Compressed Sensing via a Deep Convolutional Auto-encoder

Hao Wu[1,*], Ziyang Zheng[1,*], Yong Li[1,*] Wenrui Dai[2,†], Hongkai Xiong[1,*]
[1]*Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China*
[2]*School of Biomedical Informatics, University of Texas Health Science Center at Houston, Houston, Texas 77030, USA*
[*]{wu_hao, doll001, marsleely, xionghongkai}@sjtu.edu.cn, [†]Wenrui.Dai@uth.tmc.edu

***Abstract*—The nonlinear recovery is not promising in accuracy and speed, which limits the practical usage of compressed sensing (CS). This paper proposes a deep learning-based CS framework which leverages a deep convolutional auto-encoder for image sensing and recovery. The utilized auto-encoder architecture consists of three components: the fully convolutional network acts as an adaptive measurement matrix generator in the encoder; while in the decoder, the deconvolution network and refined reconstruction network are learned for intermediate and final recovery, respectively. Different from most previous work focusing on the block-wise manner to reduce implementation cost but result in blocky artifacts, our adaptive measurement matrix is applicable to any size of scene image and the decoder network reconstructs the whole image efficiently without any blocky artifacts. Moreover, dense connectivity is leveraged to combine multi-level features and alleviate the vanishing-gradient problem in the refined reconstruction network which boosts the performance on image recovery. Compared to the state-of-the-art methods, our algorithm improves more than 0.8 dB in average PSNR.**

***Index Terms*—Compressed sensing, Adaptive measurement matrix, Convolutional auto-encoder, Image reconstruction**

I. Introduction

Compressed sensing (CS) [1] is a well-studied research topic in signal processing. The CS theory indicates that robust signal recovery can be obtained from far fewer samples than required by the Shannon-Nyquist sampling theorem. The sparse representation of the sampled signal can be exploited by solving the optimization problem in underdetermined linear systems.

A variety of compressed sensing and recovery methods have been developed for images and videos. The "single pixel camera" [2] is one of the most impressive inventions based on the compressed sensing, which uses only one detector to collect the linear projection and recovers the entire image using compressed sensing reconstruction algorithms. However the reconstruction often takes an amount of time due to the vectorization of high-dimensional signals. Thus, block based CS (BCS) [3] was proposed to sense and reconstruct the scene image in block-wise manner, where the scene image was divided into non-overlapping blocks with the same size. For refined improvement, a multi-scale version of BCS-SPL that deploys BCS within the domain of a wavelet transform was proposed in [4]. For the sensing phase, random Gaussian measurement

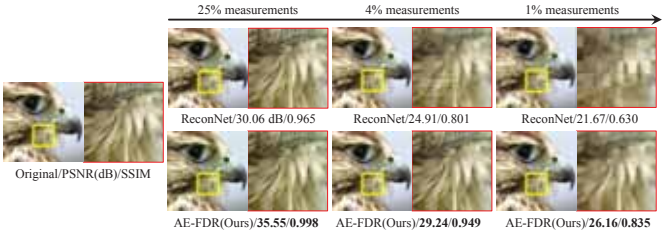

Fig. 1. Proposed reconstruction results compared to ReconNet [5] and ground truth. It is clearly seen that blocky artifacts could be eliminated and the reconstruction performence is boosted using our auto-encoder network.

matrix is often used to obtain the linear measurements. To our best knowlege, there are two mainly obvious drawbacks of the above CS methods. One is high computational complexity caused by non-linear optimization problem for image recovery. The other is blocky artifacts caused by block-wise manner.

Recently, convolutional neural networks (CNNs) [6], [7] have been leveraged to overcome the drawback of time cost in image reconstruction from CS measurements [5], [8]. ReconNet [5] was proposed to learn the mapping from CS measurements back to the original image by utilizing CNNs [6], which replaces the traditional iterative reconstruction algorithm. However, ReconNet does not involve any method of designing measurement matrix. Although the non-iterative reconstruction is fast, the reconstruction performance is unsatisfactory and suffers from blocky artifacts. Later in [8], the fully connected network was used in both sensing and reconstruction, which successfully reconstructed high quality image. However, since the fully connected layers fix the input dimension, the work still maintains block-wise CS manner and suffers from blocky artifacts.

