Fast Reconstruction of 1D Compressive Sensing Data Using a Deep Neural Network

Youhao Yu[1,2] and Richard M. Dansereau[1]
[1]Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada
[2]School of Information Engineering, Putian University, Fujian, China
Email: {youhaoyu, rdanse}@sce.carleton.ca

***Abstract*—A deep neural network is used to recognize the nonzero positions of a one-dimensional signal in its sparse domain. Unlike classical data reconstruction methods in Compressive Sensing (CS), such as basis pursuit or recast as a linear programming problem and solved with Primal-Dual Interior Point Method (PDIPM), the proposed data reconstruction method is inspired by the performance of Convolutional Neural Networks (CNNs) on image edge detection. A CNN is expected to find the nonzero positions of a sequence in the sparse domain. The proposed method trains the CNN with deep residual learning [1] and takes the Half-Mean-Squared-Error (HMSE) as the loss function. It is difficult with a CNN to get accurate amplitude of nonzero points directly, but the CNN finds the nonzero positions efficiently. When the nonzero positions are found, Lower–Upper (LU) matrix factorization with partial pivoting can be used to acquire accurate CS reconstruction. The experiments show that the proposed method operates with higher speed and reconstruction accuracy than competing methods.**

***Index Terms*—compressive sensing, CNN, primal-dual interior point, data reconstruction**

I. INTRODUCTION

According to the Shannon-Nyquist theorem, the sampling rate of an analog signal must be at least twice the highest analog frequency component [2], otherwise aliasing will distort the reconstructed signal. Compressive sensing has been proposed in recent years by Donoho [3], Candes [4], et al. With CS, data is collected and compressed simultaneously, which breaks through the barrier of Nyquist sampling rate of the classical sampling theorem. CS saves on memory storage and the original signal can be reconstructed with fewer measurements. Though CS has been widely used in many areas, reconstruction accuracy and speed are still challenges [5], [6].

There are numerous data reconstruction approaches for compressive sensing, such as basis pursuit [7], matching pursuit [8], iterative thresholding [9], and interior point method [10]. These approaches are computationally expensive due to the necessary iterations [11], which limits the applicability of these approaches for real-time signal processing. Data reconstruction based on neural networks is non-iterative after training, so may be faster than traditional methods [12]-[14]. Li and Wei converted the basis pursuit denoising model into a quadratic programming problem and implemented signal reconstruction based on recurrent neural networks [15], but they did not analyze its overall performance. In [12], a modified U-Net CNN architecture is used for image data, but the data reconstruction accuracy and robustness are limited. A deep residual reconstruction network is proposed in [11] to reconstruct an image from its CS measurements; it has low signal-to-noise ratio (SNR). In [16], an autoencoder architecture has been applied to data reconstruction. The method has poor 1D data reconstruction performance even though we tried implementing it with additional fully connected layers.

In this paper, the goal is to develop a network with faster speed than traditional CS reconstruction methods but still with good reconstruction quality. We propose a data reconstruction algorithm based on a deep neural network with residual learning and convolutional layers. To achieve the higher speed and quality, the neural network is used only to identify nonzero positions in the signal and LU matrix factorization with partial pivoting is used to acquire an accurate reconstruction.

The rest of the paper is organized as follow. Section II briefly reviews the reconstruction of compressive sensing and neural networks. In Sec. III, we proposed a neural network-based reconstruction method. The experimental results are reported in Sec. IV and conclusions given in Sec. V.

II. RELATED WORK

A. Compressive Sensing

According to compressive sensing theory, a measurement y of a signal x can be represented as

$$y = \Phi x \tag{1}$$

where $x \in \mathbb{R}^{N\times 1}$ is the original signal, $y \in \mathbb{R}^{M\times 1}$ is the measurement data, and M and N are integers where $M << N$. $\Phi \in \mathbb{R}^{M\times N}$ is the measurement matrix which should satisfy the restricted isometry property (RIP) [17]. x should be sparse or compressible on some domain such that

$$x = \Psi s \tag{2}$$

Manuscript received November 28, 2019; revised February 28, 2020.

doi: 10.18178/ijsps.8.1.26-31

where $\Psi \in \mathbb{R}^{N \times N}$ is a transform matrix for a sparsifying domain, and $s \in \mathbb{R}^{N \times 1}$ is k-sparse with only k non-zero elements [3].

