



An Image Compression Method Based on Compressive Sensing and Convolution Neural Network for Massive Imaging Flow Cytometry Data

Long Cheng and Yi Gu^(✉)

School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, Jiangsu, China

8202101437@jiangnan.edu.cn

Abstract. Imaging flow cytometer provides high throughput cell imaging capability and is an essential tool for cell biology research. However, the high throughput also dramatically increases the cell image datasets in cell analysis. In this paper, we propose an image compression method combining compressive sensing and convolution neural network for massive imaging flow cytometer data. In the proposed workflow, data compression is done using compressive sensing technique and the image reconstruction is done using convolution network to improve speed. We demonstrate the proposed method on imaging flow cytometer cell dataset and evaluate the performance.

Keywords: Image compression · Image reconstruction · Compressive sensing · Convolution neural network

1 Introduction

High throughput analysis has been an essential tool in cell biology research [1–3]. Conventional intensity based flow cytometer has been widely used in high throughput cell analysis [4]. In intensity based flow cytometer, cells are suspended and flow through laser illumination region at high speed, then fluorescence intensity and scattering intensity of large amount of cells are measured by photodetectors and stored for statistical analysis [5]. Since the invention of the first imaging flow cytometer (IFC) in 2000s [6], imaging capability are integrated with flow cytometer, and researchers are able to acquire images for each single cell using IFC. Thanks to the development of IFC, large amount of cell images can be recorded in a short time period for phenotype study that benefits drug discovery [7], immunology research [8], cancer research [9], etc. However, the data size of high throughput cell analysis also increased dramatically due to the imaging capability of IFC [10]. In conventional cytometer, only light intensity is recorded, and there are only a few parameters for each cell. However, in IFC based cell analysis, images are recorded for each cell. Furthermore, due to the multi-modal and multi-fluorescence-color capability of IFC, usually 3–4 images are recorded for each single cell in real applications.

The transfer and storage of large cell image datasets remains to be the bottleneck of IFC cell analysis, and efficient data compression method needs to be developed.

Compressive sensing [11] is a technique for data compression. In compressive sensing process, sensing matrix are designed to project high dimensional data to low dimensional space. To realize successful data reconstruction, the original data needs to meet sparsity requirement. IFC datasets consist of large number of cell images, and each image is a single cell image. Due to the single cell image characteristics, either the cell image itself is sparse or the total variation of cell image is sparse, which makes compressive sensing suitable for IFC image compression [12]. The data compression can be realized using simple matrix multiplication, which makes the compression process time efficient. In addition, the compressive sensing algorithm is compatible with other existing image compression techniques, meaning that the cell images can be first compressed by other image compression method, then further compressed using compressive sensing method.

However, the image reconstruction process of compressive sensing is time consuming. Typically, reconstructing one image takes several seconds to several minutes [13]. There are usually hundreds of thousands of single cell images in a IFC dataset, which makes the time consuming reconstruction process unacceptable in real IFC applications. Another disadvantage of compressive sensing reconstruction is that predefined constraints such as image sparsity is required for successful image reconstruction. And designing the predefined constraints requires manual experience of cell property.

Convolution neural network [14] is a powerful technology for image reconstruction. Since convolution neural network is a forward generative model, the image reconstruction speed is much improved. Here, we combine compressive sensing and convolution neural network to propose a novel workflow for IFC cell image compression. The proposed workflow consists of two major steps. The first step is image compression. In the image compression step, random generated sensing matrix are generated for compressive sensing. The second step is image reconstruction. In the reconstruction step, we design a neural network to bypass the compressive sensing based image reconstruction. Using the neural network generative model, not only the image reconstruction speed is significantly improved, but also the requirement for predefined constraints is released. To demonstrate the proposed method, we compressed the Jurkat cell image using compressive sensing method. And for the image reconstruction step, we compared the proposed neural network based method to the compressive sensing method. From the experimental results, we can conclude that the proposed method is significantly faster, and reconstructed image quality is improved.

This paper will be arranged as follows. First, related works are introduced. Second, the proposed method is described in detail. Third, experiments and experimental results are discussed.

