Optics Communications 520 (2022) 128450

Contents lists available at ScienceDirect

Optics Communications

journal homepage: www.elsevier.com/locate/optcom

0.8% Nyquist computational ghost imaging via non-experimental deep learning

Haotian Song [a,b], Xiaoyu Nie [a,b], Hairong Su [c], Hui Chen [d], Yu Zhou [b], Xingchen Zhao [a],
Tao Peng [a,*], Marlan O. Scully [a,e]

[a] *Texas A&M University, College Station, TX, 77843, USA*
[b] *School of Physics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China*
[c] *School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China*
[d] *Electronic Materials Research Laboratory, Key Laboratory of the Ministry of Education & International Center for Dielectric Research, Xi'an Jiaotong University, Xi'an, 710049, China*
[e] *Baylor University, Waco, 76706, USA*

ARTICLE INFO

Keywords:
Deep learning
Ghost imaging
Sub-Nyquist sampling

ABSTRACT

We present a framework for computational ghost imaging based on deep learning and customized pink noise speckle patterns. The deep neural network in this work, which can learn the sensing model and enhance image reconstruction quality, is trained merely by simulation. The conventional computational ghost imaging results, deep learning-based ghost imaging results with white and pink noise are compared under multiple sampling ratios at different noise conditions. The experiments are done with digits, English letters, and Chinese characters. We show that the proposed scheme can provide high-quality images with a sampling ratio as low as 0.8% even when the object is outside the training dataset and robust to noisy environments. The method can be applied to a wide range of applications, including those requiring a low sampling ratio, fast reconstruction, or experiencing strong noise interference.

1. Introduction

Ghost imaging (GI) [1–5] is an innovative method for measuring the spatial correlations between light beams. In GI, the signal light field interacts with the object and is collected by a single-pixel detector, and the reference light field, which does not interact with the object, falls on the imaging detector. Thus, neither beam alone contains the image information; it is only revealed by their correlation. Computational ghost imaging (CGI) [6,7] was proposed to further simplify and improve this framework. In CGI, The reference arm that records the speckles is replaced by loading pre-generated patterns directly onto the spatial light modulator or the digital micromirror device (DMD). The unconventional image is then revealed by correlating the sequentially recorded intensities at the single-pixel detector with the corresponding patterns. CGI finds many applications, including wide spectrum imaging [8–10], remote sensing [11], and quantum-secured imaging [12].

However, CGI generally requires a large number of samplings to reconstruct a high-quality image, or the signal would have been submerged under correlation fluctuations and environmental noise. To suppress the environmental noise and correlation fluctuations, the required minimum number of sampling is proportional to the total pixel number of the pattern applied on DMD, *i.e.*, the Nyquist sampling limit [13,14]. The image could have a meager quality with a limited sampling number. This demanding requirement hindered CGI from fully replacing conventional photography. A large number of schemes have been proposed to improve CGI's speed and decrease the sampling ratio (SR) (sub-Nyquist). For instance, compressive sensing imaging can reconstruct images with a relatively low SR by exploiting the sparsity of the objects [15–18]. It nevertheless largely depends on the sparsity of objects and is sensitive to noise [19]. Orthonormalized speckle patterns can be used to suppress the noise and improve the image's quality under a limited sampling number [20,21]. In particular, the orthonormalized colored noise speckle patterns can break the Nyquist limit down to $\sim$ 5% [21]. Fourier and sequence-ordered Walsh–Hadamard patterns, which are orthogonal to each other in time or spatial domain, were also applied to the sub-Nyquist imaging [22–24]. The Russian doll [25] and cake-cutting [26] ordering of Walsh–Hadamard patterns can minimize the SR to 5%–10% Nyquist limit.

Recently, the deep learning (DL) technique has been employed to identify images [27,28] and improve the quality of images with deep neural networks (DNN) [29–37]. Computing ghost imaging via deep learning (CGIDL) has demonstrated a minimum Nyquist limit of $\sim$

---

* Corresponding author.
E-mail address: taopeng@tamu.edu (T. Peng).

https://doi.org/10.1016/j.optcom.2022.128450
Received 22 January 2022; Received in revised form 4 May 2022; Accepted 8 May 2022

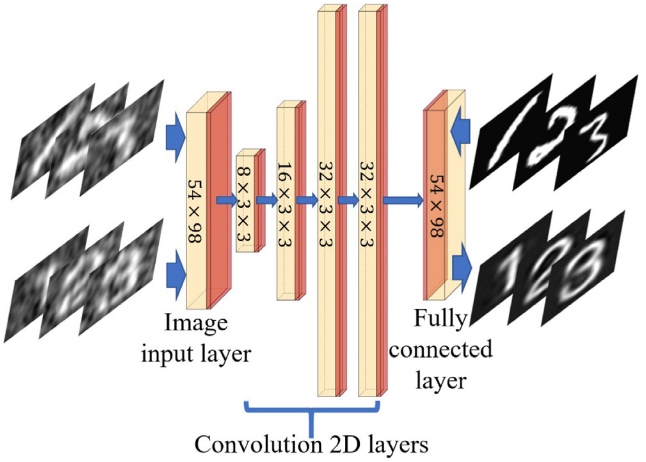

Fig. 1. Architecture of DNN. It consists of four convolution layers, one image input layer, one fully connected layer (yellow), the rectified linear unit, and the batch normalization layers (red). In the upper line are CGI results (training inputs) and handwriting ground truths (training labels); In the bottom line are CGI results from the experiment (test inputs) and CGIDL results (test outputs) with block style.

