www.nature.com/scientificreports

SCIENTIFIC REPORTS
natureresearch

OPEN

A residual-based deep learning approach for ghost imaging

Tong Bian[1,2], Yuxuan Yi[2], Jiale Hu[2], Yin Zhang[3], Yide Wang[2] & Lu Gao[1]✉

Ghost imaging using deep learning (GIDL) is a kind of computational quantum imaging method devised to improve the imaging efficiency. However, among most proposals of GIDL so far, the same set of random patterns were used in both the training and test set, leading to a decrease of the generalization ability of networks. Thus, the GIDL technique can only reconstruct the profile of the image of the object, losing most of the details. Here we optimize the simulation algorithm of ghost imaging (GI) by introducing the concept of "batch" into the pre-processing stage. It can significantly reduce the data acquisition time and create reliable simulation data. The generalization ability of GIDL has been appreciably enhanced. Furthermore, we develop a residual-based framework for the GI system, namely the double residual U-Net (DRU-Net). The imaging quality of GI has been tripled in the evaluation of the structural similarity index by our proposed DRU-Net.

Ghost imaging (GI) is an unconventional imaging method compared with traditional optical imaging methods. It was first proposed as a quantum entangled phenomenon by making use of the entangled two-photon light source generated by the spontaneous parameter down-conversion (SPDC) process[1]. However, GI was demonstrated to be accomplished by classical incoherent light source soon later[2]. It led to controversy focus on whether the quantum entangled is necessary for the GI system. The GI system contains two distributed light beams, which are the test beam and the reference beam. In the test beam, the light illuminates the object directly and then be collected into a bucket measurement. In the reference beam, the light travels freely to a high-resolution detector without interacting with the object. The image can then be reconstructed through the correlation measurement between the two light beam signals[3]. In terms of computational ghost imaging (CGI), the reference beam becomes unnecessary as one can obtain the image by calculating the correlation between the test beam intensity and the knowledge of the random patterns displayed on the spatial light modulator (SLM)[4–6].

CGI usually needs a large number of random patterns to avoid noise disturbance to achieve images of high resolution. This requirement leads to a long data acquisition time, which is the main issue preventing CGI from practical use. Improved correlation method such as normalized ghost imaging[7] and differential ghost imaging[8] was later proposed to increase imaging efficiency. Notably, the Gerchberg–Saxton algorithm and compressive sensing ghost imaging (CSGI) regard GI as an optimization problem. In[9], the Gerchberg-Saxton-like technique takes the integral property of Fourier transform into full consideration to provide a different perspective of image reconstruction of GI. CSGI can reduce measurement times and boost imaging quality[10–14].

Recently, deep learning (DL) has achieved widespread use in various fields, such as image denoising, image inpainting, natural language processing, to name a few[15]. DL has also made a comprehensive application in computational imaging[16] such as digital holography[17], lensless imaging[18], imaging through scattering media and turbid media[19–21], optical encryption[22,23] and ghost imaging[24–28]. The term of ghost imaging using deep learning (GIDL) was first proposed by Lyu et al.[24]. Their research shows that GIDL can accomplish the reconstruction of binary images of high quality, even with significantly less sampling rate $\beta = N/M$, where N represents the number of measurements, and M stands for the number of pixels of the image. GIDL generally pairs up thousands of GI images as inputs and their corresponding images of the object as targets to train the deep neural network (DNN). Here we use the term "GI image" to represent the image reconstructed by traditional ghost imaging. GIDL can reconstruct objects from severely disrupted images in which the object can hardly be recognized. In the evaluation of the peak signal to noise ratio (PSNR), structural similarity (SSIM) index, and root mean squared error (RMSE), GIDL shows excellent results[25,26,28]. Also, GIDL needs far less processing time than non-deep-learning-based methods[25]. Previous research demonstrates that deep-learning-based solvers for general computational imaging can be trained by simulation data to make it more practical[28].

[1]School of Science, China University of Geosciences, Beijing 100083, China. [2]School of Information Engineering, China University of Geosciences, Beijing 100083, China. [3]School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, China. ✉email: gaolu@cugb.edu.cn

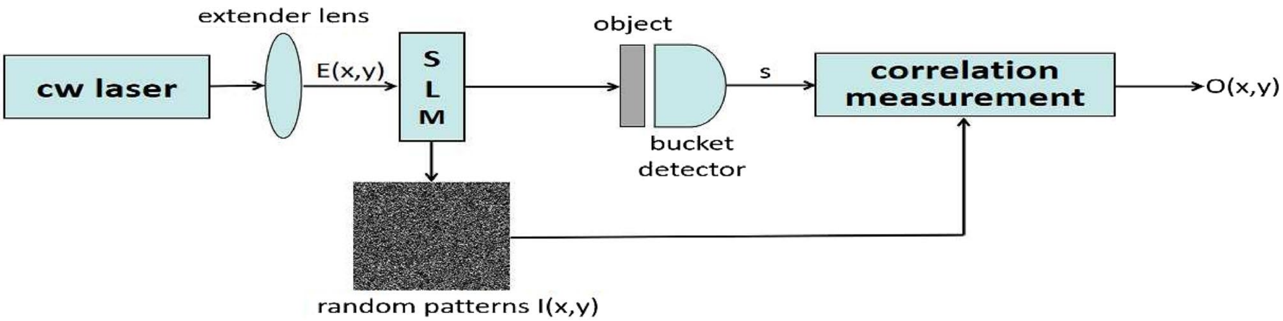

Figure 1. Schematic illustration of the CGI setup.

