This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2021.3076504, IEEE Communications Letters

Lightweight Convolutional Neural Networks for CSI Feedback in Massive MIMO

Zheng Cao, Wan-Ting Shih, Jiajia Guo, Chao-Kai Wen, *Senior Member, IEEE*, Shi Jin, *Senior Member, IEEE*

***Abstract*—In frequency division duplex mode of massive multiple-input multiple-output systems, the downlink channel state information (CSI) must be sent to the base station (BS) through a feedback link. However, transmitting CSI to the BS is costly due to the bandwidth limitation of the feedback link. Deep learning (DL) has recently achieved remarkable success in CSI feedback. Realizing high-performance and low-complexity CSI feedback is a challenge in DL-based communication. We develop a DL-based CSI feedback network in this study to complete the feedback of CSI effectively. However, this network cannot be effectively applied to the mobile terminal due to its excessive number of parameters and high computational complexity. Therefore, we further propose a new lightweight CSI feedback network based on the developed network. Simulation results show that the proposed CSI network maintains a few parameters and parameter complexity while exhibiting better reconstruction performance than existing works. These findings suggest the feasibility and potential of the proposed techniques.**

***Index Terms*—Massive MIMO, FDD, CSI feedback, deep learning, lightweight neural network.**

I. Introduction

Massive multiple-input multiple-output (MIMO) systems are widely regarded as the main technology of 5G wireless communication systems [1]. These systems refer to a communication system that uses hundreds of antennas at the same time-frequency resource to serve tens of user equipment (UE) simultaneously. In frequency division duplex (FDD) mode, a massive MIMO system must feedback the downlink channel state information (CSI) to the base station (BS) to realize its potential gain. However, transmitting CSI to the BS is costly due to the bandwidth limitation. The conventional compressive sensing-based methods have several shortcomings. First, they are difficult to select an appropriate sparsifying basis to guarantee high reconstruction accuracy. Second, the channel structure is not fully utilized in these schemes. Finally, some iterative-based CS reconstruction methods have low reconstruction speed [2].

In recent years, deep learning (DL) technology has been widely applied in the field of wireless communication [3], [4], [5]. The authors in [2] used DL technology to build a new type of CSI compression and recovery neural network called CsiNet. This network can learn how to use the channel architecture effectively to convert from CSI to codeword and vice versa. The reconstruction performance of CsiNet is superior to that of traditional compressed sensing methods. Subsequent related studies expanded the original scope of the network. [6] proposed a novel neural network based on multi-resolution architecture. [7] proposed a DL-based novel CSI feedback scheme by utilizing non-local block and dense connectivity. [8] proposed a multiple-rate DL-based CSI feedback framework that can switch under different compression ratios. The DL-based CSI feedback is further evaluated under practical channel measurements in [9].

Previous studies have shown that the CSI feedback network architecture is roughly divided into feedforward neural networks such as CsiNet and convolutional neural networks (CNN). Given that CSI has an image structure, CNNs are more suitable for CSI feedback because they handle images more efficiently and can adapt to inputs of different sizes. In this study, we use an autoencoder to develop the network structure and complete the CSI feedback, which improves the reconstruction performance and can adapt to input size with different antennas and subcarriers. However, it is challenging to implement this CSI feedback network on a mobile terminal because it is based on the CNN architecture and has high computational complexity and many parameters. Therefore, we adopted a lightweight structure based on the developed network to build a new structure to reduce the computational complexity and number of parameters.

The contribution of this work is as follows. We propose a DL-based CSI feedback structure called ConvCsiNet for FDD MIMO systems. This structure uses convolutional layers to extract features, while the mean-pooling and upsampling layers are used to compress and expand the size of the matrix. Considering mobile devices' application in modern communications, we also propose a DL-based CSI feedback lightweight structure called ShuffleCsiNet. This structure can acquire accurate downlink CSI while consuming low memory space and core computing power. The experimental results show that ConvCsiNet is superior to CsiNet and existing works considering reconstruction performance. Moreover, ConvCsiNet improves the network performance at the cost of complexity while adapting to input size with different antennas and subcarriers. Compared with ConvCsiNet, the parameter number and algorithm complexity of ShuffleCsiNet are greatly

The work was supported in part by the National Key Research and Development Program 2018YFA0701602, the National Natural Science Foundation of China (NSFC) for Distinguished Young Scholars with Grant 61625106, and the NSFC under Grant 61941104. The work of W.-T. Shih and C.-K. Wen was supported in part by the Ministry of Science and Technology of Taiwan under grant MOST 108-2628-E-110-001-MY3. (Corresponding author: Shi Jin.)

