Deep-learned speckle pattern and its application to ghost imaging

TAO PENG[1,‡], HAOTIAN SONG[2,3], XINGCHEN ZHAO[1], ZHEDONG ZHANG[4,†], AND MARLAN O. SCULLY[1,5,6]

[1] Texas A&M University, College Station, Texas, 77843, USA
[2] School of Physics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China
[3] College of Physics & Astronomy, University of Manchester, Manchester M13 9PL, UK
[4] Department of Physics, City University of Hong Kong, Kowloon, Hong Kong SAR
[5] Baylor University, Waco, 76706, USA
[6] Princeton University, Princeton, New Jersey 08544, USA
[†] zzhan26@cityu.edu.hk
[‡] taopeng@tamu.edu

Compiled October 23, 2021

In this paper, we present a method for speckle pattern design using deep learning. The speckle patterns possess unique features after experiencing convolutions in Speckle-Net, our well-designed framework for speckle pattern generation. We then apply our method to the computational ghost imaging system. The standard deep learning-assisted ghost imaging methods use the network to recognize the reconstructed objects or imaging algorithms. In contrast, this innovative application optimizes the illuminating speckle patterns via Speckle-Net with specific sampling rates. Our method, therefore, outperforms the other techniques for ghost imaging, particularly its ability to retrieve high-quality images with extremely low sampling rates. It opens a new route towards non-trivial speckle generation by referring to a standard loss function on specified objectives with the modified deep neural network. It also has great potential in other areas using speckle patterns such as dynamic speckle illumination microscopy, structured illumination microscopy, x-ray imaging, photo-acoustic imaging, and optical lattices.

http://dx.doi.org/10.1364/ao.XX.XXXXXX

1. INTRODUCTION

Typical speckle patterns are generated when light is scattered or diffused from the inhomogeneous rough media [1]. The statistics of the speckles depends on the incident light field [2]. In particular, scattered laser speckle is known as the Rayleigh speckle since the amplitude of the scatted filed obey the Rayleigh statistics [3]. Speckle patterns can also be produced by sources such as x-rays [4], microwaves [5], and Terahertz radiation [6] besides visible light. The study of speckle patterns has been conducted in many scenarios such as waveguides [7], fibers [8], and nanowires [9]. The wide range of applications of the speckle patterns include spectroscopy [10], microscopy [11, 12], interferometry [13], metrology techniques [14, 15], and correlated disorder in optical lattices [16–18]. In these applications, the speckle patterns act as efficient random carriers of encoding the spatial information within the systems and later on being decoded. Therefore, to retain well-performed data carriers, manipulation of its inherent statistical properties is highly demanding from the perspective of efficiency, accuracy, and robustness.

Speckle pattern also plays an essential role in ghost imaging [19, 20]. Standard Rayleigh speckles have been used for ghost imaging for decades [21, 22]. Later on, the spatial light modulator (SLM) and the digital micromirror device (DMD) are used as convenient and powerful tools for speckle pattern formation. Various synthesized speckle patterns [23–26] are generated by customizing and regulating amplitude and phase of the electromagnetic field or directly designing and adjusting the power spectrum of the speckle patterns. Recently, efforts have been made to generate orthonormalized [27], Walsh-Hadamard [28–30], and colored noise [31] speckle patterns for sub-Nyquist sampling imaging. To date, the synthesized speckle patterns are typically generated from customizing the power spectrum, vortex, amplitude of either the intensity or field distribution to finally justify their spatial correlations. Therefore, tremendous work must be done, from complicated theoretical calculations and many experimental attempts to decide the parameters discussed above. Besides, the speckle patterns used for sub-Nyquist imaging are not optimal for any specific sampling rate (SR).

In this work, we introduce a universally applicable speckle pattern generating method based on deep learning (DL), namely Speckle-Net. We design a specific deep neural network (DNN) customized to speckle pattern generation by utilizing the convolution concept in the convolutional neural network (CNN). The kernels in CNN are used to adjust the second-order correlation of the speckle patterns. The Speckle-Net training is a pre-processing technique only based on the optical system and the loss function. During the training process, Speckle-Net continuously improves the kernel values in each training epoch until they reach the optimum values referring to the loss function. We then implement this technique on the ghost imaging system, in which the speckle patterns are optimized for any given SR. The optimized speckle pattern can then be applied to any computational ghost imaging (CGI) system, resulting high-quality images even at extremely low SRs. Speckle-Net can also be applied to other illumination systems that require optimizing the speckle patterns of illumination by selecting suitable evaluators as the loss function in a one-time training process.

2. PRINCIPLES OF SPECKLE-NET

A. Correlation Modulation by Kernels

Kernel, a popular concept in DL, is usually applied in CNN containing each convolutional layer. It functions as a matrix that makes convolution on speckle patterns to minimize the size of patterns and localizes the feature within areas in the pattern. The output patterns are mainly modulated by kernels convoluting the initial pattern. In our strategy, multiple unique kernels are designed to act on initial speckle patterns $P_i(x,y)$ in each layer of DL, after which multiple different speckle patterns $P'_i(x,y)$ $(i = 1, \cdots, N)$ are generated, as is shown in Fig. 1. Speckle patterns after multiple convolution transformations ought to be distinct from each other, and their spatial intensity fluctuation correlation distribution will be modulated by multiple kernels in multiple layers with the instruction of standard loss function in DL.