GIDL usually takes a large number of images of the object as ground-truth, and then they are multiplied with a series of random patterns to create corresponding GI images as the network inputs. However, among most GIDLs proposed so far, the same set of specific random patterns is used unchangingly in both the training and test set. This drawback may lead to unsatisfactory performance in practical applications with weakened generalization ability[29]. Here we optimize the simulation algorithm of GI with batch processing of different random patterns to create datasets similar to real physical condition. We apply this optimized simulation algorithm in DNNs for the GI scenario, and the generalization ability of existing GIDL is enhanced obviously. In order to improve the imaging quality of GIDL, we make DNN learn the residual image but not the image of the object. We divide the residual of GI into two and further develop a double residual (DR) framework by learning two different kinds of residual images. Through combining the DR framework with the U-Net, we propose a new GIDL, namely the double residual U-Net (DRU-Net). The image quality has been significantly improved with the triple SSIM index.

Simulation

Optimized simulation algorithm with batch processing.

The configuration for CGI is shown in Fig. 1. Here we simulate it on the computer. A series of random patterns is created and multiplied with the image of the object. It is then summed up as a single pixel to accomplish the bucket measurement. Images can be reconstructed by the correlation between the bucket measurement point intensity in the test beam and the random patterns in the reference beam. The bucket measurement function is given by

$$T_{GI}(x,y) = \left\langle S_m I_m(x,y) \right\rangle - \langle S_m \rangle \left\langle I_m(x,y) \right\rangle, \tag{1}$$

where $\langle \cdot \rangle$ denotes the average of the multiple measurements.

The pseudothermal light random patterns are generated through the simulation to train the network[28]. In the plain serial simulation algorithm, images are processed one by one to achieve the corresponding images of GI. And creating a different set of random patterns for each image will lead to huge input/output (I/O) overhead. Since DL usually needs datasets of thousands of images, it will take tens of hours to generate them with the plain serial simulation algorithm. In previous research, a general method is to make use of the same set of random patterns to generate datasets?. It can reduce both the random pattern generating time and I/O overhead at the cost of the reliability of data and network's generalization ability. Here we optimize the simulation with batch processing to create more reliable data with different sets of random patterns. We introduce the concept of batch in stochastic gradient descent (SGD)[30] into the pre-processing stage to make full use of the Compute Unified Device Architecture (CUDA). Images are divided into batches and then combined in the same batch into a 3-dimensional (3D) array. Each 3D array will be taken as the smallest operation unit and a different set of random patterns are created for the specific 3D array. Thus, images in the same batch share the same set of random patterns, whereas different sets of random patterns are used among batches. The GI images are achieved from the dot product of the 3D object distributions and their corresponding random patterns. By setting batch size as 256, 512 or even larger number, our optimized simulation algorithm significantly reduces the I/O overhead. We then normalized the GI simulation data according to the formula as

$$T_{GI}\left(x,y\right) = \frac{T_{GI}\left(x,y\right) - T_{GI\,\min}\left(x,y\right)}{T_{GI\,\max}\left(x,y\right) - T_{GI\,\min}\left(x,y\right)} \tag{2}$$

In this way, the generalization ability of GIDL has been improved. Our optimized algorithm can deal with grayscale image as well as binary images. Both the plain serial simulation algorithm and optimized simulation algorithm are implemented by PyTorch, which is a useful python machine learning library. It has been optimized for matrix, hence most of the matrix calculation time can be saved. For the training set of 128×128 pixels images, it takes almost 28 h to process them with the plain serial simulation algorithm but only 23 minutes with our optimized simulation algorithm by the laptop. The calculation efficiency is increased by 70 times.

Proposed double residual learning method.

In existing GIDLs, images of the object are reconstructed end-to-end from GI images. This process can be expressed as

$$O(x,y) = F\{T_{GI}(x,y)\}, \tag{3}$$

where O stands for the output of the network, and $F\{\cdot\}$ represents the network that maps the GI image to the corresponding image of the object. This specific training process is given by

$$F = \underset{\theta \in \Theta}{\arg\min} \sum_{i=1}^{K} L(T^i(x,y), F_\theta\{T^i_{GI}(x,y)\}) + W(\theta), \tag{4}$$

where Θ is the set of network parameters, $L(\cdot)$ is the loss function used to measure the distance between the network outputs and their corresponding images of the object, and the superscript i denotes the ith in-out pair. Here we have $i = 1 \ldots K$, enumerating the total K in-out pairs. The last term, $W(\theta)$, means the regularization of parameters in case of overfitting[29,31].

