat{h}_i^H h_i\right|}{\left\|\widehat{h}_i\right\|_2 \|h_i\|_2}\right\}, \tag{8}$$

where $\widehat{h}_i$ denotes the reconstructed channel vector of the i_{th} subcarrier. The CSI feedback can be used for computing beamforming vector. The BS can use $\mathbf{v}_i$ as a beamforming vector which is defined below so that ρ can be used to indicate beamforming gain.

$$\mathbf{v}_i = \widehat{h}_i / \left\|\widehat{h}_i\right\|_2 \tag{9}$$

The results of comparison of NMSE and ρ are showed in Table II. From the table, we can see that our model and CsiNet perform significantly better than other CS-based method. Neither in indoor nor outdoor scenario, our model also outperforms CsiNet 1-3dB in NMSE and can obtain state-of-the-art among all the methods. When CR=1/64 or 1/32, our model can even perform better than CsiNet with CR=1/32 or 1/16. In Fig. 4, there are some pseudo-gray plots of reconstruction samples at different compression ratios. Both CsiNet and our model can get good performance but ours are better more.

TABLE II
SIMULATION RESULTS(NMSE IN DB AND COSINE SIMILARITY ρ)

CR	Method	Indoor		Outdoor	
		NMSE	ρ	*NMSE*	ρ
1/4	LASSO	-7.59	0.91	-5.08	0.82
	BM3D-AMP	-4.33	0.80	-1.33	0.52
	TVAL3	-14.87	0.97	-6.90	0.88
	CsiNet	-17.36	0.99	-8.75	0.91
	Attention CsiNet	**-20.29**	**0.99**	**-10.43**	**0.94**
1/16	LASSO	-2.72	0.70	-1.01	0.46
	BM3D-AMP	0.26	0.16	0.55	0.11
	TVAL3	-2.61	0.66	-0.43	0.45
	CsiNet	-8.65	0.93	-4.51	0.79
	Attention CsiNet	**-10.16**	**0.95**	**-6.11**	**0.85**
1/32	LASSO	-1.03	0.48	-0.24	0.27
	BM3D-AMP	24.72	0.04	22.66	0.04
	TVAL3	-0.27	0.33	0.46	0.28
	CsiNet	-6.24	0.89	-2.81	0.67
	Attention CsiNet	**-8.58**	**0.93**	**-4.57**	**0.79**
1/64	LASSO	0.14	0.22	-0.06	0.12
	BM3D-AMP	0.22	0.04	25.45	0.03
	TVAL3	0.63	0.11	0.76	0.19
	CsiNet	-5.84	0.87	-1.93	0.59
	Attention CsiNet	**-6.32**	**0.89**	**-3.27**	**0.71**

Moreover, we compared the convergence speed of our model with CsiNet. We choosed the outdoor scenario with CR=1/64. From the Fig. 5, we can see that our network can converge more quickly and smoothly.

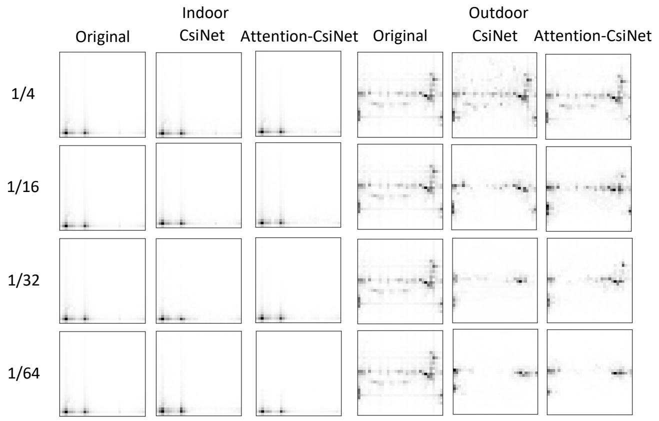


Fig. 4. Reconstruction images for different compression ratios by CsiNet and Attention-CsiNet in indoor and outdoor scenarios.

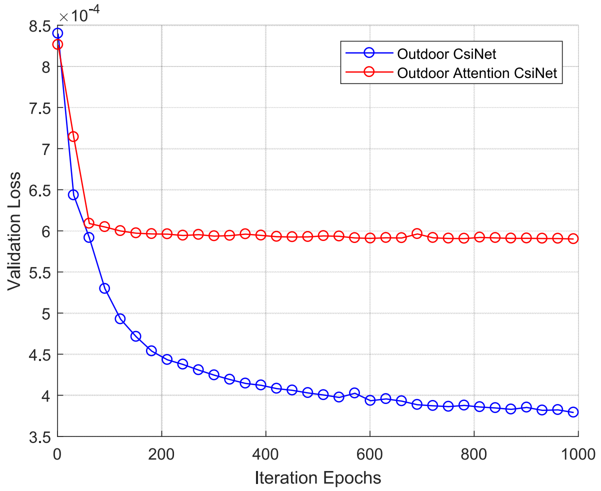


Fig. 5. The convergence speed between CsiNet and Attention-CsiNet

Finally, we find methods based on deep learning can save much time which benefits from GPU acceleration. Because of the feedforward and fast matrix vector computation, DL-based methods perform approximately thousands of times faster than other CS-based algorithms. The results are showed in Table III.

