

Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels

Tianqi Wang, Chao-Kai Wen[✉], Shi Jin[✉], and Geoffrey Ye Li

Abstract—Massive multiple-input multiple-output (MIMO) systems rely on channel state information (CSI) feedback to perform precoding and achieve performance gain in frequency division duplex networks. However, the huge number of antennas poses a challenge to the conventional CSI feedback reduction methods and leads to excessive feedback overhead. In this letter, we develop a real-time CSI feedback architecture, called CsiNet-long short-term memory (LSTM), by extending a novel deep learning (DL)-based CSI sensing and recovery network. CsiNet-LSTM considerably enhances recovery quality and improves tradeoff between compression ratio (CR) and complexity by directly learning spatial structures combined with time correlation from training samples of time-varying massive MIMO channels. Simulation results demonstrate that CsiNet-LSTM outperforms existing compressive sensing-based and DL-based methods and is remarkably robust to CR reduction.

Index Terms—Massive MIMO, FDD, CSI feedback, compressive sensing, deep learning.

I. INTRODUCTION

MASSIVE multiple-input multiple-output (MIMO) systems have been recognized as a critical development for future wireless communications. With downlink channel state information (CSI), a base station (BS) with massive antennas can use channel-adaptive techniques to eliminate inter-user interference and increase channel capacity. In frequency division duplex (FDD) networks, downlink CSI can only be estimated at user equipment (UE) and fed back to the BS. The excessive overhead has motivated many feedback reduction techniques, such as vector quantization and codebook-based approaches [1]. However, quantization errors pose a challenge to CSI-sensitive applications, whereas the huge number of antennas complicates the codebook design and accordingly increases feedback overhead.

The compressive sensing (CS)-based CSI feedback approaches proposed recently address the aforementioned problems by using the spatial and temporal correlation of

CSI. These methods sparsify CSI under certain bases to apply CS for feedback and reconstruction [2] or distributed compressive channel estimation [4]. In reality, CSI is only approximately sparse under elaborate base selection or sparsity modeling. Many existing CS algorithms experience difficulty in CSI compression and recovery if there is a model mismatch.

Time correlation property of slow-varying channels has been considered in [2] to further reduce feedback quantity. This method reuses the previously retained channel information for subsequent CSI recovery if the error is under a certain threshold. However, the reused information only provides an estimate and is hard to update in real time. As a result, resolution degrades and the feedback overhead cannot reduce any more in fast-changing channels.

Recently, deep learning (DL) methods have been successfully applied in wireless communications [5]–[7]. A CSI feedback network, called CsiNet [8], uses an autoencoder-like architecture to mimic the CS and reconstruction processes. It uses an encoder to obtain compressed representation (codewords) by directly learning channel structures from the training data and a decoder to recover CSI via one-off feedforward multiplication. CsiNet remarkably outperforms the CS-based methods. But it reconstructs CSI independently, and ignores time correlation in time-varying channels.

In this letter, we improve the architecture by considering time correlation. This letter is motivated by the recurrent convolutional neural network (RCNN) architecture that has been successfully used in video representation and reconstruction [9]. The basic idea is to use a convolutional neural network (CNN) and a recurrent neural network (RNN) to extract spatial features and interframe correlation, respectively. Our contribution in this letter is summarized as follows.

- We propose an DL-based CSI feedback protocol for FDD MIMO systems by extending CsiNet with a long short-term memory (LSTM) network, which is a classic type of RNN. The proposed network, called CsiNet-LSTM, modifies the CNN-based CsiNet for CSI compression and initial recovery and uses LSTM to extract time correlation for further improvement in resolution.
- The experiment results demonstrate that CsiNet-LSTM achieves the best recovery quality and outperforms state-of-the-art CS methods in terms of complexity. CsiNet-LSTM exhibits remarkable robustness to compression ratio (CR) reduction and enables real-time and extensible CSI feedback applications without considerably increasing overhead compared with CsiNet.