Z. Cao, J. Guo and S. Jin are with the National Mobile Communications Research Laboratory, Southeast University, Nanjing, 210096, P. R. China (e-mail: {zheng_cao, jiajiaguo, jinshi}@seu.edu.cn).

W.-T. Shih is with the Institute of Electrical Control Engineering, National Chiao Tung University, Hsinchu 30010, Taiwan (email: sydney2317076@gmail.com).

C.-K. Wen is with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan (e-mail: chaokai.wen@mail.nsysu.edu.tw).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2021.3076504, IEEE Communications Letters

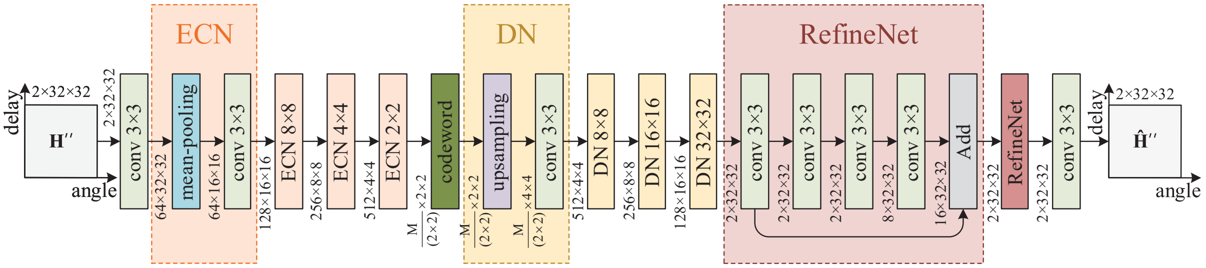

Fig. 1. Architecture of ConvCsiNet: An encoder structure constructed with convolutional layers and ECN units, which include mean-pooling and convolutional layers; a decoder structure with convolutional layers, two series-connected RefineNet units, and DN units, which include upsampling and convolutional layers (**conv**: convolutional layer)

reduced, while the reconstruction performance is only slightly degraded.

II. System model

This study considers a single-cell downlink massive MIMO system with a BS and a UE. The BS is configured with a uniform linear antenna array, the number of transmitting antennas is $N_t \gg 1$, and the UE uses a single receive antenna. An OFDM system, which transmits information on N_c orthogonal subcarriers, is also considered. The signal on the n-th ($n = 1, 2, \ldots, N_c$) subcarrier received by the UE can be expressed as follows:

$$y_n = \mathbf{h}_n^T \mathbf{v}_n x_n + z_n, \tag{1}$$

where $\mathbf{h}_n \in \mathbb{C}^{N_t \times 1}$, $\mathbf{v}_n \in \mathbb{C}^{N_t \times 1}$, $x_n \in \mathbb{C}$, and $z_n \in \mathbb{C}$ respectively denote the instantaneous channel vector in the frequency domain, precoding vector, data symbol transmitted in the downlink and additive Gaussian white noise. The channel vector on all subcarriers is the instantaneous downlink CSI matrix necessary for the UE to facilitate feedback. At this time, the CSI matrix can be denoted as $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_{N_c}]^T \in \mathbb{C}^{N_c \times N_t}$. A total of $N_t N_c$ complex numbers must be transmitted for a complete CSI matrix due to a large number of antennas. These data occupy a substantial amount of feedback resources, which is undesirable for massive MIMO FDD systems in practical situations. Using a 2D discrete Fourier transform (2D-DFT), the CSI matrix $\mathbf{H}$ can be converted into an angular-delay domain matrix denoted as $\mathbf{H}' = \mathbf{F}_{\mathsf{d}} \mathbf{H} \mathbf{F}_{\mathsf{a}}$, where $\mathbf{F}_{\mathsf{d}} \in \mathbb{C}^{N_c \times N_c}$ and $\mathbf{F}_{\mathsf{a}} \in \mathbb{C}^{N_t \times N_t}$ are two DFT matrices [2]. The elements of CSI matrix $\mathbf{H}'$ only contain a small fraction of the large components, and other components are close to zero. Only the first N_c' rows of $\mathbf{H}'$ in the delay domain contain values because the time delay between multipath arrivals is within a limited period. Therefore, we only keep the first N_c' rows of $\mathbf{H}'$ and obtain $\mathbf{H}'' \in \mathbb{C}^{N_c' \times N_t}$. The total number of parameters for the feedback is then reduced to $N = N_c' N_t$.