B. *Data Reconstruction*

Many iterative recovery algorithms exist for CS, such as matching pursuit [18], orthogonal matching pursuit [19], iterative soft-thresholding [20], and iterative hard-thresholding [21]. To recast the problem as a linear program and use the primal dual interior point method (PDIPM) to solve it is a mature technology for CS reconstruction [22]-[24]. The problem can be described as

$$\min \|s\|_1 \quad s.t. \;\; y = As \tag{3}$$

where $A = \Phi\Psi$ and $\|s\|_1 = \sum_i |s_i|$; (3) is also known as basis pursuit. After finding the vector s with the smallest l_1 norm, x is determined using (2). Since y, A, and s are real, the problem can be recast as the linear program

$$\min_{s,u} \sum_i u_i \;\; s.t. \; -u_i \le s_i \le u_i, y = As \tag{4}$$

which can be solved using the primal-dual algorithm [10].

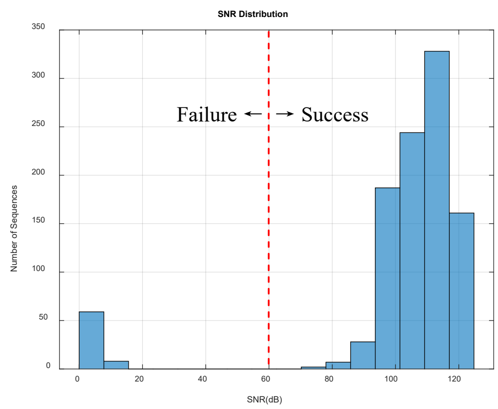

Figure 1. SNR distribution of sequences reconstructed by PDIPM.

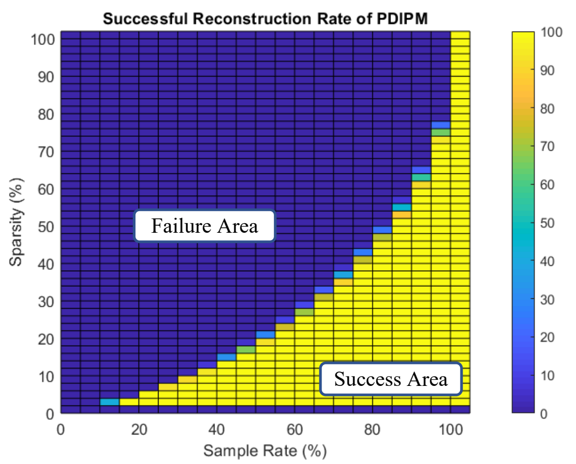

Figure 2. Reconstruction success with different sampling rate and sparsity.

Experiments show that when a signal is successively reconstructed by the PDIPM method, SNR is usually near 100 dB. If reconstruction fails, SNR is usually less than 30 dB. For example, we produced 1024 sequences with 1024 random points that are sparse in time domain. Using a compressive sample rate of 50% and sparsity rate of 18%, Fig. 1 shows that SNR has clear separation between successful and failed reconstructions. Fig. 2 shows the relationship of the compressive sensing sampling rate with the sparsity when PDIPM is used to reconstruct the original signal over 10,000 sequences. If we only use PDIPM to reconstruct the original signal, experiments show that after about 10 iterations the surrogate duality gap (SDG) will be small and the SNR will be around 100 dB. In Table I, when sparsity is 20% and tolerance for primal-dual algorithm set to 1e-8, the mean of SDG is 0.002 after 15 iterations. It could not always converge to accurate results since the reciprocal of the condition number of the matrix in the method becomes too small (less than 1e-14), but the SNR is near 100 dB. Experiments show that after 9 iterations the non-zero positions of all the sequences can be found by choosing the largest 20% of values, even though SDG is still large. It is well known that iterations are time consuming, so if the nonzero positions of the sparse signal were found, we can use LU matrix factorization with partial pivoting to get results. Fig. 3 shows the maximum number of iterations that PDIPM needs to find the non-zero positions over sparsity.

