Transformer Empowered CSI Feedback for Massive MIMO Systems

Yang Xu[1], Mingqi Yuan[1] and Man-On Pun[1, 2,†]
[1]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China, 518172
[2]Shenzhen Research Institute of Big Data, Shenzhen, China, 518172

2021 30th Wireless and Optical Communications Conference (WOCC) | 978-1-6654-2772-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/WOCC53213.2021.9602863

***Abstract*—This work investigates the problem of CSI feedback in massive multiple-input multiple-output (MIMO) systems. It is well-known that accurate CSI plays a crucial role in realizing the beamforming gain promised by the MIMO technology. However, CSI feedback often incurs excessive feedback overhead. To cope with this problem, this work proposes an effective and robust CSI feedback scheme called *CsiTransformer* by leveraging a recently developed machine learning (ML) model named *transformer* in its encoder and decoder. In contrast to the existing ML approaches, CsiTransformer can more effectively exploit the correlation among elements in the channel matrix, which leads to improved CSI reconstruction quality. Furthermore, motivated by the fact that cellphones are power-limited, we propose a further reduced-complexity scheme called *MixedCsiNet* by using the less computationally expensive conventional convolutional neural networks (CNNs) as the encoder while the transformer model as the decoder. Extensive computer simulation is performed to confirm that CsiTransformer and MixedCsiNet are able to achieve substantially better CSI feedback at different compression rates as compared to existing CSI feedback schemes.**

***Index Terms*—Multiple-input multiple-output (MIMO), Channel State Information (CSI), Machine Learning, Transformer.**

I. Introduction

The sixth-generation (6G) wireless communication systems have been envisaged to provide high-quality data services simultaneously to a large number of devices. To accomplish this demanding goal, massive multiple-input multiple-output (MIMO) that endows base stations (BSs) with a massive number of antennas is well regarded as one of the most promising technologies for 6G. By fully exploiting the channel state information (CSI), MIMO can perform sophisticated beamforming, precoding and interference suppression, leading to significantly improved network throughput. The acquisition of CSI is usually performed in a training process using pilot signals. More specifically, the user equipment (UE) receives the pilot signals transmitted from its serving BS before returning its estimated CSI to the BS. Unfortunately, the CSI feedback incurs excessive feedback overhead that increases with the numbers of antennas, receivers and subcarriers, which has become a major obstacle in deploying massive MIMO systems in practice.

To reduce the CSI feedback overhead, compressive sensing (CS) was introduced to devise CSI feedback protocols with compressive channel estimation by exploiting the temporal and spatial correlation of CSI [1]. Upon receiving the compressed CSI, the BS utilizes the LASSO ℓ_1-solver [2] and the AMP algorithm [3] to recover the compressed CSI using iterative thresholding and the ℓ_1 penalty. However, these algorithms make use of oversimplified assumptions on a simple sparsity prior and a perfectly sparse channel matrix, which can be an issue of concern in practice. Recently, more advanced algorithms such as TVAL3 [4] and BM3D-AMP [5] were proposed to capitalize on elaborate priors in CSI recovery. Unfortunately, the quality of the recovered CSI was shown to be highly sensitive to the accuracy of the priors [4], [5], which renders these algorithms impractical. In addition to the aforementioned challenges, CS-based methods assume that the channel can be transformed into sparse forms in some bases. However, in practice, channels are usually not sparse or even mathematically inexplicable in many real-world scenarios. Furthermore, the random projection commonly employed in CS cannot fully comprehend the channel matrix structure, resulting in information loss and poor performance in CSI recovery. Finally, most CS-based methods are computationally inefficient as they require iterative algorithms for CSI recovery.

To cope with these challenges, the machine learning (ML) approach has been proposed for CSI feedback [6], [7]. In sharp contrast to the CS-based approach, the ML approach exploits massive data retrieved from the network without explicitly deriving a mathematical model to compress CSI. In [6], the deep neural network (DNN) was leveraged to conduct CSI encoding in a closed-loop MIMO system. Simulation results showed that the DNN-based encoder outperformed the traditional singular value decomposition (SVD)-based encoder for MIMO spatial multiplexing systems. In [7], both CSI encoding and recovery were achieved by using a novel DNN model called *CsiNet*. In particular, CsiNet uses convolutional neural networks (CNNs) to encode and decode CSI by leveraging similar techniques commonly adopted in image reconstruction. The resulting CsiNet can significantly improve the CSI reconstruction quality, even at low compression rates. Despite its good performance, CsiNet suffers from large computational complexity due to its excessive parameters. This problem becomes even more prominent as the numbers of antennas, receivers and subcarriers increase, which incurs prohibitively expensive computational complexity for training CsiNet. Finally, CsiNet was not capable of capturing the long-term dependency across data samples fed into CNN.

