Towards reduced CNNs for de-noising phase images corrupted with speckle noise

Marie Tahon
Silvio Montresor
Pascal Picart

02/20/2025

Table des matières

I. Présentation 3

II. Cours 4

1. Databases....7
   - 1.1. HOLODEEP database....7
   - 1.2. DATAEVAL database....8
   - 1.3. NATURAL database....11
2. Baseline approaches....12
   - 2.1. Signal processing approaches for speckle de-noising....13
   - 2.2. Deep learning approach for speckle de-noising....13
3. Experimental protocols....15
   - 3.1. Data pre-processing and implementation....15
   - 3.2. Network architecture....16
   - 3.3. Evaluation network depth and architecture....17
   - 3.4. Over-fitting and influence of the number of layers of the CNN....17
   - 3.5. Evaluation of a pre-trained network....18
   - 3.6. Protocole....18
4. Results and discussion....18
   - 4.1. Network depth and architecture....18
   - 4.2. Pre-training....19
   - 4.3. Evaluation on target images....21

III. Conclusion 22

IV. Acknowledgment 23

Ressources annexes 24

Bibliographie 25

Webographie 27

I.Présentation

Module :

Towards reduced CNNs for de-noising phase images corrupted with speckle noise

Auteur(s) :

Marie Tahon - LIUM - Le Mans Université
Silvio Montresor - LAUM - Graduate School (IA-GS) - Le Mans Université
Pascal Picart - LAUM - Graduate School (IA-GS) - Le Mans Université

Résumé :

Digital holography is a very efficient technique for 3D imaging and the characterization of changes at the surfaces of any objects. However, during the process of holographic interferometry, the reconstructed phase images suffer from speckle noise. In this paper, de-noising is addressed with phase images corrupted with speckle noise. To do so, DnCNN residual networks with different depths were built and trained with various holographic noisy phase data. The possibility of using a network pre-trained on natural images with Gaussian noise is also investigated. All models are evaluated in terms of phase error with HOLODEEP benchmark data and with 3 unseen images corresponding to different experimental conditions. The best results are obtained using a network with only 4 convolutional blocks, and trained with a wide range of noisy phase patterns.

Mots-clés :

Digital holography, image de-noising, deep learning, DnCNN, fine-tuning

Pré-requis :

-

Objectif(s) pédagogique(s) :

-

Plan du cours :

- Introduction
- Databases
- Baseline approaches
- Experimental protocols
- Results and discussion
- Conclusion

Conception & production :

Le Mans Université

Licence :

II.Cours

Digital holography and related speckle based methods are very efficient techniques for the measurement of displacement fields and surface shape [1] [Digital Holography]. Due to contact-less measurements, characterization of objects can be obtained with very good accuracy with speckle patterns. Numerical back propagation yields the reconstruction of the amplitude and phase images of the object. Although this speckle pattern is quite useful for encoding, its drawback is that the reconstructed amplitude image suffers from speckle noise. Speckle noise in holographic phase data is very particular because it has non-Gaussian statistics and exhibits non-stationary property, whereas generally in amplitude images, this noise is considered as multiplicative noise.

The digital holography technique is a laser based optical process. It allows measurements of displacement fields and surface shape. Due to contactless measurements, rough surfaces characterization of objects can be obtained with very good accuracy.

Digital holography is based on a coherent mixing of a reference wave and an object wave which results from the light diffraction from any object. When the object surface is rough, speckle is included in the digital hologram. In the case of digital holographic microscopy, objects are generally transparent and thus there is no speckle in phase images. In this paper, the case of rough object surface producing speckle in phase extracted from holograms is considered.