For the convenience of the processing in the neural network (NN), we split $\mathbf{H}''$ into two parts, i.e., real and imaginary parts, and stack these two parts on the third dimension. Then, we normalize all the entries of the processed channel matrix to (0, 1). The schematic diagram of this process is shown in Fig. 2. In our proposed networks, the input of the CSI feedback network is a tensor with a size of $2 \times 32 \times 32$, and the exact amount of feedback is determined by the compression ratio. For example, at a compression rate of 1/32, the actual feedback data is a tensor with a size of $16 \times 2 \times 2$.

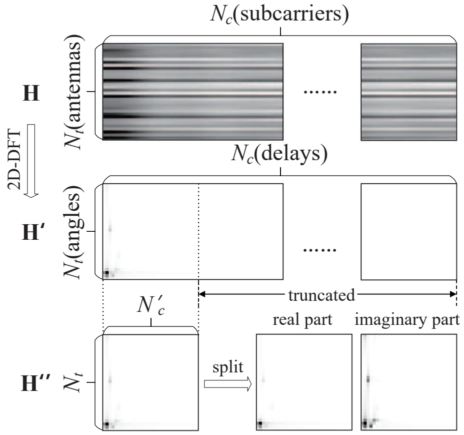

Fig. 2. The schematic diagram of preprocessing.

We mainly consider the autoencoder network for the downlink CSI feedback. The channel matrix $\mathbf{H}''$ is input to the autoencoder network, and the encoder compresses this matrix into codeword s according to a given compression ratio. After s is fed back to the BS, the decoder reconstructs s to $\mathbf{H}''$.

III. Two proposed DL-based CSI networks

We develop the DL-based CSI feedback network in this section using an autoencoder. On this basis, we then propose a new lightweight structured CSI feedback network. The specific model structure and parameter details are presented in the following subsections.

A. CSI feedback neural network architecture based on the convolutional autoencoder

The CsiNet in [2] demonstrates superior performance to traditional CS algorithms considering CSI sensing and recovery. However, the biggest problem of using a fully connected layer in a CSI feedback network is that it can only be utilized for the specified input size, i.e., for a given number of transmit antennas and subcarriers. Moreover, this network cannot effectively retain the characteristics of 2D image signals, which means its performance cannot be further improved. Therefore, we propose a CSI feedback network based on a convolutional autoencoder called ConvCsiNet to address these issues. The neural network of ConvCsiNet uses convolutional layers instead of fully connected ones to extract features. In theory, the network trained under this architecture can improve

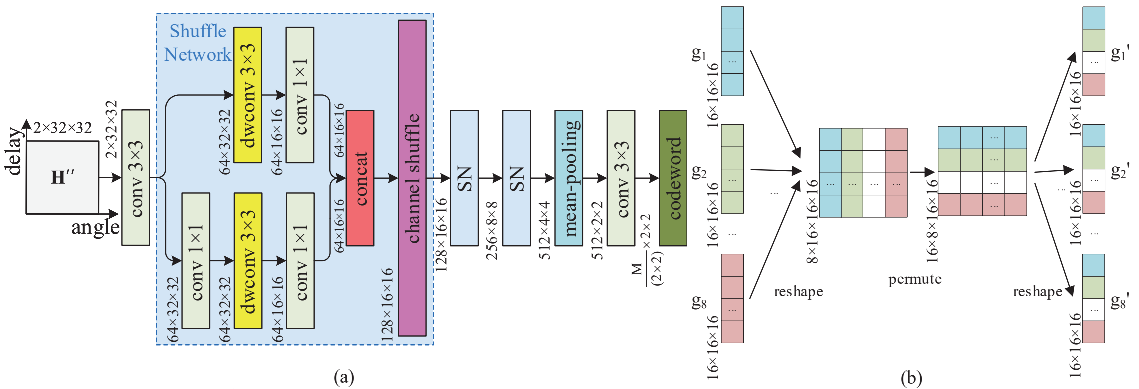

Fig. 3. (a) Encoder structure of ShuffleCsiNet constructed with convolutional and mean-pooling layers and SN units, including convolutional, depthwise convolutional, concat, and channel shuffle layers. (b) Schematic of the channel shuffle layers (**conv**: convolutional layer; **dwconv**: depthwise convolutional layer)

the reconstruction performance while adapting to input size with different antennas and subcarriers.