5% [30,34]. However, lots of the DNNs are trained by experimental CGI results, in which case only when the training environment and the reconstruction environment are highly similar can DNNs be effective. As a result, its universal application is limited and it cannot perform quick reconstructions. At least thousands of inputs have to be generated for the training, which would be extremely time-consuming if conducted every time. Studies have been also conducted to test the effectiveness of non-experimental CGI training DNN [32,33,36,37]. However, the SR is much higher for objects outside of the training dataset. Therefore, despite the proliferation of numerous algorithms, retrieving high-quality images outside of the training group with a low Nyquist limit ratio by non-experimental training remains a challenge for the CGIDL system.

The purpose of this work is to minimize the sampling number and improve the imaging quality using DL and colored noise CGI. It has recently been demonstrated that synthesized colored noise speckle patterns possess unique non-zero correlations between neighbor pixels via amplitude modulation in the spatial frequency domain [38,39]. In particular, The pink noise CGI gives a good image quality under a boisterous environment or pattern distortion when the traditional CGI method fails. Combining DL with pink noise CGI shows that the imaging can be retrieved under an extremely low SR (∼0.8%). We also show that the object used in the experiment can be independent of the training dataset, without reducing the image quality significantly, which can greatly benefit CGIDL in real applications.

2. Deep learning

2.1. Deep neural network

DNN is a basic model for deep learning, but its complexity is enough for analyzing simple datasets in this study. Our DNN model, as shown in Fig. 1, uses four convolution layers, one image input layer, and one fully connected layer. Small 3 × 3 receptive fields are applied throughout the whole convolution layers for better performance [40]. Batch normalization layers (BNL), rectified Linear Unit (ReLU) layers and zero padding are added between each convolution layer. The BNL is functioned to avoid internal covariate shift during the training process and speed up the training of DNN [41]. The ReLU layer applies a threshold operation to each element of the inputs [42]. The zero padding part was designed to maintain the characteristic of input images' boundaries. To customize the size of training pictures, both the input and output layers are set to be 54 × 98. The solver for training is employed by the Stochastic Gradient Descent with Momentum Optimizer (SGDMO) to reduce the oscillation via using momentum. The parameter vector can be updated via equation Eq. (1), which demonstrates the updating process during the iteration.

$$\theta_{\ell+1} = \theta_{\ell} - \alpha \nabla E\left(\theta_{\ell}\right) + \gamma\left(\theta_{\ell} - \theta_{\ell-1}\right), \tag{1}$$

where ℓ is the iteration number, α is the learning rate, θ is the parameter vector, and $E(\theta)$ is the loss function, mean square error (MSE). The MSE is defined as

$$\mathrm{MSE} = \frac{1}{N_{\text{pixel}}} \sum_{i=1}^{N_{\text{pixel}}} \left[\frac{G_i - X_i}{\langle G_{(o)} \rangle}\right]^2. \tag{2}$$

Here, N_{pixel} is the total number of pixels in the pattern. G represents the pixel value of the resulted imaging. $G_{(o)}$ represents pixels that the light ought to be transmitted, *i.e.*, the object area, while $G_{(b)}$ represents pixels that the light ought to be blocked, *i.e.*, the background area. X is the ground truth calculated by

$$X_i = \begin{cases} \langle G_{(o)} \rangle, & \text{Transmission} = 1 \\ \langle G_{(b)} \rangle, & \text{Transmission} = 0 \end{cases} \tag{3}$$

The third part on the right hand side of the equation is the feature of SGDMO, analog to the momentum where γ determines the contribution of the previous gradient step to the current iteration [43]. Two strategies are applied to avoid over-fitting of training images. At the end of DNN, a dropout layer is applied with probability of dropping out input elements being 0.2, which is aimed to reduce the connection between convolution layers and the fully connected layer [44]. Meanwhile, we adopted a step decay schedule for the learning rate. The learning rate dropped from 10^{-3} to 10^{-4} after 75 epochs, which constrain the fitting parameters within a reasonable region. Lower the learning rate could avoid overfitting significantly with constant maximum epochs.

2.2. Network training

The proposed CGIDL scheme requires a training process based on pre-prepared dataset. After training in simulation, it owns ability to reconstruct the images. We chose 3 sets of images as training images. There are 10,000 digits from the MNIST handwritten digit database [45], 80,000 print letters and 50,000 print Chinese characters. All images are resized and normalized to 54 × 98. In this study, we used These training images are reconstructed by the CGI algorithm. The training images and reconstruction training images then feed the DNN model as inputs and outputs, respectively, as shown in Fig. 2(a). The maximum epochs are set as 100, and the training iteration is 31200. The program is implemented via MATLAB R2019a Update 5 (9.6.0.1174912, 64-bit), and the DNN is implemented through DL Toolbox. The GPU-chip NVIDIA GTX1050 is used to accelerate the speed of the computation.