II. SYSTEM MODEL

Consider an FDD downlink massive MIMO-orthogonal frequency division multiplexing (OFDM) system with N_c

Manuscript received July 16, 2018; revised September 8, 2018; accepted September 24, 2018. Date of publication October 5, 2018; date of current version April 9, 2019. This work was supported in part by the National Science Foundation (NSFC) for Distinguished Young Scholars of China under Grant 61625106, and in part by NSFC under Grant 61531011. The work of C.-K. Wen was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-110-019 and in part by ITRI in Hsinchu, Taiwan. The associate editor coordinating the review of this paper and approving it for publication was A. Liu. (*Corresponding author: Shi Jin.*)

T. Wang and S. Jin are with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: wangtianqi@seu.edu.cn; jinshi@seu.edu.cn).

C.-K. Wen is with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan (e-mail: ckwen@ieee.org).

G. Y. Li is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: liye@ece.gatech.edu).

Digital Object Identifier 10.1109/LWC.2018.2874264

2162-2345 © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

subcarriers. The BS deploys a uniform linear array (ULA) with N_t transmit antennas. In a time-varying channel caused by UE mobility, the received signal at time t on the n th subcarrier for UE with a single receiver antenna can be modeled as,

$$y_{n,t} = \mathbf{h}_{n,t}^T \mathbf{v}_{n,t} x_{n,t} + z_{n,t}, \quad (1)$$

where $\mathbf{h}_{n,t} \in \mathbb{C}^{N_t \times 1}$, $x_{n,t} \in \mathbb{C}$, and $z_{n,t} \in \mathbb{C}$ denote the instantaneous channel vector in the frequency domain, transmit data symbol, and additive noise, respectively, $\mathbf{v}_{n,t} \in \mathbb{C}^{N_t \times 1}$ is the beamforming or precoding vector designed by the BS based on the received downlink CSI. We denote the CSI matrix at time t in the spatial-frequency domain as $\mathbf{H}_t = [\mathbf{h}_{1,t}, \dots, \mathbf{h}_{N_c,t}]^T \in \mathbb{C}^{N_c \times N_t}$. In practice, the UE continuously estimates and feeds instantaneous CSI (i.e., $\mathbf{H}_t, \mathbf{H}_{t+1}, \dots$) back to the BS to track the time-varying characteristics of the channel. To reduce feedback overhead, we can exploit the following observations.

Observation 1 (Angular-Delay Domain Sparsity): \mathbf{H}_t can be transformed into an approximately sparsified matrix \mathbf{H}'_t in the angular-delay domain via 2D discrete Fourier transform (2D-DFT) [8] by $\mathbf{H}'_t = \mathbf{F}_d \mathbf{H}_t \mathbf{F}_a$, where $\mathbf{F}_d \in \mathbb{C}^{N_c \times N_c}$ and $\mathbf{F}_a \in \mathbb{C}^{N_t \times N_t}$ are two DFT matrices. First, due to limited multipath time delay, performing DFT on frequency domain channel vectors (i.e., column vectors of \mathbf{H}_t) can transform \mathbf{H}_t into a sparse matrix in the delay domain, with only the first N'_c ($< N_c$) rows having distinct non-zero values. Secondly, as proved in [10], the channel matrix is sparse in a defined angle domain by performing DFT on spatial domain channel vectors (i.e., row vectors of \mathbf{H}_t) if the number of transmit antennas, $N_t \rightarrow +\infty$, is very large. Usually, \mathbf{H}'_t is only approximately sparse for finite N_t , which challenges conventional CS methods. Therefore, we will propose a DL-based feedback architecture without sparsity prior constraint. We perform sparsity transformation to decrease parameter overhead and training complexity. We retain the first N'_c non-zero rows and truncate \mathbf{H}'_t to a $N'_c \times N_t$ matrix, \mathbf{H}''_t , which reduces the total number of parameters for feedback to $N = N'_c N_t$.