The principle of the second order correlation modulation is briefly explained here. We use in total N kernels $C_i(m,n)$, where m,n are coordinates of the kernel. The speckle pattern P after convolution can be expressed as $P'_i(x,y) = \sum_{m,n} C_i(m,n)P(x+m,y+n)$, where x,y are coordinates in the pattern. The average value of the resulted patterns $P'_i(x,y)$ is

$$\begin{aligned}\bar{P}'(x,y) &= \frac{1}{N}\sum_{i=1}^{N}\sum_{m,n} C_i(m,n)P(x+m,y+n) \\ &= \sum_{m,n}\bar{C}(m,n)P(x+m,y+n). \end{aligned} \quad \textbf{(1)}$$

We then have

$$\begin{aligned}\Delta P'_i(x,y) &\equiv P'_i(x,y) - \bar{P}'(x,y) \\ &= \sum_{m,n}(C_i(m,n) - \bar{C}(m,n)) - P(x+m,y+n) \\ &= \sum_{m,n}\Delta C_i(m,n)P(x+m,y+n), \end{aligned} \quad \textbf{(2)}$$

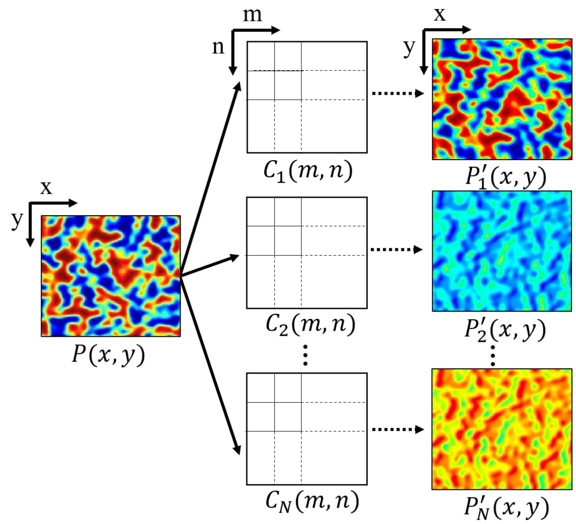

Fig. 1. One layer convolution in our featured neural network. Multiple kernels are attached on a single speckle pattern. P is the initial or convoluted pattern from the former layer, and P's are the output patterns from the current layer. Each subscript indicates the correspondence between convoluted speckle patterns and kernels.

and the correlation function of the resulted patterns is

$$\begin{aligned}\Gamma^{(2)}(\Delta x,\Delta y) &= \langle \Delta P'_i(x_1,y_1)\Delta P'_i(x_2,y_2)\rangle \\ &= \Big\langle [\sum_{m_1,n_1}\Delta C_i(m_1,n_1)P(x_1+m_1,y_1+n_1)] \\ &\quad \times [\sum_{m_2,n_2}\Delta C_i(m_2,n_2)P(x_2+m_2,y_2+n_2)]\Big\rangle \\ &= \sum_{m_{1,2},n_{1,2}}\langle \Delta C(m_1,n_1)\Delta C(m_2,n_2)\rangle \\ &\quad \times P(x_1+m_1,y_1+n_1)P(x_2+m_2,y_2+n_2) \\ &= \sum_{m_{1,2},n_{1,2}}\Gamma^{(2)}_C(\Delta m,\Delta n) \\ &\quad \times P(x_1+m_1,y_1+n_1)P(x_2+m_2,y_2+n_2), \end{aligned} \quad \textbf{(3)}$$

where $\Delta x \equiv x_1 - x_2$, $\Delta y \equiv y_1 - y_2$. It is clear shown in Eq. (3) that the correlation function of the generated speckle patterns depends on the correlation function of the kernel $\Gamma^{(2)}_C(\Delta m,\Delta n)$ and the initial pattern. Thus, the process of adjustment on each kernel in DL is aimed at producing desired correlations with respect to the initial speckle pattern, which can be seen as weight parameters. The convolution process of a single pattern can be understood as a re-distribution of the spatial correlation from different kernels.

B. Structure of the Speckle-Net

Speckle-Net consists of multi-branch and simplified layers, as shown in Fig. 2 (a)[1]. Single pattern padded with reflection of their boundaries plays the role of input. To provide the flexibility of correlation adjustment, convolution layers with a relatively large kernel size of 10×10, a Rectified Linear Unit (ReLU) [32], and a Batch Normalization Layer (BNL) [33], are combined into a series of processes in each layer. The layers share similarities

[1]The raw codes of Speckle-Net can be found on https://github.com/XJTU-TAMU-CGI/PatternDL

