Deep Learning for Massive MIMO CSI Feedback

Chao-Kai Wen, Wan-Ting Shih, and Shi Jin

***Abstract*—In frequency division duplex mode, the downlink channel state information (CSI) should be sent to the base station through feedback links so that the potential gains of a massive multiple-input multiple-output can be exhibited. However, such a transmission is hindered by excessive feedback overhead. In this letter, we use deep learning technology to develop CsiNet, a novel CSI sensing and recovery mechanism that learns to effectively use channel structure from training samples. CsiNet learns a transformation from CSI to a near-optimal number of representations (or codewords) and an inverse transformation from codewords to CSI. We perform experiments to demonstrate that CsiNet can recover CSI with significantly improved reconstruction quality compared with existing compressive sensing (CS)-based methods. Even at excessively low compression regions where CS-based methods cannot work, CsiNet retains effective beamforming gain.**

***Index Terms*—Massive MIMO, FDD, compressed sensing, deep learning, conventional neural network.**

I. Introduction

THE MASSIVE multiple-input multiple-output (MIMO) system is widely regarded as a major technology for fifth-generation wireless communication systems. By equipping a base station (BS) with hundreds or even thousands of antennas in a centralized [1] or distributed [2] manner, such a system can substantially reduce multiuser interference and provide a multifold increase in cell throughput. This potential benefit is mainly obtained by exploiting channel state information (CSI) at BSs. In current frequency division duplexity (FDD) MIMO systems (e.g., long-term evolution Release-8), the downlink CSI is acquired at the user equipment (UE) during the training period and returns to the BS through feedback links. Vector quantization or codebook-based approaches are usually adopted to reduce feedback overhead. However, the feedback quantities resulting from these approaches need to be scaled linearly with the number of transmit antennas and are prohibitive in a massive MIMO regime.

The challenge of CSI feedback in massive MIMO systems has motivated numerous studies [3], [4]. These works have mainly focused on reducing feedback overhead by using the spatial and temporal correlation of CSI. In particular, correlated CSI can be transformed into an uncorrelated sparse vector in some bases; thus, one can use compressive sensing (CS) to obtain a sufficiently accurate estimate of a sparse vector from an underdetermined linear system. This concept has inspired the establishment of CSI feedback protocols based on CS [3] and distributed compressive channel estimation [4]. The use of several algorithms, including LASSO ℓ_1-solver [5] and AMP [6], has also been proposed in CS. However, these algorithms [5], [6] struggle to recover compressive CSI because they use a simple sparsity prior while their channel matrix is not perfectly but is *approximately* sparse. Moreover, the changes among most adjacent elements in the channel matrix are subtle. These properties complicate modeling their priors. Although researchers have designed advanced algorithms (e.g., TVAL3 [7] and BM3D-AMP [8]) that can impose elaborate priors on reconstruction, these algorithms do not significantly boost CSI recovery quality because hand-crafted priors remain far from practice.

Summarily, three central problems are inherent in CS-based methods. First, they rely heavily on the assumption that channels are sparse in some bases. However, channels are not exactly sparse in any basis and may *not* even have an interpretable structure. Second, CS uses random projection and does not fully exploit channel structures. Third, existing CS algorithms for signal reconstruction are often iterative approaches, which have slow reconstruction. In the present study, we address the above problems using deep learning (DL). DL attempts to mimic the human brain to accomplish a specific task by training large multilayered neural networks with vast numbers of training samples. Our developed **CSI** sensing (or encoder) and recovery (or decoder) network is hereafter called CsiNet. CsiNet has the following features.

- *Encoder:* Rather than using random projection, CsiNet learns a transformation from original channel matrices to compress representations (codewords) through training data. The algorithm is agnostic to human knowledge on channel distribution and instead directly learns to effectively use the channel structure from training data.
- *Decoder:* CsiNet learns inverse transformation from codewords to original channels. Inverse transformation is non-iterative and multiple orders of magnitude faster than iterative algorithms.

