Check for updates

## OPTICA

## Deep-learning-based fringe-pattern analysis with uncertainty estimation

## Shijie Feng,<sup>1,2,3</sup> <sup>(i)</sup> Chao Zuo,<sup>1,2,\*</sup> <sup>(i)</sup> Yan Hu,<sup>1,2</sup> <sup>(i)</sup> Yixuan Li,<sup>1,2</sup> <sup>(i)</sup> and Qian Chen<sup>2,4</sup> <sup>(i)</sup>

<sup>1</sup>Smart Computational Imaging (SCI) Laboratory, Nanjing University of Science and Technology, Nanjing, Jiangsu Province 210094, China <sup>2</sup>Jiangsu Key Laboratory of Spectral Imaging & Intelligent Sense, Nanjing University of Science and Technology, Nanjing, Jiangsu Province 210094, China

<sup>3</sup>e-mail: shijiefeng@njust.edu.cn

<sup>4</sup>e-mail: chengian@njust.edu.cn

\*Corresponding author: zuochao@njust.edu.cn

Received 16 June 2021; revised 1 November 2021; accepted 1 November 2021; published 23 November 2021

Deep learning has gained increasing attention in the field of optical metrology and demonstrated great potential in solving a variety of optical metrology tasks, such as fringe analysis and phase unwrapping. However, deep neural networks cannot always produce a provably correct solution, and the prediction error cannot be easily detected and evaluated unless the ground-truth is available. This issue is critical for optical metrology, as the reliability and repeatability of the measurement are of major importance for high-stakes scenarios. In this paper, for the first time to our knowledge, we demonstrate that a Bayesian convolutional neural network (BNN) can be trained to not only retrieve the phase from a single fringe pattern but also produce uncertainty maps depicting the pixel-wise confidence measure of the estimated phase. Experimental results show that the proposed BNN can quantify the reliability of phase predictions under conditions of various training dataset sizes and never-before-experienced inputs. Our work allows for making better decisions in deep learning solutions, paving a new way to reliable and practical learning-based optical metrology. © 2021 Optical Society of America under the terms of the OSA Open Access Publishing Agreement

https://doi.org/10.1364/OPTICA.434311

Fringe-pattern analysis is key to many optical metrology applications [1], such as optical interferometry, fringe projection profilometry, digital holography, moiré interferometry, shearography, and corneal topography. The purpose of the fringe-pattern analysis is to extract the underlying phase information of test objects from one or several fringe pattern(s). Normally, a fringe pattern I can be expressed as

$$I(x, y) = A(x, y) + B(x, y) \cos \varphi(x, y), \tag{1}$$

where (x, y) is the pixel coordinate, A is the background signal, B is the modulation, and  $\varphi$  is the phase of test objects. As A and B are unknown, it is an ill-posed problem to extract  $\varphi$  if only one fringe image is at hand. Single-shot phase demodulation approaches, e.g., Fourier transform profilometry (FTP) [2], resort to the assistance of a spatial carrier to handle the ill-posed issue. Although they are of high efficiency, they are susceptible to complex surfaces that

can easily cause spectral aliasing during the phase demodulation. On the contrary, multi-shot phase demodulation approaches, such as phase-shifting (PS) algorithms [3], can carry out pixel-wise phase measurements with high accuracy. However, they are fragile for disturbances and vibrations due to the limited efficiency resulting from the multi-frame nature.

Recently, the deep learning technique has been introduced to the fringe-pattern analysis [4]. It is reported that the phase information can be extracted from a single fringe pattern with substantially enhanced phase accuracy for complex objects by a trained deep neural network (DNN). Therefore, the learning-based fringe analysis has great potential in realizing high-efficiency and highaccuracy phase demodulation. However, as most DNNs are driven by data completely, the reasoning process is quite different from that of a traditional physical model. Actually, when the training data are insufficient or the testing data are rare, the output of DNN may not be reliable enough. A recent example in computer vision has shown a disastrous prediction where an image classification network mistakenly identified two African Americans as gorillas, giving rise to concerns of racial discrimination [5]. Therefore, *how to trust the prediction of a DNN is still a big challenge*.