Observation 2 (Correlation Within Coherence Time): UE motion during communication results in a Doppler spread, i.e., time-varying characteristics of wireless channels. With the maximum movement speed denoted as v , coherence time can be calculated as

$$\Delta t = \frac{c}{2v f_0}, \quad (2)$$

where f_0 is the carrier frequency and c is the speed of light [3]. The CSI within Δt is considered correlated with one other. Therefore, instead of independently recovering CSI, the BS can combine the feedback and previous channel information to enhance the subsequent reconstruction. We set the feedback time interval as δt and place T adjacent instantaneous angular-delay domain channel matrices into a channel group, i.e., $\{\mathbf{H}''_t\}_{t=1}^T = \{\mathbf{H}''_1, \dots, \mathbf{H}''_t, \dots, \mathbf{H}''_T\}$. The group exhibits correlation property, as long as T satisfies $0 \leq \delta t \cdot T \leq \Delta t$.

In this letter, we design an encoder, $s_t = f_{\text{en}}(\mathbf{H}''_t)$, at the UE to compress each complex-valued \mathbf{H}''_t of $\{\mathbf{H}''_t\}_{t=1}^T$ into an M -dimensional real-valued codeword vector s_t ($M < N$). If two real number matrices are used to represent the real and imaginary parts of \mathbf{H}''_t , then CR will be $M/2N$. We also design a decoder with a memory that can extract time correlation from the previously recovered channel matrices, $\hat{\mathbf{H}}''_1, \dots, \hat{\mathbf{H}}''_{t-1}$ and combine them with the received s_t for current reconstruction, $\hat{\mathbf{H}}''_t = f_{\text{de}}(s_t; \hat{\mathbf{H}}''_1, \dots, \hat{\mathbf{H}}''_{t-1})$, where $1 \leq t \leq T$. Then, inverse

2D-DFT is performed to obtain the original spatial frequency channel matrix.

III. CSI-NET-LSTM

The CsiNet in [8] demonstrates remarkable performance in CSI sensing and reconstruction. However, the resolution degrades at low CR because the it only focuses on angular-delay domain sparsity (Observation 1) and ignores the time correlation (Observation 2) of time-varying massive MIMO channels. The two observations in Section II are similar to the spatial structure and interframe correlation of videos, respectively. Motivated by RCNN that excels in extracting spatial-temporal features for video representation [9], we will extend CsiNet with LSTM to improve CR and recovery quality trade-off. We will also introduce the multi-CR strategy in [9] to implement variable CRs on different channel matrices.

The proposed CsiNet-LSTM is illustrated in Fig. 1(b) with CsiNet shown in Fig. 1(a). Our model includes the following two steps: angular-delay domain feature extraction, correlation representation and final reconstruction.

1) *Angular-Delay Domain Feature Extraction:* We apply CsiNet with two different CRs to $\{\mathbf{H}''_t\}_{t=1}^T$ to learn the angular-delay domain structure and perform sensing and initial reconstruction. A high-CR CsiNet transforms the first channel \mathbf{H}''_1 into an $M_1 \times 1$ codeword vector that retains sufficient structure information for high resolution recovery. A low-CR CsiNet encoder performs on the remaining $T - 1$ channel matrices to generate a series of $M_2 \times 1$ codewords ($M_1 > M_2$), given that less information is required due to channel correlation. The $T - 1$ codewords are all concatenated with the first $M_1 \times 1$ codeword before being fed into the low-CR CsiNet decoder to fully utilize feedback information. Each CsiNet outputs two matrices with size $N'_c \times N_t$ as extracted features from the angular-delay domain.