This work was supported by National Key Research and Development Program of China (2020YFB1807700).

† Corresponding author, email: SimonPun@cuhk.edu.cn.

Inspired by the discussions above, we propose to leverage a new DNN model called *transformer* [8] to achieve robust CSI feedback with reduced feedback overhead. More specifically, the transformer was originally developed in [9] to overcome the drawbacks of recurrent neural networks (RNNs) for natural language processing (NLP) by characterizing both short-distance and long-distance dependencies among data samples using a self-attention mechanism [9]. As a result, transformer can significantly outperform the traditional CNN and RNN in semantic feature extraction, long-distance feature capture and comprehensive feature extraction. Since the transformer was first proposed, it has been successfully applied in computer vision applications with improved model accuracy and reduced computation complexity. The main contributions of this work are summarized as follows:

- We propose a novel transformer-based CSI feedback scheme called *CsiTransformer* for efficient and robust CSI feedback. Empowered by the transformer model, CsiTransformer consists of an encoder and a decoder that are responsible for CSI compression and reconstruction, respectively;
- To reduce the computational complexity imposed on the encoder, we propose a mixed CNN-and-transformer feedback scheme called MixedCsiNet by adopting CNN-based encoder while keeping the transformer-based decoder design;
- Extensive simulation results confirm that the proposed CsiTransfomer and MixedCsiNet achieve high-quality CSI feedback at all compression rates tested.

Notation: Uppercase boldface and lowercase boldface letters are used to denote matrices and vectors, respectively. $\boldsymbol{A}^T$ and $\boldsymbol{A}^H$ are the transpose and conjugate transpose of $\boldsymbol{A}$, respectively. In addition, $\|\boldsymbol{A}\|$ stands for the ℓ_2 norm of $\boldsymbol{A}$.

II. System Model and Problem Formulation

A. System Model

Similar to [7], we consider a single-cell downlink massive MIMO system, in which the CSI feedback is sent from a single-antenna UE to a BS equipped with N_t antennas. Furthermore, we assume that the system operates in the frequency division duplexing (FDD) mode with $\tilde{N}_c$ subcarriers. Denote by $x \in \mathbb{C}$ the data symbol, the received signal at the n-th subcarrier is given by:

$$y_n = \tilde{\mathbf{h}}_n^H \mathbf{v}_n x_n + \epsilon_n, \tag{1}$$

where $\tilde{\mathbf{h}}_n \in \mathbb{C}^{N_t \times 1}, \mathbf{v}_n \in \mathbb{C}^{N_t \times 1}$ and $\epsilon_n \in \mathbb{C}$ are the channel vector, precoding vector and additive noise of the n-th subcarrier, respectively.

We then define the CSI stack matrix in the spatial frequency domain as

$$\tilde{\mathbf{H}} = [\tilde{\mathbf{h}}_1, \cdots, \tilde{\mathbf{h}}_{\tilde{N}_c}]^H \in \mathbb{C}^{\tilde{N}_c \times N_t}. \tag{2}$$

After the UE derives $\tilde{\mathbf{H}}$ by exploiting the pilot signals, the UE is required to return $\tilde{\mathbf{H}}$ to the BS through feedback links in a timely manner.

To reduce feedback overhead, we follow the compression method proposed in [7] to sparsify $\tilde{\mathbf{H}}$ in the angular-delay domain using the 2D discrete Fourier transform (DFT) as follows:

$$\mathbf{H}' = \mathbf{F}_d \tilde{\mathbf{H}} \mathbf{F}_a, \tag{3}$$

where $\mathbf{F}_d$ and $\mathbf{F}_a$ are $\tilde{N}_c \times \tilde{N}_c$ and $N_t \times N_t$ DFT matrices, respectively.