For the task of single-shot fringe-pattern analysis, the uncertainty estimation of the predicted phase is indispensable as it is an ill-posed problem to retrieve the phase from Eq. (1) with a single image. Inspired by recent successful applications of Bayesian deep learning approaches [6], we demonstrate for the first time, to the best of our knowledge, that a Bayesian convolutional neural network (BNN) can be trained to not only demodulate the phase from a single fringe pattern, but also evaluate two uncertainties of the prediction. They are the data uncertainty and the model uncertainty. The data uncertainty is also referred to as the aleatoric uncertainty that can quantify the randomness of the prediction due to the noise and data imperfection. The model uncertainty can be referred to as the epistemic uncertainty, which captures the robustness and the uncertainty of the model. The proposed BNN is easy to construct and can be extended to traditional DNNs readily. Experimental results on fringe projection profilometry show that the uncertainty maps predicted by BNN can indicate the actual error distribution faithfully in the absence of standard reference data.

According to Eq. (1), the phase can be retrieved by

$$\varphi(x, y) = \arctan \frac{M(x, y)}{D(x, y)} = \arctan \frac{c B(x, y) \sin \varphi(x, y)}{c B(x, y) \cos \varphi(x, y)},$$
(2)

where the numerator M(x, y) characterizes the phase sine  $[\sin \varphi(x, y)]$  and the denominator D(x, y) characterizes the phase cosine  $[\cos \varphi(x, y)]$ . *c* is a constant parameter that depends on the used phase demodulation approach [2,3]. To emulate this process, a DNN can be trained to learn M(x, y) and D(x, y), which are then fed into the arctangent function for retrieving the phase.

Here, we present a BNN that uses the Concrete dropout [7] to approximate Bayesian inference in deep Gaussian processes for learning the numerator M(x, y) and the denominator D(x, y) statistically. We assume that **X** is a set of input fringe images, which can be represented as  $\mathbf{X} = \{\mathbf{x}^k\}_{k=1}^K$ , where  $\mathbf{x}^k$  is the *k*th input fringe pattern and *K* is the size of the set. **Y** is a set of ground-truth labels corresponding to the training data, which can be written as  $\mathbf{Y} = \{\mathbf{y}^k\}_{k=1}^K$ , where  $\mathbf{y}^k$  consists of the ground-truth numerator and denominator  $(M^k, D^k)$ . **w** represents the weight matrix of the BNN. To investigate the distribution of the output of BNN, we model the predictive distribution  $p(\mathbf{y}^* | \mathbf{x}^*, \mathbf{X}, \mathbf{Y})$  as

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w}) p(\mathbf{w}|\mathbf{X}, \mathbf{Y}) d\mathbf{w}, \qquad (3)$$

where  $p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w})$  is the probability of the output  $\mathbf{y}^*$  given the input  $\mathbf{x}^*$ , the weights  $\mathbf{w}$ , and  $p(\mathbf{w}|\mathbf{X}, \mathbf{Y})$  the probability of the weights  $\mathbf{w}$  given the training data  $(\mathbf{X}, \mathbf{Y})$ . The distribution  $p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w})$  describes the data uncertainty, and the distribution  $p(\mathbf{w}|\mathbf{X}, \mathbf{Y})$  characterizes the model uncertainty.

To measure the data uncertainty, we assume that  $\mathbf{y}^k$  has N pixels, and  $p(\mathbf{y}^k | \mathbf{x}^k, \mathbf{w})$  can then be written as

$$p(\mathbf{y}^k|\mathbf{x}^k,\mathbf{w}) = \prod_{i=1}^N p(\mathbf{y}_i^k|\mathbf{x}^k,\mathbf{w}).$$
(4)

Assuming that the distribution of  $y^k$  is Gaussian for each pixel, the data uncertainty can be captured by minimizing the negative log-likelihood function at the training stage,

$$-\frac{1}{K}\sum_{k}\log p\left(\mathbf{y}^{k}|\mathbf{x}^{k},\mathbf{w}\right)$$
$$=\frac{1}{K}\sum_{k=1}^{K}\left[\frac{1}{2(\sigma^{k})^{2}}\left\|\mathbf{y}^{k}-\hat{\mathbf{y}}^{k}\right\|^{2}+\frac{1}{2}\log\left(\sigma^{k}\right)^{2}\right],\quad(5)$$

where **y** is the ground-truth label,  $\hat{\mathbf{y}}$  is the result predicted by BNN, and  $\sigma^2$  is the predicted variance.