In recent years, we have witnessed the widespread applications of residual learning in many image processing fields[32,33]. Inspired by the DnCNN[34,35], we make DNN learn the residual image but not the image of the object. When applied in GIDL, the residual, denoted as *Res*, can be written as

$$Res(x,y) = T(x,y) - T_{GI}(x,y), \tag{5}$$

DnCNN is a deep residual network able to handle several general image denoising tasks, including Gaussian denoising, SISR and JPEG image deblockin[34,35]. It can yield better results than other state-of-the-art methods. Using R to denote the DnCNN, the training process can be formulated as

$$R = \underset{\theta \in \Theta}{\arg\min} \sum_{i=1}^{K} L(Res^i(x,y), R_\theta\{T^i_{GI}(x,y)\}). \tag{6}$$

In this way, we can express the reconstructed image as

$$O(x,y) = T_{GI}(x,y) + R\{T_{GI}(x,y)\}. \tag{7}$$

The residual mapping is usually easier to optimize than the original mapping as the residual image has small enough pixel values compared to those of the object's image[33,34]. To the extreme, if the identity mapping is optimal, it would be easier for the network to push the residuals to zero than fit the identity mapping. However, GI is an exception as its residual is of too high value making it hard to learn. In GI images, such as shown in Fig. 2b, the correlation measurement leads to the blurred image with the averaged pixel value. It turns out that the residual between Fig. 2a,b is in the range of − 209 to 145. The interval length exceeds the upper limit of the grayscale. It means that the residual is much more complex than the image of the object. Thus residual learning does not appear adaptive to GI. Here we produce two different kinds of residuals, namely the up-residual (up-Res) and down-residual (down-Res), to reduce the range of the pixel value to make it consistent with the classical nature of residual learning. The learning process becomes easier by dividing the residual into two parts. Figure 2c,d show the up-Res and down-Res images, respectively.

The main body of the DR framework consists of two CNNs. We refer to our proposed network as double residual U-Net (DRU-Net). The U-Net is a deep CNN with a U-shaped structure, where the max-pooling layers and the up-sampling layers are symmetrical to each other[36]. It was initially designed for image segmentation, but it can also be used in image denoising problem and ghost imaging[26]. Its variant ResUNet-a has also achieved significant improvement in remotely sensed data[37]. As schematically outlined in Fig. 3, one CNN is trained with up-Res images, and the other is trained with down-Res images. The DRU-Net reflects the statistical property of GI and makes the residual of GI easy enough to learn. After training, our network obtains much better results than both the U-Net and DnCNN. The DRU-Net not only reconstructs high-quality images with an amount of details restored but also shows strong generalization ability on the test set.

The DRU-Net is made up of the up-CNN and down-CNN, which intends to learn the up-Res images and down-Res images, respectively. Here we define the up-Res and down-Res as follows

$$Res_{up} = \begin{cases} T(x,y) - T_{GI}(x,y), & T(x,y) > T_{GI}(x,y), \\ 0, & T(x,y) \le T_{GI}(x,y). \end{cases} \tag{8}$$

$$Res_{down} = \begin{cases} T_{GI}(x,y) - T(x,y), & T_{GI}(x,y) > T(x,y), \\ 0, & T_{GI}(x,y) \le T(x,y). \end{cases} \tag{9}$$

We replace the negative pixel values by zeros to make it more adaptive to convolution operation. We feed those highly corrupted GI images to networks as inputs and use the two residuals as targets separately. In this case, the training process can be written as

$$R_{up} = \underset{\theta \in \Theta_{up}}{\arg\min} \sum_{i=1}^{K} L(Res^i_{up}(x,y), R_{up\theta}\{T^i_{GI}(x,y)\}), \tag{10}$$