2 Related Works

2.1 Compressive Sensing

Compressed sensing, also known as compressed sampling or sparse sampling, is a technique for data compression based on the assumption that the original data is sparse in certain space [15]. As shown in Fig. 1, the concept of compressive sensing can be

expressed as Eq. (1). Where s is the original data(signal/image) to be compressed, Ψ is the sparse basis matrix, and Φ is the sensing matrix. Assuming s is a one dimensional $n \times 1$ matrix, then Ψ is a $n \times n$ matrix. Ψ projects s to a sparse basis using $x = \Psi s$. Then the sensing matrix Φ is introduced for data compression. The size of Φ is $k \times n$, where $n \gg k$. Φ projects a higher dimensional matrix x onto a lower dimensional space, and $y = \Phi x$ is the compressed data of length k . Let $A = \Phi\Psi$, we can simplify Eq. (1) as $y = As$.

$$y = \Phi\Psi s \tag{1}$$

In the reconstruction process, y and A are known, the purpose of reconstruction is to solve s . However, as discussed above, $n \gg k$, the problem is an ill-posed problem. Based on the sparsity assumption, a regularized term $\gamma(s)$ is introduced. So, s is reconstructed by solving Eq. (2), where $\gamma(s)$ is the regularization term designed based on the sparsity assumption, and λ is the coefficient of the regularization term.

$$\hat{s} = \operatorname{argmin} \|y - As\|_2^2 + \gamma(s) \tag{2}$$

Compressive sensing solves the reconstruction problem by iteratively optimizing Eq. (2). There are several compressive sensing algorithms. L1-Magic [16] and TwIST [17] are representative algorithms, and we compare the performance of both algorithms to our proposed method. L1-Magic provides high quality reconstruction but is time consuming. TwIST provides relatively high reconstruction speed but sacrifice image quality. As shown by our results, the proposed method outperforms L1-Magic and TwIST in both speed and reconstruction quality.

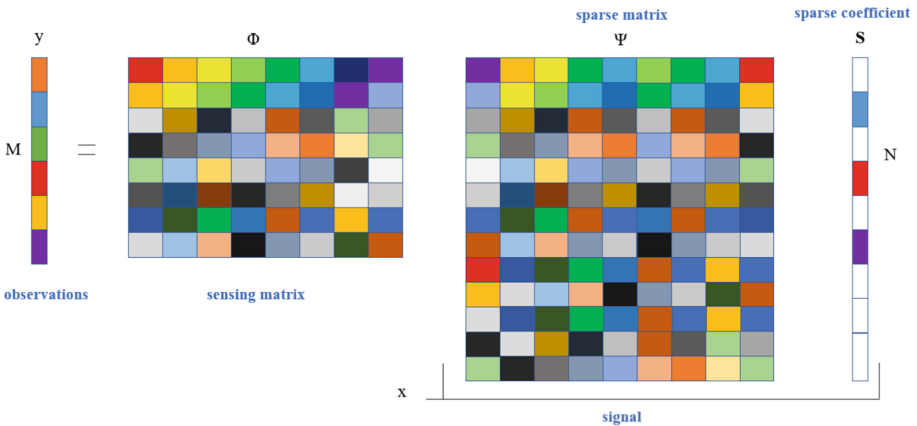


Fig. 1. Principle of compressed sensing

2.2 Convolutional Neural Network

Convolution neural network [18] is a feedforward neural network, whose artificial neurons can respond to part of the surrounding elements within the coverage area and

perform well in large-scale image processing. Through the cooperation of convolution, pooling, activation and other operations, the convolution neural network can better learn the features of spatial association. Convolution neural network can be used as a forward generative model for image generation. And in this paper, we designed a network for IFC compressive sensing image reconstruction to significantly improve the image reconstruction speed.

3 Image Compression Based on Compressive Sensing and CNN

The proposed method consists of two major steps. The first step is data compression step, and the second is image reconstruction steps. We will describe both steps in details in this session.