TABLE I. ITERATIONS VS. SURROGATE DUALITY GAP

Primal Dual Interior Point Method[a]	
Number of iterations	Mean of surrogate duality gap
9	220.604
15	0.002

a. Sparsity=20%, Tolerance=1e-8

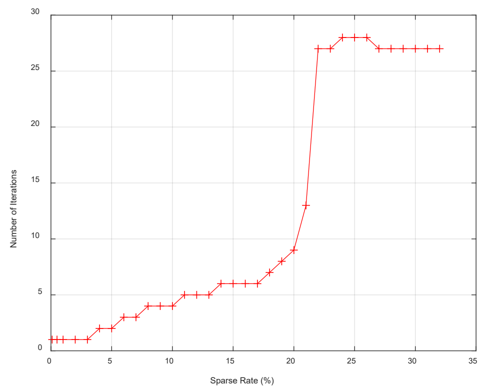

Figure 3. Number of iterations needed to find nonzero positions.

C. *Deep Residual Network*

The deep residual network (ResNet) has shown promising performance in image recognition [25]. Fig. 4 shows the basic framework [1] where a secondary link allows the layers to learn only residual changes. The stacked nonlinear layers fit a residual mapping $F(x)$. The original desired underlying mapping $H(x)$ is $F(x)+x$. It has been shown that residual networks are easier to optimize and can gain accuracy from considerably increased depth [1].

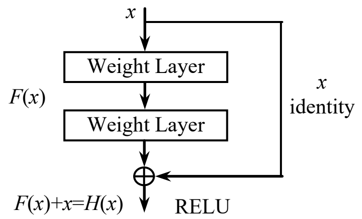

Figure 4. Residual learning network: a building block.

III. PROPOSED CS RECONSTRUCTION METHOD

Without loss of generality, we assume that the original signal is sparse, so the transform matrix Ψ is unitary. The CNN shown in Fig. 5 is developed to reconstruct the original signal. After the input layer, the measurement data with length M is converted to length N by linear mapping, which is a custom layer without learnable parameters. It is implemented by multiplying the input sequence with the measurement matrix Φ to generate correlations between y and the columns of Φ. If the original signal is very sparse, the non-zero positions can often be determined as the k largest values from this layer. In an experiment with 20,000 sequences with 5% sparsity, results show that this first layer determines the non-zero positions in more than 95% of these sequences for a Φ with i.i.d. Guassian elements. When sparsity increases to 13%, only 51.61% of the sequences have all non-zero positions found correctly.

To refine the location of non-zero positions, additional layers are used in Fig. 5, including convolution, batch normalization, and parametric rectified linear unit (PReLU) [26] layers. The convolution layer convolves the input with filters along the sequence with stride of 1, adds a bias term, and includes zero padding so that the output has the same size as the input. A normalization layer is used to reduce network training time and sensitivity to network initialization [27]. For optimal output, normalization also shifts and scales the activations. The PReLU layer is implemented by a custom deep learning layer. It performs as a threshold operator since any input value less than zero is weighted by a scalar learned at training time.

Because x (or s) is k-sparse ($k << M$), a custom layer is used before the output of the network to keep the M largest values. A ReLU layer beforehand only keeps positive values and makes many elements zero before selecting the M largest values. The k nonzero values are usually included in the M largest values, so using the measurement matrix, compressed data and LU matrix factorization, accurate reconstruction can be done with high probability. Only the nonzero positions (number between k and M) are needed using LU matrix factorization with partial pivoting to acquire the reconstruction. It contributes to the signal reconstruction by shrinking the size of matrix, reducing the probability of a singular matrix and making the operation faster.

IV. EXPERIMENTS

We carried out the following experiments on a laptop with Intel Core i7-6700HQ 2.60 GHz CPU, NVIDIA GeForce GTX 960M display adapters and 16 GB memory. All experiments were performed using MATLAB version R2018b.

A. Produce Data for Training and Testing

Suppose a sparse signal is a sequence with length 1024 and only 10% of its elements are non-zero. The non-zero positions follow a random uniform distribution in the sequence. Assume M=608, which means the sampling rate nearly equal to 60%. For convenience, a measurement matrix is built with all orthonormal rows [10] and size 608×1024.

B. Building and Training A Residual CNN

Fig. 5 shows the residual CNN we propose and has 28 layers. If we were to remove the two addition layers and their connection lines labeled *a* and *b*, the structure would be a plain network instead of a residual network. The input are sequences with length 608 and the output will be the corresponding sparse sequences with length 1024. There are 5 convolution layers and each has four filters of size 1×5. Although the residual network has additional connections than a plain non-residual network, they have the same complexity and both have 4,199,844 learnable parameters. Most of these parameters are concentrated in the two fully connected (FC) layers. Grouping every 64 sequences as a mini batch, the training data is shuffled after every epoch. If selecting M points from a sparse sequence includes the k nonzero points, then the signal is correctly reconstructed with high probability.