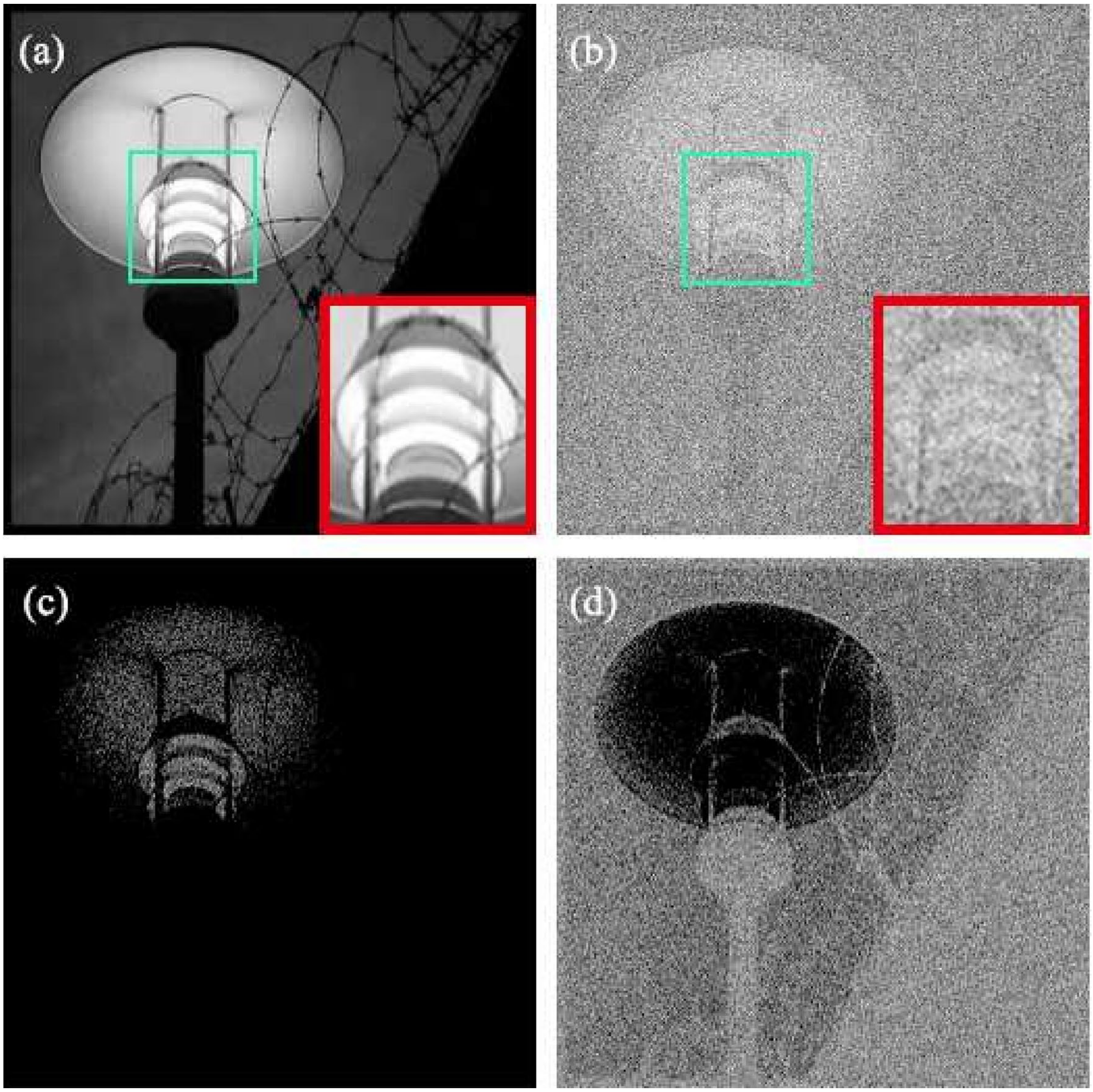

Figure 2. A comparison between image of the object and GI simulation. The image of the object is shown in (**a**), while the corresponding GI image is shown in (**b**), the up-Res image is shown in (**c**) and the down-Res image is shown in (**d**).

$$R_{down} = \underset{\theta \in \Theta_{down}}{\arg\min} \sum_{i=1}^{K} L(Res^{i}_{down}(x, y), R_{down_\theta}\{T^{i}_{GI}(x, y)\}), \tag{11}$$

where R_{up} denotes the up-CNN, and R_{down} denotes the down-CNN. We also convert the floating-point number into an integer so that we can save the residuals as images without explicit loss of information. Now the reconstructed image can be expressed here as

$$O(x, y) = T_{GI}(x, y) + R_{up}\{T_{GI}(x, y)\} - R_{down}\{T_{GI}(x, y)\}. \tag{12}$$

The MSRA10K dataset[38] was used as the image of the object to train all three GIDLs in our study. We use the models of the U-Net and DnCNN in previous research[26,34,35]. We selected 5120 images and resized them into 128 × 128 pixels. After that, we divided them into batches and obtained corresponding GI images with the optimized simulation algorithm. According to Eqs. (8) and (9), we calculated the up-Res and down-Res image paired up with corresponding GI images to train the DRU-Net. During the training period, Adam optimizer[39] was adopted to minimize the mean square error (MSE). The batch size of the training set was 16. The learning rate was set to 0.00002, and we had the weight decay equal to 0.0003. We used PyTorch 1.3.1 and a laptop with an NVIDIA GTX 1080M graphic card to train our network.