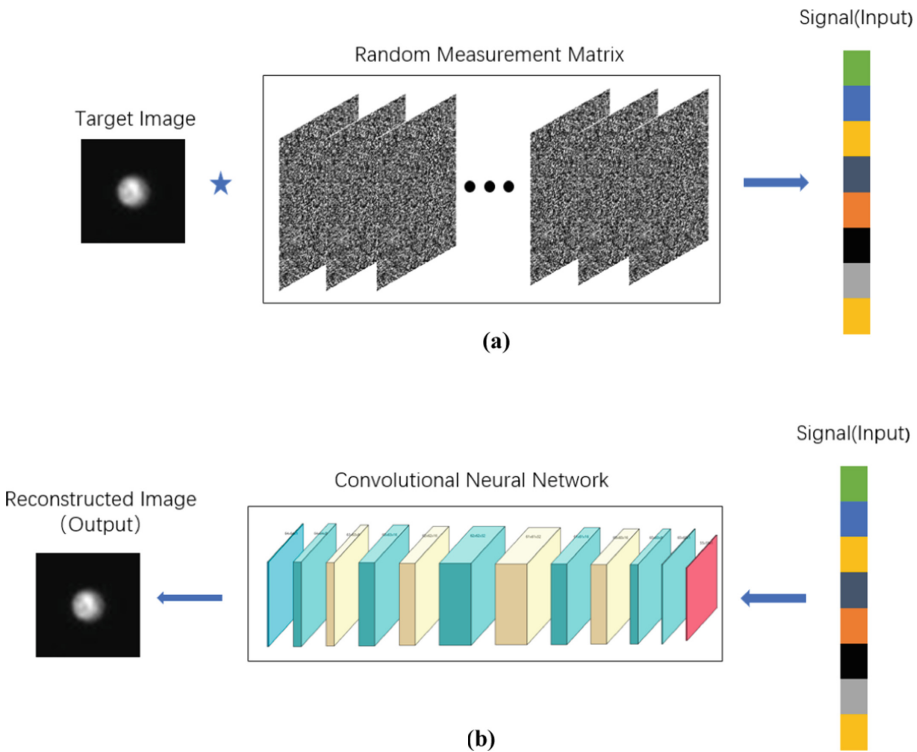


Fig. 2. (a) Data compression process. (b) Image reconstruction process

3.1 Data Compression

The data compression step is shown in Fig. 2(a), where cell image is projected to a low dimensional space using compressive sensing method. Assume the image size is

$n \times n$. And k sensing matrix are generated for data compression, and in this paper, we use Gaussian random matrix as the sensing matrix since it satisfies the incoherence requirement. For each sensing matrix, we can calculate the correlation between image and the sensing matrix as shown by Eq. (3),

$$S(i) = \sum_{x=1}^N \sum_{y=1}^N I(x, y) M^i(x, y) \quad (3)$$

where I is the image, M^i is the i^{th} sensing matrix. And S is a k dimensional vector. So, when $n \times n \gg k$, S is significantly compressed. In this paper, $n = 55$ and 2 different values of k are tested.

In this paper, we propose convolution neural network for image reconstruction. To compare performance, we also employed the conventional compressive sensing algorithm for image reconstruction. For compressive sensing, we use Total variation (TV) as the regularization term in Eq. (2). And the TV regularization term can be described as Eq. (4),

$$\psi(I) = \sum_x \sum_y \sqrt{D_x(I)^2 + D_y(I)^2} \quad (4)$$

where $\psi(I)$ is the TV regularization term, $D_x(I)$ and $D_y(I)$ are gradients of I in x and y axis.

3.2 Image Reconstruction

The image reconstruction step is shown in Fig. 2(b). The purpose of image reconstruction step is to generate cell images using the compressed low dimensional data. We deigned a CNN structure convolution neural network as the generator. The designed network is a forward generative model which takes the compressed data as input and outputs the reconstructed cell image, thus significantly improves the reconstruction speed. Also, shown by our experimental results, the convolutional layers efficiently extract the data features and improves the reconstructed image quality. The designed convolution neural network structure is shown in Fig. 3, which consists of six convolutional layers, four maximum pooling layers and one full connection layer. The input dimension is determined by the compression ratio. In this paper, we test 2 different compression ratio 20% and 5%, corresponding to the input dimension as 606 and 151. The input signal is stretched to 4096×1 through a fully connected layer and then reshaped to 64×64 . The features are extracted by the sub-sampling network, which is composed of six sub-conversion layers, which are composed of activation layer and convolution layer. The activation function uses ReLU function. The convolution layer is composed of 3×3 convolution blocks, whose step size is 1. We set four maximum pooling layers at the back of the first four convolution layers with a size of 2×2 and step size of 1. The pooling layer selects features and filters information from the output feature maps. Finally a 55×55 output image is obtained through the last convolutional layer. The generated model is trained in the way of supervised learning. The loss function is the absolute value of the difference between the generated image and the original image, and can be described as Eq. (5), where I is the groundtruth image and \hat{I} is the predicted

image. Once the generative model is trained, it takes the compressed data as input and directly outputs the reconstructed image.

