Fringe pattern analysis using deep learning: Supplementary Information

Shijie Feng[a,b,c,d], Qian Chen[a,b,*], Guohua Gu[a,b], Tianyang Tao[a,b], Liang Zhang[a,b,c], Yan Hu[a,b,c], Wei Yin[a,b,c], and Chao Zuo[a,b,c,*]

[a]School of Electronic and Optical Engineering, Nanjing University of Science and Technology, No. 200 Xiaolingwei Street, Nanjing, Jiangsu Province 210094, China
[b]Jiangsu Key Laboratory of Spectral Imaging & Intelligent Sense, Nanjing, Jiangsu Province 210094, China
[c]Smart Computational Imaging Laboratory (SCILab), Nanjing University of Science and Technology, Nanjing, Jiangsu Province 210094, China
[d]shijiefeng@njust.edu.cn
[*]Corresponding author: zuochao@njust.edu.cn (Chao Zuo) and chenqian@njust.edu.cn (Qian Chen)

ABSTRACT

This document provides the information about the optical set-up, the collection of training data, details of neural networks training, analysis of fringe patterns with relatively low frequency, and other supplementary contents to the primary manuscript ”Fringe pattern analysis using deep learning”.

1 Optical set-up

Our fringe projection system includes a projector (DLP 4100, Texas Instruments) with resolution of 1024 × 768 and a CMOS camera (V611, Vision Research Phantom) with resolution of 1280 × 800 and with pixel depth of 8 bits. The camera is equipped with a lens of 24 mm focal length. The distance between the measured object and our system is about 1.5 meters. The arrangement is shown in Fig. S1. To achieve a better performance, 12-step phase-shifting algorithm is used to calculate the ground truth data for our neural networks. The 12-step phase-shifting fringe patterns are generated as

$$I_n^p(x^p,y^p) = a + b\cos\left(2\pi f x^p - \frac{2\pi n}{12}\right) \tag{1}$$

where (x^p,y^p) is the pixel coordinate of the projector, and index $n = 0,1,2,...,11$. Parameters a, b, f are the mean value, amplitude, and spatial frequency, respectively. In our experiments, we set $f = 160$ and $a = b = 127.5$ (for the projection of 8-bit images).

2 Collection of training data

The projector projected the generated 12-step phase-shifting fringe patterns onto different measured objects. The camera captured the reflected fringe pattern simultaneously from a different angle and transferred them to our computer. As the performance of the deep neural network largely depends on the quality of collected training data, we captured 80 different scenes including simple and complex objects. For each scene, we recorded 12 phase-shifting fringe patterns, thus entirely collecting 960 fringe images for all of the scenes. The collection of training data is demonstrated in Fig. S2. The first row and the second row show the measured scenes and captured fringe images of the scenes, respectively. Through the 12-step phase-shifting algorithm, we then calculated the corresponding ground truth data. The third row of Fig. S2 shows the background images, which were used as ground truth to train CNN1. The fourth and the fifth rows demonstrate the ground truth numerator and denominator for the training of CNN2. It is noted that before being fed into the networks, the raw images (captured fringe patterns and the predicted background images) were divided by 255 for normalization, which made the learning easier for the networks. Moreover, for a preferable selection of training objects, one is suggested choosing objects without very dark or shiny surfaces to insure captured fringe images with enough signal-to-noise ratio or without saturated points.

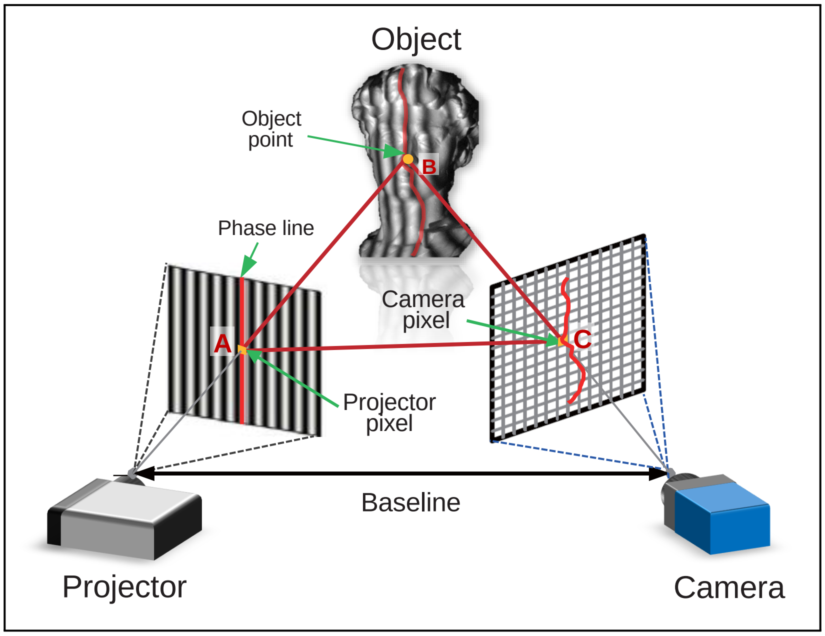

Figure S1. Arrangement of our fringe projection system.