We assume that the first $N_c \leq \tilde{N}_c$ rows of $\mathbf{H}'$ have non-negligible values. Recalling that $\mathbf{H}'$ is complex, we can retain the first N_c rows of $\mathbf{H}'$ before concatenating the real and imaginary part of the truncated matrix to form a real-valued matrix $\mathbf{H}$ of dimension $2N_c \times N_t$. $\mathbf{H}$ is fed into the encoder as shown in Fig. 1.

B. Problem Formulation

In this work, we consider building an encoder f_e and a decoder f_d responsible for CSI encoding and recovery, respectively. More specifically, the encoder transforms the channel matrix $\mathbf{H}$ into a codeword:

$$\mathbf{s} = f_e(\mathbf{H}), \tag{4}$$

where $\mathbf{s} \in \mathbb{R}^{M \times 1}$ with $M < 2N_cN_t$. This results in a compression ratio of $\eta = \frac{M}{2N_cN_t}$. The codeword $\mathbf{s}$ is then sent to the BS. Assuming that $\mathbf{s}$ is perfectly received, the BS recovers the original channel matrix using the following decoder:

$$\widehat{\mathbf{H}} = f_d(\mathbf{s}). \tag{5}$$

Equipped with the definitions above, we are ready to formulate the optimization problem:

$$\underset{f_d, f_e \in \mathcal{F}}{\operatorname{argmin}} \|\mathbf{H} - \widehat{\mathbf{H}}\|^2, \tag{OP1}$$

where $\mathcal{F}$ is the enclosed set of functions defined in $\mathbb{R}$.

Unfortunately, it is non-trivial to find the optimal f_e^* and f_d^* through conventional model-based or existing ML-based approaches. The following section proposes novel encoder and decoder functions by leveraging the transformer model.

III. Proposed Schemes

A. CsiTransformer

Fig. 1 shows the proposed CSI feedback architecture coined as *CsiTransformer*. We begin with the discussions on the encoder inside the transmitter. $\mathbf{H}$ enters into the transformer layer whose output is a matrix of size $S_1 \times S_2$. After that, the matrix is converted into a vector of length S_1S_2 before the vector is fed into a fully-connected layer. The resulting encoded vector $\mathbf{s}$ is of length M with $M \leq S_1S_2$.

Fig. 2 depicts the detailed structure of the transformer layer modified based on the model proposed from [8] inside the encoder in which $\mathbf{H}$ is first passed through a 1-by-1 convolution layer and reshaped into a matrix of enlarged size $4N_c \times N_t$ with $d = 2N_c$. Next, the enlarged matrix is entered into a multi-head self-attention layer in which different elements in $\mathbf{H}$ are jointly attended. After that, the input and output of the self-attention layer are added together and

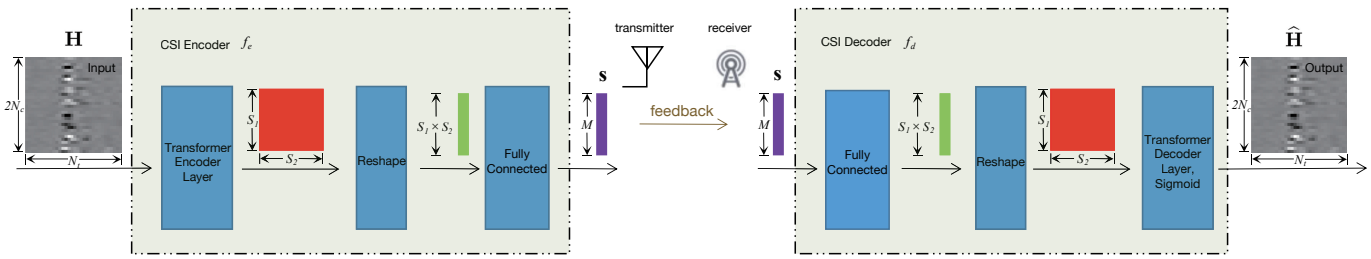

Fig. 1. Architecture of the proposed CsiTransformer with transformer-based encoder and decoder.