For the plain CNN, we produced 2^{18} training sequences and 2^{16} test sequences. The maximum epoch is 80. The initial learning rate is 0.001 and is dropped by 10% after every 30 epochs. The network needed around 16 hours to train and 99.90% of the test sequences had all k nonzero points included in the M points by the neural network.

For the residual CNN, we produced 2^{15} training sequences and 2^{14} test sequences. The maximum epoch is 30. The initial learning rate is 0.001 and is dropped by 10% after every 25 epochs. The network needed only about 1 hour to train and all the test sequences had the k nonzero points included in the M points by the neural network. So, the architecture improves training speed and has greater reliability.

C. Experiments with Five Competing Methods

Five methods are used to implement data reconstruction. The number of sequences used ranges from 100 to 1000 in steps of 100. All methods can reconstruct the signal successfully, but Fig. 6 and Fig. 7 show different elapsed times and SNRs.

1) PDIPM

Only use the PDIPM method to reconstruct the original sparse signal [10]. The row vector of the measurement multiplied with the measurement matrix is used as the initial point. The method needs more than ten iterations and the SNR reaches about 120 dB. The elapsed time of data reconstruction are linearly increasing with number of sequences.

2) CNN-PDIPM

CNN network is used to predict the original signal first. The regression is not accurate, although the M largest values include the k non-zero positions can be found. Taking the predicted result as the initial value of the PDIPM method, the elapsed time of data reconstruction still increases linearly with the number of sequences and it is a little slower than only use PDIPM method.

3) PDIPM-LUP

Using PDIPM method first, after at most 5 iterations the non-zero positions will be found just as the k largest values. Then using lower–upper (LU) matrix factorization and partial pivoting the non-zero values could be solved accurately. The elapsed time of data reconstruction reduces significantly.

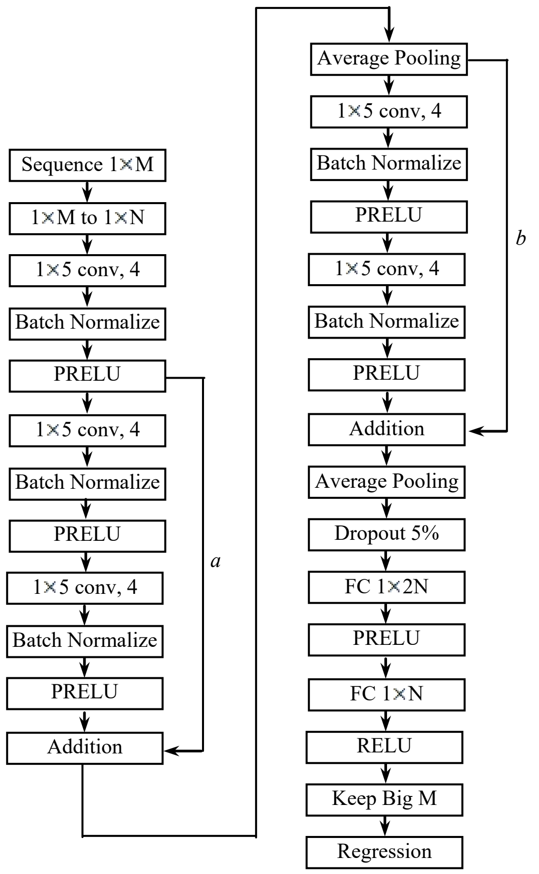

Figure 5. The architecture of residual CNN.

4) FPC-AS

The fixed-point continuation active set algorithm (FPC-AS) is used. It based on shrinkage, subspace optimization and continuation [5]. It has two stages, which estimates the non-zero positions and solves the values repeatedly. The algorithm has excellent performance in terms of speed and ability to reconstruct sparse signals.

5) CNN-LUP

The proposed CNN network is used to forecast the non-zero positions by keeping the M largest values of the output. Although the CNN could not predict the value with high precision, it can find the positions of nonzero points very quickly. When the positions are determined, the accurate value is solved effectively with LU matrix factorization and partial pivoting method. The elapsed time is small compared with other methods. It is quicker than FPC-AS and has better precision than FPC-AS, but based on other experiments, we are not able to reconstruct a signal with as high sparsity and low sampling rate as FPC-AS.