A UE uses the encoder to transform channel matrices into codewords. Once the codewords are returned to the BS, it recovers the original channel matrices by using the decoder. The methodology can be used in FDD MIMO systems as a feedback protocol. In fact, CsiNet is closely related to the autoencoder [9, Ch. 14] in DL, which is used to learn a representation (encoding) for a set of data typically for dimensionality reduction. Recently, several DL architectures have been proposed to reconstruct natural images from CS measurements [10]–[12]. Although DL exhibits state-of-the-art performance in natural-image reconstruction, whether DL

Manuscript received March 1, 2018; accepted March 18, 2018. Date of publication March 22, 2018; date of current version October 11, 2018. The work of C.-K. Wen and W.-T. Shih 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, Hsinchu, Taiwan. The work of S. Jin was supported in part by the National Science Foundation for Distinguished Young Scholars of China under Grant 61625106, and in part by the National Natural Science Foundation of China under Grant 61531011. The associate editor coordinating the review of this paper and approving it for publication was Y. Gao. *(Corresponding author: Chao-Kai Wen.)*

C.-K. Wen and W.-T. Shih are with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung 804, Taiwan (e-mail: chaokai.wen@mail.nsysu.edu.tw; sydney2317076@gmail.com).

S. Jin is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: jinshi@seu.edu.cn).

Digital Object Identifier 10.1109/LWC.2018.2818160

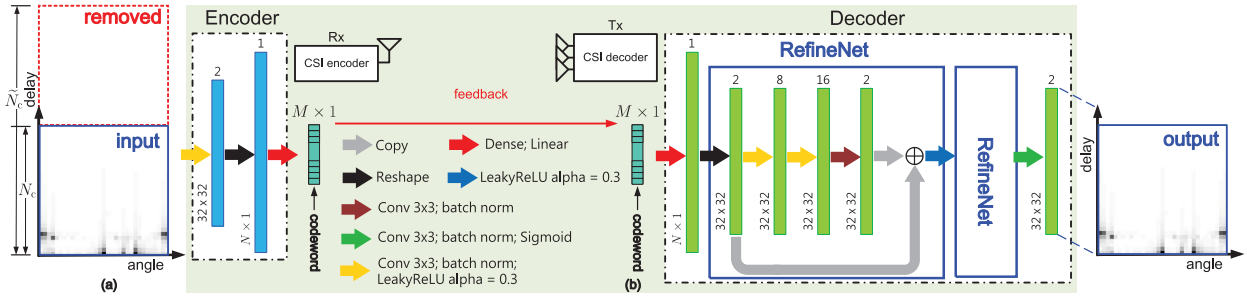

Fig. 1. (a) Pseudo-color plot of the strength of $\mathbf{H} \in \mathbb{C}^{32\times32}$. (b) Architecture of CsiNet, which includes the encoder and decoder.

can also show its ability in wireless channel reconstruction is unclear because this reconstruction is more sophisticated than image reconstruction. The present work is the first to suggest a DL-based CSI reduction and recovery approach.[1] The most relevant work appears to be [14], in which DL-based CSI encoding has been used in a closed-loop MIMO system. Different from [14], which has not considered CSI recovery, we show that CSI can be recovered with significantly improved reconstruction quality through DL compared with existing CS-based approaches. Even reconstructions at an excessively low compression rate retain sufficient content that allows effective beamforming gain.

II. System Model and CSI Feedback

We consider a simple single-cell downlink massive MIMO system with $N_t \gg 1$ transmit antennas at a BS and a single receiver antenna at a UE. The system is operated in OFDM over $\tilde{N}_c$ subcarriers. The received signal at the nth subcarrier is provided as follows:

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

where $\tilde{\mathbf{h}}_n \in \mathbb{C}^{N_t\times1}$, $\mathbf{v}_n \in \mathbb{C}^{N_t\times1}$, $x_n \in \mathbb{C}$, and $z_n \in \mathbb{C}$ denote the channel vector, precoding vector, data-bearing symbol, and additive noise of the nth subcarrier, respectively. Let $\tilde{\mathbf{H}} = [\tilde{\mathbf{h}}_1 \dots \tilde{\mathbf{h}}_{\tilde{N}_c}]^H \in \mathbb{C}^{\tilde{N}_c\times N_t}$ be the CSI stacked in the spatial frequency domain. The BS can design the precoding vectors $\{\mathbf{v}_n,\ n = 1, \dots, \tilde{N}_c\}$ once it receives $\tilde{\mathbf{H}}$ feedback. In the FDD system, the UE should return $\tilde{\mathbf{H}}$ to the BS through feedback links. The total number of feedback parameters is $N_t\tilde{N}_c$, which is not allowed for limited feedback links. Although downlink channel estimation is challenging, this topic is beyond the scope of this letter. We assume that perfect CSI has been acquired through pilot-based training [15] and focus on the feedback scheme.

To reduce feedback overhead, we propose that $\tilde{\mathbf{H}}$ can be sparsified in the angular-delay domain using a 2D discrete Fourier transform (DFT) as follows:

$$\mathbf{H} = \mathbf{F}_d \tilde{\mathbf{H}} \mathbf{F}_a^H, \tag{2}$$

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. To clarify this concept, a realization of the absolute values of $\mathbf{H}$ with the COST 2100 channel model [16] is depicted in Fig. 1(a). Parameterization is performed using a uniform linear array (ULA) with half-wavelength spacing in an indoor environment. The elements of $\mathbf{H}$ contain only a small fraction of large components, and the other components are close to zero. In the delay domain, only the first N_c rows of $\mathbf{H}$ contain values because the time delay between multipath arrivals lies within a limited period. Therefore, we can retain the first N_c rows of $\mathbf{H}$ and remove remaining rows. By an abuse of notation, we continuously use $\mathbf{H}$ to denote the $N_c \times N_t$ truncated matrix. The total number of feedback parameters can be reduced to $2N_cN_t$, which remains a large number in the massive MIMO regime.

In this letter, we are interested in designing the encoder

$$\mathbf{s} = f_{\mathsf{en}}(\mathbf{H}), \tag{3}$$

which can transform the channel matrix into an M-dimensional vector (codeword), where $M < N$. The data compression ratio is $\gamma = M/N$. In addition, we have to design the inverse transformation (decoder) from the codeword to the original channel, that is,

$$\mathbf{H} = f_{\mathsf{de}}(\mathbf{s}). \tag{4}$$

The CSI feedback approach is as follows. Once the channel matrix $\tilde{\mathbf{H}}$ is acquired at the UE side, we perform 2D DFT in (2) to obtain the truncated matrix $\mathbf{H}$ and then use the encoder (3) to generate a codeword $\mathbf{s}$. Next, $\mathbf{s}$ is returned to the BS, and the BS uses the decoder (4) to obtain $\mathbf{H}$. The final channel matrix in the spatial-frequency domain can be obtained by performing inverse DFT.

III. CsiNet

We exploit the recent and popular conventional neural networks (CNNs) for the encoder and decoder they can exploit spatial local correlation by enforcing a local connectivity pattern among the neurons of adjacent layers. The overview of the proposed DL architecture, named CsiNet, is shown in Fig. 1(b), in which the values $S_1 \times S_2 \times S_3$ denote the length, width, and number of feature maps, respectively. The first layer of the encoder is a convolutional layer with the real and imaginary parts of $\mathbf{H}$ being its input. This layer uses kernels with dimensions of 3×3 to generate two feature maps. Following the convolutional layer, we reshape the feature maps into a vector and use a fully connected layer to generate the codeword $\mathbf{s}$, which is a real-valued vector of size M. The first two layers mimic the projection of CS and serve as encoders. However, in contrast to random projections in CS, CsiNet attempts to translate the extracted feature maps into a codeword.