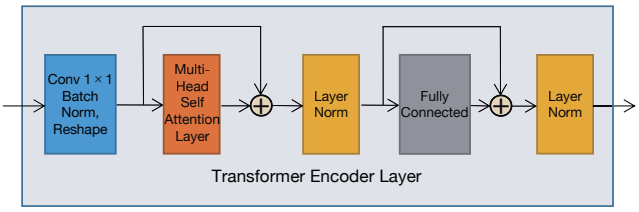

Fig. 2. Structure of the transformer layer inside the *encoder*

subsequently normalized. The normalized data is then fed into a fully-connected layer for linear transformation. Finally, the input and output of the fully-connected layer are also summed together and normalized. The output of the transformer layer is a matrix of size $S_1 \times S_2$.

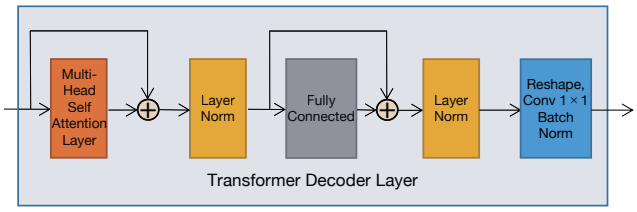

Fig. 3. Structure of the transformer layer inside the *decoder*

Next, we discuss the decoder structure inside the receiver. For simplicity of presentation, we assume that the received signal is noise-free. The more general case for noisy received signal can be extended from our following discussions in a straightforward manner. Under such a noiseless assumption, the input into the decoder is the compressed codeword $\mathbf{s}$. As shown in Fig. 1, the first layer of the decoder includes a fully-connected layer that takes $\mathbf{s}$ as input and outputs a vector of length $S_1 S_2$. The vector is then converted into a matrix of size $S_1 \times S_2$ before the matrix is fed into the transformer decoder layer. The transformer decoder layer as shown in Fig. 3 has the same structure with the encoder, except that the reshaping operation and convolution operation are performed at the end of the layer. The operations are designed to reduce the matrix size from $S_1 \times S_2$ to $2N_c \times N_t$. The output of the transformer layer is scaled to $[0, 1]$ by a sigmoid function.

End-to-end training is employed to train all the networks in the encoder and the decoder shown in Fig. 1. Network parameters are updated with the AdamW algorithm using the following loss function known as the normalized mean squared error (NMSE) [7]:

$$\text{NMSE} = E\Big\{\frac{\|\mathbf{H} - \widehat{\mathbf{H}}\|^2}{\|\mathbf{H}\|^2}\Big\}. \tag{6}$$

B. MixedCsiNet

In the CSI feedback problem, the transmitter is usually battery-powered. Thus, it is more desirable to have an encoder of lower computational complexity. Considering a task of embedding a sequence of length n into a d-dimensional vector, the computational complexity of using a transformer layer is estimated as $O(n^2 \cdot d + n \cdot d^2)$ while using a convolutional layer $O(n \cdot d^2)$. Motivated by this observation, we propose a variant entitled *MixedCsiNet* by leveraging one convolutional layer to replace the transformer layer of the encoder in CsiTransformer. Fig. 4 illustrates the architecture of MixedCsiNet in which the reshaped $\mathbf{H}$ is input into a convolutional layer composed of 1×1 kernels. The output of the convolutional layer contains $S_3 = 2$ feature maps of height S_1 and width S_2. After being converted into a vector of length $S_1 S_2 S_3$, the feature maps are loaded into a fully-connected layer to generate the compressed codeword $\mathbf{s}$. The decoder of MixedCsiNet is the same as that of CsiTransformer.

TABLE I
COMPARISON ON ARCHITECTURES

Method	Encoder	Decoder
CS-CsiNet [7]	CS-based Random Projection	CNN
CsiNet [7]	CNN ($n = d = 32$)	CNN
CsiTransformer	Transformer ($n = 32, d = 64$)	Transformer
MixedCsiNet	CNN ($n = d = 32$)	Transformer

Table I compares the four machine learning-based CSI feedback schemes discussed above. In particular, CS-CsiNet proposed in [7] uses random linear projections for encoding and a CNN-based decoder. In contrast, CsiNet uses CNN in both its encoder and decoder. Furthermore, the proposed