TABLE III
COMPLEXITY/RUNTIME COMPARISON IN OUTDOOR/INDOOR SCENARIOS

	CR	LASSO	BM3D-AMP	TVAL3	CsiNet	Attention-CsiNet
Out	1/16	0.2471	0.3454	0.3148	0.0001	0.0002
	1/32	0.2137	0.5556	0.3148	0.0001	0.0002
	1/64	0.2479	0.6047	0.2860	0.0001	0.0002
In	1/16	0.2122	0.4210	0.3145	0.0001	0.0002
	1/32	0.2409	0.6031	0.2985	0.0001	0.0002
	1/64	0.0166	0.5980	0.2850	0.0001	0.0002

V. CONCLUSION

In this paper, an improved compressed sensing attention recovery model have been proposed for Massive MIMO CSI feedback which can achieve state-of-the-art. We extend the CsiNet with attention mechanism and use bi-LSTM replace fully connected encoder. Our attention model achieved remarkable recovery performance and reduced training steps. It can be used for CSI feedback in communication system to replace traditional method with CQI/PMI/RI feedback which can only send back limited information to the BS. We believe that this attention model has the potential for being used on real systems. We hope this study can help those interested in this direction.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 61601047, 61671080, 61771066), and Science and Technology on Communication Networks Laboratory Open Project.

REFERENCES

[1] P. H. Kuo, H. Kung, and P. A. Ting, Compressive sensing based channel feedback protocols for spatially-correlated massive antenna arrays, In Proc. IEEE WCNC, Shanghai, China, Apr. 2012, pp. 492C497.

[2] X. Rao and V. K. Lau, Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems, IEEE Trans. Signal Process, vol. 62, no. 12, pp. 3261C3271, Jun. 2014.

[3] I. Daubechies, M. Defrise, and C. D. Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, Comm. Pure and Applied Math, vol. 75, pp. 1412C1457, 2004.

[4] C. Li, W. Yin, and Y. Zhang, Users guide for tval3: Tv minimization by augmented lagrangian and alternating direction algorithms, CAAM report, vol. 20, pp. 46C47, 2009.

[5] C. A. Metzler, A. Maleki, and R. G. Baraniuk, From denoising to compressed sensing, IEEE Trans. Inf. Theory, vol. 62, no. 9, pp. 5117C5144, 2016.

[6] A. Mousavi et al. "A Deep Learning Approach to Structured Signal Recovery," arXiv:1802.04741, 2018

[7] H. Yao et al., DR2-Net: Deep residual reconstruction network for image compressive sensing, arXiv:1702.05743, 2017

[8] K. Xu and F. Ren, Csvideonet: A real-time end-to-end learning framework for high-frame-rate video compressive sensing, In Proc. IEEE WACV, NV, USA, Mar. 2018, pp. 1680C1688.

[9] T. N. Canh and B. Jeon, Multi-Scale Deep Compressive Sensing Network, arXiv:1809.05717, 2018

[10] C. K. Wen, W. T. Shih, and S. Jin, Deep learning for massive MIMO CSI feedback, IEEE Wireless Commun. Lett, to be published, DOI10.1109/LWC.2018.2818160

[11] S. Hochreiter and J. Schmidhuber. "Long short-term memory," Neural computation, 9(8):1735C1780, 1997.

[12] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate," arXiv preprint arXiv:1409.0473, 2014.

[13] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks, arXiv preprint arXiv:1709.01507, 2017.

[14] Jeffrey L. Elman. "Finding structure in time," Cognitive Science, 14(2):179 C 211, 1990.

[15] S. Ioffe and C. Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift," In ICML, 2015.

[16] MAAS, A. L., HANNUN, A. Y., AND NG, A. Y. "Rectifier nonlinearities improve neural network acoustic models," In Proc. ICML, vol. 30, 2013.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," arXiv:1512.03385, 2015.

[18] Kingma, Diederik P and Ba, Jimmy Lei. "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[19] L. Liu et al., The COST 2100 MIMO channel model, IEEE Wireless Commun, vol. 19, no. 6, pp. 92C99, Dec. 2012.