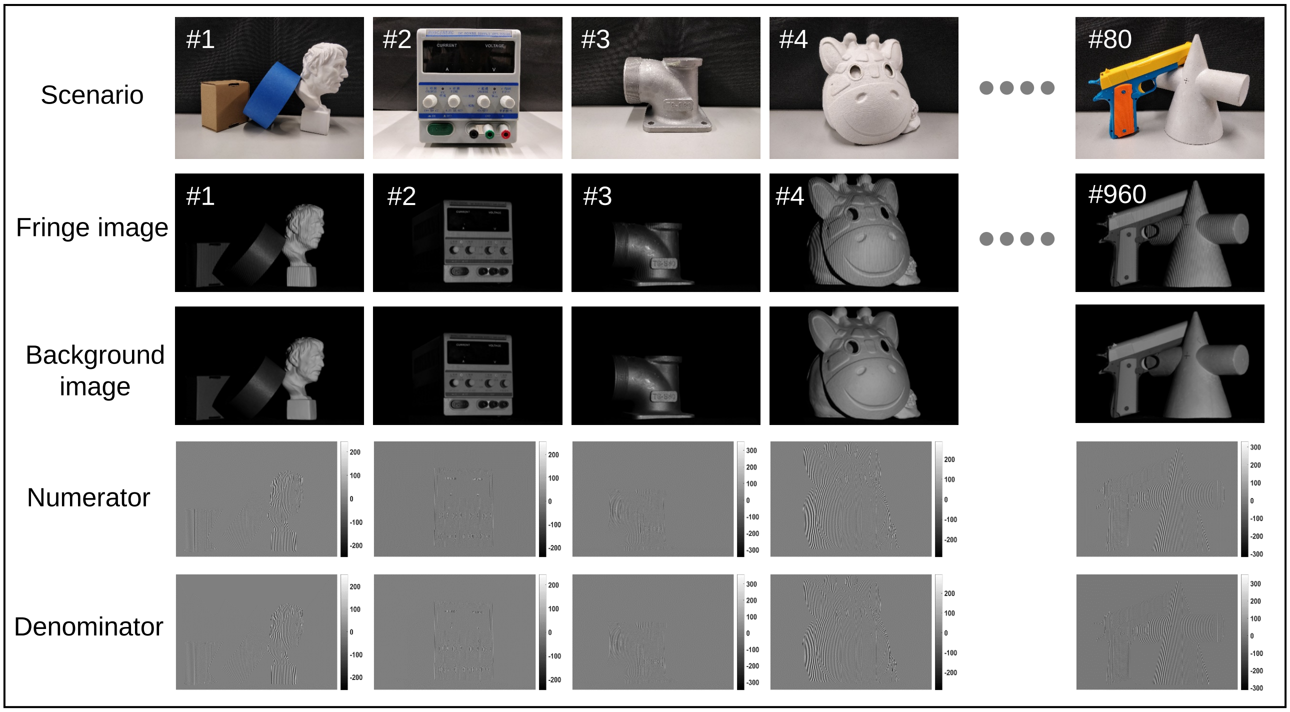

Figure S2. Collection of the training data. The first row shows different tested scenarios. For each of them, we captured 12 fringe patterns and totally obtained 960 training input images, as demonstrated in the second row. The third row shows the ground truth background image calculated by Eq. (4) of the main manuscript for the training of CNN1. The fourth and fifth row display the ground truth numerator and denominator computed through Eqs. (6) and (7) of the main manuscript for the training of CNN2.

3 Neural networks training

As can be seen in Figs. 2 and 3 of the primary manuscript, we used convolutional layers to construct our learning model. For all of the convolutional layers in CNN1 and CNN2, the kernel size is 3×3 and convolution stride is one. Zero-padding is used

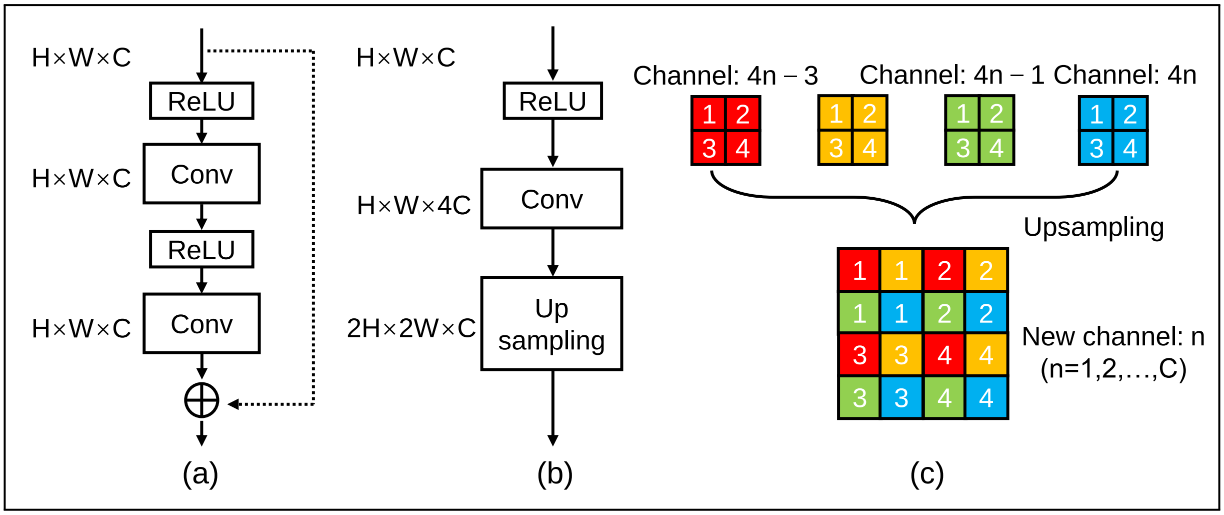

Figure S3. Architecture of the residual block and the upsampling block. (a) Residual block; (b) upsampling block; (c) diagram of the upsampling process in (b).