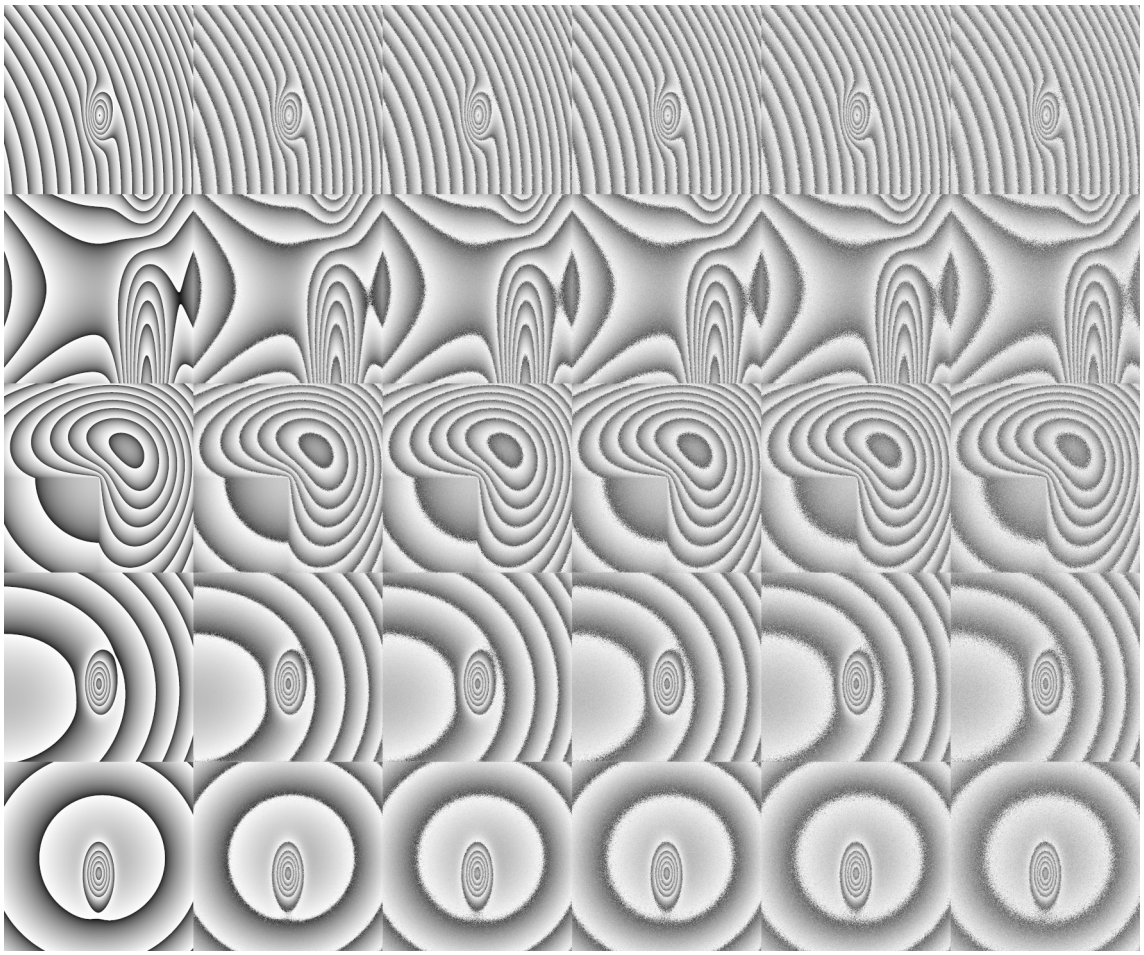


Figure 1 : An image sensor (CMOS or CCD) records the fringe patterns delivered by the coherent mixture.

Figure 1 exhibits the digital holography principle.

If the hologram is recorded~\cite{A3} in accordance with Shannon conditions, we obtain an image of the object wave-front from digital reconstruction of the hologram. The complex field of the object is computed by Single Fast Fourier Transform (S-FFT) or Double Fast Fourier Transform (D-FFT) algorithms using adjustable magnification ~\cite{B1}. The numerical calculation of the diffracted field yields a complex amplitude $O_0(x,y)$ sampled on a grid corresponding to the number of reconstruction points of the algorithm. The amplitude image (modulus), A_r, and phase image, ψ_r (argument of the field) are derived from the numerical calculation of the object diffracted field when the hologram is recorded.

namely:

$$A_r(x,y) = | O_0(x,y) |, \qquad (1)$$

$$\psi_r(x,y) = \operatorname{atan2}\{\frac{\Im m[O_0(x,y)]}{\Re e[O_0(x,y)]}\} \bmod(2\pi). \qquad (1)$$

The phase of the field is calculated by an arctangent function, and consequently result is wrapped in the interval $[-\pi,+\pi]$, modulo 2π.

Consequently to computation by the means of the arctangent function,ψ_r is wrapped in the interval $[-\pi,+\pi]$, modulo 2π.

In most cases [2] [Deep residual learning for image recognition] the randomness of the phase, which is reconstructed from the object diffracted field, is due to the roughness of the object. The reconstructed object is then subject to the phenomenon of speckle noise. Phase estimation of the reconstructed optical field is a key issue for a large number of applications of digital holography.