The trained DNN is then tested by simulation and used for retrieving CGI results in the experiments. In the testing part, the CGI algorithm generates reconstructed images from testing images with both the MNIST handwritten digits and block style digits, where the later set is different from images in the training group. As shown in Fig. 2(b), the trained DNN, fed with reconstruction testing images, generates CGIDL results. Comparing the difference between CGIDL and testing images, we could measure the quality of the trained DNN. Well-performed DNN can be used for retrieving CGI in the experiment.

The schematic of the experiment is shown in Fig. 2(c). A CW laser is used to illuminate the DMD, on which the speckle patterns are loaded. The pattern generated by the DMD is then projected onto the object. In our experiment, the size of the speckle patterns is 54 × 98 independent DMD pixels. Each DMD pixel is 16 μm × 16 μm in size. Therefore, the full sampling number in our experiment should be $54 \times 98 = 5292$.

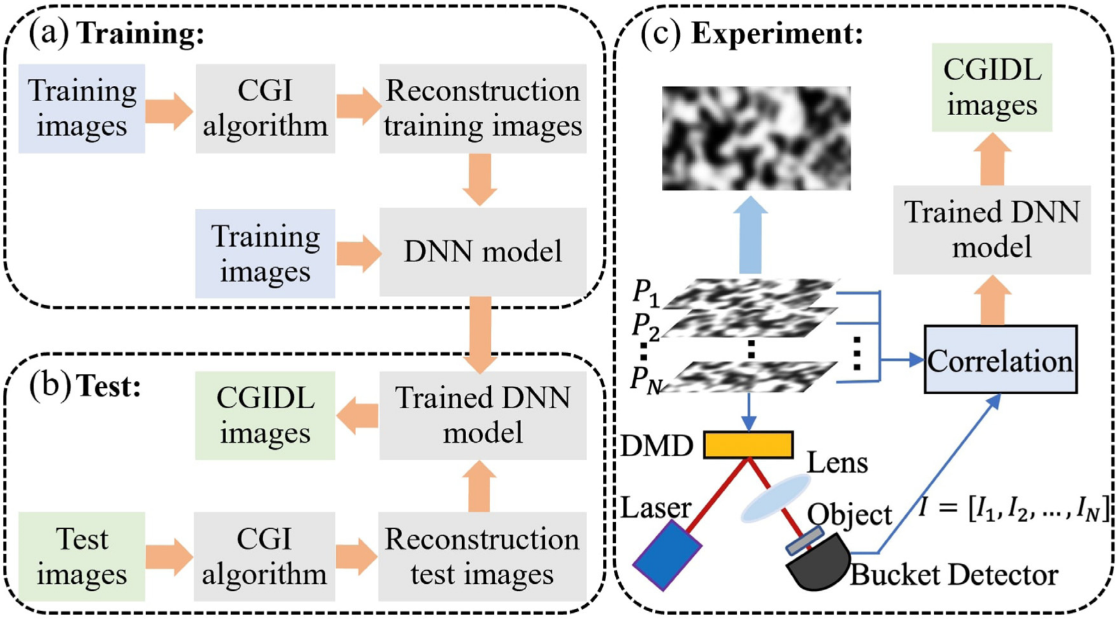

Fig. 2. The flow chart of CGIDL consists of three parts: (a) training, (b) test, and (c) experiment. The DNN model is trained with CGI results from database via simulation. Each DNN model is associated with a specific SR and pattern type. The simulation testing process and experimental measurements use both the handwriting digits and block style digits. The experimental part for CGI uses pink noise and white noise speckle patterns, and their CGI results are ameliorated by trained DNN model.

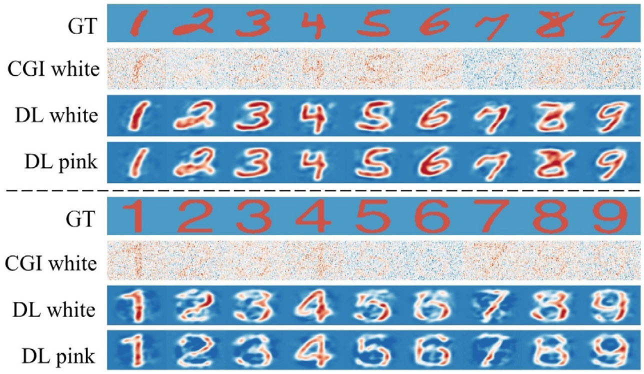

Fig. 3. Simulation results without noise. The upper part used handwriting digits 1–9 from the training dataset, and the lower part used block style digits 1–9, which is outside the training dataset. All the simulations are done at $\beta = 5\%$. GT: ground truth.