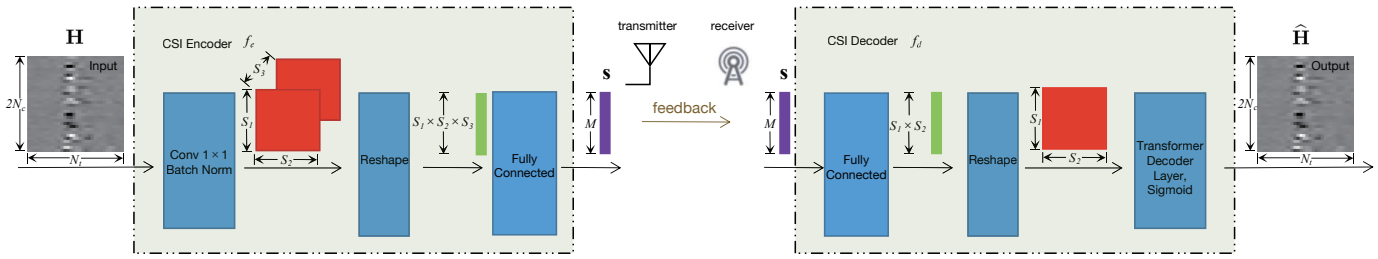

Fig. 4. Architecture of MixedCsiNet with CNN encoder and transformer decoder.

CsiTransformer employs the transformer model in both its encoder and decoder. Finally, the proposed MixedCsiNet adopts the CNN structure in its encoder to reduce its computational complexity while using the transformer layer in its decoder to harvest performance gains.

IV. Experiments & Numerical Results

We use the datasets provided online [10] as our training and test datasets whose sizes are $256,000$ and $64,000$ samples, respectively. The data was generated with a BS equipped with a uniform linear array (ULA) of $N_t = 32$ antennas and $\tilde{N}_c = 256$ subcarriers [10]. The system operates in the 3.5 GHz frequency band. Furthermore, only the first 16 rows of the channel matrix were retained in the angular-delay domain. In other words, $\mathbf{H}$ is of the size of 32×32 after being reshaped into a 2D real-valued matrix. In addition, the training was performed with a batch size of 32 and 40 epochs. Finally, the learning rate was initially set to 0.001 and reduced to 0.0001 after the first 30 epochs.

Besides using NMSE defined in Eq. (6) to quantify the difference between the original $\mathbf{H}$ and the recovered $\widehat{\mathbf{H}}$, we also use the following cosine similarity defined in [7] to evaluate the performance of different feedback schemes:

$$\rho = E\left\{\frac{1}{\tilde{N}_c}\sum_{n=1}^{\tilde{N}_c}\frac{|\widehat{\tilde{\mathbf{h}}}_{\mathbf{n}}^H \tilde{\mathbf{h}}_{\mathbf{n}}|}{\|\widehat{\tilde{\mathbf{h}}}_{\mathbf{n}}\| \cdot \|\tilde{\mathbf{h}}_{\mathbf{n}}\|}\right\}, \tag{7}$$

where $\widehat{\tilde{\mathbf{h}}}_n$ represents the recovered channel vector of the n-th subcarrier. Note that $|\widehat{\tilde{\mathbf{h}}}_{\mathbf{n}}^H \tilde{\mathbf{h}}_{\mathbf{n}}|/\|\widehat{\tilde{\mathbf{h}}}_{\mathbf{n}}\|\|\tilde{\mathbf{h}}_{\mathbf{n}}\|$ measures the correlation between $\widehat{\tilde{\mathbf{h}}}_{\mathbf{n}}$ and $\tilde{\mathbf{h}}_{\mathbf{n}}$.

The performance of all fours schemes discussed is compared in terms of NMSE and ρ in Table II with the best results being highlighted. Inspection of Table II reveals that the proposed transformer-based schemes outperformed the conventional CS-CsiNet and CsiNet for all compression rates between $1/32$ and $1/4$. In particular, CsiTransformer showed the best performance in terms of both cosine similarity and NMSE for all compression rates we tested. Fig. 5 provides visual comparison on the CSI recovery performance at different compression rates using the four schemes. In particular, the recovered channels provided by CS-CsiNet showed noticeable distortions as the compression rate decreased. Fig. 6 shows the NMSE performance as a function of epochs during training at compression rate of $1/32$. From Fig. 6, we can observe that the NMSE of all three schemes shown decreased smoothly and steadily. In particular, the NMSE performance for CsiTransformer significantly outperformed CsiNet. Note that the decrease in NMSE at the 30-th epoch was caused by the adjustment of the learning step size.