Metrology applications require optical phase, so this paper focuses on phase changes over time. The quantity of interest is a phase difference between two instants, allowing to follow-up the evolution of a phenomenon over time. Taking into account Doppler effect, the phase difference is proportional to the displacement field of the object between the two instants. As the optical phase is calculated from the arctangent function, then it is wrapped. Phase must be unwrapped in order to access physical kinematic quantities of the object [3] [Two-dimensional phase unwrapping: theory, algorithms, and software]. For example, digital holography permits to investigate complex acoustic phenomena by using the method of ultra-fast digital holography with sampling rate-up to 100kHz [4] [High-speed holographic metrology: principle, limitations, and application to vibroacoustics of structures].

Regarding image de-noising, algorithms are generally designed with the assumption of additive Gaussian noise and there is a real need for new de-noising approaches able to cope with speckle noise and complex fringe patterns. For a decade, the reference algorithms are related to non-local patch based methods such as BM3D [3] [Two-dimensional phase unwrapping: theory, algorithms, and software], wavelet based methods such as DTDWT [5] [Experimental and theoretical investigation of the pixel saturation effect in digital holography], and short-term Fourier transform algorithms as the WFT2F [6] [Windowed Fourier transform for fringe pattern analysis: theoretical analyse].

Machine learning algorithms exhibit growing interest for signal and image processing since the last 10 years. In particular, neural networks are able to learn very complex functions from databases.

At the heart of these algorithms, convolutional networks (CNNs) have been extensively used in place of traditional image processing approaches.

In contrast with these traditional approaches, machine learning based solutions such as convolutional neural networks (CNNs) use dataset examples and are able to learn how to invert very complex degradation functions [7] [Natural image denoising with convolutional networks].

They have been used to simulate wavelets and multiresolution analysis, shrinking and thresholding algorithms, sparse representations, block matching and dictionary learning [8] [Computational image speckle suppression using block matching and machine learning].

Many neural architectures have been developed for Gaussian noise such as residual learning for image recognition [2] [Deep residual learning for image recognition], generative adversarial networks (GANs) [9] [Generative adversarial nets].

Note that in the field of digital holography and digital holography microscopy, several papers related to applications of CNN were published [10] [On the use of deep learning for computational imaging].

Currently, state of the art image de-noising systems are dominated by DnCNN [8] [Computational image speckle suppression using block matching and machine learning] and its recent modifications such as hierarchical residual learning HRLNet [11] [Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising].

Residual networks learn to predict the residual image between the clean and noisy inputs. It includes skip-connections which consist of an identity mapping placed between two non-adjacent layers and helps to avoid the vanishing gradient problem when the network depth is high [2] [Deep residual learning for image recognition]. With residual learning very deep networks can be easily trained and improved accuracy has been achieved for image classification and object detection.

Several approaches were proposed in optical coherence tomography [12] [On the use of deep learning for computational imaging], in hyperspectral imaging or using multiscale decompositions [10] [Speckle noise reduction for digital holographic images using multi-scale convolutional neural networks]. The problem of speckle decorrelation has also been approached with deep learning networks with conditional GANs [13] [Speckle noise reduction in optical coherence tomography images based on edge-sensitive cGAN].

While the amount and the diversity of natural images are huge and thus allow to train deep networks with many parameters, when moving to phase data processing in digital holography, the quantity and the diversity are clearly reduced.

Indeed, there is currently no way to get experimental phase data with speckle noise together with its clean version. That is the reason why simulated data is required.

Such simulation can foster the diversity of fringe patterns and frequency.

Image de-speckling ground truth clean images have been generated from outputs of commercial optical coherent tomography scanners [13] [Speckle noise reduction for digital holographic images using multi-scale convolutional neural networks].

In [14] [Quantitative appraisal for noise reduction in digital holographic phase imaging], a database including 25 fringe patterns divided into 5 patterns and 5 different signal-to-noise ratios was generated with a realistic noise simulator [15] [Error analysis for noise reduction in 3D deformation measurement with digital color holography] to foster the diversity of phase fringe patterns.

To improve de-noising performances, one solution is to go deeper, *i.e.* to add more layers to the network. However, with higher capacity two problems emerge: overfitting and vanishing or exploding gradients.

The latter can be controlled by batch normalization and the use of skip connections such as in residual networks.

Overfitting can be controlled by regularization.

However, the amount of data is crucial to avoid overfitting even with regularization techniques. The use of data augmentation usually helps in artificially increasing the amount of training data [16] [The effectiveness of data augmentation in image classification using deep learning].