$$Loss = \iint |I - \hat{I}| dx dy \quad (5)$$

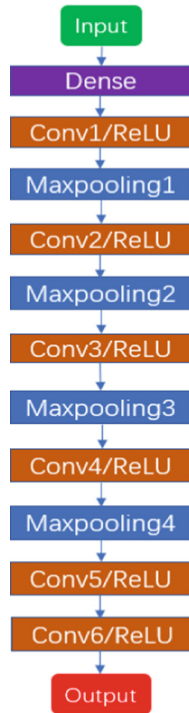


Fig. 3. CNN structure diagram

4 Experimental Results

The flow chart of experiment is shown in Fig. 4. First, sensing matrix are generated and cell images are compressed. Following is the reconstruction part. In the reconstruction part, the compressed data is used as convolution neural network input to generate reconstructed image. First, the reconstruction model is trained using training set. Then we optimize the model until the best performance is achieved. The trained and optimized model is used for image reconstruction. Finally, the performance is evaluated. The details of the experiment and the final results will be explained in a later section.

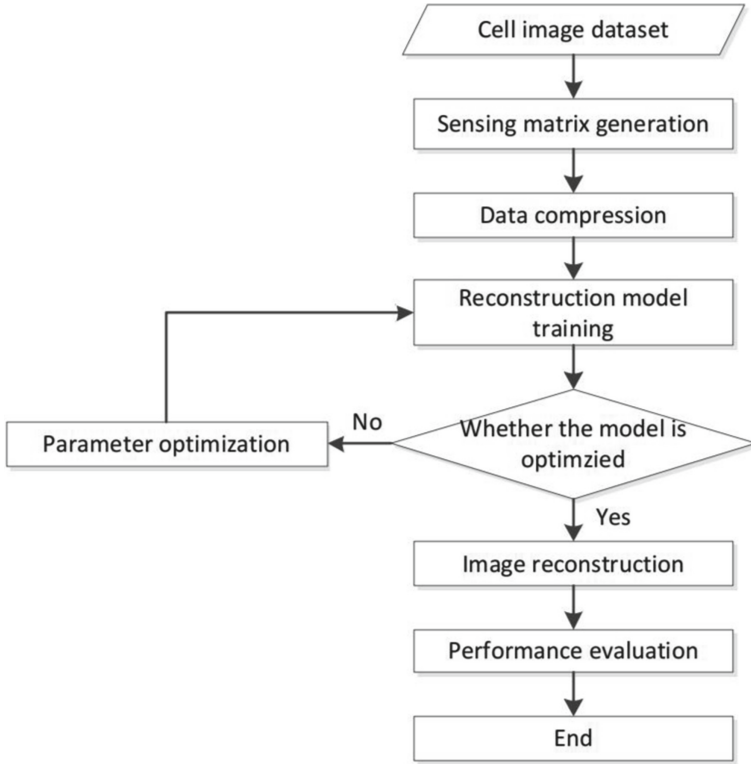


Fig. 4. The experimental process

4.1 Dataset

In our experiment, we used the Jurkat cell image dataset in the paper published by T Blasi et al. [19]. The dataset is generated using an imaging flow cytometer platform named ImageStreamX. Before image acquisition, Jurkat cells are stained with PI antibody and MPM2 antibody. Then the stained cells are flow through the imaging flow cytometer. Bright field, dark field and both fluorescence channels are imaged. For demonstration purpose, we choose the PI stained fluorescence imaging channel. These cell images are 55 by 55 pixels in size, with a pixel size of 0.33 μm . Among them, 80% of the images in the data set are used as our training set and 20% of the samples are used as our test set.

4.2 Experimental Setup

Our compressed sensing image reconstruction based on supervised learning is completed using TensorFlow [20] on an NVIDIA GeForce RTX 3060 Ti GPU. We used Adam optimizer [21] to train our model, set the epoch to 150, and the batch size to 20, and the learning rate to be fixed at 5×10^{-5} . As mentioned in the previous content, we will compare with TwIST algorithm and L1-Magic algorithm in the same data set and experimental environment to test the effect of our image reconstruction.