TABLE II
Performance comparison in terms of NMSE and ρ

Compression Rate	Scheme	NMSE	ρ
1/4	CS-CsiNet	0.0682	0.964
	CsiNet	0.0203	0.989
	CsiTransformer	**0.0069**	**0.996**
	MixedCsiNet	0.0100	0.995
1/8	CS-CsiNet	0.1996	0.890
	CsiNet	0.0598	0.969
	CsiTransformer	**0.0339**	**0.982**
	MixedCsiNet	0.0440	0.977
1/16	CS-CsiNet	0.4701	0.716
	CsiNet	0.1368	0.925
	CsiTransformer	**0.1055**	**0.943**
	MixedCsiNet	0.1286	0.930
1/32	CS-CsiNet	0.5732	0.626
	CsiNet	0.3401	0.825
	CsiTransformer	**0.2214**	**0.875**
	MixedCsiNet	0.2635	0.849

TABLE III
Average runtime

Compression Rate	Scheme	Average Running Time
1/16	CsiNet	3.01ms
	CsiTransformer	3.02ms
	MixedCsiNet	2.89ms
1/32	CsiNet	2.89ms
	CsiTransformer	2.94ms
	MixedCsiNet	2.36ms

Finally, we measured the average runtime of CsiNet, CsiTransformer and MixedCsiNet using our test dataset under compression rates of $1/16$ and $1/32$. The results in Table III confirmed that MixedCsiNet achieved the highest speed while CsiTransformer outperformed CsiNet in terms of runtime.

V. Conclusion

In this work, we have introduced the transformer model into the CSI feedback scheme by proposing two efficient CSI