However, one solution can also be not to go deeper.

https://www2.cs.duke.edu/courses/fall20/compsci371d/slides/s_10_CNNGeneralization.pdf[1]

While it is known that a relation does exist between the network depth and the size of the convolutional filters (and consequently the receptive field) [17] [Very deep convolutional networks for large-scale image recognition], the question of the necessity of depth has not

1 - To improve de-noising performances, one solution is to go deeper, \textit{i.e.} to add more layers to the network. However, with higher capacity two problems emerge: overfitting and vanishing or exploding gradients.

been much investigated. In [18] [Depth selection for deep ReLU nets in feature extraction and generalization], the authors propose quantification of the correspondence between features learnt by the network and its depth.

DnCNN [2] [Deep residual learning for image recognition] has been designed following this approach.

Following this approach, for natural images Gaussian de-noising with a certain noise level, DnCNN receptive field has been set to 35×35 with corresponding 17 convolutional layers [2] [Deep residual learning for image recognition].

A recent study has shown that the different trained layers of a network are relatively uncorrelated, which could explain why neural networks are resistant to overfitting. This study also conclude that large networks trained on datasets with a small number of training samples also generalize well and avoid overfitting.

The generalization power of machine learning algorithms is the ``ability to perform well on previously unobserved inputs'' [19] [Deep Learning]. To do so, data is usually split into training, development and test sets, the reminder consisting of the unobserved inputs.

In previous work, authors trained a DnCNN for holographic phase data with speckle de-noising [20] [Computational de-noising based on deep learning for phase data in digital holographic interferometry]. This network reaches good performances with the benchmark data in comparison to other de-noising techniques such as BM3D or WFT2F on most of the evaluated phase images.

In the present paper, networks are evaluated in terms of phase errors and generalization power defined as the ``ability to perform well on previously unobserved inputs'' [19] [Deep Learning]. The aim is to reduce the training time while reaching similar performances. To do so, databases for development and validation are presented in section Databases. Baseline de-noising algorithms and results are summarized in section Baseline approaches. The training protocols include networks with different depths, on various phase image data (voir section Experimental Protocols). With the advantage of fine-tuning with phase data corrupted with speckle noise, a network previously trained on natural noisy images is also investigated. Experimental results are discussed in section Results and discussion.

1. Databases

1.1. HOLODEEP database

This database consists in 5 different types of noise-free phase fringe patterns and used to train the models and for development purposes.

Each pattern is degraded with realistic speckle decorrelation noise with statistics described in [14] [Quantitative appraisal for noise reduction in digital holographic phase imaging]. From each noise-free fringe pattern, five noisy fringe patterns controlled with a parameter, namely Δ , are generated with the simulator presented [14] [Quantitative appraisal for noise reduction in digital holographic phase imaging] , corresponding to different signal-to-noise-ratios (SNR) in the range [3dB-12dB].

The parameter Δ is used to mimic strongly degraded experimental phase data. $The\ higher\ \Delta, the\ smaller\ the\ SNR.$ In real conditions, there are several degradation sources which may induce more decorrelation noise than expected if all would be perfect. As examples, the reconstruction of holographic data might be not perfectly in focus [21] [Refocus criterion based on maximization of the coherence factor in digital three-wavelength holographic interferometry] , the pixels could have a large active surface [4] [High-speed holographic metrology: principle, limitations, and application to vibroacoustics of structures] , the recording could have low number of pixels or saturated pixel [5] [Experimental and theoretical investigation of the pixel saturation effect in digital holography] , the number of useful

quantization bits could be insufficient [22] [Quality assessment of combined quantization-shot-noise-induced decorrelation noise in high-speed digital holographic metrology] , or also there could be wavelength changes between exposures [23] [Improvement of accuracy in digital holography by use of multiple holograms]. All these degradation source have for consequence the increase of the speckle decorrelation and then the increase of the noise. Thus, using Δ is a useful way to get data with more noise in order to mimic possible experimental conditions. In the simulator described [14] [Quantitative appraisal for noise reduction in digital holographic phase imaging], Δ corresponds to small changes in the wavelength between the two exposures. So adjusting Δ is useful to increase the speckle decorrelation and thus to decrease the SNR in phase data. The simulated images, sized 1024×1024 pixels, were generated using Matlab and are available in the Matlab `mat` format or as `tiff` images. The 25 images used for training the models are shown in Figure 2.