In order to test the reconstruction effect of our experiment, we tested and evaluated the supervised learning method, TwIST algorithm, L1-Magic algorithm, and the performance are compared. For TwIST and L1-Magic, we followed the same experiment environment as [16, 17].

4.3 Performance Evaluation Index

In order to evaluate the quality of image reconstruction, PSNR [22] and SSIM [23] are introduced. Given a clean image I and a noisy image K of size $m \times n$, mean square error is defined as follows:

$$MSE = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} [I(x, y) - \hat{I}(x, y)]^2 \tag{6}$$

PSNR (Peak Signal-to-Noise Ratio) is defined as follows:

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \tag{7}$$

where MAX_I^2 is the maximum possible pixel value of the picture. If each pixel is represented by an 8-bit binary, 255.

SSIM (Structure Similarity) formula is based on three comparative measures between samples x and y , brightness, contrast and structure.

$$l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \tag{8}$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \tag{9}$$

$$s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3} \tag{10}$$

where x is the original image, y is the reconstructed image; μ_x, μ_y is the mean value of x, y ; σ_x^2, σ_y^2 is the variance of x, y ; σ_{xy} is the covariance of x, y ; $c_1 = (k_1L)^2, c_2 = (k_2L)^2$, in general we take $c_3 = c_2/2$; L is the range of pixel values, we set $k_1 = 0.01, k_2 = 0.02$ as the default value.

SSIM formula is defined as follows:

$$SSIM(x, y) = l(x, y)^\alpha c(x, y)^\beta s(x, y)^\gamma \tag{11}$$

Set $\alpha = \beta = \gamma = 1$, and new SSIM formula is shown in Eq. (12):

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{12}$$

4.4 Image Reconstruction Result

In this section, we discuss the performance of image reconstruction. To quantitatively evaluate the reconstruction performance, three evaluation indexes are used, including PSNR, SSIM and image reconstruction speed. To compare performance, we also employ the TwIST compressive sensing algorithm and L1-Magic algorithm. First, we set the compression ratio to 20%. The example reconstructed images are shown in Fig. 5, the first column are the original images, the second column are the images reconstructed using our proposed method, the third column are the images reconstructed using L1-Magic method, the fourth column are the image reconstructed using the TwIST algorithm, and the fifth column is the compressed low dimensional data corresponding to the original images. As can be seen from the Fig. 5, the reconstruction performance of the proposed method and L1-Magic method is relatively good, while the images reconstructed using the TwIST algorithm have obviously worse quality. Then we compared the SSIM, PSNR and image reconstruction speed of the three methods, and the results are shown in Table 1, Our proposed method and L1-Magic method have better reconstruction quality, SSIM is 0.9932 and 0.9812, PSNR is 45.18 dB and 40.10 dB, respectively. As we can see, the reconstruction quality of TwIST algorithm is not as good as the two methods. SSIM and PSNR were only 0.8592 and 31.57 dB. From Table 1, we can conclude that our proposed method is much faster in reconstruction speed.