In the CGI process, the quality of the images is proportional to the SR, which is the ratio between the number of illumination patterns N_{pattern} and N_{pixel} [29,34,46]:

$$\beta = N_{\text{pattern}}/N_{\text{pixel}}. \tag{4}$$

Following, we evaluate the reconstruction performance of the trained networks using white noise speckle patterns (DL white) and pink noise speckle patterns (DL pink), as well as conventional CGI (CGI white), with different SRs. Specifically, β of 50% (2646 patterns), 5% (265 patterns), and 1% (53 patterns) are used for DL white, 5%, 0.8% (42 patterns), and 0.5% (26 patterns) for DL pink, 100% (5292 patterns), 50%, and 10% (529 patterns) for CGI white. Each network was associated with a specific SR and pattern type in the training and testing process to achieve the best performance.

3. Simulation

We performed a set of simulations to test the robustness of our method, *i.e.*, with different datasets, under different noise conditions, and at different SRs. Two sets of testing images are used in the simulation. One of which is the handwriting digits 1–9 from the training set, the other is the block style digits 1–9, which are independent of the training set. These images have 28 × 28 pixels and are resized into 54 × 98 by widening and amplification. We started our simulation from the comparison of the CGI white, DL white and DL pink without noise at $\beta = 5\%$, as shown in Fig. 3. The upper part is with the handwriting digits 1–9, the lower part is with the block style digits 1–9. Apparently, at this low SR, the traditional CGI method fails to retrieve the images in both cases. On the other hand, both DL methods work much better than the traditional CGI. For digits from the training dataset, both methods work almost equally well. For digits from outside the training dataset, DL pink works already better than DL white. For example, the DL white barely can distinguish digits '3' and '8', but DL pink can retrieve all the digits images.

In real application, there always exist noise in the measurement. It is therefore worthwhile checking the performances of different methods under the influence of noise. We then performed another set of simulations with added grayscale random noise. The signal-to-noise ratio (SNR) in logarithmic decibel scale is defined as

$$\text{SNR} = 10 \log \frac{P_s}{P_b}, \tag{5}$$

where P_s is the average signal and P_b is the average noise background. In the experiment, the environmental and electronics noise are measured by the bucket detector with the laser blocked. The signals are collected, in addition to the noises, while conducting the imaging experiments, *i.e.*, when speckle patterns illuminate the objects. Both the signal and noise follow the Gaussian distribution. Thus, we calculate the SNR by their mean values. We then adjust the SNR by adjusting the incident laser power, and therefore the average signal P_s. Here the SNR is set to be 4.77 dB. In the simulation, we use the signal and noise levels precisely as measured in the experiment. As shown in Fig. 4, the upper part is the simulation with digits 2, 3, 5, and 6 from the training dataset, and the lower part is the simulation with digits 2, 3, 5, and 6 from the block style dataset. For both datasets, β of 100%, 50%, and 10% are chosen for CGI white, 50%, 5%, and 1% for DL white, 5%, 0.8%, and 0.5% for DL pink. The image quality is better with the increase of β for all cases, as expected. As for the CGI white case, it

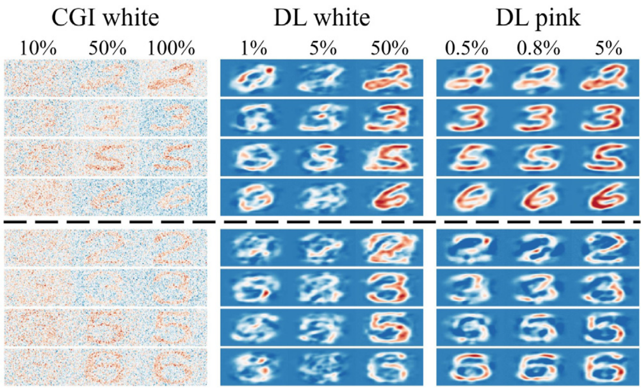

Fig. 4. Simulation results of handwriting (top) and block style (bottom) digits 2, 3, 5, 6 with the SNR of 4.77 dB. The results of CGI white are done at β of 10%, 50%, and 100%, DL white with β of 1%, 5%, and 50%, and DL pink with β of 0.5%, 0.8%, and 5%.

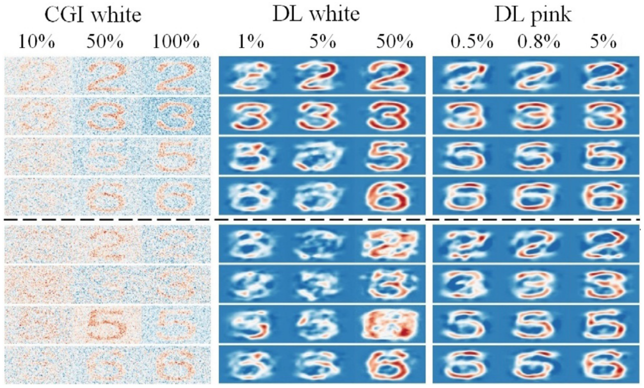

Fig. 5. Experimental results with the SNR of 14.90 dB (upper) and 4.77 dB (lower). Objects are block style digits 2, 3, 5, 6. Different SRs are shown for different methods: CGI white are done at β of 10%, 50%, and 100%, DL white with β of 1%, 5%, and 50%, and DL pink with β of 0.5%, 0.8%, and 5%.

can only give marginally visible images when the SR is beyond 50%. The DL white, can retrieve the digits from the training dataset when $\beta = 1\%$. However, for the block style digits, it fails to do so even when $\beta = 5\%$. Unlike the previous case with no noise, there is a significant difference between objects from the training dataset and outside the training dataset. Lastly, we note that the DL pink trained network can retrieve the training dataset when $\beta = 0.5\%$. It can also retrieve clear images for the block style digits at $\beta = 0.8\%$. If we compare the block style images at $\beta = 5\%$ for both DL white and DL pink with the no noise case in Fig. 3, it obvious that the quality of the former is largely affected by the noise, and the latter is barely affected.