to control the spatial size of the output data, so that the input and output height and width are the same. The output of the convolutional layer is a 3-D tensor of shape (H,W,C), where H and W are the height and width in pixels of the input fringe pattern. C is the number of filters used in the convolutional layer and equals the number of channels of output data. The filter is used to extract a feature map (channel) for the output tensor. Therefore with more filters, the convolutional network can perceive more details of measured surfaces. But the cost is that the network will consume more time for training. Thus, we use 50 filters in the work to achieve a balance. Except for the last convolutional layer of CNN2 which is activated linearly, the rest ones use the rectified linear unit (ReLU) as activation function, i.e., $ReLU\left(x\right) = max\left(0,x\right)$. Compared with other activation functions, e.g., sigmoid function, it has been demonstrated to enable better training of deeper networks[1]. The reason to activate the last convolutional layer of CNN2 linearly is that it predicts the values of the numerator and the denominator which can be negative practically. In the networks, we also used residual blocks whose architecture is shown in Fig. S3(a). The residual framework is composed of 2 sets of convolutional layer (Conv) activated by ReLU stacked one above the other. It creates a shortcut between the input and output and can solve the degradation of accuracy as the network becomes deeper, thus easing the training process.

In CNN1, the input fringe pattern is successively processed by a convolutional layer, a group of residual blocks (which contains four residual blocks) and two convolutional layers. The last layer estimates the gray values of the background image. The parameters of the network, i.e., the weights, bias and convolutional kernels, are trained using backpropagation on mean-squared-errors between the background intensity of the network output with respect to that of the object's ground truth image, obtained by Eq. 4 of the main manuscript using the 12-step phase-shifting algorithm. The loss function is computed as

$$Loss_1(\theta_1) = \frac{1}{H \times W}\left\|Y_A^{\theta_1} - G_A\right\|^2 \tag{2}$$

where G_A is the actual background intensity, and $Y_A^{\theta_1}$ the background intensity predicted by CNN1 with the parameter space θ_1 in CNN1.

In CNN2, the input images are down-sampled by $\times 1$ and $\times 2$ in two different paths. The first data flow path is similar to the one in CNN1 in which the image size keeps unchanged. In the second path, the data is first downsamped for a high-level perception and then upsampled to match the original dimensions. The downsampling is achieved through a pooling layer. For each channel of the input, the pooling layer finds the maximum value in a 2×2 neighborhood. It then replaces the pixels in the 2×2 window with the found pixel of the maximum value. Therefore, the size of output is reduced by half for both the height and the width. To match the dimension of the original image, we upsample the data using the upsampling block as shown in Fig. S3(b). The input data first passes through a convolutional layer with ReLU activation. We then use quadruple filters to extract features from the input for providing rich information for the following upsampling, whose schematic is shown in Fig. S3(c). For the upsampled channel n, it is generated by original channels from $4n-3$ to $4n$, thus allowing the output data with $\times 2$ spatial resolution. Next, the outputs of these two data flow paths in CNN2 are concatenated into a tensor with doubled channels. Finally, the last convolutional layer yields two channels: one for the numerator $M(x,y)$ and the other for the denominator $D(x,y)$. To train CNN2, we minimize the mean-squared-errors of the output numerator and denominator with respect to the ones of the object' ground truth, obtained using the 12-step phase-shifting algorithm according to Eqs. 6 and 7 of the main manuscript. The loss function of CNN2 is computed as

$$Loss_2(\theta_2) = \frac{1}{H \times W}\left[\left\|Y_M^{\theta_2} - G_M\right\|^2 + \left\|Y_D^{\theta_2} - G_D\right\|^2\right] \tag{3}$$

where G_M and G_D are the actual numerator and denominator, and $Y_M^{\theta_2}$ and $Y_D^{\theta_2}$the numerator and denominator predicted by CNN2 with the parameter space θ_2 which includes the weights, bias and convolutional kernels in this layer.

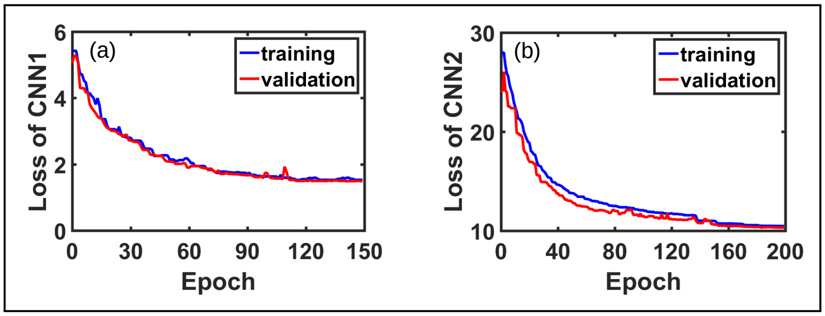

Figure S4. Loss curves of the training and validation set for the neural networks. (a) CNN1 trained over 150 epochs; (b) CNN2 trained over 200 epochs.