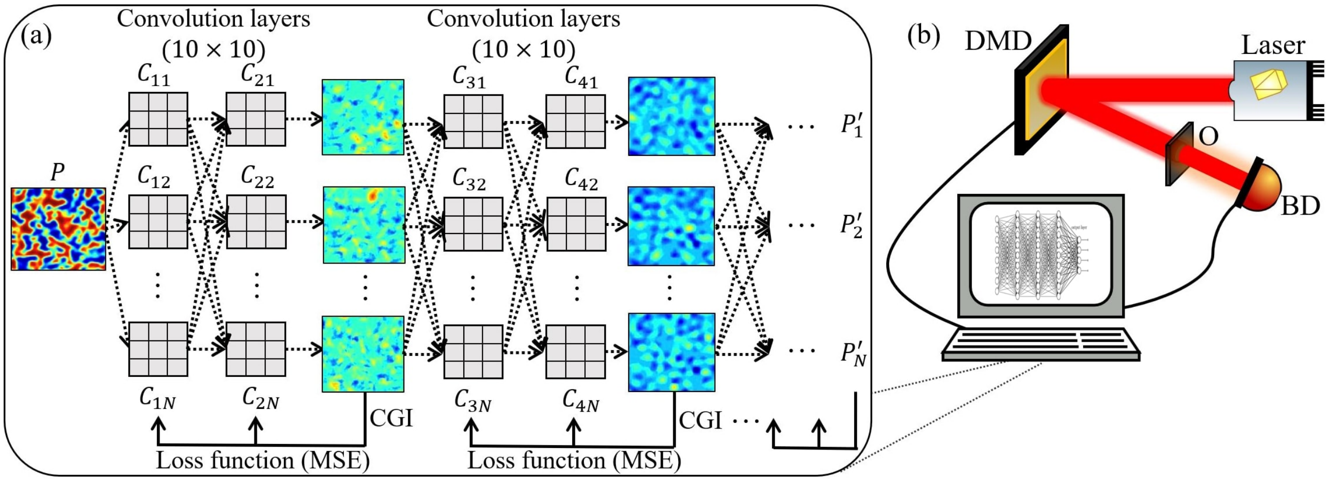

Fig. 2. (a) Diagram of Speckle-Net. A multi-branch structure with two convolution layers within each branch is used in the model, in which 10×10-sized kernels are adopted. The subscripts j and i in C_{ji} denote the *j-th* layer and *i-th* kernel in each layer corresponding to the deep-learned speckle patterns. A loss function feedback is applied at the end of each branch to modify the parameters in kernels. The deep-learned speckle patterns generate the CGI results in training at each training epoch. (b) Schematic of the experimental setup. The deep-learned speckle patterns P_i' are applied to the DMD for the CGI measurement. The laser illuminated patterns are projected onto the object (O). Light passing through object is collected by a bucket detector (BD).

with Branch Convolutional Neural Network [34], and the outputs of all layers are padded again by boundary reflections to maintain the size of their origin. The ReLU could improve the sensitivity to the activation sum input, and BNL is implied to reduce internal covariate shift.

Speckle-Net has higher effectiveness and efficiency than conventional CNN, has no overfitting concerns, and is adaptable to other systems. Firstly, the multiple backward methods significantly improve the performance of the network. It is difficult to analyze and enhance the original pattern and aimed imaging systems from a single or a few intermediate layers. At the same time, too many layers have poor directional of amelioration [35], therefore losing the characteristics of the original pattern and sought imaging systems. Nevertheless, our multi-branches neural network boosts the feedback gradient adjustment at each epoch from the loss function, avoiding the loss function of output patterns trapped in a local minimum. Every two layers' parameters in one branch are adjusted independently. Therefore, getting the optimum parameters in our model is more efficient and effective than single-branch CNN with multiple layers and single loss function feedback. Meanwhile, this Multi-branches learning process has great performance because various training complexities are required for different sampling rates β. For example, when small β is adopted, fewer patterns lead to fewer required parameters and less time for training. Therefore, only two rounds of training are necessary to get desired speckle patterns. Otherwise, more branches can be used for a larger sampling rate, as shown in Supplement 1, section 1. Thus, this Multi-branches Speckle-Net enables us to select the most efficient number of training branches according to looking at the loss functions of previous results. If the loss in two (or more, to ensure) neighboring training branches go closely to the same minimum, we can conclude that the speckle patterns reach to the global optimum. Secondly, we abandon the fully connected (FC) layers and dropout layers. FC layers in this structure demand large RAM[2], and is useless in that the convolution parts aim to adjust the correlation of patterns rather than get the CGI results. On the other hand, the dropout layer is functioned to avoid over-fitting in convolutional layers. However, in a deep-learned speckle pattern scheme, the optimum patterns are our ultimate goal which remains intact for various training and testing images. A constant input image means that over-fitting does not exist in our model. Therefore, the epoch number can be determined based on the convergence of the loss function in each branch, as shown in Supplement 1, section 1. Moreover, the loss function in our model can be adjusted according to the feature of the physical process, and the CGI algorithm can be substituted by other physical processes as well. In imaging and spectroscopic systems, the mean square error (MSE), contrast-to-noise ratio, correlation-coefficient, *etc.*, can be applied to the loss function independently or in combination to achieve good visibility, high contrast, and optimized similarities.

[2]For instance, if the image size of patterns is 112×112 and sampling rate $\beta = 0.5\%$, the number of patterns is 62. Then the size of parameters in the FC layer is around 9,000 TB, which is unrealistic for training.

3. IMPLEMENTATION: COMPUTATIONAL GHOST IMAGING

Ghost imaging [19, 20, 36], a single pixel imaging technique, reconstructs the object through second-order correlation between reference and object light paths. CGI [22, 37] substitutes the reference path by preparing speckles in advance. Therefore, one only needs to record the intensity of object light path and correlate them with speckles in sequence.

One of the main disadvantages of CGI is the large sampling rate, and therefore long sampling time. CGI have to project a large number of speckle patterns on objects and then collect light intensity sequentially for the ensemble correlation. When the object pixel size is large, the required number of speckle patterns is tremendous. Many ameliorated techniques have been proposed to minimize the sampling rate, such as orthonormalization method [27, 31], Fourier and sequency Walsh-Hadamard speckles [28, 29, 38], and compressive sensing [39, 40].