The architecture of ConvCsiNet shown in Fig. 1 is divided into the following two modules: encoder and decoder. The encoder includes a convolutional layer and four encoded convolution network (ECN) units. The ECN is used to complete the dimensionality reduction and further feature extraction. The number of channels of the convolutional layer in the last ECN unit is adjusted by the specific compression ratio. Each convolutional layer is added to the batch normalization (BN) layer and uses the Leaky ReLU activation function. The M-dimensional codeword s outputted by the encoder is presented in the form of a feature map.

The decoder part comprises four deconvolution network (DN) units, two RefineNet units, and a convolutional layer. The DN unit is used to restore codeword s to the original channel dimension and complete the preliminary reconstruction. The convolutional layer of the last DN unit generates a two-channel feature map, that is, two real number matrices, as the initial estimates of the real and imaginary parts of the original CSI matrix. These estimates then enter two similar RefineNet units. RefineNet is used to improve the quality of reconstruction, which is composed of four convolutional layers. RefineNet is inspired by the deep Residual Network and adds the output of the first convolutional layer to the last layer as the output of the entire structure to avoid the problem of gradient disappearance caused by too many layers. The last convolutional layer uses a Sigmoid activation function to normalize the output elements to the [0,1] interval. The remaining convolutional layers are added to the BN layer, and the Leaky ReLU activation function is used.

B. CSI feedback neural network architecture based on the lightweight structure

The ConvCsiNet aims to optimize the CSI feedback network theoretically and adapt to CSI information input of different sizes, which causes its high complexity. Today, mobile terminals and embedded devices are widely used in the field of modern communications. The parameters and calculations of the ConvCsiNet network model are excessively large. Therefore, we propose a new autoencoder-based CSI feedback network called ShuffleCsiNet. This method replaces the average pooling and the convolutional layers with a lightweight structure to reduce the parameter number and algorithm complexity.

The encoder and decoder parts of the CSI feedback network are located at the UE and BS, respectively. The mobile end has more computing and memory limitations than those of the BS. Therefore, ShuffleCsiNet focuses on optimizing the structure of the encoder part, and the decoder part is the same as that in ConvCsiNet.

Fig. 3(a) shows that the encoder part of ShuffleCsiNet includes a pooling layer, two convolutional layers, and three shuffle network (SN) units. The SN is used to complete the dimensionality reduction and further feature extraction. The depthwise and the next convolutional layers with respective sizes of 3×3 and 1×1 can form a depthwise separable convolutional layer [10]. Afterward, the two branches pass the concat layer, which halves the size of the feature space and doubles the number of channels to achieve feature reuse. Finally, channel shuffle can ensure information exchange between the two branches. Each depth separable convolutional layer is added to the BN layer, and each convolutional layer is added to the BN layer and uses the Leaky ReLU activation function. The M-dimensional codeword s outputted by the encoder is presented in the form of a feature map. This structure can achieve almost the same effect as the traditional convolution structure but with considerably reduced parameters.

The two branches of the Shuffle Network can use different receptive fields to learn different features of the input information. Both branches which end with a convolutional layer are superimposed through the concat layer. If the convolution outputs are superimposed directly without the channel shuffle layer, a side effect occurs: the output of a channel comes from only a small fraction of the input channel. This property blocks the flow of information between channel groups and weakens the representation capability. This problem can be solved efficiently and elegantly by a channel shuffle layer [10]. The structure of the channel shuffle is shown in Fig. 3(b). The channel shuffle after the concat layer can reduce element-level operations and help the information flowing across feature channels, and thus the overall network performance is improved. After several experiments, we set the number of shuffle groups to 8 to obtain satisfactory reconstruction performance.