In the training, the networks use the score of loss function as a feedback signal to adjust the parameters in θ_1 and θ_2 by a little bit, in a direction that would lower the loss score. To this end, the adaptive moment estimation (ADAM) is used in our networks to tune the parameters to find the minimum of the loss function[2]. In the implementation of ADAM, we start the training with a learning rate of 10^{-4}. We would drop it by a factor of 2 if the validation loss has stopped improving for 10 epochs, which helps the loss function get out of local minima during training. To characterize the training, we plot the progression of the training and validation loss over training epochs, i.e., the number of iterations in the backpropagation over all of the dataset. Figure S4(a) shows the curve for CNN1 which converges after 120 epochs, and Fig. S4(b) the curve for CNN2 which shows convergence after 160 epochs. From both curves, we can see there is not overfitting to our training dataset. As to the time cost, the training of CNN1 over 150 epochs took 4.15 hours and that of CNN2 over 200 epochs 9.7 hours. More time to be required for the training of CNN2 is mainly due to its more sophisticated architecture.

In practice, prior to the implementation of the training, a baseline method was used to estimate a baseline that we will beat to demonstrate the effectiveness of the proposed deep learning models. Since the problem of phase estimation is a regression problem, we employed the mean value of the ground truth of each training data as the predicted result of the baseline method. For instance, for CNN1 the fringe image is $I(x,y)$, and its ground truth is $A(x,y)$. We calculated the mean value $\bar{A} = \sum_{x=1}^{W}\sum_{y=1}^{H} A(x,y)$, where (x,y) is the pixel coordinate, and W and H is the image width and height. Then, we let $A_{base}(x,y) = \bar{A}$ be the result predicted by the baseline method for this fringe image. The same strategy was employed to predict the result of baseline method for CNN2. Equations (2) and (3) of the supplement material were used to compute the difference between the results of baseline method and the ground truth. We find that the loss of CNN1 is 7719 and that of CNN2 is 27156 for the baseline methods. On contrary, with proper training the losses of CNN1 and CNN2 can reduce greatly. As indicated by Fig. S4, the accuracy of both models has been improved significantly with the proposed method.

4 Analysis of fringe patterns with relatively low frequency

In this experiment, the projected fringe pattern has frequency $f = 60$, which is almost reduced by 2/3 compared with the one ($f = 160$) in Fig. 4(a) of the main manuscript. The captured fringe images is shown in Fig. S5(a), which can be observed with relatively wide stripes. Our neural networks were first trained using fringe patterns with the reduced frequency and then predicted the phase of Fig. S5(a). For comparison, we also implemented FT and WFT. The phase errors of FT and WFT are shown in Figs. S5(b) and (c), respectively. We can see the overall phase distributions were calculated with evident distortion for both methods. Especially, the phase errors are very serious for the regions with rich details, e.g., the areas of curly hair. Quantitatively, Table S1 shows the mean absolute errors (MAE) of these methods. We can see the error of FT and WFT increases to 0.28 rad and 0.26 rad. As to the proposed method, we found it is less sensitive to the variation of the density of fringes as can be observed in Fig. S5(d). Except for some hairs on the left side, most regions were measured with higher accuracy than FT and WFT. From the quantitative result, the MAE of our method is 0.10 rad which is much smaller than the ones of FT and WFT.

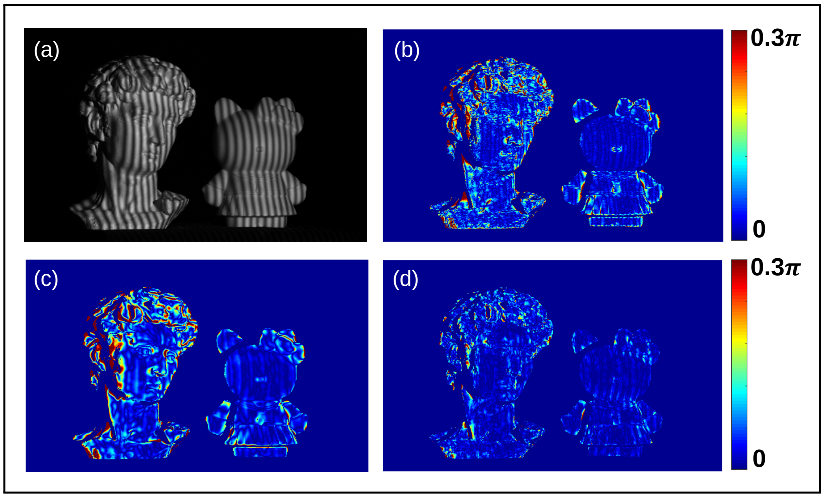

Figure S5. Comparison of the phase errors for different methods when the projected pattern has frequency $f = 60$. (a) Analyzed fringe pattern; (b)-(d) phase errors of FT, WFT and our method, respectively.