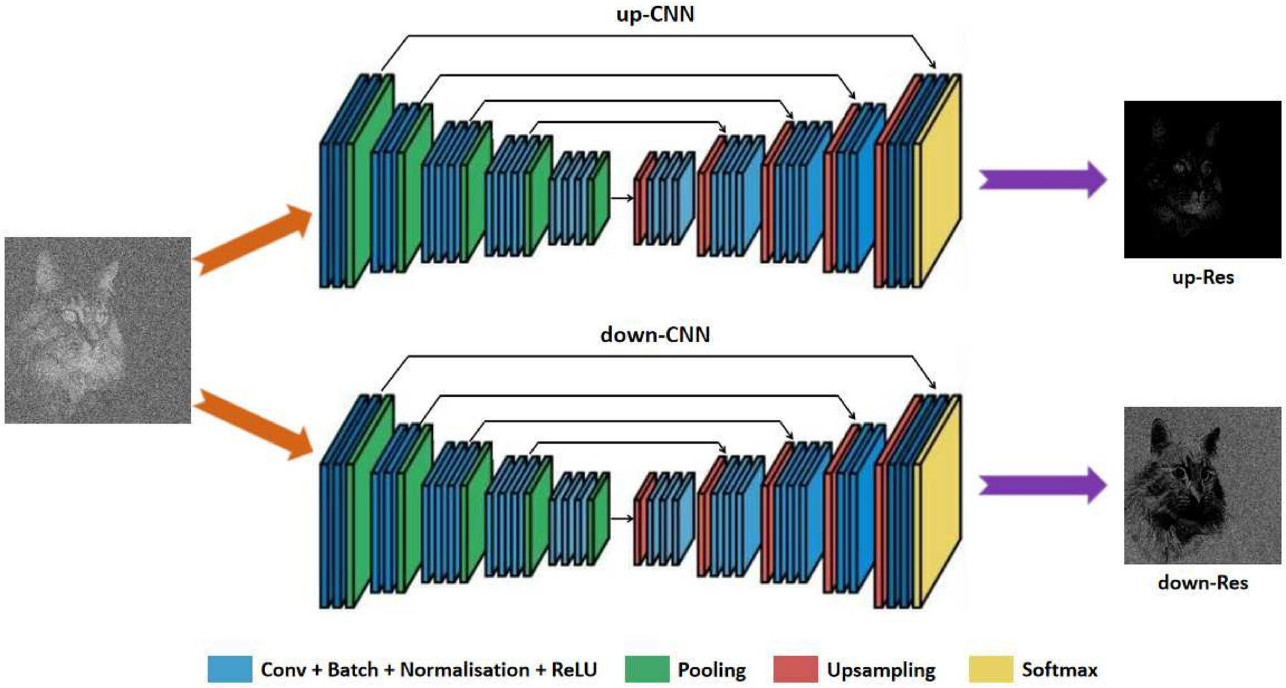

Figure 3. Detailed schematic illustration of the DRU-Net. The image of the object is multiplied with a series of, say M, random patterns. After that, we can get a real-valued 2-dimensional matrix whose size is the same as the image of the object. We then normalize the data and feed them to the DRU-Net. The DRU-Net consists of two networks, which are the up-CNN and down-CNN. The up-CNN has to reconstruct the up-Res image, and the down-CNN aims to learn the down-Res image.

Results

Figure 4 shows a comparison between the results of different GIDLs. The first column shows the four images of the object that are not included in the training set, while the second column shows the images reconstructed by GI simulation, the third shows the results produced by the DnCNN, the fourth shows the results of the U-Net and the fifth shows the results obtained by the DRU-Net.

As clearly shown in Fig. 4, both the U-Net and DRU-Net improve the quality of imaging significantly. The DnCNN performs not well, which is owing to the residual complexity of GI. It is obvious that our proposed DRU-Net achieves much better performance than other methods. In the image of Cat and Yii, more facial details are restored with the DRU-Net compared to the U-Net. In the image of Torre di Pisa, the gray value of DRU-Net is almost close to the image of the object, while the gray value of the U-Net is higher than the image of the object thus its dark background appears quite bright. As to the image of Rose, the image reconstructed using the DRU-Net is almost the same as the image of the object.

The DnCNN, U-Net and DRU-Net parameters for GI are shown in Table 1. The DRU-Net achieves the highest score under the measurement of both the PSNR and SSIM index and the lowest of the RMSE. The DnCNN performs poorly compared to the other two GIDLs. In the image of Yii, the RMSE of the DnCNN is higher than GI, which proves that the residual of GI is different from other image processing.

The SSIM index can evaluate image quality more accurately than the PSNR[26,40–42]. The average SSIM index of GI, DnCNN, U-Net, and DRU-Net are 0.183, 0.328, 0.482, and 0.555 respectively. The SSIM index of the image has been tripled with our proposed DRU-Net.

Discussion

To further examine the robustness and generalization ability of the DRU-Net, we let them reconstruct images with different sampling rates, as $\beta = 24.4\%, 61\%, 122\%, 183\%$. The results are shown in Fig. 5. Here we mainly focus on the U-Net and DRU-Net, dismissing the results of DnCNN because it performs relatively weak compared with the other two networks. The first row shows the reconstructed images of GI; the second row shows that of the U-Net, and the third shows the results of the DRU-Net.

The quality of the reconstructed image continues to grow as the sampling rate increases. If β equals to 24.4%, the reconstructed image of the road sign with the U-Net or DRU-Net are either highly blurred. When β equals 61%, the profile of the object becomes clearer. However, the characters still cannot be reconstructed. When β reaches as high as 183%, the word "STOP" can be recognized clearly. In the reconstructed images with the DRU-Net, more details of the water were restored than the image with U-Net. Nevertheless, the DRU-Net still cannot accomplish a very high-quality image since its PSNR is only 15.21dB. The noise in the process of GI has severely corrupted the image, which causes significant information loss. Since now, most GIDLs make use of supervised learning, and the GI process is taken as a denoising problem. In our future project, we plan to carry out the