In this paper, we propose a deep convolutional auto-encoder for image compressed sensing and reconstruction. By removing the non-linear modules such as batch normalization [9] layers and non-linear activation layers from the encoder network, we achieve generating the adaptive measurement matrix by a 4-layer fully convolutional network. The adaptive measurement matrix and the decoder network can apply to any size of the scene image so that sensing and recovery always perform on the whole image which avoids blocky artifacts as shown in Fig. 1. In our refined reconstruction network, dense connectivity is used to combine multi-level features for final reconstruction. It alleviates the vanishing-gradient problem and

978-1-5386-4458-4/18/$31.00 ©2018 IEEE

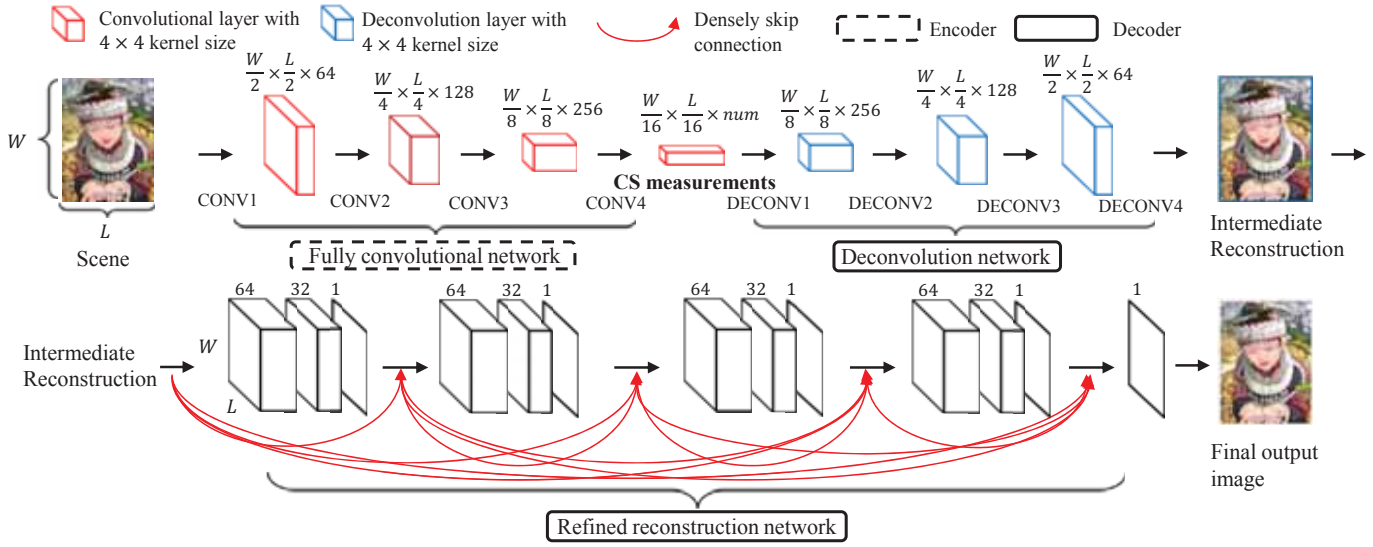

Fig. 2. Three components of our auto-encoder. The fully convolutional network contains 4 convolutional layers to learn adaptive measurement matrix. Symmetric with the fully convolutional network, the deconvolution network is employed for intermediate reconstruction. The refined reconstruction network contains 13 convolutional layers with dense connectivity. The relation of num with measurement rate is defined by Equation 3. All these three components are described in section II.

makes our architecture more effective for image recovery. Compared to the state-of-the-art method, the performance of proposed auto-encoder with fully convolutional network, deconvolutional network and refined reconstruction network (**AE-FDR**) improves more than 0.8dB in terms of PSNR. Even without the refined reconstruction network, the auto-encoder with fully convolutional network and deconvolutional network (**AE-FD**) achieves generating high quality reconstruction with faster speed and lower computational cost.

II. Proposed Method

Given vectorized image $x \in \mathbb{R}^N$, and linear measurement matrix $\Phi \in \mathbb{R}^{M \times N}(M < N)$, the measurements of the image are obtained by $y = \Phi x$.

Instead of using random Gaussian matrix, this paper aims to learn an adaptive measurement matrix Φ with a deep convolutional auto-encoder. Fig. 2 depicts the proposed auto-encoder framework for compressed sensing. Section II-A elaborates that the encoder leverages fully convolutional networks to adaptively generate measurements y. The decoder consists of the deconvolution network and the refined reconstruction network for intermediate and final reconstruction, shown in Section II-B and II-C.