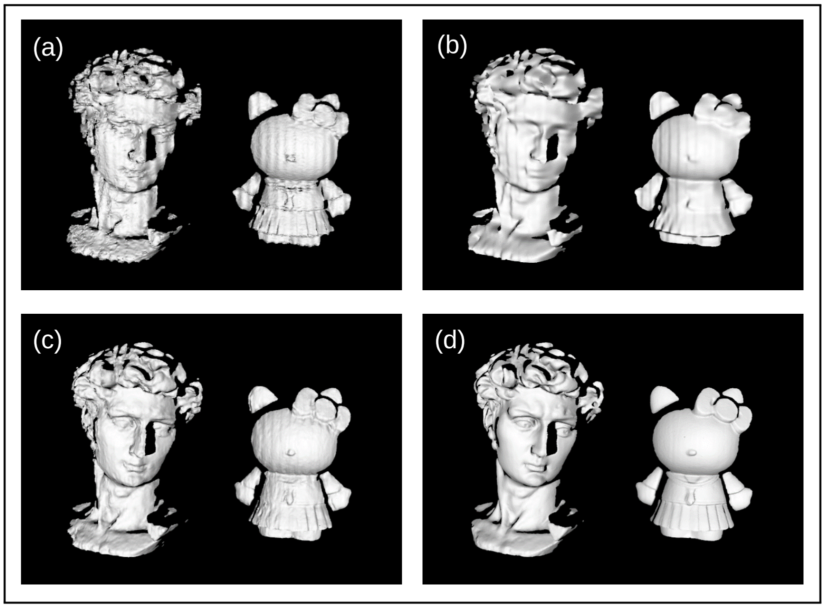

Figure S6. Comparison of the 3-D reconstruction results for different methods: (a) FT, (b) WFT, (c) our method, (d) ground truth obtained by 12-step phase-shifting profilometry.

Table S1. Phase error of FT, WFT, and our method when $f = 60$

Method	FT	WFT	Our
MAE (rad)	0.28	0.26	0.10

Moreover, we converted the phase information into 3-D reconstructions as shown in Fig. S6 for an intuitive comparison. The reduction of spatial frequency significantly influences the demodulation of the phase for FT and WFT as can be seen in Figs. S6(a) and S6(b), where both the complex surfaces of the hair and the smooth area of the face were poorly reconstructed. As to our result shown in Fig. S6(c), we can see the object was reconstructed with much higher accuracy through deep learning. This experiment validates that the fringe analysis using deep neural networks is more robust than FT and WFT in terms of the sensitivity to the variation of spatial frequency of recorded fringes.

5 Selection of the optimal fringe frequency

In order to choose the fringe with optimal frequency for training, we trained our neural networks with 11 sets of patterns with different frequencies $f = 1, 40, 80, 120, 160, ..., 360, 400$. Figure S7 shows the training and validation errors of CNN1, from which we can see the higher the frequency the smaller the errors in training. Then, we tested the performance of CNN2, and the errors are shown in Fig. S8. Analogous to the results of CNN1, the error of CNN2 is decreasing with the increase of the fringe frequency as well.

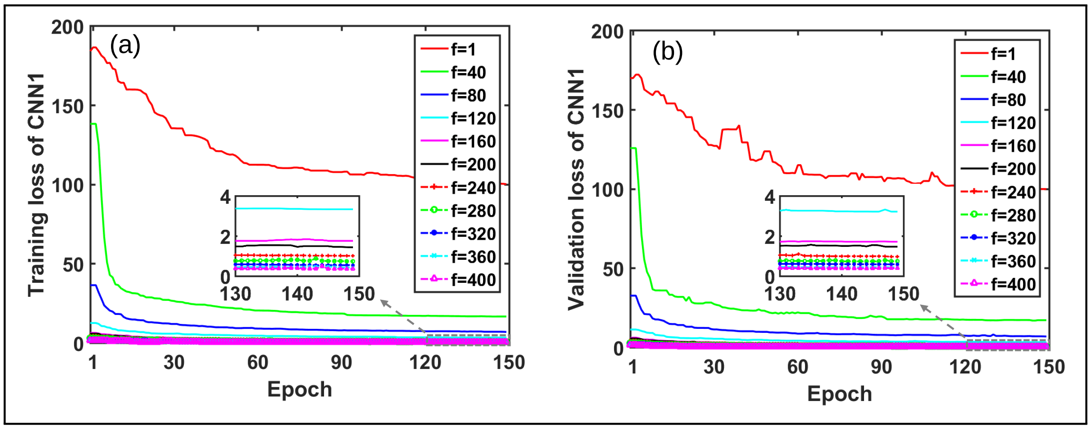

Figure S7. Loss curves of CNN1 when it is trained to analyze the fringe image with different frequencies. (a) Training loss obtained with fringes of various frequencies; (b) validation loss obtained with fringes of various frequencies.

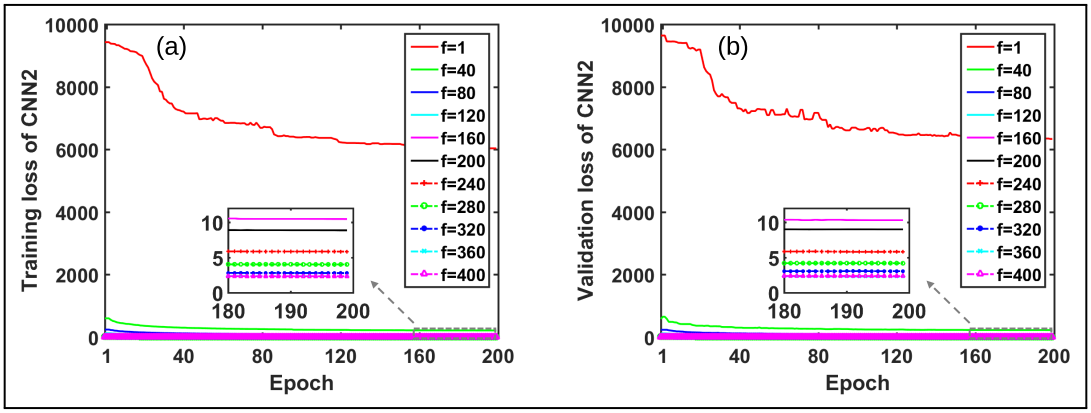

Figure S8. Loss curves of CNN2 when it is trained to analyze the fringe image with different frequencies. (a) Training loss obtained with fringes of various frequencies; (b) validation loss obtained with fringes of various frequencies.