To measure the model uncertainty, the Concrete dropout network is applied. By placing the Concrete dropout before every weight layer, we can use a simple variational distribution  $q(\mathbf{w})$  to approximate  $p(\mathbf{w}|\mathbf{X}, \mathbf{Y})$ , which is usually hard to calculate analytically. By using the Monte Carlo (MC) integration over T samples satisfying  $\mathbf{w}^{(t)} \sim q(\mathbf{w})$ , Eq. (3) can be approximated as

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) \approx \int p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w}) q(\mathbf{w}) d\mathbf{w} \approx \frac{1}{T} \sum_{t=1}^T p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w}^{(t)})$$
(6)

At the prediction stage, the dropout layers in our BNN randomly set input neurons to zero with a learned dropout rate. By collecting the results of stochastric forward propagation through the trained model, the predictive mean can be computed and be used as the prediction of the BNN,

$$\hat{\mu} = E(\mathbf{y}^* | \mathbf{x}^*, \mathbf{X}, \mathbf{Y}) \approx \frac{1}{T} \sum_{t=1}^T E(\mathbf{y}^* | \mathbf{x}^*, \mathbf{w}^{(t)}) = \frac{1}{T} \sum_{t=1}^T \mathbf{y}^{*(t)},$$
(7)

where E is the expectation. The model uncertainty is measured by the variance of the predicted results:

$$\hat{\sigma}^{\text{model}} = \sqrt{E\left[\operatorname{Var}\left(\mathbf{y}^{*} \mid \mathbf{w}, \mathbf{x}^{*}, \mathbf{X}, \mathbf{Y}\right)\right]} \approx \sqrt{\frac{1}{T} \sum_{t=1}^{T} \left(\mathbf{y}^{*(t)} - \hat{\mu}\right)^{2}}.$$
(8)

Then, the data uncertainty is quantified by the average of the estimated variance:

$$\hat{\sigma}^{\text{data}} = \sqrt{Var\left(E\left[\mathbf{y}^{*} \mid \mathbf{w}, \mathbf{x}^{*}, \mathbf{X}, \mathbf{Y}\right]\right)} \approx \sqrt{\frac{1}{T} \sum_{t=1}^{T} \left(\sigma^{2}\right)^{(t)}}.$$
 (9)

Our BNN follows the architecture of the U-Net. In the training stage, the dropout rate of each layer is not fixed and can be learned automatically by BNN. More details about the theory, the structure, and the learned dropout rates of BNN are provided in Supplement 1.

The diagram of the testing process of our method is shown in Fig. 1. With an input fringe pattern, the trained BNN outputs T different sets of data including the numerator, the denominator, and their variance maps. The mean numerator and the mean denominator are obtained for calculating the final wrapped phase  $\hat{\mu}_{\varphi}$  by Eq. (2). To obtain the data/model uncertainty of the phase, we calculate the data/model uncertainty of the numerator and the denominator using Eqs. (9) and (8) first, and then apply the propagation of uncertainty:

$$\hat{\sigma}_{\varphi}^{\text{model/data}} = \sqrt{\left(\frac{\partial\varphi}{\partial M}\hat{\sigma}_{M}^{\text{model/data}}\right)^{2} + \left(\frac{\partial\varphi}{\partial D}\hat{\sigma}_{D}^{\text{model/data}}\right)^{2}}.$$
 (10)

More details on the calculation of the phase and its uncertainties are provided in Supplement 1.

We tested the proposed method under the scenario of fringe projection profilometry. Our system consisted of a projector (DLP 4100, Texas Instruments) and a camera (V611, Vision Research Phantom). The projector illuminated test objects with pre-designed fringe patterns and the camera captured 8-bit grayscale images simultaneously from a different perspective. The spatial frequency of the projected fringes was f = 160. To collect training data, we captured many fringe images of different kinds of objects and generated the ground-truth labels by a 12-step PS algorithm. The BNN was implemented by using the Keras and computing on a graphic card (GTX Titan, NVIDIA). Further details about the optical setup, implementation of BNN, and tests with fringe patterns of different spatial frequencies are provided in Supplement 1.

The test scene shown in Fig. 2(a) contains two plaster statues that are not present in the training stage. The trained BNN used the fringe image as an input and made T = 50 predictions. The mean of the numerator and the denominator, and the wrapped phase, are shown in Figs. 2(b)–2(d), respectively. The corresponding uncertainties are demonstrated in Figs. 2(e)–2(h), respectively.