DL-based CGI technique has also shown sub-Nyquist imaging ability. It can retrieve images at a few percentage sampling rates, which is much lower than other techniques [41–45]. Nevertheless, almost every work uses post-processing techniques, and their adaptive objects are limited to categories from training groups. Therefore, they don't work or work much worse when objects are outside the training group. In general, these works focus on using DL to suppress the noise fluctuation via matrix restoration or array amelioration algorithm, which does not touch the core concept of ghost imaging. CGI is the linear aggregation of correlation from each pixel where light passes through. To solve the problem fundamentally and universally, we should be direct to the correlation.

we conceive that applying DL technique will optimize the cross- and auto-correlation of the speckles. Mean Squared Error (MSE) is one of the most frequently appeared evaluators in DL to evaluate picture quality. We therefore choose MSE here as the training loss functions for each branch, to compare the CGI results and their ground-truths. The MSE is defined as

$$MSE = \frac{1}{N_{\text{pixel}}} \sum_{i=1}^{N_{\text{pixel}}} [\frac{G_{\text{i}} - X_{\text{i}}}{\langle G_{(\text{o})} \rangle}]^2 \quad \textbf{(4)}$$

Here, X is the reference matrix calculated by

$$X_i = \begin{cases} \langle G_{(\text{o})} \rangle & \text{, Transmission = 1} \\ \langle G_{(\text{b})} \rangle & \text{, Transmission = 0} \end{cases} \quad \textbf{(5)}$$

G represents pixels in the correlation results, $G_{(\text{o})}$ is where the light ought to be transmitted, *i.e.*, the object area, while $G_{(\text{b})}$ is where the light ought to be blocked, *i.e.*, the background area. N_{pixel} is corresponded to the total pixel number in the speckle patterns ($N_{\text{pixel}} = 112 \times 112$ in our experiment).

This way of correlation adjustment to improve CGI is not limited by training database categories, one-time, and can let the sampling rate reach to 0.5%. To demonstrate the ability of Speckle-Net, only the MNIST dataset is adopted as training and part of testing images. A total of 60,000 handwritten digits resized to 112×112 pixels are used. The optimizer for training process is Stochastic Gradient Descent with Momentum Optimizer (SGDMO) [46]. The momentum of optimizer was set to 0.9 as suggested and weights decay factor was 10^{-3} to avoid exploding gradient. After network predicts manipulation on speckles, we utilize training images and patterns to obtain temporary CGIs. The loss function is the MSE between temporary CGIs and original training images, a general loss function for DL problem. Losses of some training images are tremendous, and we adopted the mean reduction of each batch as losses. Then the backwards adjust parameters in the network via manipulation patterns. Generally speaking, the network only relates directly to the speckle patterns instead of training images as in the traditional CNN. As mentioned before, the over-fitting effect is not obvious in our network. Therefore, the network was trained for 200 epochs before which the loss stopped declining. This program is implemented via Pytorch 1.7.1 and CUDA 11.0 on Python 3.8.5, and we imply GPU-chip NVIDIA GTX1050 for computation acceleration.

The convoluted speckle patterns can then be directly used in the CGI experiment. A typical CGI experiment setup is presented in Fig. 2(b). The convoluted speckle patterns from three-step training output are loaded onto digital micromirror device (DMD). With the illumination from laser, the speckle patterns are projected to objects, and light passing through object is collected by the bucket detector (BD). The images can then be retrieved using the standard CGI algorithm.

4. CHARACTERISTICS OF THE DEEP-LEARNED SPECKLE PATTERNS

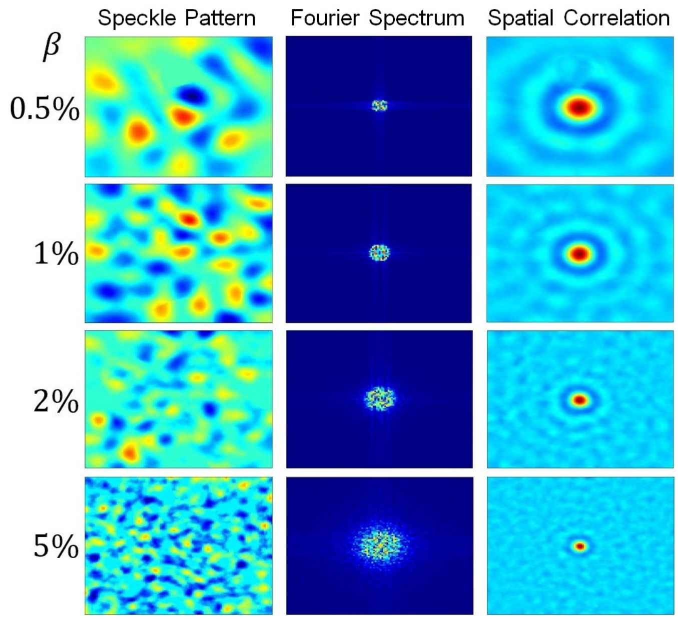

Fig. 3. Left column: Typical speckle patterns experiencing three rounds DNN training with sampling rate $\beta = 0.5\%, 1\%, 2\%$, and 5%; Middle column: The Fourier spectra of corresponding convoluted speckle patterns; Right column: The spatial intensity fluctuation correlation distributions of corresponding speckle patterns.