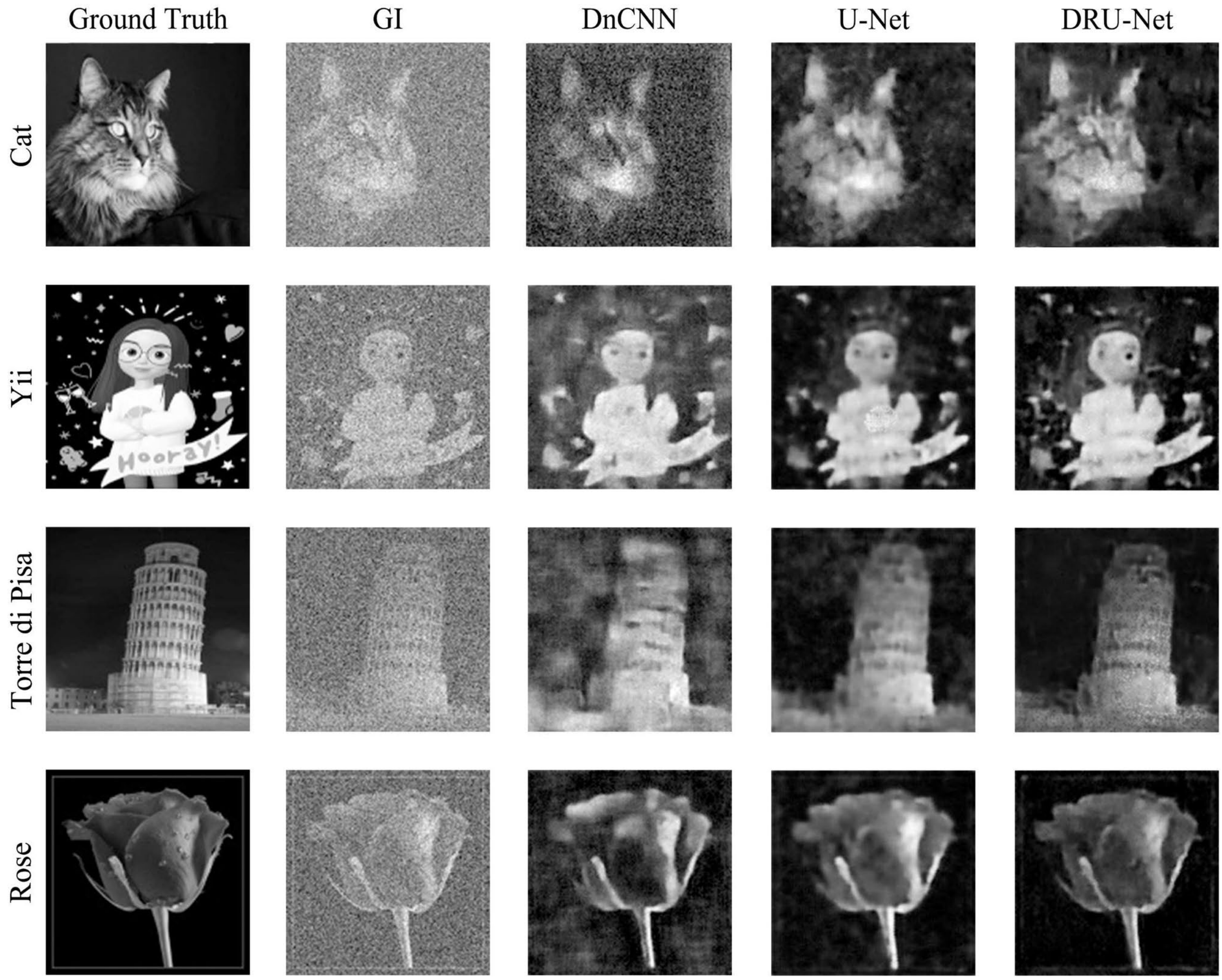

Figure 4. A comparison between the results of GI, DnCNN, U-Net and DRU-Net.

	Images	Cat	Yii	Torre di Pisa	Rose
PSNR	GI	9.463	9.549	12.011	10.228
	DnCNN	15.565	14.262	16.601	18.764
	U-Net	22.007	16.087	21.827	20.072
	DRU-Net	22.409	17.165	22.702	26.544
SSIM	GI	0.179	0.204	0.200	0.172
	DnCNN	0.239	0.338	0.312	0.340
	U-Net	0.535	0.411	0.564	0.482
	DRU-Net	0.584	0.460	0.606	0.684
RMSE	GI	10.564	10.453	10.508	10.746
	DnCNN	10.167	10.679	10.205	9.702
	U-Net	8.843	9.620	8.828	8.767
	DRU-Net	8.668	9.432	8.767	6.783

Table 1. PSNRs, SSIMs, and RMSEs for GI, DnCNN, U-Net, and DRU-Net. Higher PSNR and SSIM and lower RMSE present the higher quality image.

unsupervised learning methods, such as inpainting algorithm[43–45] to "guess" the lost part of the object instead of learning since the significant information loss can hardly be viewed as denoising problem.

In summary, we propose the concept of batch in the pre-processing stage and optimize the simulation algorithm of GI. This allows more reliable simulation data with different sets of random patterns to be acquired within less time. The generalization ability of GIDL has been enhanced with the optimized simulation algorithm.