However, we find the smallest training error or validation error may not guarantee the highest measuring precision. The reason is that the ground-truth data obtained with the phase-shifting method becomes inaccurate due to the decrease of fringe contrast when the frequency is excessively high. In our experiments, we measured the pair of standard spheres with trained models of different frequencies and calculate the average MAE of the radii of the spheres. The result is shown in Fig. S9 where the 12-step phase-shifting method was applied for comparison. On one hand, we can see the error of the 12-step phase-shifting method begins to increase when the frequency is larger than 160. As the accuracy of the deep learning based method relies on that of the training data, the result of the proposed method is affected when the ground truth of the training data is becoming less accurate. The most likely reason for this is that the phase-shifting technique is a pointwise method and is sensitive to noise. While in our deep-learning based method, each pixel is connected and highly correlated to its neighbors, providing greater flexibilities in signal localization and noise suppression especially in low signal-to-noise ratio (SNR) condition. On the other hand, we can see that the accuracy is also adversely influenced if the frequency is too low, e.g., $f = 1$. The reason could be the fact that it is difficult for CNN1 to correctly extract the background image from the fringe image, as most of the useful information are easily hidden in the wide and dark stripe. According to the results above, we therefore employed the fringe images of $f = 160$ to measure objects in our experiments.

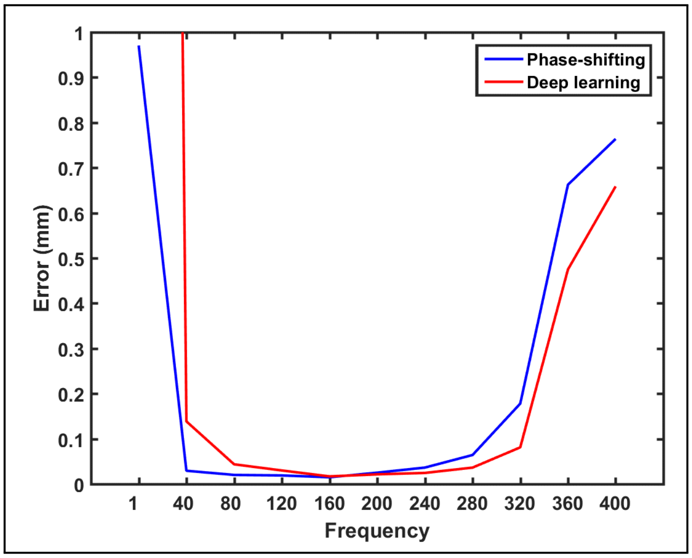

Figure S9. 3-D reconstruction error of standard spheres when the fringe pattern has different frequencies.

In summary, to find fringes of the optimal frequency, one is suggested choosing the fringe with high frequency, which is dense enough while does not affect the contrast of captured patterns that guarantees the accuracy of the calculated ground truth.

6 Ablation analysis

To conduct the ablation study of the proposed method, we additionally trained three neural networks to estimate: (1) the phase $\phi(x,y)$ directly; (2) $D(x,y)$ and $M(x,y)$ without $A(x,y)$; (3) $A(x,y)$, $D(x,y)$ and $M(x,y)$ simultaneously. First, for a concise description, these methods are named as "Phi", "DM", and "ADM", respectively. Then, to simplify the comparison, we used the same structure to train all of the methods, which is the same as CNN2. The only difference lies in their different output as we design. We compared them with our developed CNN2 and Fig. S10 shows the training and validation errors. From Fig. S10(a), we can see the errors during training are very large and do not converge for Phi. On contrary, the methods DM, ADM and CNN2 have much better performance as their losses converge rapidly as can be seen in Figs. S10(b)-S10(d). It is noted that the output of these models are not in the same scale. For example, the phase ranges from $-\pi$ to $+\pi$ while D and M can reach several hundreds. Therefore, instead of investigating the testing error directly, we converted the output of these methods into the phase for comparison.

We exploited these methods to measure several scenes of which the fringe image is shown in Fig. S11. The phase error was calculated against the result computed by 12-step phase-shifting algorithm. Figure S12 shows the phase errors of these methods. We find Phi has the highest error for the tested scenes, which can be deduced from its poor performance in the training process. The rest approaches performed much better than Phi and show similar performance. From the magnified view, we can see that our CNN2 has the smallest errors compared with DM and ADM. The reason should be the fact that CNN2 exploits an extra