All low-CR CsiNets shown in Fig. 1(b) share the same network parameters, i.e., weights and bias, because they perform the same work. which dramatically reduces parameter overhead. Furthermore, the architecture can be easily rescaled to perform on channel groups with different T if the value of T changes to adapt to the channel-changing speed and feedback frequency. In practice, a low-CR CsiNet will be reused $T - 1$ times instead of making $T - 1$ copies. The grey blocks in Fig. 1(b) load parameters from the original CsiNets as pre-training before end-to-end training with the entire architecture. This method can alleviate vanishing gradient problems due to long paths from CsiNets to LSTMs.

2) *Correlation Representation and Final Reconstruction:* We use LSTMs to extend the CsiNet decoders for time correlation extraction and final reconstruction. LSTMs have inherent memory cells and can keep the previously extracted information for a long period for later prediction. In particular, the outputs of the CsiNet decoders form a sequence of length T before being fed into three-layer LSTMs. Each LSTM has $2N'_c N_t$ hidden units, which is the same as the output dimension. The final outputs are then reshaped into two $N'_c \times N_t$ matrices as the final recovered $\hat{\mathbf{H}}''_t$. The spatial frequency domain CSI can then be obtained via inverse 2D-DFT. At each time step, the LSTMs implicitly learn time correlation from the previous inputs and then merge them with the current inputs to increase low CR recovery quality. Correlation information is updated after each step due to the nature of LSTM. The experimental results show that the highly compressed $T - 1$ matrices can achieve better recovery accuracy than \mathbf{H}''_1 as a benefit from LSTMs.

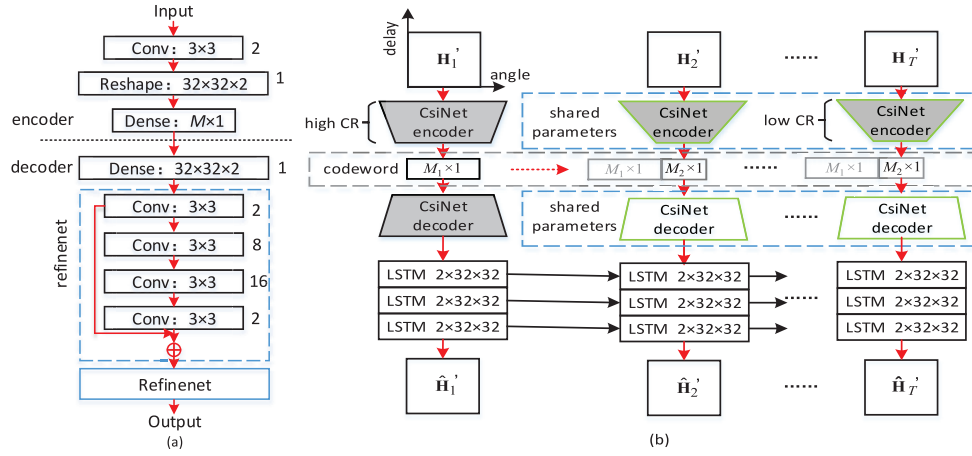


Fig. 1. (a) CsiNet architecture presented in [8]. It comprises an encoder with a 3×3 conv layer and an M -unit dense layer for sensing and a decoder with a $2N'_c N_t$ -unit dense layer and two RefineNet for reconstruction. Each RefineNet contains four 3×3 conv layers with different channel sizes. (b) Overall architecture of CsiNet-LSTM. H'_1 and the remaining $T - 1$ channel matrices are compressed by high-CR and low-CR CsiNet encoders, respectively. Codewords are concatenated before being fed into the low-CR CsiNet decoder, and final reconstruction is performed by three $2N'_c N_t$ -unit LSTMs.