**Fig. 1.** Schematic of the proposed method. With the Monte Carlo dropout sampling, *T* samples of the BNN's prediction are obtained for an input fringe pattern. Each prediction outputs a set of data including  $M^{*(r)}$ ,  $(\sigma_M^2)^{(r)}$ ,  $D^{*(t)}$ , and  $(\sigma_D^2)^{(r)}$ . The wrapped phase  $\hat{\mu}_{\varphi}$  is obtained by feeding the mean  $\hat{\mu}_M$  and the mean  $\hat{\mu}_D$  into the arctangent function. To obtain the phase uncertainties, we first calculate the uncertainties of the numerator and the denominator and then apply the propagation of uncertainty (PU).



**Fig. 2.** Test of the trained BNN. (a) The input fringe pattern. (b)–(d) Mean numerator, mean denominator, and wrapped phase, respectively. (e) and (f) Data uncertainty and the model uncertainty of the estimated numerator, respectively. (g) and (h) Corresponding uncertainties of the denominator.

Our BNN is well-calibrated, and the evaluation of the predicted uncertainties is provided in Supplement 1. To investigate the phase accuracy, we unwrapped the phase by using the temporal phase unwrapping approach [8] and calculated the phase error against a ground-truth phase map, which was obtained by the 12-PS algorithm. In Supplement 1, the unwrapped phase has been converted into the 3D reconstruction for better investigation of recovered surface details.

To demonstrate the efficacy of the uncertainties, we also trained the BNN with only half of the training data. For comparison, a convolutional U-Net (termed as "CNN") that had no dropout layers was trained as well. Figures 3(a) and 3(b) show the absolute phase error when both models were trained with all of the data. The two networks demonstrated similar performance on the phase measurement as the BNN followed the main structure of the U-Net. Two regions of interest (ROIs) were selected, and their error distributions are shown in Figs. 3(i) and 3(j). For both the CNN and BNN, the phase errors are small for smooth areas, such as the statues' faces. But, the error begins to increase rapidly for the sharp regions, e.g., the hairs of the statues. From Figs. 3(c) and 3(d), we can see that the distribution of uncertainties faithfully indicate the error distribution, where the areas with large errors have been labeled with large uncertainties. We find the model uncertainty is



**Fig. 3.** Analysis of the phase error and uncertainties of the BNN in two differenct cases. For the first case where the full training dataset was used for training: (a) bsolute phase error of CNN; (b) absolute phase error of BNN; (c) and (d) BNN's data uncertainty and model uncertainty of the phase. (e)–(h) Corresponding results for the second case where only half of the dataset was used for training. (i) and (j) show the errors and uncertainties of the two ROIs in the first case and the second case, respectively.

small, implying that the phase prediction can be performed consistently by the BNN. The data uncertainty is more significant, which is the result of the image noise in the captured images. In fringe projection, dense fringe patterns (e.g., f = 160) are usually captured with compromised fringe contrast. Next, the errors of both CNN and BNN increase when only half of the data were used, as can be seen in Figs. 3(e) and 3(f). We can see the data uncertainty almost does not change as the data reduction did not affect the data noise. However, the model uncertainty rises significantly. Its mean value surges from 0.029 rad to 0.062 rad, as can be seen from Figs. 3(d) and 3(h). The reduction of training data has an adverse effect on the robustness of the model, thus increasing its doubt about the prediction.

Further, we tested the BNN by using a tough sample that is a complex industrial part with screw thread shown in Fig. 4(a). The absolute phase error of the BNN is shown in Fig. 4(b). It can be