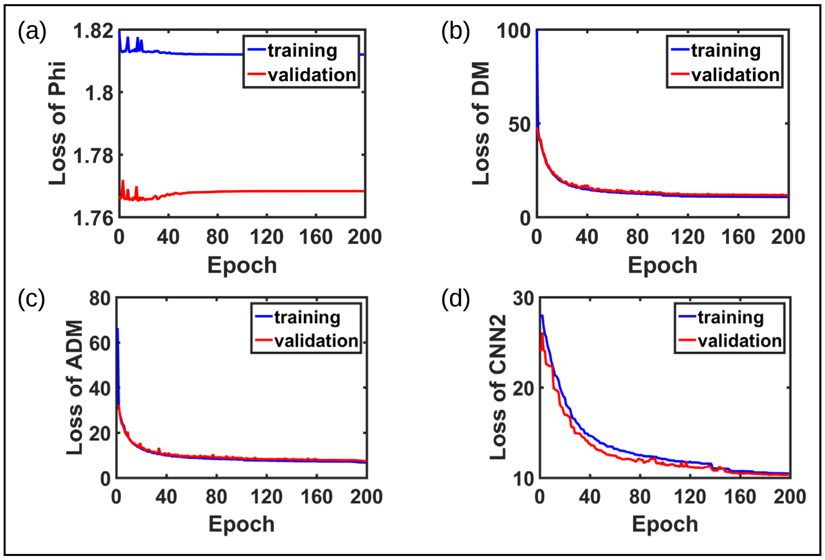

Figure S10. Loss curves of the training and validation set for different neural networks. (a) Phi which estimates the phase ϕ directly; (b) DM which predicts the numerator D and the denominator M without the background image A; (c) ADM which calculates A, D and M simultaneously; (d) the proposed CNN2.

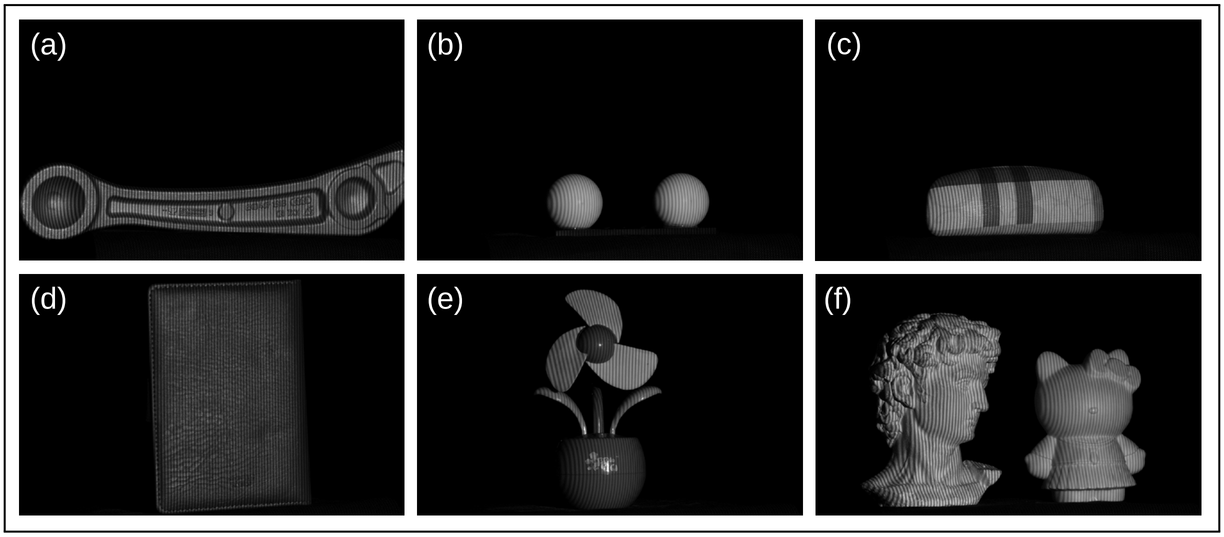

Figure S11. Different scenes to test the performance of methods Phi, DM, ADM, and CNN2. (a) Scene 1: an automobile part; (b) scene 2: a pair of standard spheres; (c) scene 3: an eyeglass case; (d) scene 4: a note book; (e) scene 5: a toy fan; (f) scene 6: two plaster models.

background image for the phase calculation, which provides additional information and thus eases the estimation of phase. The ablation analysis validates the effectiveness of the proposed method.

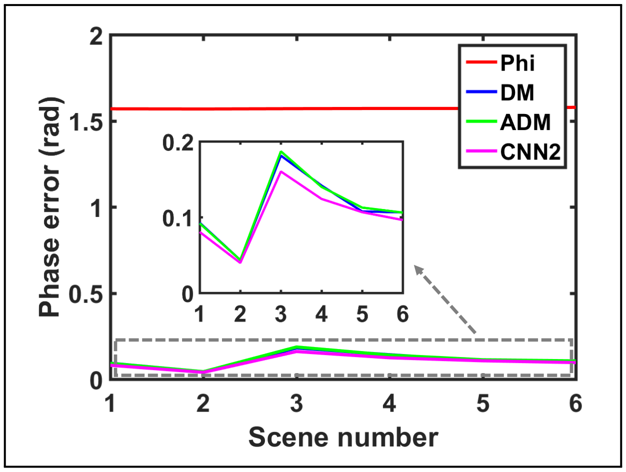

Figure S12. Comparison of the phase error of different methods for different scenes. The area in the dotted box is magnified to show the details of the phase error for each method.

7 Measurements of different kinds of object

In this section, we tested the performance of our trained neural networks with more objects to investigate the adaptability to scenes they never see. The first column of Fig. S13 shows the fringe images of the tested scenes. The second to the fourth column demonstrates the absolute phase error of FT, WFT and our method respectively, which was calculated by referring to the phase computed by 12-step phase-shifting method. The average of the absolute error is calculated and listed in Table S2.