A. Adaptive measurement matrix

Different from most previous deep auto-encoders, our encoder network consists of a 4-layer fully convolutional network, from which we extract non-linear modules for linear sensing. According to [10] convolutional operation in deep neural network is equivalent to multiply the vectorized input by the weight matrix. Denote by $F_k(\alpha)$ the output of the k^{th} convolutional layer with weight matrix w_k and bias vector b_k.

$$F_k(\alpha) = w_k\alpha + b_k \tag{1}$$

where α is the vectorized input of k^{th} layer. Thus, the output $F(x)$ of the 4-layer fully convolutional network without any non-linear module is formulated by:

$$\begin{aligned} F(x) &= w_4\left(w_3\left(w_2\left(w_1x + b_1\right) + b_2\right) + b_3\right) + b_4 \\ &= w_4w_3w_2w_1x + w_4w_3w_2b_1 + w_4w_3b_2 + w_4b_3 + b_4 \end{aligned} \tag{2}$$

Equation (2) implies that the measurement matrix can be adaptively determined by $\Phi = w_4w_3w_2w_1$ while $w_4w_3w_2b_1 + w_4w_3b_2 + w_4b_3 + b_4$ is a constant vector after the network convergence. Consequently, CS measurements can be obtained from the 4-layer fully convolutional network. It is worth mentioning that the proposed compressed sensing method can suppress blocky artifacts, as it can avoid block-wise processing by adapting to any size of scene image. In practice, we set kernel size to 4×4 and stride to 2 in each layer of the fully convolutional network and adopt zero-padding for each side of input by one pixel. The input downscales through each convolutional layer with scale factor of $2\times$. Thus, the size num of output feature maps of the fully convolutional network can be determined based on the measurement rate R.

$$num = \lfloor 256 \times R + 0.5 \rfloor \tag{3}$$

Here, $\lfloor \cdot \rfloor$ stands for the rounding down operation.

B. Deconvolution Network

We design the deconvolution network symmetric to the fully convolutional network for intermediate reconstruction. We utilize a 4-layer deconvolution network to learn up-scaling filters and obtain an intermediate image reconstruction $\bar{x}$ from measurements y:

$$\bar{x} = D(y) \tag{4}$$

TABLE I
AVERAGE PSNR AND SSIM FOR 10 TEST IMAGES AT DIFFERENT MEASUREMENT RATES ($R = \frac{M}{N}$) OBTAINED BY THE PROPOSED METHOD (AE-FD, AE-FDR), MS-BCS-SPL [4], BCS-DL [8] AND RECONNET [5], RESPECTIVELY. THE RESULTS IS REPRESENTED AS PSNR|SSIM.

Algorithm	R=0.25	R = 0.10	R = 0.04	R = 0.01
MS-BCS-SPL	31.14|0.874	27.34|0.757	24.64|0.626	23.32|0.573
BCS-DL	32.15|0.976	28.21|0.916	25.13|0.806	22.36|0.607
ReconNet	26.93|0.876	24.54|0.774	22.51|0.650	19.68|0.483
AE-FD (Ours)	32.07|0.976	28.58|0.922	26.04|0.834	23.52|0.678
AE-FDR (Ours)	**32.55|0.976**	**28.91|0.926**	**26.27|0.839**	**23.61|0.683**

TABLE II
AVERAGE RUN-TIME(IN SECONDS) FOR RECONSTRUCTING AN IMAGE OF 512×512 PIXELS.

Algorithm	R=0.25	R=0.1	R=0.04	R=0.01
MS-BCS-SPL	5.5593	12.8645	20.0781	7.2037
BCS-DL	0.1061	0.1053	0.1045	0.1002
ReconNet	0.0251	0.0254	0.0251	0.0243
AE-FD (Ours)	**0.0246**	**0.0248**	**0.0227**	**0.0237**
AE-FDR (Ours)	0.1036	0.1029	0.1031	0.0997

where $D(y)$ indicates the projection from y onto $\bar{x}$ using the proposed deconvolution network. It should be noted that, contrary to the fully convolutional network for linear sensing, the deconvolution network is used for non-linear reconstruction from CS measurements back to original image. Therefore, Batch Normalization layers and ReLU layers are introduced to accelerate deep network training and nonlinear mapping. Given the training set of N pairs of images and measurements (x_i, y_i), $i = 1, \cdots, N$, the intermediate reconstruction is derived from the deconvolution network by minimizing the Mean Squared Error (MSE):

$$L(\{W^F, W^D\}) = \frac{1}{N}\sum_i^N \|D(y_i) - x_i\|^2 \tag{5}$$

Here W^F and W^D are the weights for the proposed fully convolutional network and deconvolution network.