We use end-to-end learning to obtain all parameters for the encoder and the decoder denoted as $\Theta = \{\Theta_{\text{en}}, \Theta_{\text{de}}\}$. Notably, H'_t are normalized with all elements scaled into the $[0, 1]$ range before being fed into the network. This normalization is required for CsiNet. For details, we refer to [8]. Let f denote the final trained network defined as

$$\hat{H}_t'' = f(H_t''; \Theta) = f_{\text{de}}(f_{\text{en}}(H_1''; \Theta_{\text{en}}), \dots, f_{\text{en}}(H_t''; \Theta_{\text{en}}); \Theta_{\text{de}}). \quad (3)$$

We select ADAM as the optimization algorithm and use mean-squared error (MSE) as the loss function, which is defined as,

$$L(\Theta) = \frac{1}{M} \sum_{m=1}^M \sum_{t=1}^T \|f(H_t''; \Theta) - H_t''\|_2^2, \quad (4)$$

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

The procedure for CsiNet-LSTM is described as follows. Multiple CR CsiNet encoders are deployed at UE, whereas the CsiNet decoders and LSTMs are deployed at the BS. Each side has a counter. At the beginning, H'_1 is compressed with high CR at the UE and recovered by a high-CR CsiNet decoder and initialized by the LSTMs at the BS. In the subsequent time step t ($2 \leq t \leq T$), H'_t is transformed into a lower-dimensional codeword s_t at the UE, which is expected to contain the learned correlation information. The lower-dimensional codeword, s_t is then concatenated with the first one s_1 and inversely transformed by the LSTMs at the BS. After each time step, the counters add by one. Similar operations continue until the counters accumulate to T and the LSTMs are reset for the subsequent channel group recovery.

IV. SIMULATION RESULTS AND ANALYSIS

We use the COST 2100 model [11] to simulate time-varying MIMO channels and generate training samples. We set the MIMO-OFDM system to work on a 20 MHz bandwidth with $N_c = 256$ subcarriers and use ULA with $N_t = 32$ antennas at the BS. The angular-delay domain channel matrix is truncated to a size of 32×32 . Two scenarios are considered: the indoor scenario at 5.3 GHz with UE speed $v = 0.0036$ km/h and the outdoor scenario at 300 MHz with UE speed $v = 3.24$ km/h. Data sets are generated by randomly setting different start

places with two uniform velocities. Therefore, Δt is 30s and 0.56s, respectively. Compressed CSI is fed back every $\delta t = 0.04$ s. We set the channel group size $T = 10$, which satisfies $\delta t \cdot T < \Delta t$ in both scenarios. We perform experiments at CR values of 1/16, 1/32, and 1/64, with the first channel H'_1 compressed under 1/4.

Training, validation, and testing sets have 75,000, 12,500, and 12,500 samples, respectively, for offline training. Some parameters are preloaded from the CsiNet for initialization. The epochs are adjusted for a convergence situation ranging from 500 to 1,000. The batch size is 100 and the learning rates are 0.001 and 0.0001 for the former and latter epochs, respectively.

We compare our architecture with three state-of-the-art CS-based algorithms, namely, LASSO ℓ_1 -solver [12], TVAL3 [13], and BM3D-AMP [14], and the DL-based CsiNet [8]. LASSO uses simple sparsity priors but achieves good performance. TVAL3 is a minimum total variation method that provides remarkable recovery quality but with high computing efficiency. BM3D-AMP achieves the most accurate recovery performance on natural images and is 10 times faster than other iterative methods.

We use the default configuration in the open source codes of the aforementioned methods for simulation. When comparing with CsiNet, we consider the slight difference between datasets and refine the CsiNet parameters on our training set for several epochs for fairness. We run the conventional CS-based methods on an Intel Core i7-6700 CPU due to the lack of a GPU solution. CsiNet and CsiNet-LSTM are trained and tested on Nvidia GeForce GTX 1080 Ti GPU.