4. Experiments

To demonstrate the advantage and applicability of DL pink, we perform experiments with the non-experimental and one-time trained model. The experiments are done with digits 2, 3, 5, and 6 with block style. The block style is chosen to better compare the different behaviors of all three methods. We manage to start from a relatively low noise level of SNR = 14.90 dB. The results are shown in upper part of Fig. 5. We can see at this noise level, the CGI white method barely can distinguish the images from the noisy background even at $\beta = 100\%$. The DL white trained network, while giving clear images at $\beta = 50\%$, fails to fully image the digits at $\beta = 5\%$. This is mainly due to the objects are outside the training set, reveal one of the shortcomings of the standard DL network. On the other hand, our DL pink trained network can still give clear results even when $\beta = 0.5\%$.

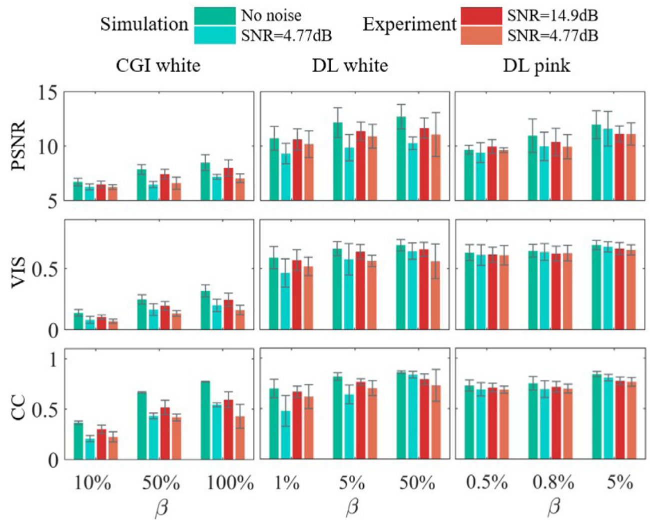

Fig. 6. PSNR, VIS, and CC for simulation and experiments of the block style digits 2, 3, 5, 6 with three categories: CGI white (β of 10%, 50%, and 100%), DL white (β of 1%, 5%, and 50%), and DL pink (β of 0.5%, 0.8%, and 5%).

We then increase the noise level to SNR = 4.77 dB, which is the same as the simulation case so we can have a fair comparison. The experimental results are shown in the lower part of Fig. 5. The CGI white completely fail to image the digits even at $\beta = 100\%$. The DL white trained network is also largely affected by the noisy environment, and not able to fully retrieve the images at $\beta = 50\%$. On the other hand, the DL pink method can still image all digits at the SR of 0.8%. If we compare these results to the corresponding low noise case, we can see that the image qualities do not change much, indicating our trained network is robust to noise. Also, the results with $\beta = 0.8\%$ is better than the standard DL white network at $\beta = 50\%$, which is about two orders higher.

To quantitatively justify the quality of reconstructed block style images, we compare three evaluating indicators of image quality, *i.e.*, the peak signal to noise ratio (PSNR), the visibility (VIS), and the correlation coefficient (CC):

$$\begin{aligned}
\mathrm{PSNR} &= 10 \times \log_{10}[\frac{(2^k-1)^2}{\mathrm{MSE}}],\\
\mathrm{VIS} &= \frac{\langle G_{(\mathrm{o})}\rangle - \langle G_{(\mathrm{b})}\rangle}{\langle G_{(\mathrm{o})}\rangle + \langle G_{(\mathrm{b})}\rangle},\\
\mathrm{CC} &= \frac{\mathrm{Cov}(G,X)}{\sqrt{\mathrm{Var}(G)\mathrm{Var}(X)}}.
\end{aligned} \tag{6}$$

Here Var( ) is the variance of its arguments, Cov( ) is the covariance of its arguments, k is the gray level of the image, and in our experiment $k \equiv 1$.

The results for all cases including simulation without and with noise, experiment with high and low SNR, are shown in Fig. 6. The PSNR, VIS, and CC all indicate that the CGIDL methods are much better than the traditional CGI method. Indeed, as shown in the simulation case, the image quality of CGIDL at 5% is already better than CGI at full SR for all the situations. When we compare the two DL methods, we see that in general DL pink is much better than DL white, as also suggested from Figs. 3, 4, and 5. Since the network is trained using MSE as the loss function, the PSNR of simulation without noise at 5% is very similar for both cases. However, when the noise increases, the PSNR of DL white starts to decrease, while the PSNR of DL pink does not change much. The VIS and CC also have similar behavior as PSNR. We note here that all three indicators suggest DL pink works better than the other two methods, in the experimental results with low SNR, DL pink of 5% SR is already better than the DL white with 50% SR.