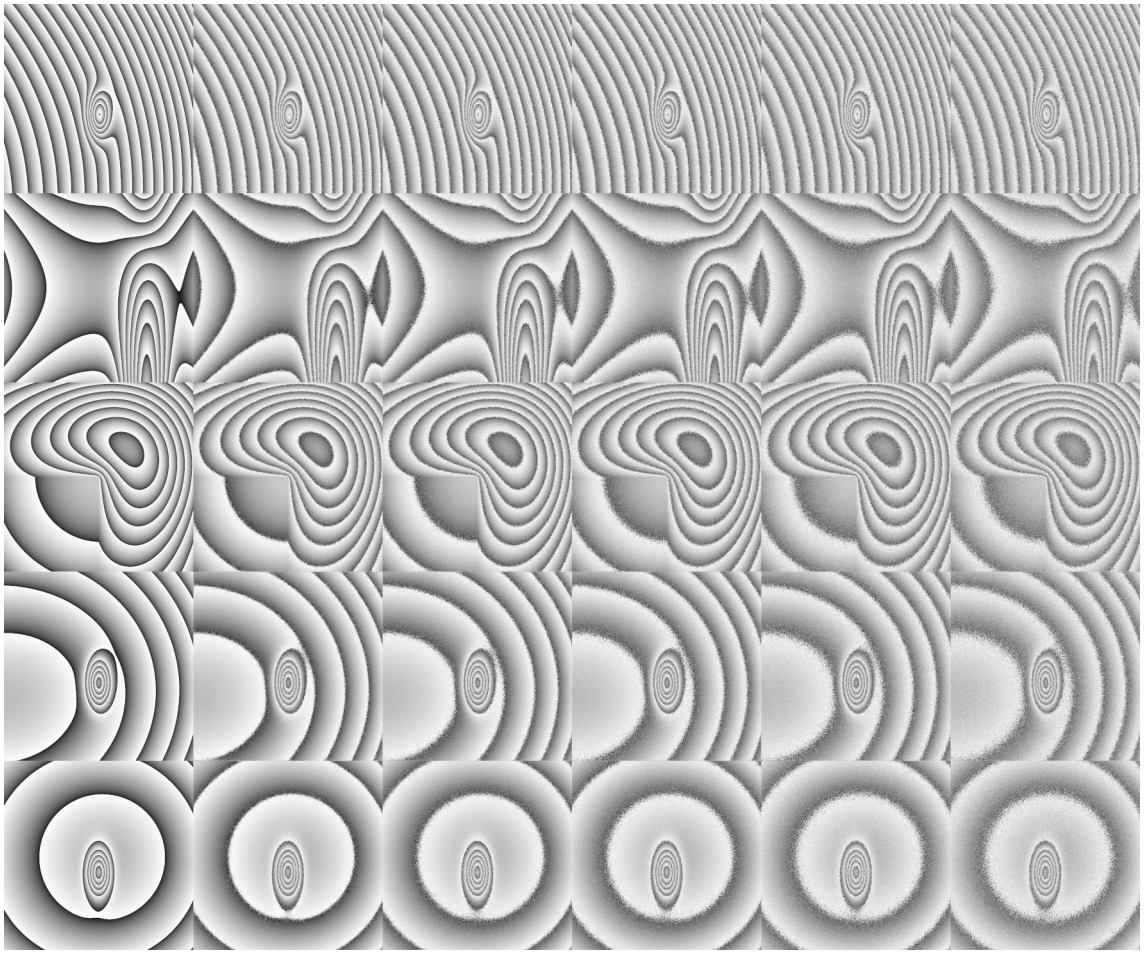


Figure 2 : HOLODEEP training phase images: 5 patterns (in lines) with simulated speckle noise with five values of Δ (in columns)

1.2. DATAEVAL database

This validation database consists in 3 images used for testing the model with images that have not been seen during the training or development process. Two phase images, namely `Test1` and `Test2` , are simulated using the simulator in reference [16] [Quantitative appraisal for noise reduction in digital holographic phase imaging], similarly as for simulating the HOLODEEP database. The SNR of the two phases are respectively $3.05\ dB$ (see Figure 3) and $1.26\ dB$ (see Figure 4). These phase maps are not included in the HOLODEEP database. The last phase is an experimental noisy phase from vibration measured at $17\,512\ Hz$, named `Test3` with a $SNR = 2.52\ dB$. The clean phase is shown in Figure 5, the noisy phase in Figure 6 and the obtained noisy phase in Figure 7. The experimental set-up and methodology to obtain

such phase images is described in references ([4] [High-speed holographic metrology: principle, limitations, and application to vibroacoustics of structures], [24] [Visualization of travelling waves propagating in a plate equipped with {2D ABH} using wide-field holographic vibrometry]). The reader is invited to have a look at the papers for further details.