Let the transformation formula and all parameters of the entire network be $f(\cdot)$ and $\Theta = \{\Theta_{\mathsf{en}}, \Theta_{\mathsf{de}}\}$, respectively, where the parameters include encoder and decoder parameters. The CSI matrix recovered from the CSI feedback network model proposed in this study can then be expressed as follows:

$$\hat{\mathbf{H}}'' = f(\mathbf{H}''; \Theta) \triangleq f_{\mathsf{de}}(f_{\mathsf{en}}(\mathbf{H}''; \Theta_{\mathsf{en}}); \Theta_{\mathsf{de}}). \quad (2)$$

We use the adaptive moment estimation (ADAM) algorithm to update the parameter set of the network. The loss function of the network is represented by the mean squared error (MSE). Therefore, the prediction loss of the model is defined as follows:

$$L(\Theta) = \frac{1}{M} \sum_{m=1}^{M} \|f(\mathbf{H}''_m; \Theta) - \mathbf{H}''_m\|_2^2, \quad (3)$$

where M is the total number of samples in the training set, and $\|\cdot\|_2$ is the Euclidean norm.

IV. Numerical results and analysis

We compare and analyze the experimental data of ConvCsiNet and ShuffleCsiNet, including their reconstruction performance, parameter numbers, and floating point operations (FLOPs). FLOPs can be used to measure the algorithmic complexity of network models. We verify the positive effect of the channel shuffle structure on ShuffleCsiNet. Besides, we apply ConvCsiNet on different datasets to verify that it can adapt to CSI information of different sizes.

We use the COST 2100 model [11] to obtain the training and testing data and generate channel matrices for two environments. The two environments include indoor cellular environments in the 5.3 GHz bands and outdoor rural environments in 300 MHz bands. We set the bandwidth of the MIMO system to 20 MHz, and the number of subcarriers is N_c = 1024. Since only a small number of lines have a signal in the delay domain, we truncated the first 32 lines of data. Therefore, we selected $N_c' = 32$ as the truncated size in the actual training and testing process.

We use a uniform linear array with N_t = 32 at the BS for the convenience of comparison. The results of [13] show that under normal circumstances, the feedback of the CSI feedback network adopts 4bit quantization. In this case, the feedback amount for compression ratios (CR) of 1/4, 1/8 is still large. So the experiments in this study will be performed with CR of 1/16 and 1/32. The training, validation, and test sets used for offline training contain 75,000, 12,500, and 12,500 samples, respectively. The entire training process is performed in the Keras framework. We utilize the ADAM optimizer using MSE loss to configure the training model. The parameters in ADAM are set as $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1 \times 10^{-8}$. This study also uses a small batch training scheme with a batch size of 200, a training round set of 1000, and a learning rate of 0.001. CsiNet, ConvCsiNet, and ShuffleCsiNet are trained and tested on NVIDIA Tesla V100.

A. CSI reconstruction performance of the networks

The normalized MSE (NMSE) is used to evaluate the reconstruction performance and can be defined as follows:

$$\text{NMSE} = \mathbb{E}\left\{\frac{\|\mathbf{H} - \hat{\mathbf{H}}\|_2^2}{\|\mathbf{H}\|_2^2}\right\}. \quad (4)$$

The following cosine similarity is also calculated to facilitate comparison with the CsiNet.

$$\rho = \mathbb{E}\left\{\frac{1}{N_c} \sum_{n=1}^{N_c} \frac{|\hat{\mathbf{h}}_n^H \mathbf{h}_n|}{\|\hat{\mathbf{h}}_n\|_2 \|\mathbf{h}_n\|_2}\right\}, \quad (5)$$

where $\hat{\mathbf{h}}_n$ denotes the reconstructed channel vector of the nth subcarrier and ρ measures the quality of the beamforming vector when the vector is set as $\mathbf{v}_n = \hat{\mathbf{h}}_n / \|\hat{\mathbf{h}}_n\|_2$.

Table I summarizes the performance comparison considering NMSE and cosine similarity. The NMSE values in the experimental data are in the form of dB. We compare the two proposed CSI feedback networks with CsiNet, CRNet [6], DS-NLCsiNet [7], and some non-DL methods. The table shows that the reconstruction performance of ConvCsiNet and ShuffleCsiNet is far superior to other DL-based and non-DL methods. Compared with other DL-based CSI feedback networks, ConvCsiNet has more advantages when the compression ratio is 1/32. Notably, ShuffleCsiNet can maintain satisfactory reconstruction performance when parameter quantity and algorithm complexity are substantially reduced. Such a phenomenon is due to the functional reuse in the SN structure of the concat layer and the channel shuffle structure, thereby increasing the network's robustness.