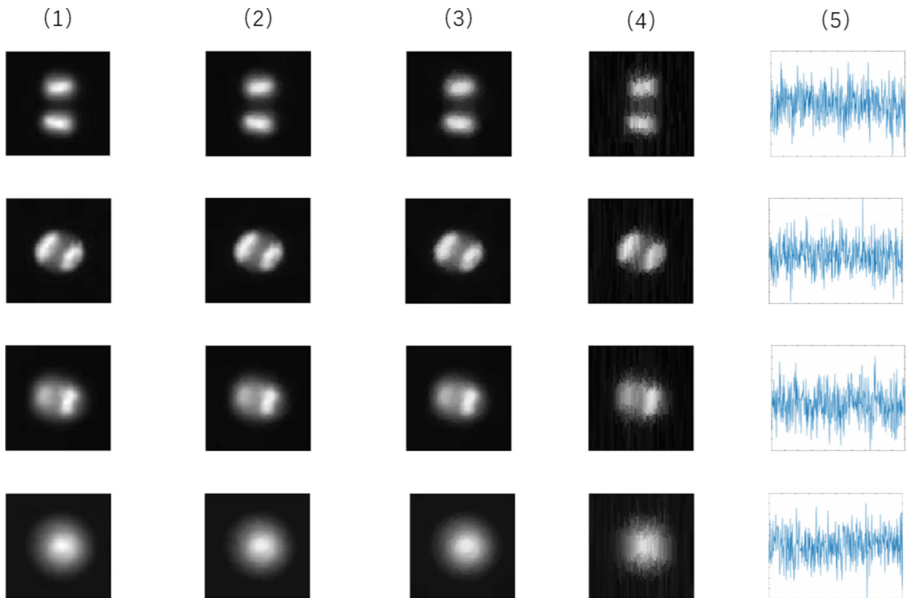
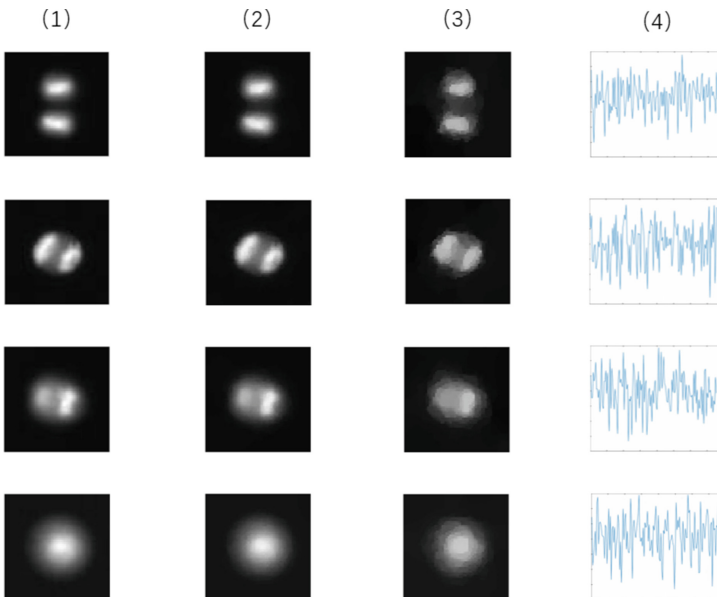


Fig. 5. Example reconstructed images (compression ratio = 20%)

Table 1. Performance evaluation of image reconstruction (compression ratio = 20%)

	Our method	L1-Magic	TwIST
PSNR	45.18 dB	40.10 dB	31.57 dB
SSIM	0.9932	0.9812	0.8592
Reconstruction speed	0.12 s	7.84 s	2.11 s

Then we reduced the compression ratio to 5%. When the compression ratio is 5%, the image reconstruction of TwIST algorithm is unsuccessful, so we will not show the reconstruction images. As can be seen from the Fig. 6, where the first column are the original images, the second column are the images reconstructed using our proposed method, the third column are the images reconstructed using L1-Magic method, and the fourth column are the compressed low dimensional data corresponding to the original images. The reconstruction quality of our proposed method is relatively good, while the images reconstructed using the L1-Magic algorithm are obviously worse. Then we compared the SSIM, PSNR and image reconstruction time of the two methods. And the results are shown in Table 2, Our proposed method has better reconstruction performance, SSIM is 0.9918, PSNR is 43.05 dB. In contrast, L1-Magic algorithm does not have such good performance. Its SSIM and PSNR are 0.9135 and 29.61 dB respectively. And the proposed method provides much faster image reconstruction. In conclusion, compared with the other methods, our proposed method not only ensures the image reconstruction

**Fig. 6.** Example reconstructed images (compression ratio = 5%)

quality, but also significantly improves the reconstruction speed, showing the superiority of our proposed method.

Table 2. Performance evaluation of image reconstruction (compression ratio = 5%)

	Our method	L1-Magic
PSNR	43.05 dB	29.61 dB
SSIM	0.9918	0.9135
Reconstruction speed	0.11 s	2.24 s

5 Conclusion

In this paper, we proposed a data compression method for imaging flow cytometer data combining compressive sensing and convolution neural network. In the proposed method, random generated sensing matrix are used for image compression and convolution neural network is used for image reconstruction. We use the Jurkat cell dataset to demonstrate our proposed method. The experimental results show that the proposed method is time efficient and provides good image reconstruction quality. We anticipate that the proposed method would improve the data compression process for massive imaging flow cytometer dataset and have broad application in cell biology research.