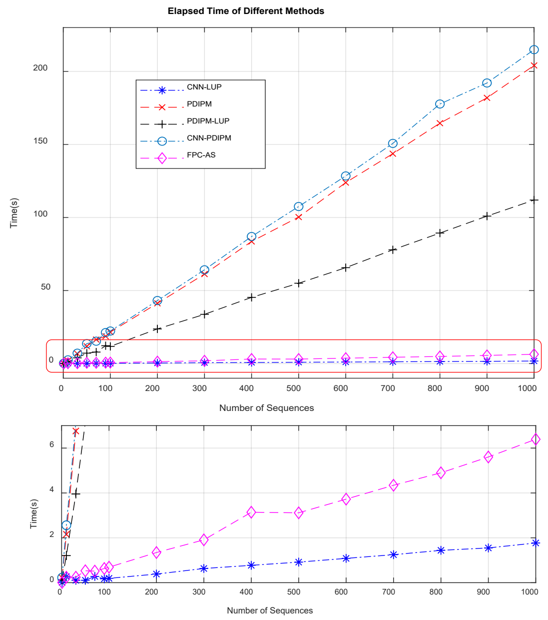

Figure 6. Elapsed time of different methods.

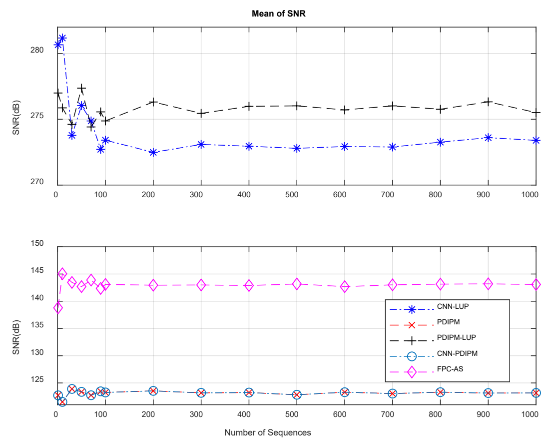

Figure 7. Reconstruction quality of different methods.

V. Conclusions and Future Work

Traditional CS reconstruction algorithms solve an optimization problem with iterations, often with expensive computation. The processing time is a bottleneck for the real time applications. In this paper, deep learning-based approaches have been applied to compressive sensing data reconstruction and work much faster than traditional algorithms. Experimental results show it is difficult to get accurate values directly by neural networks, but it is convenient to recognize the nonzero positions of the sparse signal or the big amplitude positions of the compressible signal in their sparse domain. Then by means of LU matrix factorization with partial pivoting, the signal can be reconstructed with higher accuracy and faster speed than the other methods. Although it takes time to train the neural network, it could not precisely predict the k non-zero positions. Usually the prediction has k to M non-zero positions. The redundancy in estimated positions nearly assures the k non-zeros are included, so it is not as accurate as the third method and the SNR is slightly worse. In the future, we would like to research the signal sparsity in other domains with noise, and we will visualize the results of every layer of the neural network to analyze which are the useful features.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Youhao Yu and Professor Dansereau conducted the research. Youhao Yu analyzed the data and wrote the paper. Professor Dansereau supervised the work and put forward valuable suggestions on how to improve it. All authors had approved the final version.

Acknowledgment

The authors wish to thank Natural Sciences and Engineering Research Council of Canada (NSERC). This work was supported through an NSERC Discovery Grant.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 770-778.

[2] H. Bai, A. Wang, and M. Zhang, "Compressive sensing for DCT image," in *Proc. Int. Conf. Comput. Asp. Soc. Networks*, 2010, pp. 378-381.

[3] J. Donoho, D. Lustig, M. Santos, and J. Pauly, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1289-1306, 2006.

[4] E. J. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406-5425, 2006.

[5] Z. Wen, W. Yin, D. Goldfarb, and Y. Zhang, "A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation," *SIAM J. Sci. Comput.*, vol. 32, no. 4, pp. 1832-1857, 2010.

[6] K. Fountoulakis, J. Gondzio, and P. Zhlobich, "Matrix-free interior point method for compressed sensing problems," *Math. Program. Comput.*, vol. 6, no. 1, pp. 1-31, Mar. 2014.

[7] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129-159, 2001.