Normalized MSE (NMSE) is used to evaluate the recovery performance, which is defined as follows:

$$\text{NMSE} = \mathbb{E} \left\{ \frac{1}{T} \sum_{t=1}^T \|H_t'' - \hat{H}_t''\|_2^2 / \|H_t''\|_2^2 \right\}. \quad (5)$$

To compare with CsiNet, the following cosine similarity is also calculated:

$$\rho = \mathbb{E} \left\{ \frac{1}{T} \frac{1}{N_c} \sum_{t=1}^T \sum_{n=1}^{N_c} \frac{|\hat{\mathbf{h}}_{n,t}^H \mathbf{h}_{n,t}|}{\|\hat{\mathbf{h}}_{n,t}\|_2 \|\mathbf{h}_{n,t}\|_2} \right\}, \quad (6)$$

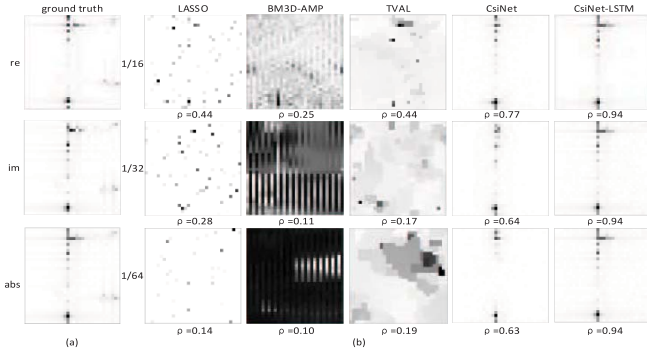


Fig. 2. (a) Pseudo-gray plots of an original channel generated by COST 2100 model in outdoor scenario, showing real part, imagine part and absolute values, respectively. (b) Absolute values of reconstructed images, which are performed by different methods on the original channel given by (a) at different CRs.

TABLE I
PERFORMANCE IN NMSE (dB), COSINE SIMILARITY ρ ,
AND RUNTIME (sec)

	CR	LASSO	BM3D-AMP	TVAL3	CsiNet	CsiNet-LSTM	
Indoor	NMSE	1/16	-2.96	0.25	-3.20	-10.59	-23.06
		1/32	-1.18	20.85	-0.46	-7.35	-22.33
		1/64	-0.18	26.66	0.60	-6.09	-21.24
	ρ	1/16	0.72	0.29	0.73	0.95	0.99
		1/32	0.53	0.17	0.45	0.90	0.99
		1/64	0.30	0.16	0.24	0.87	0.99
	runtime	1/16	0.2471	0.3454	0.3148	0.0001	0.0003
		1/32	0.2137	0.5556	0.3148	0.0001	0.0003
		1/64	0.2479	0.6047	0.2860	0.0001	0.0003
	NMSE↓	1/16-1/64	94%	105	1.19	42%	8%
Outdoor	NMSE	1/16	-1.09	0.40	-0.53	-3.60	-9.86
		1/32	-0.27	18.99	0.42	-2.14	-9.18
		1/64	-0.06	24.42	0.74	-1.65	-8.83
	ρ	1/16	0.49	0.23	0.46	0.75	0.95
		1/32	0.32	0.16	0.28	0.63	0.94
		1/64	0.19	0.16	0.19	0.58	0.93
	runtime	1/16	0.2122	0.4210	0.3145	0.0001	0.0003
		1/32	0.2409	0.6031	0.2985	0.0001	0.0003
		1/64	0.0166	0.5980	0.2850	0.0001	0.0003
	NMSE↓	1/16-1/64	94%	60	2.40	54%	10%

where $\hat{\mathbf{h}}_{n,t}$ denotes the reconstructed channel vector of the n th subcarrier at time t . ρ can measure the quality of beamforming vector when the vector is set as $\mathbf{v}_{n,t} = \hat{\mathbf{h}}_{n,t} / \|\hat{\mathbf{h}}_{n,t}\|_2$ since the UE will achieve the equivalent channel $\hat{\mathbf{h}}_{n,t}^H \hat{\mathbf{h}}_{n,t} / \|\hat{\mathbf{h}}_{n,t}\|_2^2$.