We choose four different sampling ratio β (0.5%, 1%, 2%, and 5%) for the Speckle-Net training. β is defined as $\beta = N_{\text{pattern}}/N_{\text{pixel}}$, where N_{pattern} is the total number of speckle patterns. When β is given, the number of kernels N_{k} in each layer is settled, $N_{\text{k}} = \beta N_{\text{pixel}} = N_{\text{pattern}}$. A group of output patterns is given after each round of training with each β. A typical pink noise speckle pattern [47] is used as the initial pattern. Since the pink noise speckle pattern favors lower spatial frequency components, therefore can in principle make the training process converge faster especially in small β cases. Three rounds are enough to generate the optimized patterns from the initial pattern via Speckle-Net for all the βs used in this work, and two rounds are sufficient for smaller βs (see supplement 1 for detail). In principle, any speckle pattern can be used as the initial input, with possibly extra training (see supplement 1, section 3 for detail).

In Fig. 3, we show the three-round convoluted patterns for various β in the first column. The Fourier spectrum distribution and spatial intensity fluctuation correlation distribution $\Gamma^{(2)}(x, y)$ of the patterns are also presented in the second column and the third column, correspondingly. From Fig. 3 we can see that the grain size of the speckle pattern gradually decreases when β increases. This is also reflected in the Fourier spectrum distribution, *i.e.*, it concentrates on low spatial frequency when β is small, and expands to higher spatial frequencies when β increases. Nevertheless, we also notice there are some high frequency components in all the β cases, which is also essential for the CGI process. Now if we check the spatial correlation of the deep-learned speckle patterns, we notice that the width of the correlation function is broad when $\beta = 0.5\%$, and approaches a delta function when $\beta = 5\%$. On the other hand, the background is smoothly distributed, irrespective of β. This is different than traditional speckle patterns when β is small. The latter case typically has a significant fluctuation and random distribution in the background due to the lack of ensemble average. Overall, for various β, the deep-learned speckle patterns always give optimized correlation function which peaked at auto-correlation with certain bandwidth and smoothly distributed cross-correlation background.

5. EXPERIMENTAL RESULTS

A. Imaging results with different sampling rates

To testify the effectiveness of the deep-learned speckle patterns in CGI system, we performed a serials of measurements using the experimental setup shown in left part of Fig. 2. The DMD is illuminated by a CW laser, and the deep-learned speckle patterns are sequentially loaded on the DMD then projected to illuminate the object. All objects are 112×112 pixels in size and placed at the imaging plane in front of the BD. Light passing through the object is collected by a BD, the recorded intensities are then used to make second-order correlations with corresponding patterns. After correlation ensemble controlled by sampling rate β, the object is reconstructed. In the experiment, we used our deep-learned speckle patterns with sampling rates of 0.5%, 1%, 2%, and 5%. We adopt four categories, in total 16 different objects (simple digits and letters, English letters, Chinese characters, and pictures) for reconstruction. In all the 16 objects, only digits '4' and '8' are from the pattern training dataset. These objects have different sizes, orientations, and complexities, in order to demonstrate the universal adaptability of the deep-learned patterns.

The main results are shown in Fig. 4. Simple objects such as the simple shape 'three lines', Greek letter 'π', digits '4' and '8', and Chinese character 'huo', can be reconstructed at the SR of only 0.5%, *i.e.*, only 62 patterns are used for the imaging process. At SR of 1%, the basic profile can be reconstructed for most of the objects already, and become much more clearer when the SR is 2%. At the SR of 5%, all objects can be clearly retrieved. We note here that, when the sampling rate is low, the deep-learned patterns possess higher cross-auto correlation ratio, as shown in Fig. 3. The images generally show higher signal to background ratio but lower resolution. When the SR is high, such as 5%, the images have much higher resolution. From Fig. 4 we can conclude that all the objects with different complexity can be reconstructed with high visibility and low noise fluctuation in the background. This boost the deep-learned speckle patterns' applicability in extremely low sampling ranges, which might be useful in moving object capture and dynamic imaging systems.

B. Imaging results under different noise conditions

Another advantage of the deep-learned speckle pattern is that, the optimized auto- and cross-correlation enables its noise-robust feature meanwhile possesses sufficient spatial resolution. To demonstrate the ability of imaging under noisy interference of the deep-learned patterns, we perform a series of measurements of four objects under different noise levels. We choose the four objects from our four catalogs: Greek letter 'π', letters 'CGI', Chinese character 'yan', and picture 'leaf'. Different noise levels are represented by different SNRs. The SNR in logarithmic decibel scale is defined as

$$\mathrm{SNR} = 10 \log \frac{P_s}{P_b}, \tag{6}$$

where P_s is the average intensity in each signal pixel and P_b is the average intensity in the noise background. Here we choose three different SNRs: 8.8dB, 6.4dB, and 3.1dB.

The results are shown in Fig. 5. It is clearly seen that at 8.8dB, all the images can be retrieved at all different SRs. When the SNR is 6.4dB, some of the images start to show noisy background. Nevertheless, all the objects can still been clearly identified. When the SNR is 3.1dB, which can be considered very noisy, most of the objects can still be identified. We also notice that, speckle patterns with lower SR are more robust to noise interference. Take the Greek letter 'π' for example, although it can be clearly imaged at 3.1dB when the SR is 5%, there exists obvious background noise in the resulted image. At 2% SR, the background noise starts to degrade. When the SR is at 1% or 0.5%, the background is almost smooth and we see nearly no difference between results at the three noise levels.

The noise-robust feature is resulted from the optimized cross-auto correlation ratio for each SR. At the extremely low SR such as 1% and 0.5%, the cross correlation is much emphasized to enhance the signal to noise ratio, and suppress the fluctuations in the correlation due to limited number of sampling. Therefore, the deep-learned speckle patterns are feasible to apply in noisy environments.