Once we obtain the codeword $\mathbf{s}$, we use several layers (as a decoder) to map it back into the channel matrix $\mathbf{H}$. The first

[1]For an overview of applying DL to the wireless physical layer, we refer the interested readers to [13].

layer of the decoder is a fully connected layer that considers $\mathbf{s}$ as input and outputs two matrices of size $N_c \times N_t$, which serve as an initial estimate of the real and imaginary parts of $\mathbf{H}$. The initial estimate is then fed into several "RefineNet units" that continuously refine the reconstruction. Each RefineNet unit consists of four layers, as shown in Fig. 1(b). In RefineNet unit, the first layer is the input layer. All the remaining 3 layers use 3×3 kernels. The second and third layers generate 8 and 16 feature maps, respectively, and the final layer generates the final reconstruction of $\mathbf{H}$. Using appropriate zero padding, the feature maps produced by the three convolutional layers are set to the same size as the input channel matrix size $N_c \times N_t$. The rectified linear unit (ReLU), $\mathsf{ReLU}(x) = \max(x, 0)$, is used as the activation function, and we introduce batch normalization to each layer.

Two features of a RefineNet unit are as follows. First, the output size of the RefineNet unit is equal to the channel matrix size. This concept is inspired by [10] and [11]. To reduce dimensionality, nearly all conventional implementations of CNNs involve pooling layers, which is a form of down-sampling. In contrast to conventional implementations, our target is refinement rather than dimensionality reduction. Second, in the RefineNet unit, we introduce identity shortcut connections that directly pass data flow to later layers. This approach is inspired by the deep Residual Network [12], [17], which avoids the vanishing gradient problem caused by multiple stacked non-linear transformations.

Experiments reveal that two RefineNet units produce good performance. Adding further RefineNet units does not significantly boost reconstruction quality but adds to computational complexity. Once the channel matrix has been refined by a series of RefineNet units, the channel matrix is input into the final convolutional layer, and the sigmoid function is used to scale values to the [0, 1] range. CsiNet can be extended to deal with cases involving multiple antennas at the UE by increasing the numbers of feature maps, i.e., S_3. We leave the exploitation of the spatial correlation across UE antennas as a topic for future studies.

To train CsiNet, we use end-to-end learning for all the kernel and bias values of the encoder and decoder. This training procedure differs from the two-step approach used in [12]. The set of parameters is denoted as $\Theta = \{\Theta_{\text{en}}, \Theta_{\text{de}}\}$. The input to CsiNet is $\mathbf{H}_i$, and the reconstructed channel matrix is denoted by $\widehat{\mathbf{H}}_i = f(\mathbf{H}_i; \Theta) \triangleq f_{\text{de}}(f_{\text{en}}(\mathbf{H}_i; \Theta_{\text{en}}); \Theta_{\text{de}})$ for the ith patch. Notably, the input and output of CsiNet are *normalized* channel matrices, whose elements are scaled in the [0, 1] range. Similar to the autoencoder, CsiNet is an unsupervised learning algorithm. The set of parameters is updated by the ADAM algorithm. The loss function is the mean squared error (MSE), which is calculated as follows:

$$L(\Theta) = \frac{1}{T}\sum_{i=1}^{T} \|f(\mathbf{s}_i; \Theta) - \mathbf{H}_i\|_2^2, \tag{5}$$

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

IV. Experiments

To generate the training and testing samples, we create two types of channel matrices through the COST 2100 channel model [16]: 1) the indoor picocellular scenario at the 5.3 GHz band, and 2) the outdoor rural scenario at the 300 MHz band.

TABLE I
NMSE IN dB AND COSINE SIMILARITY ρ

γ	Methods	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
	CS-CsiNet	-11.82	0.96	-6.69	0.87
	CsiNet	**-17.36**	**0.99**	**-8.75**	**0.91**
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
	CS-CsiNet	-6.09	0.87	-2.51	0.66
	CsiNet	**-8.65**	**0.93**	**-4.51**	**0.79**
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
	CS-CsiNet	-4.67	0.83	-0.52	0.37
	CsiNet	**-6.24**	**0.89**	**-2.81**	**0.67**
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
	CS-CsiNet	-2.46	0.68	-0.22	0.28
	CsiNet	**-5.84**	**0.87**	**-1.93**	**0.59**

All parameters follow their default setting in [16]. The BS is positioned at the center of a square area with lengths of 20 and 400m for indoor and outdoor scenarios, respectively, whereas the UEs are randomly positioned in the square area per sample. We use the ULA with $N_t = 32$ antennas at the BS and $\tilde{N}_c = 1024$ subcarriers. When transforming the channel matrix into the angular-delay domain, we retain the first 32 rows of the channel matrix. That is, $\mathbf{H}$ is 32×32 in size. The training, validation, and testing sets contain 100,000, 30,000, and 20,000 samples, respectively. All testing samples are excluded from the training and validation samples. We train several parameter sets with Glorot uniform initialization and then select the parameter set that provides minimal loss in the validation test. The epochs, learning rate, and batch size are set as 1000, 0.001, and 200, respectively.