[8] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655-4666, 2007.

[9] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265-274, Nov. 2009.

[10] E. J. Candès and J. Romberg. (2005). l_1-MAGIC: Recovery of sparse signals via convex programming. *Tech. Rep*. [Online]. Available: http://www.acm.caltech.edu/l1magic

[11] H. Yao, F. Dai, D. Zhang, Y. Ma, and S. Zhang, "DR^2-Net: Deep residual reconstruction network for image compressive sensing," *arXiv Preprint*, arXiv:1702.05743, pp. 1-10, 2017.

[12] C. M. Sandino and J. Y. Cheng, "Deep convolutional neural networks for accelerated dynamic magnetic resonance imaging," Stanford Univ., CS231N, Course Proj., 2017.

[13] A. Mousavi and R. G. Baraniuk, "Learning to invert: Signal recovery via deep convolutional networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2017, pp. 2272-2276.

[14] L. Sun, Z. Fan, Y. Huang, X. Ding, and J. Paisley, "Compressed sensing MRI using a recursive dilated network," in *Proc. of AAAI Conf. Artificial Intelligence*., 2018, pp. 2444-2451.

[15] Y. M. Li and D. Wei, "Signal reconstruction of compressed sensing based on recurrent neural networks," *Optik (Stuttg).*, vol. 127, no. 10, pp. 4473-4477, 2016.

[16] L. Tian, G. Li, and C. Wang, "A data reconstruction algorithm based on neural network for compressed sensing," in *Proc. 5th Int. Conf. Adv. Cloud Big Data*, 2017, pp. 291-295.

[17] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure & Applied Mathematics*, vol. 59, no. 8, pp. 1207-1223, 2006.

[18] Z. Zhang and S. G. Mallat, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process*., vol. 41, no. 12, pp. 3397-3415, 1993.

[19] J. A. Tropp and A. C. Gilbert, "Via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655-4666, 2007.

[20] I. Daubechies, M. Defrise, and C. D. Mol, "An iterative thresholding algorithm for linear problems with a sparsity constraint," *Comm. Pure Appl. Math*., vol. 57, pp. 1413-1457, 2004.

[21] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal*., vol. 27, no. 3, pp. 265-274, 2009.

[22] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203-4215, 2005.

[23] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact recovery from highly incomplete fourier information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489-509, 2006.

[24] K. Fountoulakis, J. Gondzio, and P. Zhlobich, "Matrix-free interior point method for compressed sensing problems," *Mathematical Programming Computation*, vol. 6, no. 1, pp. 1-31, 2014.

[25] H. H. Pham, L. Khoudour, A. Crouzil, P. Zegers, and S. Velastin, "Learning to recognise 3D human action from a new skeleton-based representation using deep convolutional neural networks," *IET Computer Vision*, vol. 13, no. 3, pp. 319-328, 2019.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. the IEEE Intern. Conf. on Computer Vision*, 2015, pp. 1026-1034, 2015.

[27] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," *NIPS Proceedings*, pp. 901-909, 2016.

Youhao Yu was born in Fujian, China. He received M.S. degree in Electrical theory and new technology from Huaqiao University, Fujian, China, in 2005. He is currently pursuing the Ph.D degree with the Department of System and Computer Engineering, Carleton University, Ottawa, Canada. The major field of study are digital signal processing. From 2005 to 2007, he worked as an instructor in Minjiang University, Fujian, China. Since 2007 he worked as a lecturer in Putian University, Fujian, China. Now he works as teaching assistant in Carleton University and research assistant of Professor Dansereau. His research interests include compressive sensing and deep neural network.

Richard M. Dansereau is Associate Dean (Student Affairs) and Professor in the Faculty of Engineering and Design at Carleton University. Dr. Dansereau received the B.Sc. and Ph.D. degrees in Computer Engineering from the University of Manitoba, Winnipeg, MB, Canada, in 1995 and 2001, respectively. From 1996 to 2000, he was a Researcher with the Telecommunications Research Laboratories and from 1999 to 2000, he was also with SpectraWorks, Inc. From 2000 to 2001, he was an Instructor and Research Engineer with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, where he was also with the Center for Signal and Image Processing. Since 2001, Dr. Dansereau has been a professor with the Department of Systems and Computer Engineering at Carleton University and held positions of Associate Chair (Undergraduate Studies) and Associate Dean (Student Affairs).