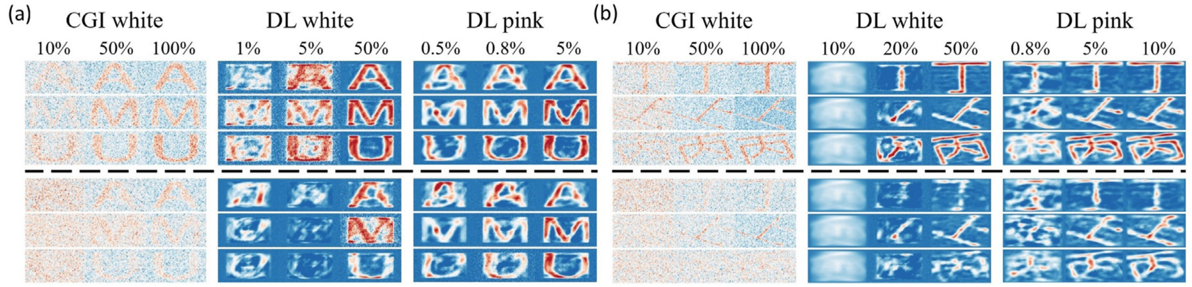

Fig. 7. Experimental results with the SNR of 14.90 dB (upper) and 4.77 dB (lower). Objects are (a) letters A, M, and U; and (b) Chinese characters 'Ding', 'shang', and 'bing'. Different SRs are shown for different methods: For the letters, CGI white are done with β of 10%, 50%, and 100%, DL white with β of 1%, 5%, and 50%, and DL pink with β of 0.5%, 0.8%, and 5%; for the Chinese characters, CGI white are done at β of 10%, 50%, and 100%, DL white with β of 10%, 20% (1058 patterns), and 50%, and DL pink with β of 0.8%, 2% (106 patterns), and 5%.

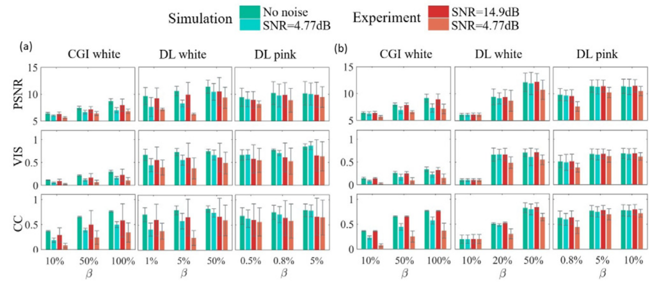

Fig. 8. PSNR, VIS, and CC for simulation and experiments of (a) letters A, M, and U; and (b) Chinese characters 'ding', 'shang', and 'bing'. There are three categories: CGI white (β of 10%, 50%, and 100%), DL white (β of 1%, 5%, and 50% for the letters; 10%, 20%, and 50% for Chinese characters), and DL pink (β of 0.5%, 0.8%, and 5% for letters; 0.8%, 5%, and 10% for Chinese characters).

To further demonstrate the universal ability of our scheme, we also experimentally tested our system with English letters and Chinese characters, as shown in Fig. 7. The same DNN model is used, and trained by simulation only. The letters A, M and U, and Chinese characters 'Ding', 'shang', and 'bing' used in the experiments are also outside the training dataset. Firstly, we notice that the quality of DL white is largely affected when the complexity of the object increases. It cannot retrieve the letters at SR of 1%. When Chinese characters are used, DL white failed to give any results even at SR of 10%. When significant noise is present, DL white can barely retrieve images of the letters at SR of 50%, but failed to retrieve the Chinese characters at such high SR. On the other hand, DL pink is less affected. It has almost the same performance for the English letters as compared to the block style digits, with both low and high noise levels. In the Chinese character case, it is only slightly affected. It can still retrieve the basic structures of these characters at 0.8%, and result clear images at 5%. Even with the present of strong noise, 10% is enough to give clear images.

We then plot PSNR, VIS, and CC for the experimental results together with simulation results, as shown in Fig. 8. We can see that, again, the PSNR, VIS, and CC all indicate that the CGIDL methods are much better than the traditional CGI method when applied to English letters. However, the traditional CGI method is already better than DL white at SR of 10%. It is when DL white can retrieve the structure of the objects, such as 50% of the SR, it outperforms CGI white again. When we compare the two DL methods, we see that DL pink is much better than DL white, and when the noise level is relatively high, DL pink is only slightly affected, as also suggested from Figs. 7.

5. Discussion

In conclusion, we have demonstrated a deep-learning imaging method with pink noise speckle patterns. The DNN is trained using only simulation data. The trained network can then be applied to various conditions, including objects outside the training set and experiments with strong noise. We have demonstrated imaging results with extremely low SRs in both simulations and experiments. Furthermore, we evaluated the PSNR, VIS, and CC quality of the images outside the training dataset for both simulation and experimental results.