C. Refined Reconstruction Network

The role of our refined reconstruction network is to refine the quality of reconstruction image. The architecture is shown in Fig. 2, which totally consists of 13 convolutional layers. We construct refined reconstruction network by applying convolutional blocks and introduce dense connectivity [7]. Each convolutional blocks comprise three convolutional layers. With the dense connectivity, the i^{th} layer uses the feature-maps of all preceding layers, $X_0, X, \cdots, X_{i-1}$ as input:

$$X_i = max(0, F_i([X_0, X_1, \cdots X_{i-1}])) \tag{6}$$

where $[X_0, X_1, \cdots X_{i-1}]$ denotes the concatenation of the feature maps of preceding layers $0, 1 \cdots, i-1$. max refers to the operation of ReLU layer. We introduce this strategy to encourage feature reuse and boost our network performance in reconstruction. Specifically, the intermediate reconstruction as low-level feature is concatenated to all the convolutional blocks and reconstruction layer by dense connectivity. Given a training image data x_i, let $\varphi(x_i)$ denotes output of our auto-encoder and W^R denotes the weights of refined reconstruction network. We minimize the following MSE to obtain the optimum weights of our network:

$$L(\{W^F, W^D, W^R\}) = \frac{1}{N}\sum_i^N \|\varphi(x_i) - x_i\|^2 \tag{7}$$

III. EXPERIMENT

In this section, we discuss the implementation details and evaluate the performance of the proposed method.

A. Datasets and Settings

The training dataset consists of 1,200,000 image patches of size 64×64 randomly extracted from ImageNet. Similar to most CS methods, we mainly focus on the reconstruction performance on the luminance component. In addition, We find that the models trained in the luminance channel can also be used in both Cb and Cr channels. The extensive experiments in color image are shown in supplemental files. The proposed methods are evaluated on the test images described in [8].

Our networks are implemented using Caffe [11] and optimized using Adam [12]. The training phase consists of two stages. In the first stage, we train the model AE-FD, in which W^F and W^D are initialized using the method [13] and learned to reconstruct an intermediate reconstruction. The initial learning rate is set to 0.0001 and step strategy is used to update learning rate with step-size 400,000, gamma 0.1 and the maximum number of iterations 800,000. In the second stage, we train the model AE-FDR and update W^F, W^D, W^R to obtain final reconstruction. The weights are initialized using pre-trained AE-FD to accelerate training. The initial learning rate is set to 0.0001. Step strategy is applied with step-size 300,000, gamma 0.1 and the maximum number of iterations 450,000. The batch-size is set to 32 during both training stages.

B. Comparison with the State-of-the-art Methods

The performance of our algorithms is compared with three state-of-the-art CS algorithms: MS-BCS-SPL [4], ReconNet [5] and BCS-DL [8]. MS-BCS-SPL is a classical CS method. We directly use the code provided by the author and the parameters are set to default values. ReconNet and BCS-DL are two deep learning-based algorithms. For ReconNet, the author trained the model with 21760 image patches of size 33×33 which are much smaller than ours. In order to eliminate the impact from different size of training datasets, we retrain the ReconNet with 5,000,000 image patches of size 33×33 randomly extracted from ImageNet. After convergence, the performance of ReconNet improves about 2dB in PSNR . As for BCS-DL, the models were trained with a large training set, so we just supplemented the experiments at measurement rate 0.01 and 0.04. Table I evaluates the proposed AE-FD and AE-FDR and the state-of-the-arts in terms of PSNR (dB) and SSIM under measurement rate 0.25, 0.1, 0.04 and 0.01. It shows that both AE-FD and AE-FDR obviously outperform ReconNet and BCS-DL. Further, we show some visual experimental results in Fig. 3. More experimental results on luminance component are shown in supplemental files.