6. CONCLUSION AND DISCUSSION

In summary, we propose a speckle pattern generation scheme, Speckle-Net, by using DL algorithms and concepts to obtain the desired feature. We then chose the standard CGI algorithm as our objective for loss function, and applied this method to generate speckle patterns for CGI. We experimentally demonstrate that the deep-learned speckle pattern can be used for the standard CGI measurement, enhance the imaging efficiency, and robust to noise. The method is unique and superior to the traditional CGI and deep-learning-based CGI focusing in image amelioration or imaging algorithms. Firstly, this featured multi-branch Speckle-Net provides with flexibility in finding global optimal solution and time-consumption in training. Secondly, since the learning process only focuses the speckle patterns, it can be used for other speckle illumination systems by changing the objective in loss function. Thirdly, even though the network is trained only using the MNIST digit dataset, the resulting pattern can retrieve images for simple letters with an extremely low sampling rate (0.5%) and can imaging complicated objects with only a 5% sampling rate. Furthermore, deep-learned speckle pattern based CGI system is insensitive to noise interference.

Although a particular example, *i.e.*, the CGI is demonstrated in this work, in the long term, we believe the pioneering work boosts a closer connection between DL and speckle pattern generation, which will pave the way for broader and practical exploitation of ghost imaging and other applications. In addition, other structures such as U-net [48], recurrent neural network (RNN) [49], transformer [50, 51], *etc.*, can be similarly explored and modified to generate aimed speckle patterns. For example, the time-dependent RNN and transformer can be modified similarly as what we do on CNN to make other types of Speckle-Net which can fabricate time-dependent speckle patterns according to the instant feedback and demand of systems during the measurement. Specifically, the n-th illumination pattern can be generated from patterns and results with $n-1$ sampling number.

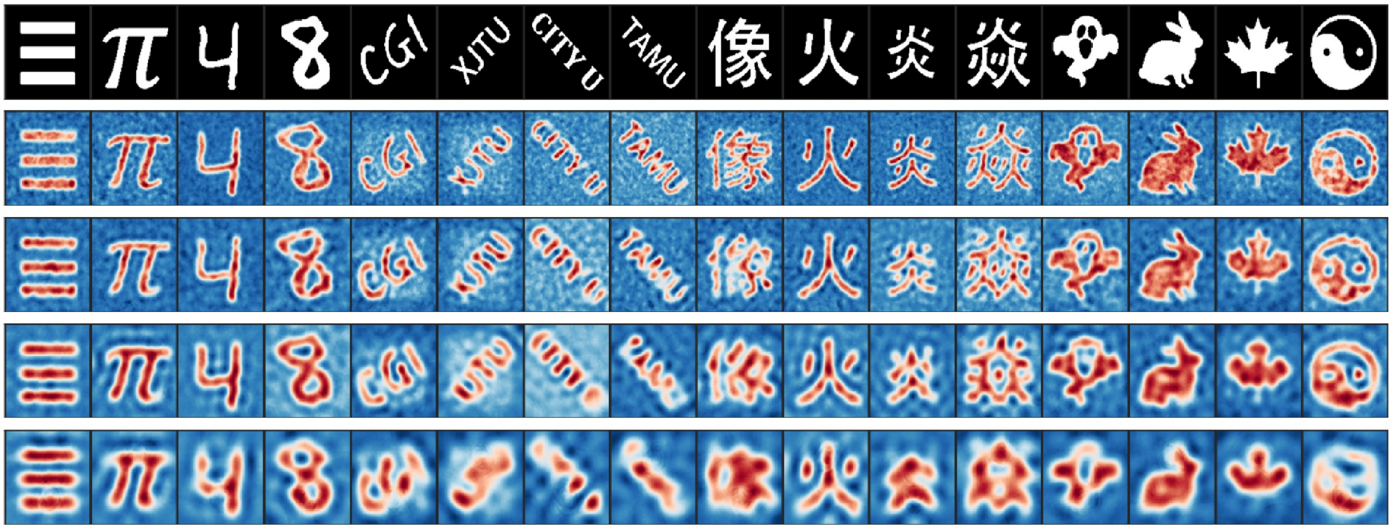

Fig. 4. Experimental results of CGI with simple symbols, words, Chinese characters, and pictures by three rounds deep-learned speckle patterns. From top to bottom: original objects, CGI results with $\beta = 5\%$, 2%, 1%, and 0.5%, respectively.