The performance comparison of NMSE, ρ , and runtime are summarized in Table I. From the table, the DL-based CsiNet and CsiNet-LSTM considerably outperform all CS-based methods. Fig. 2 shows a reconstruction result of the 5th channel matrix of a certain channel group in outdoor scenario as an example, which represents the average performance at different CRs. Apparently, CsiNet and CsiNet-LSTM continue to offer adequate beamforming gain at low CRs, where CS-based methods fail to work. In particular, CsiNet-LSTM achieves the lowest NMSE at all CRs and is multiple times lower than CsiNet, especially when CR is low.

Notably, CsiNet-LSTM has the least performance loss as CR decreases, with only 8% and 10% for indoor and outdoor, respectively. The simulation results indicate that the remaining channel matrices $\{\mathbf{H}_t''\}_{t=2}^T$ recovered from a low CR exhibit similar recovery quality and are better than the first channel matrix \mathbf{H}_1'' from a high CR, which is -14.74 dB

and -8.35 dB in average for the indoor and outdoor scenarios, respectively. This result is mainly attributed to the correlation of the channel matrices in time, which can be inherently retained by LSTMs. Moreover, the remaining $T - 1$ channel matrices achieve better recovery quality since code-words are concatenated to offer more information before fed into the low-CR decoder. Furthermore, the DL-based methods benefit from GPU acceleration due to the feedforward and fast matrix vector multiplication nature, which perform approximately thousandfold faster than the CS-based methods. Compared with CsiNet, CsiNet-LSTM slightly loses time efficiency. However, its NMSE and ρ are significantly improved. In addition, runtime is considerably shorter than the feedback interval $\delta t = 0.04$ s, which makes real-time reconstruction possible.

V. CONCLUSION

In this letter, we have proposed a real-time and end-to-end CSI feedback framework by extending the DL-based CsiNet with LSTM. CsiNet-LSTM achieves a remarkable trade-off among CR, recovery quality, and complexity by utilizing the time correlation and structure properties of time-varying massive MIMO channels. Thus, CsiNet-LSTM outperforms CsiNet and CS-based methods. We believe that this framework has the potential for practical deployment on real systems.

REFERENCES

- [1] D. J. Love *et al.*, "An overview of limited feedback in wireless communication systems," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1341–1365, Oct. 2008.
- [2] P.-H. Kuo, H. T. 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. 492–497.
- [3] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
- [4] X. Rao and V. K. N. Lau, "Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3261–3271, Jun. 2014.
- [5] T. Wang *et al.*, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017.
- [6] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
- [7] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmwave massive MIMO systems," *IEEE Wireless Commun. Lett.*, to be published, doi: [10.1109/LWC.2018.2832128](https://doi.org/10.1109/LWC.2018.2832128).
- [8] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Commun. Lett.*, to be published, doi: [10.1109/LWC.2018.2818160](https://doi.org/10.1109/LWC.2018.2818160).
- [9] 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*, Mar. 2018, pp. 1680–1688.
- [10] C.-K. Wen, S. Jin, K.-K. Wong, J.-C. Chen, and P. Ting, "Channel estimation for massive MIMO using Gaussian-mixture Bayesian learning," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1356–1368, Mar. 2015.
- [11] L. Liu *et al.*, "The COST 2100 MIMO channel model," *IEEE Wireless Commun.*, vol. 19, no. 6, pp. 92–99, Dec. 2012.
- [12] I. Daubechies, M. Defrise, and C. D. Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 75, no. 11, pp. 1412–1457, Nov. 2004.
- [13] C. Li, W. Yin, and Y. Zhang, "User's guide for TVAL3: TV minimization by augmented Lagrangian and alternating direction algorithms," *CAAM Rep.*, vol. 20, pp. 46–47, 2009.
- [14] C. A. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 5117–5144, Sep. 2016.