We compare CsiNet with three state-of-the-art CS-based methods, namely, LASSO ℓ_1-solver [5], TVAL3 [7], and BM3D-AMP [8]. In all experiments, we assume that the optimal regularization parameter of LASSO is given by an oracle. Among these algorithms, LASSO provides the bottom-line result of the CS problem by considering only the simplest sparsity prior. TVAL3 is a remarkably fast total variation-based recovery algorithm that considers increasingly elaborate priors. BM3D-AMP is the most accurate compressive recovery algorithm in natural-image reconstruction. We also provide the corresponding results for CS-CsiNet, which only learns to recover CSI from CS measurements (or random linear measurements). The architecture of CS-CsiNet is identical to that of the decoder of CsiNet.

The difference between the recovered channel $\widehat{\mathbf{H}}$ and original $\mathbf{H}$ is quantified by a normalized MSE, which is defined as follows:

$$\text{NMSE} = \mathsf{E}\left\{\|\mathbf{H} - \widehat{\mathbf{H}}\|_2^2 / \|\mathbf{H}\|_2^2\right\}. \tag{6}$$

The feedback CSI serves as a beamforming vector. Let $\widehat{\tilde{\mathbf{h}}}_n$ be the reconstructed channel vector of the nth subcarrier. If $\mathbf{v}_n = \widehat{\tilde{\mathbf{h}}}_n / \|\widehat{\tilde{\mathbf{h}}}_n\|_2$ is used as a beamforming vector, then we achieve the equivalent channel $\widehat{\tilde{\mathbf{h}}}_n^H \tilde{\mathbf{h}}_n / \|\widehat{\tilde{\mathbf{h}}}_n\|_2$ at the UE side.

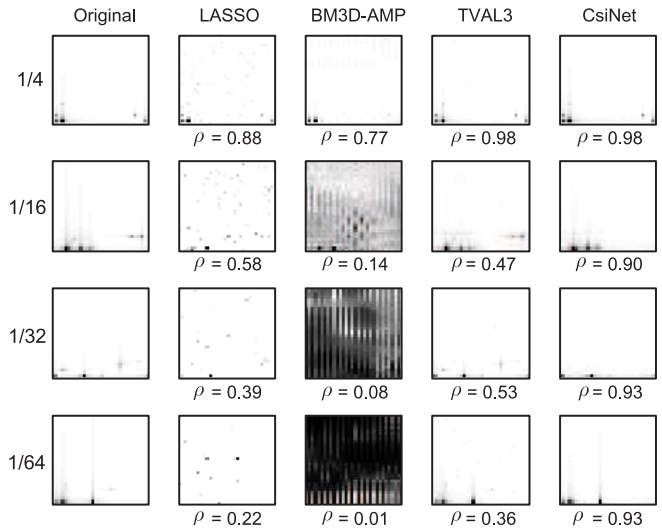

Fig. 2. Reconstruction images for different compression ratios by different algorithms in indoor picocellular scenarios.

To measure the quality of the beamforming vector, we also consider the cosine similarity

$$\rho = \mathsf{E}\left\{\frac{1}{\tilde{N}_{\mathrm{c}}}\sum_{n=1}^{\tilde{N}_{\mathrm{c}}}\frac{|\widehat{\tilde{\mathbf{h}}}_n^H\tilde{\mathbf{h}}_n|}{\|\widehat{\tilde{\mathbf{h}}}_n\|_2\|\tilde{\mathbf{h}}_n\|_2}\right\}. \tag{7}$$

Notably, when evaluating NMSE and ρ, we recover the output of CsiNet (i.e., the normalized channel matrix) back to their original levels.