seen that the error of the smooth cylindrical area is small but that of the screw thread region is quite large. The data uncertainty and the model uncertainty are demonstrated in Figs. 4(c) and 4(d). We can see the BNN has faithfully indicated the overall error distribution. For detailed investigations, we have a magnified view of the screw thread region as shown in Fig. 4(e), where A represents the internal area and B represents the screw thread. A background image without fringes was also captured, and the selected area is shown in Fig. 4(f), which demonstrates that the internal area A is smooth without any screw structure. As the smooth surface is common and has been seen by the BNN during training, the uncertainty maps indicated high credibility, and the error is small, as can be seen in Fig. 4(g). For region B, however, the error shown in Fig. 4(h) is very serious. By comparing Figs. 4(e) and 4(f), we can see the projected fringe patterns happened to couple with the structure of the screw thread at region B, forming an approximate low-frequency moiré pattern. As a result, it is difficult for the neural network to handle this rare case, thus resulting in the significant model uncertainty. We also find that the moiré pattern has also been captured by the data uncertainty, which implies that it may also be treated as a kind of image noise by BNN. Moreover, an out-of-distribution (OOD) fringe image that has a different spatial frequency (f = 80) was also tested. The corresponding results are shown in Figs. 4(i)-4(p), where the phase error and the predictive uncertainties are more severe for the whole scene. For region A, the mean data uncertainty and model uncertainty rise to 0.14 rad and 0.12 rad from 0.074 rad and 0.025 rad, respectively. For region B, they increase to 0.55 rad and 0.48 rad from 0.45 rad and 0.31 rad, respectively. We can see that the model is very suspicious of its prediction for the OOD data. Further, if considered in a quality control setting, this experiment would provide a typical example of how the BNN allows for making better decisions. When using deep learning methods for detecting surface defects, one may face the risk of incorrectly classifying an industrial part as a defective product due to a failure of the DNN. By converting the phase results into 3D reconstructions [Figs. 4(q)-4(s)], we can see that the 12-step PS method successfully measured the profile of the complex threaded region B, while the network produced inconsistent and distorted reconstructions. In this case, the "defect" is caused by the network rather than the object itself. It is worth noting that the estimated uncertainty maps have captured this problem by showing high uncertainties for this region. Consequently, instead of blindly believing that the product is defective, we should resort to alternative (preferably more reliable) methods to further check this dubious result. More experimental results of the BNN's performance in handling never-experienced input data are provided in Supplement 1.

In this work, we have presented a fringe-pattern analysis framework using a BNN that can not only demodulate the phase information from a single fringe image but also output pixelwise uncertainty maps describing the confidence of the neural network on its prediction. The BNN is developed by using the MC Concrete dropout approximation. This strategy is easy to implement and can be extended to other existing neural networks by simply adding extra Concrete dropout layers. To validate the proposed method, we tested the performance of the BNN in the conditions of varying training dataset size, rare test inputs, and OOD data, respectively. Experimental results have shown that the predicted uncertainty maps can successfully indicate the distribution of real phase errors without using any ground-truth data. In the future, error-reduction methods based on the estimated



**Fig. 4.** Uncertainty analysis of a measured complex industrial part with screw thread. (a) A captured fringe image (f = 160). (b) Absolute phase error of BNN. (c) Data uncertainty of BNN. (d) Model uncertainty of BNN. (e) Magnified view of the fringe image for the selected area, where A indicates the internal area and B the area with screw thread. (f) Magnified view of a background image (without projected fringes) for the same selected area. (g) and (h) phase errors and uncertainty maps of region A and region B, respectively. (i)–(p) Corresponding results when the frequency of the projected fringes is f = 80. (q)–(s) 3D reconstructions obtained by the 12-PS method (ground-truth method, GT) and the BNNs.

uncertainty maps will be further investigated. We believe that a DNN that can provide confidence measure of the estimated phase is crucial to fringe-pattern analysis and that it has great potential for inspiring novel and reliable learning-based optical metrology approaches.

**Funding.** National Natural Science Foundation of China (62075096); Leading Technology of Jiangsu Basic Research Plan (BK20192003); Jiangsu Provincial "One belt and one road" innovation cooperation project (BZ2020007); Fundamental Research Funds for the Central Universities (30921011208).

**Disclosures.** The authors declare no conflicts of interest.

**Data availability.** Data underlying the results presented in this Letter may be obtained from the authors upon reasonable request.

**Supplemental document.** See Supplement 1 for supporting content.

## REFERENCES

- M. Servin, J. A. Quiroga, and M. Padilla, Fringe Pattern Analysis for Optical Metrology: Theory, Algorithms, and Applications (Wiley, 2014).
- 2. M. Takeda and K. Mutoh, Appl. Opt. 22, 3977 (1983).
- C. Zuo, S. Feng, L. Huang, T. Tao, W. Yin, and Q. Chen, Opt. Laser Eng. 109, 23 (2018).
- S. Feng, Q. Chen, G. Gu, T. Tao, L. Zhang, Y. Hu, W. Yin, and C. Zuo, Adv. Photon. 1, 025001 (2019).
- A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" arXiv:1703.04977 (2017).
- 6. Y. Xue, S. Cheng, Y. Li, and L. Tian, Optica 6, 618 (2019).
- Y. Gal, J. Hron, and A. Kendall, "Concrete dropout," arXiv:1705.07832 (2017).
- C. Zuo, L. Huang, M. Zhang, Q. Chen, and A. Asundi, Opt. Laser Eng. 85, 84 (2016).