References

1. Reece, A., Xia, B., Jiang, Z., Noren, B., McBride, R., Oakey, J.: Microfluidic techniques for high throughput single cell analysis. *Curr. Opin. Biotechnol.* **40**, 90–96 (2016)
2. Gu, Q., et al.: LOC-based high-throughput cell morphology analysis system. *IEEE Trans. Autom. Sci. Eng.* **12**(4), 1346–1356 (2015)
3. Kellogg, R.A., Gómez-Sjöberg, R., Leyrat, A.A., Tay, S.: High-throughput microfluidic single-cell analysis pipeline for studies of signaling dynamics. *Nat. Protoc.* **9**(7), 1713–1726 (2014)
4. McKinnon, K.M.: Flow cytometry: An overview. *Curr. Protoc. Immunol.* **120**(1), 511–5111 (2018)
5. Adan, A., Alizada, G., Kiraz, Y., Baran, Y., Nalbant, A.: Flow cytometry: Basic principles and applications. *Crit. Rev. Biotechnol.* **37**(2), 163–176 (2017)
6. Basiji, D.A., Ortyñ, W.E., Liang, L., Venkatachalam, V., Morrissey, P.: Cellular image analysis and imaging by flow cytometry. *Clin. Lab. Med.* **27**(3), 653–670 (2007)
7. Elliott, G.S.: Moving pictures: Imaging flow cytometry for drug development. *Comb. Chem. High Throughput Screening* **12**(9), 849–859 (2009)
8. Mastoridis, S., Bertolino, G.M., Whitehouse, G., Dazzi, F., Sanchez-Fueyo, A., Martinez-LLordella, M.: Multiparametric analysis of circulating exosomes and other small extracellular vesicles by advanced imaging flow cytometry. *Front. Immunol.* **9**, 1583 (2018)
9. Ricklefs, F.L., et al.: Imaging flow cytometry facilitates multiparametric characterization of extracellular vesicles in malignant brain tumours. *J. Extracellular Vesicles* **8**(1), 1588555 (2019)

10. Han, Y., Gu, Y., Zhang, A.C., Lo, Y.H.: Imaging technologies for flow cytometry. *Lab Chip* **16**(24), 4639–4647 (2016)
11. Donoho, D.L.: Compressed sensing. *IEEE Trans. Inf. Theory* **52**(4), 1289–1306 (2006)
12. Ota, S., et al.: Ghost cytometry. *Science* **360**(6394), 1246–1251 (2018)
13. Di Carlo, D., et al.: Comment on “Ghost cytometry.” *Science* **364**(6437), eaav1429 (2019)
14. Traore, B.B., Kamsu-Foguem, B., Tangara, F.: Deep convolution neural network for image recognition. *Eco. Inform.* **48**, 257–268 (2018)
15. Qaisar, S., Bilal, R.M., Iqbal, W., Naureen, M., Lee, S.: Compressive sensing: From theory to applications, a survey. *J. Commun. Netw.* **15**(5), 443–456 (2013)
16. Candes, E., Romberg, J.: *l1-magic: Recovery of sparse signals via convex programming* (2005). [Www Magic.org](http://www.magic.org)
17. Bioucas-Dias, J.M., Figueiredo, M.A.: A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Process.* **16**(12), 2992–3004 (2007)
18. Lawrence, S., Giles, C.L., Tsoi, A.C., Back, A.D.: Face recognition: A convolutional neural network approach. *IEEE Trans. Neural Networks* **8**(1), 98–113 (1997)
19. Blasi, T., et al.: Label-free cell cycle analysis for high-throughput imaging flow cytometry. *Nat. Commun.* **7**(1), 1–9 (2016)
20. Abadi, M., et al.: {TensorFlow}: A system for {Large-Scale} machine learning. In: *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2016)*, pp. 265–283 (2016)
21. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2014). [arXiv preprint arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
22. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of PSNR in image/video quality assessment. *Electron. Lett.* **44**(13), 800–801 (2008)
23. Hore, A., Ziou, D.: Image quality metrics: PSNR vs. SSIM. In: *Proceedings of the 2010 20th International Conference on Pattern Recognition*, pp. 2366–2369. IEEE (2010)