The corresponding NMSE and ρ of all the concerned methods are summarized in Table I, with the best results presented in bold font. CsiNet obtains the lowest NMSE values and significantly outperforms CS-based methods at all compression ratios. Compared with CS-CsiNet, CsiNet also provides significant gains, which are due to the sophisticated DL architecture in the encoder and decoder. When the compression ratio is reduced to $1/16$, the CS-based methods can no longer function, whereas CsiNet and CS-CsiNet continue to perform well. Fig. 2 shows some reconstruction samples at different compression ratios along with the corresponding pseudo-gray plots of the strength of $\mathbf{H}$. CsiNet clearly outperforms the other algorithms.

Furthermore, CSI recovery through CsiNet can be executed with a relatively lower overhead than that through CS-based algorithms because CsiNet requires only several layers of simple matrix-vector multiplications. Specifically, the average running times (in seconds) of LASSO, BM3D-AMP, TVAL3, and CsiNet are 0.1828, 0.5717, 0.3155, and 0.0035, respectively. CsiNet performs approximately 52 to 163 times faster than CS-based methods.

Finally, we provide some observations without showing their experimental details. First, the DFT matrix $\mathbf{F}_{\mathrm{a}}$ that is used to transform $\tilde{\mathbf{H}}$ from the spatial domain into the angular domain is unnecessary. CsiNet can also exhibit similar performances without employing $\mathbf{F}_{\mathrm{a}}$ when retraining entire layers. This finding implies that CsiNet can be applied in other antenna configurations. Second, angular (or spatial) resolution increases with the number of antennas at the BS. Accordingly, the reconstruction performances of all the algorithms improve because $\mathbf{H}$ becomes sparser. CsiNet can be significantly improved because it is more capable of exploiting subtle changes among adjacent elements than CS-based methods.

V. Conclusion

We used DL in CsiNet, a novel CSI sensing and recovery mechanism. CsiNet performed well at low compression ratios and reduced time complexity. We believe that its reconstruction quality can be further improved by applying advance DL technology, and we hope this letter encourages future research in this direction.

References

[1] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[2] J. Zhang, C.-K. Wen, S. Jin, X. Gao, and K.-K. Wong, "On capacity of large-scale MIMO multiple access channels with distributed sets of correlated antennas," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 133–148, Feb. 2013.

[3] 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.

[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] 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. 57, no. 11, pp. 1413–1457, 2004.

[6] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 45, pp. 18914–18919, 2009.

[7] C. Li, W. Yin, and Y. Zhang. (2009). *TVAL3: TV Minimization by Augmented Lagrangian and Alternating Direction Algorithms*. [Online]. Available: http://www.caam.rice.edu/∼optimization/L1/TVAL3/

[8] 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.

[9] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[10] S. Lohit *et al.* (2017). *Convolutional Neural Networks for Non-Iterative Reconstruction of Compressively Sensed Images*. [Online]. Available: http://arxiv.org/abs/1708.04669

[11] A. Mousavi, G. Dasarathy, and R. G. Baraniuk. (2017). *DeepCodec: Adaptive Sensing and Recovery via Deep Convolutional Neural Networks*. [Online]. Available: http://arxiv.org/abs/1707.03386

[12] H. Yao *et al.* (2017). *DR*^{2}*-Net: Deep Residual Reconstruction Network for Image Compressive Sensing*. [Online]. Available: http://arxiv.org/abs/1702.05743

[13] T. Wang *et al.* (2017). *Deep Learning for Wireless Physical Layer: Opportunities and Challenges*. [Online]. Available: https://arxiv.org/abs/1710.05312

[14] T. J. O'Shea *et al.* (2017). *Deep Learning Based MIMO Communications*. [Online]. Available: https://arxiv.org/abs/1707.07980

[15] J. Choi, D. J. Love, and P. Bidigare, "Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 802–814, Oct. 2014.

[16] L. Liu *et al.*, "The COST 2100 MIMO channel model," *IEEE Wireless Commun.*, vol. 19, no. 6, pp. 92–99, Dec. 2012.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Las Vegas, NV, USA, 2016, pp. 770–778.