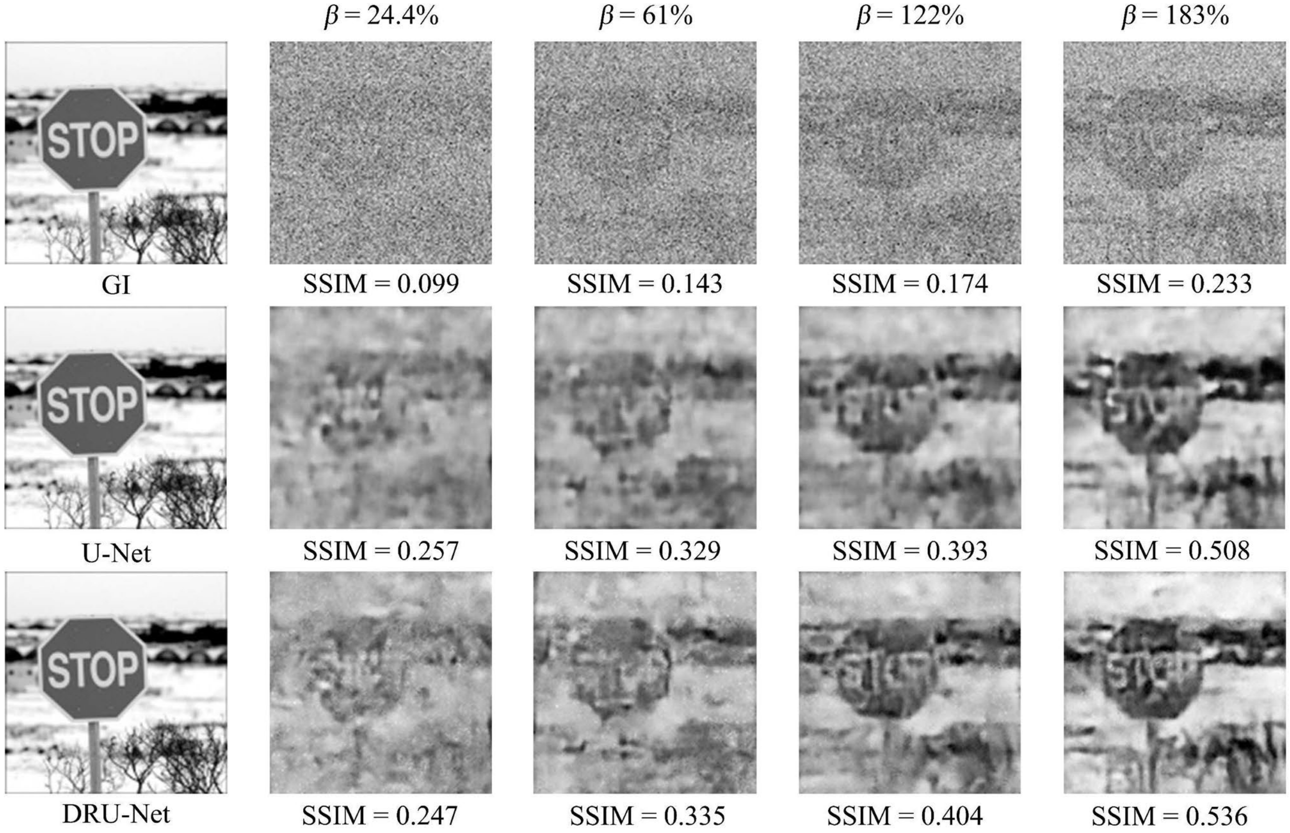

Figure 5. Network results under different sampling rate.

However, we plan to involve the experimental research in our following study. Through careful analysis of the statistical property of GI, we further simplified the residual by dividing it into up-Res and down-Res images to form a new GIDL framework, namely the DR. Also, detailed comparisons of performances between the DnCNN, U-Net, and DRU-Net were discussed. DRU-Net showed much better reconstruction compared to the existing GIDLs. It has tripled the imaging quality of GI in the evaluation of the SSIM index.

Received: 16 April 2020; Accepted: 6 July 2020
Published online: 22 July 2020

References

1. Pittman, T. B., Shih, Y. H., Strekalov, D. V. & Sergienko, A. V. Optical imaging by means of two-photon quantum entanglement. *Phys. Rev. A* **52**, R3429–R3432 (1995).
2. Gatti, A., Brambilla, E., Bache, M. & Lugiato, L. A. Correlated imaging, quantum and classical. *Phys. Rev. A* **70**, 013802 (2004).
3. Bromberg, Y., Katz, O. & Silberberg, Y. Ghost imaging with a single detector. *Phys. Rev. A* **79**, 053840 (2009).
4. Shapiro, J. H. Computational ghost imaging. *Phys. Rev. A* **78**, 061802 (2008).
5. Shirai, T., Setälä, T. & Friberg, A. T. Ghost imaging of phase objects with classical incoherent light. *Phys. Rev. A* **84**, 041801 (2011).
6. Clemente, P., Durán, V., Tajahuerce, E., Torres-Company, V. & Lancis, J. Single-pixel digital ghost holography. *Phys. Rev. A* **86**, 041803 (2012).
7. Sun, B., Welsh, S. S., Edgar, M. P., Shapiro, J. H. & Padgett, M. J. Normalized ghost imaging. *Opt. Express* **20**, 16892–16901 (2012).
8. Ferri, F., Magatti, D., Lugiato, L. A. & Gatti, A. Differential ghost imaging. *Phys. Rev. Lett.* **104**, 253603 (2010).
9. Wang, W. *et al.* Gerchberg-Saxton-like ghost imaging. *Opt. Express* **23**, 28416–28422 (2015).
10. Welsh, S. S. *et al.* Fast full-color computational imaging with single-pixel detectors. *Opt. Express* **21**, 23068–23074 (2013).
11. Zhao, C. *et al.* Ghost imaging lidar via sparsity constraints. *Appl. Phys. Lett.* **101**, 141123 (2012).
12. Katz, O., Bromberg, Y. & Silberberg, Y. Compressive ghost imaging. *Appl. Phys. Lett.* **95**, 131110 (2009).
13. Katkovnik, V. & Astola, J. T. Compressive sensing computational ghost imaging. *J. Opt. Soc. Am. A* **29**, 1556–1567 (2012).
14. Jiying, L., Jubo, Z., Chuan, L. & Shisheng, H. High-quality quantum-imaging algorithm and experiment based on compressive sensing. *Opt. Lett.* **35**, 1206–1208 (2010).
15. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
16. Barbastathis, G., Ozcan, A. & Situ, G. On the use of deep learning for computational imaging. *Optica* **6**, 921–943 (2019).
17. Wang, H., Lyu, M. & Situ, G. eHoloNet: A learning-based end-to-end approach for in-line digital holographic reconstruction. *Opt. Express* **26**, 22603–22614 (2018).
18. Sinha, A., Lee, J., Li, S. & Barbastathis, G. Lensless computational imaging through deep learning. *Optica* **4**, 1117–1125 (2017).
19. Horisaki, R., Takagi, R. & Tanida, J. Learning-based imaging through scattering media. *Opt. Express* **24**, 13738–13743 (2016).
20. Situ, G., Lyu, M., Zheng, S., Wang, H. & Li, G. Learning-based lensless imaging through optically thick scattering media. *Adv. Photonics* **1**, 036002 (2019).
21. Zhou, L., Xiao, Y. & Chen, W. Imaging through turbid media with vague concentrations based on cosine similarity and convolution neural network. *IEEE Photonics J.* **11**, 7801315 (2019).