B. Parameter numbers and FLOPs of the networks

The ShuffleCsiNet aims to obtain a light network structure and low algorithm complexity. Therefore, the corresponding reconstruction performance of ShuffleCsiNet is slightly lower than that of ConvCsiNet. To reflect the overall benefits brought by the lightweight structure, we compare ConvCsiNet as a high complexity benchmark with ShuffleCsiNet. Note that our lightweight structure is only used in the encoder part, and the decoder part of ConvCsiNet and ShuffleCsiNet are the same. Therefore, we only calculate the parameter numbers and FLOPs in the encoder part.

Table II shows the parameter numbers and FLOPs of the encoder networks of our two proposed CSI feedback networks. ConvCsiNet trades the increase in computational complexity and number of parameters for excellent reconstruction performance. Besides, the number of parameters and complexity of ShuffleCsiNet are much lower than those of ConvCsiNet due to the use of lightweight network structures.

We assume that the compression rate is 1/16, and the period of CSI compression and feedback is 1ms. The ConvCsiNet encoder network needs to store 1,697,144 float data, occupying about 6MB of storage space. The total FLOPs of the encoder network is 58,515,456, which indicates that the computing power required by the mobile device for CSI feedback is about 59G floating point operations per second (FLOPS). [12] points out that Kirin 970, which is the mid- and high-end mobile SoC today, has a total peak compute capability of 244.8 G FLOPS. However, the GPU of the mobile phone SoC still needs to complete many other tasks, such as graphics rendering and voice recognition. Therefore, the lightweight structured CSI feedback network is of practical significance.

TABLE I
NMSE IN dB AND COSINE SIMILARITY.

CR	Method	Indoor		Outdoor	
		NMSE	ρ	NMSE	ρ
1/16	LASSO	-2.72	0.70	-1.01	0.46
	TVAL3	-2.61	0.66	-0.43	0.45
	CsiNet	-8.65	0.93	-4.51	0.79
	CRNet	-11.35	0.95	-5.44	0.80
	DS-NLCsiNet	-12.93	0.97	-4.98	0.81
	ConvCsiNet	**-13.79**	**0.98**	**-6.00**	**0.85**
	ShuffleCsiNet	-12.14	0.97	-5.00	0.82
1/32	LASSO	-1.03	0.48	-0.24	0.27
	TVAL3	-0.27	0.33	0.46	0.28
	CsiNet	-6.24	0.89	-2.81	0.67
	CRNet	-8.93	0.94	-3.51	0.71
	DS-NLCsiNet	-8.64	0.93	-3.35	0.73
	ConvCsiNet	**-10.10**	**0.95**	**-5.21**	**0.82**
	ShuffleCsiNet	-9.41	0.94	-3.50	0.74

TABLE II
PARAMETER NUMBER AND FLOPs OF THE ENCODER.

CR	Method	Parameter number	FLOPs
1/16	CsiNet	262,308	561,152
	ConvCsiNet	1,697,144	58,515,456
	ShuffleCsiNet	**415,528**	**24,313,856**
1/32	CsiNet	131,236	299,008
	ConvCsiNet	1,623,416	58,220,544
	ShuffleCsiNet	**341,800**	**24,018,944**

TABLE III
NMSE PERFORMANCE OF SHUFFLECSINET-N AND SHUFFLECSINET.

CR	Method	Indoor		Outdoor	
		NMSE	ρ	NMSE	ρ
1/16	**ShuffleCsiNet**	**-12.14**	**0.97**	**-5.00**	**0.82**
	ShuffleCsiNet-n	-10.99	0.98	-4.69	0.79
1/32	**ShuffleCsiNet**	**-9.41**	**0.94**	**-3.50**	**0.74**
	ShuffleCsiNet-n	-9.06	0.93	-3.12	0.70

TABLE IV
NMSE PERFORMANCE OF CONVCSINET ON TWO DATASETS OF DIFFERENT SIZES.