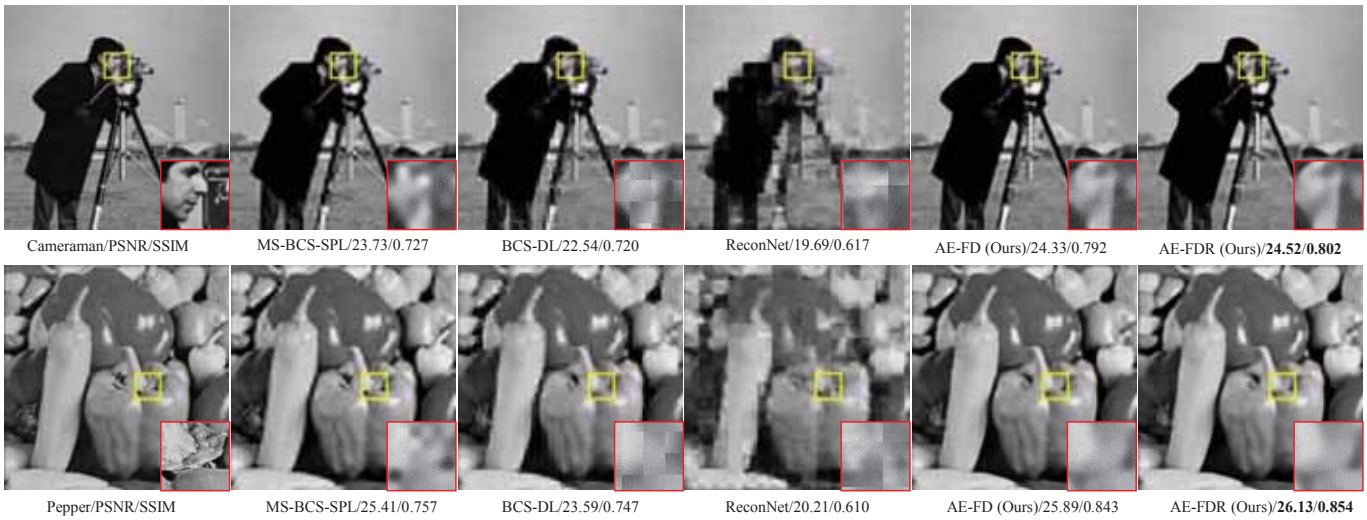

Fig. 3. Reconstruction results of 'cameraman' and 'peppers' at measurement rate 0.01. Compared to the state-of-the-art methods, we reconstruct more smooth results (last two columns) at the low measurement rate.

Time complexity is evaluated on the average run-time for reconstructing a test image of 512×512 pixels shown in Table. II. For MS-BCS-SPL, An Intel(R) Core(TM) i5-4590 CPU is used to perform the implementations provided by the authors. And we run the deep learning-based algorithms (BCS-DL, ReconNet, AE-FD, AE-FDR) using a NVIDIA Titan XP GPU. Compared to MS-BCS-SPL, the average speedup of our method (AE-FD, AE-FDR) is over 400 times and 50 times, respectively. One reason of speedup is based on the parallel computing using GPU. But the most important reason is because our methods perform non-iterative reconstruction process. Next, we compare our methods with state-of-the-art deep learning-based algorithms (BCS-DL, ReconNet). It is obviously concluded that our algorithm AE-FD outperforms the state-of-the-art CS reconstruction methods in time complexity.

IV. Conclusions

In this paper, we propose a deep convolutional auto-encoder to perform compressed sensing. By removing non-linear modules, we achieve learning adaptive measurement matrix using a 4-layer fully convolutional network. The adaptive measurement matrix allows our method to avoid block-wise processing and blocky artifacts. The model AE-FD achieves generating high quality reconstruction with fast speed which meet the need for real-time reconstruction. By introducing the dense connectivity into refined reconstruction network, the reconstruction performance of our model AE-FDR exceeds the state-of-the-art methods by more than 0.8 dB in average PSNR.

V. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 61425011, Grant 61720106001, Grant 61529101, and in part by Shanghai High Technology Project under Grant 17511106603 and the Program of Shanghai Academic Research Leader under Grant 17XD1401900.

References

[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[2] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 83–91, 2008.

[3] G. Lu, "Block compressed sensing of natural images," in *International Conference on Digital Signal Processing*. IEEE, 2007, pp. 403–406.

[4] J. E. Fowler, S. Mun, and E. W. Tramel, "Multiscale block compressed sensing with smoothed projected landweber reconstruction," in *Signal Processing Conference, 2011 19th European*. IEEE, 2011, pp. 564–568.

[5] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "Reconnet: Non-iterative reconstruction of images from compressively sensed measurements," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 449–458.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[7] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, vol. 1, 2017, p. 3.

[8] A. Adler, D. Boublil, and M. Zibulevsky, "Block-based compressed sensing of images via deep learning," in *IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, 2017, pp. 1–6.

[9] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, 2015, pp. 448–456.

[10] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *arXiv preprint arXiv:1603.07285*, 2016.

[11] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[12] D. Kinga and J. B. Adam, "A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.