Table S2. Phase error of FT, WFT and our method for different scenes

	Scene 1	Scene 2	Scene 3	Scene 4	Scene 5
FT	0.26	0.069	0.28	0.32	0.21
WFT	0.099	0.077	0.16	0.10	0.21
Our	0.075	0.032	0.10	0.094	0.094

First, Scene 1 is an automobile part which is made of aluminium alloy having long and narrow plat surface with small bumps at the central area of the model number. FT shows the most significant error distribution which can be observed on the plane surface and the regions of bumps. Compared with FT, the phase error of WFT reduces obviously for the flat regions. As to our method, it performed better than WFT as can be observed from the further decreased errors for both flat and convex surfaces. Then, for Scene 2 which is a pair of smooth porcelain spheres, FT and WFT have similar phase errors which are 0.069 rad and 0.077 rad. By contrast, our method shows the smallest error 0.032 rad which is half of those errors. Next, the third scene is an eyeglass case made of plastic and the fourth is a note book with a cover of artificial leather. For both of them, their surfaces are manufactured with lot of subtle granular structures. From the map of phase error, we can see FT failed to recover the phase of these surfaces. In comparison, WFT and our method were more appropriate. In particular, the proposed method is superior to WFT according to the smallest errors. Last, for Scene 5 which is a toy fan also made of plastic, FT and WFT have similar performance, and their phase error is the same which is 0.21 rad. From the distribution of phase error, points with large error mainly exist on the edges of blades of the fan for methods of FT and WFT. Therefore, due to the abrupt depth change occurred in these areas, we can see the measuring difficulty increases for the traditional methods. On contrary, the errors on the boundaries have been reduced when our method was applied, which implies that our method is less sensitive to those regions. According to the quantitative comparison, the phase error of the proposed method is 0.094 rad which is half of that of FT and WFT.

Based on the results above, we can see that the deep-learning based method has good adaptability to different kinds of objects. As a new technique using a single fringe pattern for phase calculation, it has higher precision than the traditional FT and WFT. In particular, it can preserve more surface details and cope better with abrupt depth variations.

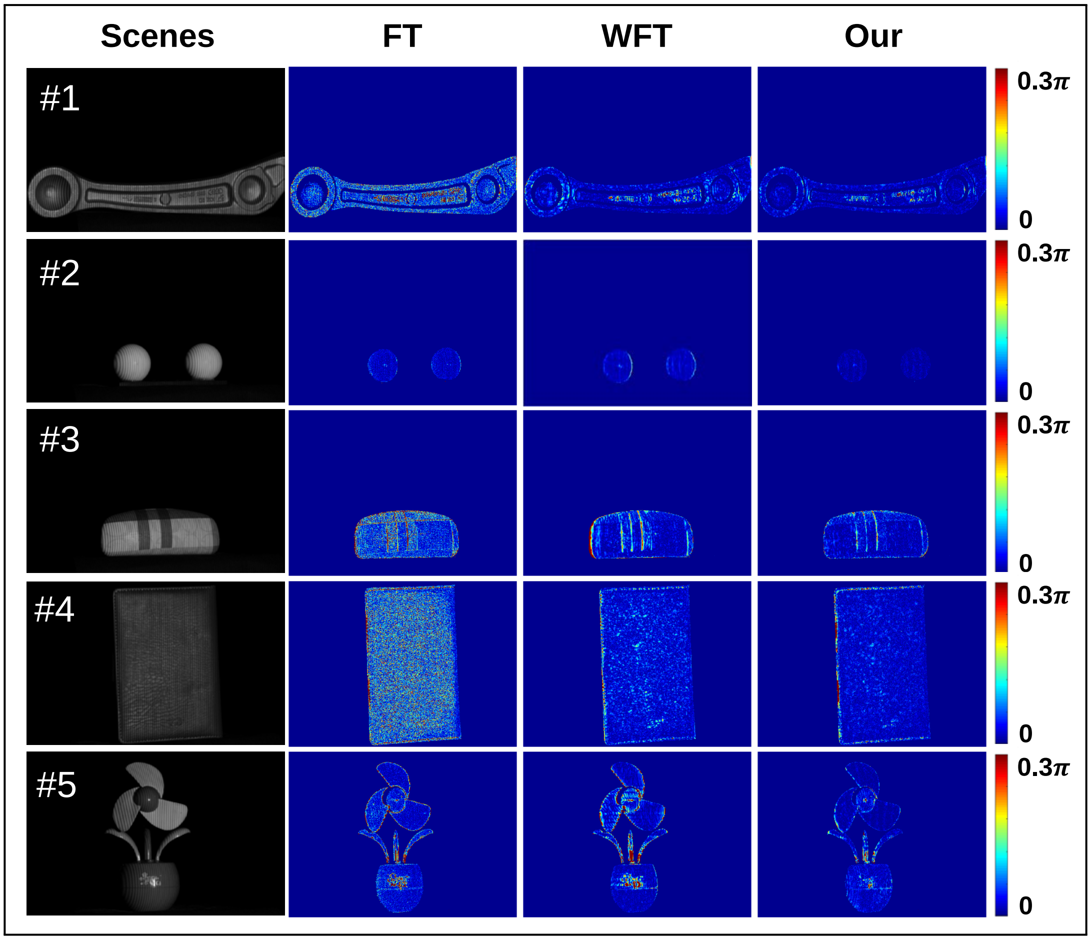

Figure S13. Comparison of the proposed method with FT and WFT in terms of different kinds of objects. The first column shows the fringe images of different scenes, and the second to the fourth column is the absolute phase error of FT, WFT and our method respectively.

References

1. Nair, V. & Hinton, G. E. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, 807–814 (Omnipress, USA, 2010).

2. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *CoRR* **abs/1412.6980** (2014).