CR	Dataset	Indoor		Outdoor	
		NMSE	ρ	NMSE	ρ
1/16	32×32	-13.79	0.98	-6.00	0.85
	48×48	-10.01	0.94	-4.97	0.81
1/32	32×32	-10.10	0.95	-5.21	0.82
	48×48	-8.70	0.92	-3.44	0.77

C. Advantage verification of the channel shuffle structure

In order to verify the advantages of the channel shuffle structure, we conducted a comparative experiment. We have built two networks: one is ShuffleCsiNet, and the other is ShuffleCsiNet without the channel shuffle structure, called ShuffleCsiNet-n. In Table III, we compared the reconstruction performance and cosine similarity of the two networks in indoor cellular environments with two compression ratios (1/16 and 1/32). From the table, we can see that the existence of the channel shuffle structure can greatly improve the overall network performance, and this structure does not increase the complexity of the network.

D. Application of ConvCsiNet on datasets of different sizes

In this experiment, we train ConvCsiNet using a dataset with 32×32 CSI size but test it in a dataset with 48×48 CSI size. The corresponding results are shown in Table IV. The results demonstrate that ConvCsiNet can adapt to different input sizes with different antennas. ConvCsiNet can be trained on a smaller system and applied to a larger system. This characteristic thus reduces the computational complexity in the training phase. Moreover, ConvCsiNet can be applied to scenarios requiring multiple antenna configurations to avoid storing multiple parameters.

V. CONCLUSION

We developed CsiNet and proposed a CSI feedback network called ConvCsiNet in this study based on a CNN autoencoder. Subsequently, we proposed a lightweight structured CSI feedback network called ShuffleCsiNet based on ConvCsiNet. Experiments show that the proposed ConvCsiNet has satisfactory reconstruction performance while adapting to input size with different antennas and subcarriers, and ShuffleCsiNet can greatly save memory space and kernel computing power while ensuring satisfactory reconstruction performance. Both CSI feedback architectures exhibit potentials for practical deployment on realistic MIMO systems. In the practical deployment, we can use NN compression and acceleration techniques [13] to reduce latency and complexity further.

REFERENCES

[1] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, “Five disruptive technology directions for 5G,” *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74–80, 2014.

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

[3] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, “Deep learning for wireless physical layer: Opportunities and challenges,” *China Commun.*, vol. 14, no. 11, pp. 92–111, 2017.

[4] M. Chen, J. Guo, X. Li, and S. Jin, “An overview of the CSI feedback based on deep learning for massive MIMO systems,” *Chin J. Internet Things*, vol. 4, no. 1, pp. 33–44, 2020.

[5] Y. Yang, F. Gao, G. Y. Li, and M. Jian, “Deep Learning-Based Downlink Channel Prediction for FDD Massive MIMO System,” *IEEE Commun. Lett.*, vol. 23, no. 11, pp. 1994–1998, 2019.

[6] Z. Lu, J. Wang, and J. Song, “Multi-resolution CSI feedback with deep learning in massive MIMO system,” in *IEEE Int. Conf. Commun.*, 2020.

[7] X. Yu, Y. Bai, H. Wu, and X. Li, “DS-NLCsiNet: Exploiting non-local neural networks for massive MIMO CSI feedback,” *IEEE Commun. Lett.*, 2020.

[8] J. Guo, C.-K. Wen, S. Jin, and G. Y. Li, “Convolutional neural network based multiple-rate compressive sensing for massive MIMO CSI feedback: Design, simulation, and analysis,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2827–2840, 2020.

[9] J. Guo, X. Li, M. Chen, P. Jiang, T. Yang, W. Duan, H. Wang, S. Jin, and Q. Yu. “AI enabled wireless communications with real channel measurements: Channel feedback,” *J. Commun. Inf. Netw.*, vol. 5, no. 3, pp. 310–317, Sep. 2020.

[10] X. Zhang, X. Zhou, M. Lin, and J. Sun, “Shufflenet: An extremely efficient convolutional neural network for mobile devices,” in *Proc. IEEE CVPR*, 2018, pp. 6848–6856.

[11] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. De Doncker. “The COST 2100 MIMO channel model,” *IEEE Wireless Commun.*, vol. 19, no. 6, pp. 92–99, 2012.

[12] S. Wang, A. Pathania, and T. Mitra, “Neural Network Inference on Mobile SoCs,” *IEEE Design & Test*, vol. 37, no. 5, pp. 50–57, 2020.

[13] J. Guo, J. Wang, C.-K. Wen, S. Jin, and G. Y. Li, “Compression and acceleration of neural networks for communications,” *IEEE Wireless Commun.*, vol. 27, no. 4, pp. 110–117, 2020.