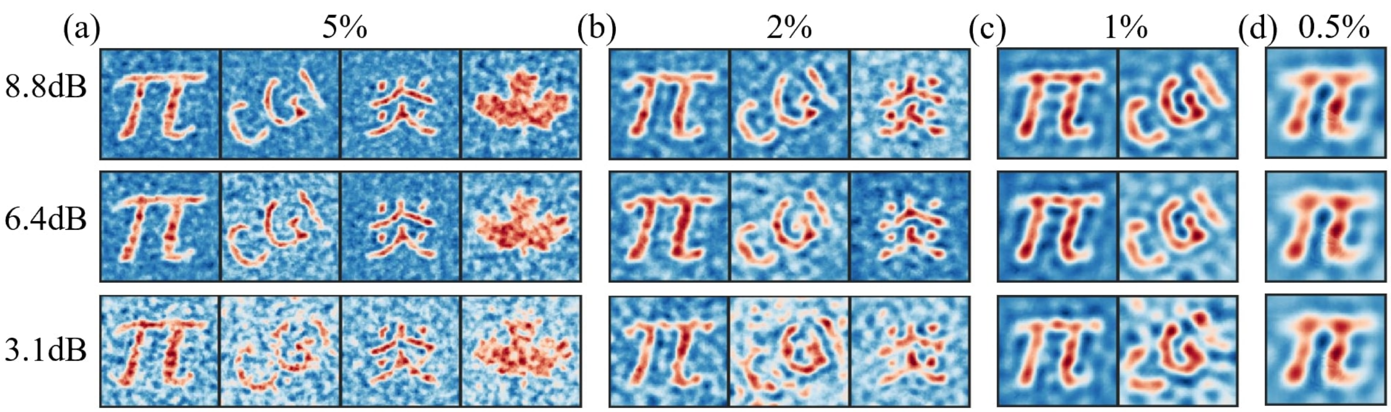

Fig. 5. Experimental results of CGI using Deep-learned speckles with different noise levels labelled in the left column. (a) CGI results with $\beta = 5\%$, (b) CGI results with $\beta = 2\%$, (c) CGI results with $\beta = 1\%$, and (d) CGI results with $\beta = 0.5\%$.

FUNDING.

Air Force Office of Scientific Research (Award No. FA9550-20-1-0366 DEF), Office of Naval Research (Award No. N00014-20-1-2184), Robert A. Welch Foundation (Grant No. A-1261), and National Science Foundation (Grant No. PHY-2013771).

DATA AVAILABILITY.

The experiment data and convoluted speckle patterns in this article are available upon reasonable request from the authors. The Speckle-Net and initial patterns can be found at https://github.com/XJTU-TAMU-CGI/PatternDL.

DISCLOSURES.

The authors declare no conflicts of interest.

SUPPLEMENTAL DOCUMENT.

See Supplement 1 for supporting content.

REFERENCES

1. D. J. Pine, D. A. Weitz, P. M. Chaikin, and E. Herbolzheimer, "Diffusing wave spectroscopy," Phys. Rev. Lett. **60**, 1134 (1988).
2. S.-W. Li, F. Li, T. Peng, and G. Agarwal, "Photon statistics of quantum light on scattering from rotating ground glass," Phys. Rev. A **101**, 063806 (2020).
3. J. W. Goodman, "Statistical properties of laser speckle patterns," in *Laser Speckle and Related Phenomena*, (Springer, 1975), pp. 9–75.
4. I. Zanette, T. Zhou, A. Burvall, U. Lundström, D. H. Larsson, M. Zdora, P. Thibault, F. Pfeiffer, and H. M. Hertz, "Speckle-based x-ray phase-contrast and dark-field imaging with a laboratory source," Phys. Rev. Lett. **112**, 253903 (2014).
5. J. Wang and A. Z. Genack, "Transport through modes in random media," Nature **471**, 345–348 (2011).
6. L. Olivieri, J. S. T. Gongora, L. Peters, V. Cecconi, A. Cutrona, J. Tunesi, R. Tucker, A. Pasquazi, and M. Peccianti, "Hyperspectral terahertz microscopy via nonlinear ghost imaging," Optica **7**, 186–191 (2020).
7. G. C. Valley, G. A. Sefler, and T. J. Shaw, "Multimode waveguide speckle patterns for compressive sensing," Opt. Lett. **41**, 2529–2532 (2016).
8. B. Redding, S. M. Popoff, and H. Cao, "All-fiber spectrometer based on speckle pattern reconstruction," Opt. Express **21**, 6584–6600 (2013).
9. T. Strudley, T. Zehender, C. Blejean, E. P. Bakkers, and O. L. Muskens, "Mesoscopic light transport by very strong collective multiple scattering

in nanowire mats,” Nat. Photonics **7**, 413–418 (2013).