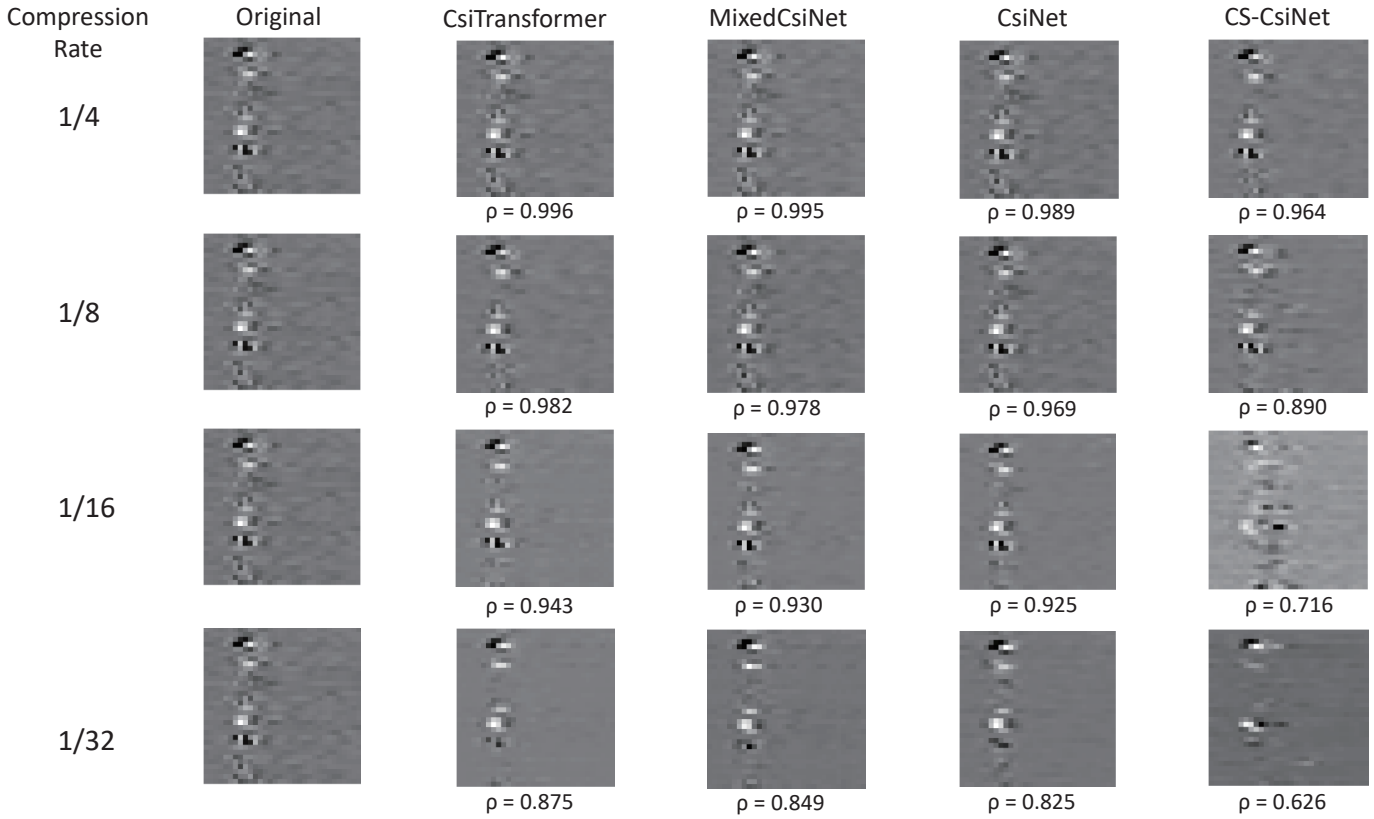

Fig. 5. Reconstruction channel images of different compressing ratios by different algorithms

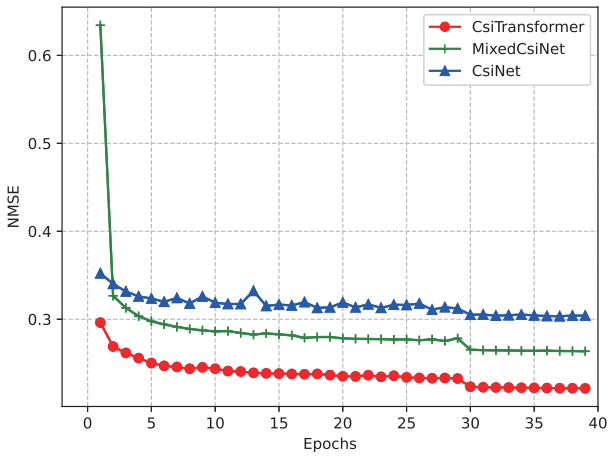

Fig. 6. The NMSE performance as a function of epochs during training for CsiTransformer, CsiNet and MixedCsiNet at a compression rate of $1/32$

feedback schemes for compression and recovery of the MIMO channel matrix, namely CsiTransformer and MixedCsiNet. Both schemes achieved significantly better performance than the original CNN-based CsiNet at all compression ratios we tested. In particular, our experiment results have suggested that the proposed CsiTransformer can achieve higher recovery accuracy while MixedCsiNet performs best in terms of runtime.

REFERENCES

[1] X. Rao and V. K. Lau, "Distributed compressive CSIT estimation and feedback for FDD multi-user massive mimo systems," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3261–3271, 2014.

[2] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 57, no. 11, pp. 1413–1457, 2004.

[3] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18914–18919, 2009.

[4] C. Li, W. Yin, and Y. Zhang, "Tval3: Tv minimization by augmented lagrangian and alternating direction agorithm 2009," 2013.

[5] C. A. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 5117–5144, 2016.

[6] T. J. O'Shea, T. Erpek, and T. C. Clancy, "Deep learning based MIMO communications," *arXiv preprint arXiv:1707.07980*, 2017.

[7] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, 2018.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[9] A. Vaswani, S. Bengio, E. Brevdo, F. Chollet, A. N. Gomez, S. Gouws, L. Jones, Ł. Kaiser, N. Kalchbrenner, N. Parmar, *et al.*, "Tensor2tensor for neural machine translation," *arXiv preprint arXiv:1803.07416*, 2018.

[10] National Mobile Communications Research Laboratory, Southeast University. http://www.china-ai.ac.cn/, 2020.