Our results suggest that the DL pink scheme has a great advantage, especially in the low sampling region. There are mainly two factors contributing to the outstanding performance of DL pink pattern speckles. Firstly, deep neural networks could be trained to analyze and construct the characteristics of images. This ability enables the network to reconstruct the image with noise suppression. Secondly, pink noise patterns offer a better platform for DNN, especially in the low sampling region and high environmental noise level, due to its unique power spectrum distribution and noise-robust feature in the CGI measurement. This one-time, noise-robust, and non-experimental training CGIDL is eligible to be implemented in various situations and has a wide range of application prospects. The pink noise speckle patterns, trained DNN with various SRs, and their raw encoding programs are encapsulated and uploaded online (https://github.com/XJTU-TAMU-CGI/CGIDL). People who need a quick sampling function on CGIDL can utilize this universal system to get ameliorated results in other CGIDL systems. Further works can reach to other imaging and spectroscopy systems by loss function adjustment and speckle pattern optimization, in order to get spatial, frequency, or time-resolution. In addition to results amelioration, DL may also have great potential to generate optimized speckle patterns for a variety of tasks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available upon reasonable request.

Acknowledgments

This research is supported by grants from: Air Force Office of Scientific Research, United States (Award No. FA9550-20-1-0366 DEF), Office of Naval Research, United States (Award No. N00014-20-1-2184), Robert A. Welch Foundation, United States (Grant No. A-1261), National Science Foundation, United States (Grant No. PHY-2013771).

References

[1] T.B. Pittman, Y.H. Shih, D.V. Strekalov, A.V. Sergienko, Optical imaging by means of two-photon quantum entanglement, Phys. Rev. A 52 (1995) R3429–R3432.

[2] R.S. Bennink, S.J. Bentley, R.W. Boyd, Two-photon coincidence imaging with a classical source, Phys. Rev. Lett. 89 (2002) 113601.

[3] A. Valencia, G. Scarcelli, M. D'Angelo, Y. Shih, Two-photon imaging with thermal light, Phys. Rev. Lett. 94 (2005) 063601.

[4] X.-H. Chen, Q. Liu, K.-H. Luo, L.-A. Wu, Lensless ghost imaging with true thermal light, Opt. Lett. 34 (2009) 695–697.

[5] F. Ferri, D. Magatti, L. Lugiato, A. Gatti, Differential ghost imaging, Phys. Rev. Lett. 104 (2010) 253603.

[6] Y. Bromberg, O. Katz, Y. Silberberg, Ghost imaging with a single detector, Phys. Rev. A 79 (2009) 053840.

[7] J.H. Shapiro, Computational ghost imaging, Phys. Rev. A 78 (2008) 061802.

[8] D. Shrekenhamer, C.M. Watts, W.J. Padilla, Terahertz single pixel imaging with an optically controlled dynamic spatial light modulator, Opt. Express 21 (2013) 12507–12518.

[9] R.S. Aspden, N.R. Gemmell, P.A. Morris, D.S. Tasca, L. Mertens, M.G. Tanner, R.A. Kirkwood, A. Ruggeri, A. Tosi, R.W. Boyd, et al., Photon-sparse microscopy: visible light imaging using infrared illumination, Optica 2 (2015) 1049–1052.

[10] Y. Klein, A. Schori, I. Dolbnya, K. Sawhney, S. Shwartz, X-ray computational ghost imaging with single-pixel detector, Opt. Express 27 (2019) 3284–3293.

[11] N.D. Hardy, J.H. Shapiro, Computational ghost imaging versus imaging laser radar for three-dimensional imaging, Phys. Rev. A 87 (2013) 023820.

[12] P. Clemente, V. Durán, E. Tajahuerce, J. Lancis, et al., Optical encryption based on computational ghost imaging, Opt. Lett. 35 (2010) 2391–2393.

[13] R.L. Cook, Stochastic sampling in computer graphics, ACM Trans. Graph. 5 (1986) 51–72.

[14] J.A. Tropp, J.N. Laska, M.F. Duarte, J.K. Romberg, R.G. Baraniuk, Beyond nyquist: Efficient sampling of sparse bandlimited signals, IEEE Trans. Inform. Theory 56 (2009) 520–544.

[15] O.S. Magana-Loaiza, G.A. Howland, M. Malik, J.C. Howell, Compressive object tracking using entangled photons, Appl. Phys. Lett. 102 (2013).

[16] O. Katz, Y. Bromberg, Y. Silberberg, Compressive ghost imaging, Appl. Phys. Lett. 95 (2009) 739.

[17] Z.-H. Xu, W. Chen, J. Penuelas, M. Padgett, M.-J. Sun, 1000 Fps computational ghost imaging using led-based structured illumination, Opt. Express 26 (2018) 2427–2434.

[18] C. Yi, C. Zhengdong, F. Xiang, C. Yubao, L. Zhenyu, Compressive sensing ghost imaging based on image gradient, Optik 182 (2019) 1021–1029.