10. B. Redding, S. F. Liew, R. Sarma, and H. Cao, “Compact spectrometer based on a disordered photonic chip,” Nat. Photonics **7**, 746–751 (2013).
11. C. Ventalon and J. Mertz, “Dynamic speckle illumination microscopy with translated versus randomized speckle patterns,” Opt. Express **14**, 7198–7209 (2006).
12. J. Mertz, “Optical sectioning microscopy with planar or structured illumination,” Nat. methods **8**, 811 (2011).
13. S. Nakadate and H. Saito, “Fringe scanning speckle-pattern interferometry,” Appl. Opt. **24**, 2172–2180 (1985).
14. H. Yilmaz, E. G. van Putten, J. Bertolotti, A. Lagendijk, W. L. Vos, and A. P. Mosk, “Speckle correlation resolution enhancement of wide-field fluorescence imaging,” Optica **2**, 424–429 (2015).
15. M. Pascucci, G. Tessier, V. Emiliani, and M. Guillon, “Superresolution imaging of optical vortices in a speckle pattern,” Phys. Rev. Lett. **116**, 093904 (2016).
16. W. McGehee, S. Kondov, W. Xu, J. Zirbel, and B. DeMarco, “Three-dimensional anderson localization in variable scale disorder,” Phys. review letters **111**, 145303 (2013).
17. D. Delande and G. Orso, “Mobility edge for cold atoms in laser speckle potentials,” Phys. review letters **113**, 060601 (2014).
18. E. Fratini and S. Pilati, “Anderson localization of matter waves in quantum-chaos theory,” Phys. Rev. A **91**, 061601 (2015).
19. R. S. Bennink, S. J. Bentley, and R. W. Boyd, ““two-photon” coincidence imaging with a classical source,” Phys. Rev. Lett. **89**, 113601 (2002).
20. X.-H. Chen, Q. Liu, K.-H. Luo, and L.-A. Wu, “Lensless ghost imaging with true thermal light,” Opt. Lett. **34**, 695–697 (2009).
21. A. Valencia, G. Scarcelli, M. D’Angelo, and Y. Shih, “Two-photon imaging with thermal light,” Phys. Rev. Lett. **94**, 063601 (2005).
22. J. H. Shapiro, “Computational ghost imaging,” Phys. Rev. A **78**, 061802 (2008).
23. Y. Bromberg and H. Cao, “Generating non-rayleigh speckles with tailored intensity statistics,” Phys. Rev. Lett. **112**, 213904 (2014).
24. H. E. Kondakci, A. Szameit, A. F. Abouraddy, D. N. Christodoulides, and B. E. Saleh, “Sub-thermal to super-thermal light statistics from a disordered lattice via deterministic control of excitation symmetry,” Optica **3**, 477–482 (2016).
25. N. Bender, H. Yılmaz, Y. Bromberg, and H. Cao, “Customizing speckle intensity statistics,” Optica **5**, 595–600 (2018).
26. Z. Li, X. Nie, F. Yang, X. Liu, D. Liu, X. Dong, X. Zhao, T. Peng, M. S. Zubairy, and M. O. Scully, “Sub-rayleigh second-order correlation imaging using spatially distributive colored noise speckle patterns,” Opt. Express **29**, 19621–19630 (2021).
27. B. Luo, P. Yin, L. Yin, G. Wu, and H. Guo, “Orthonormalization method in ghost imaging,” Opt. Express **26**, 23093–23106 (2018).
28. L. Wang and S. Zhao, “Fast reconstructed and high-quality ghost imaging with fast walsh–hadamard transform,” Photonics Res. **4**, 240–244 (2016).
29. Z. Zhang, X. Wang, G. Zheng, and J. Zhong, “Hadamard single-pixel imaging versus fourier single-pixel imaging,” Opt. Express **25**, 19619–19639 (2017).
30. W.-K. Yu, “Super sub-nyquist single-pixel imaging by means of cake-cutting hadamard basis sort,” Sensors **19**, 4122 (2019).
31. X. Nie, X. Zhao, T. Peng, and M. O. Scully, “Sub-nyquist computational ghost imaging with orthonormalized colored noise pattern,” arXiv preprint arXiv:2012.07250 (2020).
32. V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Icml,* (2010).
33. S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *International conference on machine learning,* (PMLR, 2015), pp. 448–456.
34. X. Zhu and M. Bain, “B-cnn: branch convolutional neural network for hierarchical classification,” arXiv preprint arXiv:1709.09890 (2017).
35. K. He, X. Zhang, S. Ren, and S. Jian, “Identity mappings in deep residual networks,” in *European Conference on Computer Vision,* (2016).
36. T. B. Pittman, Y. Shih, D. Strekalov, and A. V. Sergienko, “Optical imaging by means of two-photon quantum entanglement,” Phys. Rev. A **52**, R3429 (1995).
37. Y. Bromberg, O. Katz, and Y. Silberberg, “Ghost imaging with a single detector,” Phys. Rev. A **79**, 053840 (2009).
38. Z. Zhang, X. Ma, and J. Zhong, “Single-pixel imaging by means of fourier spectrum acquisition,” Nat. communications **6**, 1–6 (2015).
39. O. Katz, Y. Bromberg, and Y. Silberberg, “Compressive ghost imaging,” Appl. Phys. Lett. **95**, 131110 (2009).
40. V. Katkovnik and J. Astola, “Compressive sensing computational ghost imaging,” JOSA A **29**, 1556–1567 (2012).
41. M. Lyu, W. Wang, H. Wang, H. Wang, G. Li, N. Chen, and G. Situ, “Deep-learning-based ghost imaging,” Sci. Reports **7**, 1–6 (2017).
42. 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**, 147–151 (2018).
43. G. Barbastathis, A. Ozcan, and G. Situ, “On the use of deep learning for computational imaging,” Optica **6**, 921–943 (2019).
44. F. Wang, H. Wang, H. Wang, G. Li, and G. Situ, “Learning from simulation: An end-to-end deep-learning approach for computational ghost imaging,” Opt. Express **27**, 25560–25572 (2019).
45. H. Wu, R. Wang, G. Zhao, H. Xiao, D. Wang, J. Liang, X. Tian, L. Cheng, and X. Zhang, “Sub-nyquist computational ghost imaging with deep learning,” Opt. Express **28**, 3846–3853 (2020).
46. S. Ruder, “An overview of gradient descent optimization algorithms,” arXiv preprint arXiv:1609.04747 (2016).
47. X. Nie, F. Yang, X. Liu, X. Zhao, R. Nessler, T. Peng, M. S. Zubairy, and M. O. Scully, “Noise-robust computational ghost imaging with pink noise speckle patterns,” Phys. Rev. A **104**, 013513 (2021).
48. O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention,* (Springer, 2015), pp. 234–241.
49. J. T. Connor, R. D. Martin, and L. E. Atlas, “Recurrent neural networks and robust time series prediction,” IEEE transactions on neural networks **5**, 240–254 (1994).
50. M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, “Spatial transformer networks,” Adv. neural information processing systems **28**, 2017–2025 (2015).
51. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” Adv. neural information processing systems pp. 5998–6008 (2017).