22. Zhou, L., Xiao, Y. & Chen, W. Machine-learning attacks on interference-based optical encryption: Experimental demonstration. *Opt. Express* **27**, 26143–26154 (2019).
23. Zhou, L., Xiao, Y. & Chen, W. Learning-based attacks for detecting the vulnerability of computer-generated hologram based optical encryption. *Opt. Express* **28**, 2499–2510 (2020).
24. Lyu, M. *et al.* Deep-learning-based ghost imaging. *Sci. Rep.* **7**, 17865 (2017).
25. He, Y. *et al.* Ghost imaging based on deep learning. *Sci. Rep.* **8**, 6469 (2018).
26. Shimobaba, T. *et al.* Computational ghost imaging using deep learning. *Opt. Commun.* **413**, 147–151 (2018).
27. Zhai, X. *et al.* Foveated ghost imaging based on deep learning. *Opt. Commun.* **448**, 69–75 (2019).
28. Wang, F., Wang, H., Wang, H., Li, G. & Situ, G. Learning from simulation: An end-to-end deep-learning approach for computational ghost imaging. *Opt. Express* **27**, 25560–25572 (2019).
29. Jin, K. W., Mccann, M. T., Froustey, E. & Unser, M. Deep convolutional neural network for inverse problems in imaging. *IEEE Trans. Image Process.* **26**, 4509–4522 (2016).
30. Ferguson, T. S. An inconsistent maximum likelihood estimate. *J. Am. Stat. Assoc.* **77**, 831–834 (1982).
31. Mccann, M. T., Jin, K. H. & Unser, M. A review of convolutional neural networks for inverse problems in imaging. *IEEE Signal Process. Mag.* **34**, 85–95 (2017).
32. Srivastava, R. K., Greff, K. & Schmidhuber, J. Highway networks. *Comput. Sci.* (2015).
33. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. *IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)* **770–778**, (2016).
34. Zhang, K., Zuo, W., Chen, Y., Meng, D. & Zhang, L. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Trans. Image Process.* **26**, 3142–3155 (2017).
35. Tian, C., Xu, Y., Fei, L. & Yan, K. Deep learning for image denoising: A survey. *Adv. Intell. Syst. Comput.* **834**, 563–572 (2019).
36. Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional networks for biomedical image segmentation. *Int. Conf. Med. Image Comput. Comput. Assist. Interv.* **9351**, 234–241 (2015).
37. Diakogiannis, F. I., Waldner, F., Caccetta, P. & Wu, C. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS-J. Photogramm. Remote Sens.* **162**, 94–114 (2020).
38. Cheng, M., Mitra, N. J., Huang, X., Torr, P. H. S. & Hu, S. Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**, 569–582 (2015).
39. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *Comput. Sci.* (2014).
40. Wang, Z., Bovik, A., Sheikh, H. R. & Simoncelli, E. P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **13**, 600–612 (2014).
41. Horé, A. & Ziou, D. Image quality metrics: PSNR vs. SSIM. *Int. Conf. pattern Recogn. (ICPR)* **2366–2369** (2010).
42. Mehra, D. R. Estimation of the image quality under different distortions. *Int. J. Adv. Trends Comput. Sci. Eng.* **8** (2016).
43. Wang, Y., Tao, X., Qi, X., Shen, X. & Jia, J. Image inpainting via generative multi-column convolutional neural networks. *Conf. Neural Inf. Process. Syst. (NIPS)* **31** (2018).
44. Liu, G. *et al.* Image inpainting for irregular holes using partial convolutions. *Eur. Conf. Comput. Vis. (ECCV)* **89–105** (2018).
45. Yu, J. *et al.* Generative image inpainting with contextual attention. *IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)* **5505–5514** (2018).

Acknowledgements

This work was supported by the Fundamental Research Funds for Central Universities of China University of Geosciences (Beijing) (No. 2652018057) and National Innovation and Entrepreneurship Training Program for College Students (No. 201911415099).

Author contributions

T.B., Y.Y. and L.G. conceived the idea. J.H. and Y.Z. created the dataset. T.B. and Y.Y. performed simulation. T.B. analysed the results with the assistance of Y.Y. and Y.W.Y.Z. drew the figures. T.B. wrote the manuscript and all authors contributed to the revision. L.G. supervised the project.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to L.G.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2020