[19] J. Du, W. Gong, S. Han, The influence of sparsity property of images on ghost imaging with thermal light, Opt. Lett. 37 (2012) 1067–1069.

[20] B. Luo, P. Yin, L. Yin, G. Wu, H. Guo, Orthonormalization method in ghost imaging, Opt. Express 26 (2018) 23093–23106.

[21] X. Nie, X. Zhao, T. Peng, M.O. Scully, Sub-nyquist computational ghost imaging with orthonormal spectrum-encoded speckle patterns, Phys. Rev. A 105 (2022) 043525.

[22] Z. Zhang, X. Ma, J. Zhong, Single-pixel imaging by means of fourier spectrum acquisition, Nature Commun. 6 (2015) 1–6.

[23] L. Wang, S. Zhao, Fast reconstructed and high-quality ghost imaging with fast walsh–hadamard transform, Photonics Res. 4 (2016) 240–244.

[24] Z. Zhang, X. Wang, G. Zheng, J. Zhong, Hadamard single-pixel imaging versus fourier single-pixel imaging, Opt. Express 25 (2017) 19619–19639.

[25] M.-J. Sun, L.-T. Meng, M.P. Edgar, M.J. Padgett, N. Radwell, A russian dolls ordering of the hadamard basis for compressive single-pixel imaging, Sci. Rep. 7 (2017) 1–7.

[26] W.-K. Yu, Super sub-nyquist single-pixel imaging by means of cake-cutting hadamard basis sort, Sensors 19 (2019) 4122.

[27] X. He, S.-M. Zhao, L. Wang, Handwritten digit recognition based on ghost imaging with deep learning, Chin. Phys. B (2020).

[28] J. Li, M. Le, J. Wang, W. Zhang, B. Li, J. Peng, Object identification in computational ghost imaging based on deep learning, Appl. Phys. B 126 (2020) 1–10.

[29] M. Lyu, W. Wang, H. Wang, H. Wang, G. Li, N. Chen, G. Situ, Deep-learning-based ghost imaging, Sci. Rep. 7 (2017) 17865.

[30] Y. He, G. Wang, G. Dong, S. Zhu, H. Chen, A. Zhang, Z. Xu, Ghost imaging based on deep learning, Sci. Rep. 8 (2018) 6469.

[31] T. Shimobaba, Y. Endo, T. Nishitsuji, T. Takahashi, Y. Nagahama, S. Hasegawa, M. Sano, R. Hirayama, T. Kakue, A. Shiraki, et al., Computational ghost imaging using deep learning, Opt. Commun. 413 (2018) 147–151.

[32] F. Wang, H. Wang, H. Wang, G. Li, G. Situ, Learning from simulation: An end-to-end deep-learning approach for computational ghost imaging, Opt. Express 27 (2019) 25560–25572.

[33] X. Zhai, Z.-d. Cheng, Y. Chen, Z.-y. Liang, Y. Wei, et al., Foveated ghost imaging based on deep learning, Opt. Commun. 448 (2019) 69–75.

[34] H. Wu, R. Wang, G. Zhao, H. Xiao, D. Wang, J. Liang, X. Tian, L. Cheng, X. Zhang, Sub-nyquist computational ghost imaging with deep learning, Opt. Express 28 (2020) 3846–3853.

[35] S. Rizvi, J. Cao, K. Zhang, Q. Hao, Deepghost: real-time computational ghost imaging via deep learning, Sci. Rep. 10 (2020) 1–9.

[36] H. Wu, R. Wang, G. Zhao, H. Xiao, J. Liang, D. Wang, X. Tian, L. Cheng, X. Zhang, Deep-learning denoising computational ghost imaging, Opt. Lasers Eng. 134 (2020) 106183.

[37] T. Bian, Y. Yi, J. Hu, Y. Zhang, Y. Wang, L. Gao, A residual-based deep learning approach for ghost imaging, Sci. Rep. 10 (2020) 1–8.

[38] X. Nie, F. Yang, X. Liu, X. Zhao, R. Nessler, T. Peng, M.S. Zubairy, M.O. Scully, Noise-robust computational ghost imaging with pink noise speckle patterns, Phys. Rev. A 104 (2021) 013513.

[39] Z. Li, X. Nie, F. Yang, X. Liu, D. Liu, X. Dong, X. Zhao, T. Peng, M.S. Zubairy, M.O. Scully, Sub-rayleigh second-order correlation imaging using spatially distributive colored noise speckle patterns, Opt. Express 29 (2021) 19621–19630.

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[41] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015, arXiv preprint arXiv:1502.03167.

[42] V. Nair, G.E. Hinton, Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th International Conference on International Conference on Machine Learning, Series and Number ICML'10, Omni Press, Madison, WI, USA, 2010, pp. 807–814.

[43] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.

[44] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (2014) 1929–1958.

[45] L. Deng, The mnist database of handwritten digit images for machine learning research [best of the web], IEEE Signal Process. Mag. 29 (2012) 141–142.

[46] W. Wang, X. Hu, J. Liu, S. Zhang, J. Suo, G. Situ, Gerchberg-saxton-like ghost imaging, Opt. Express 23 (2015) 28416–28422.