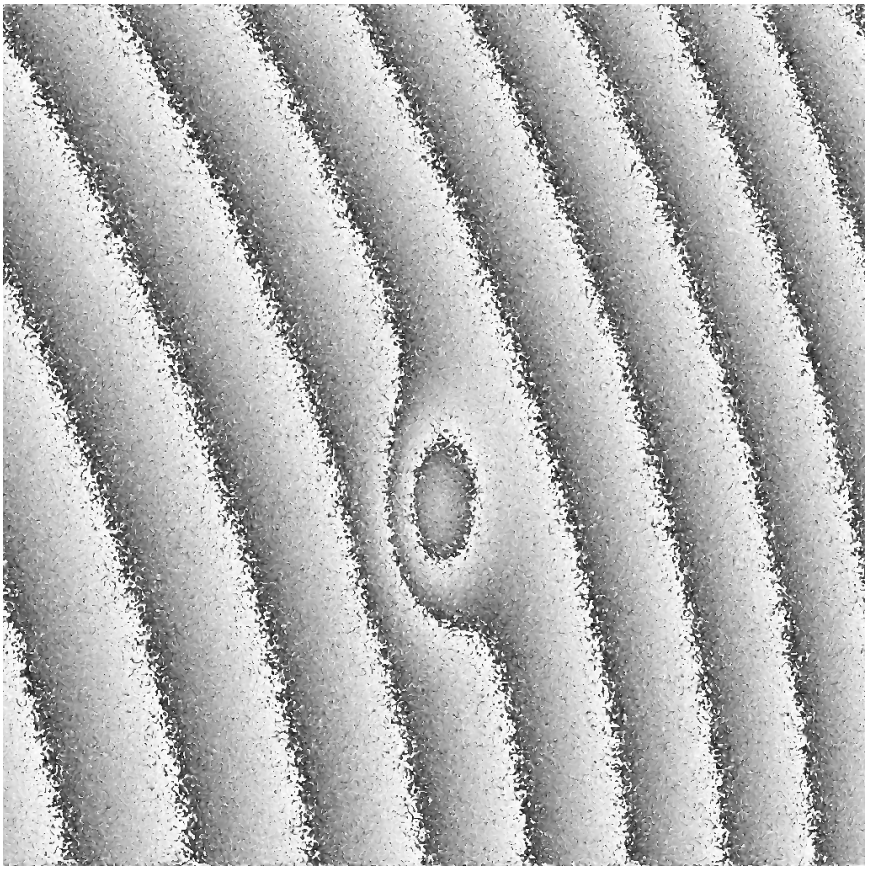


Figure 3 : Noisy Test 1 phase

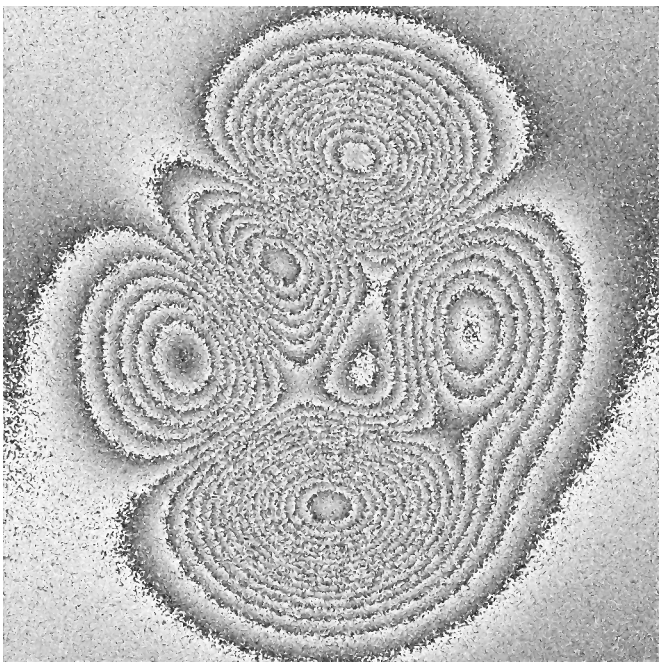


Figure 4 : Noisy Test 2 phase

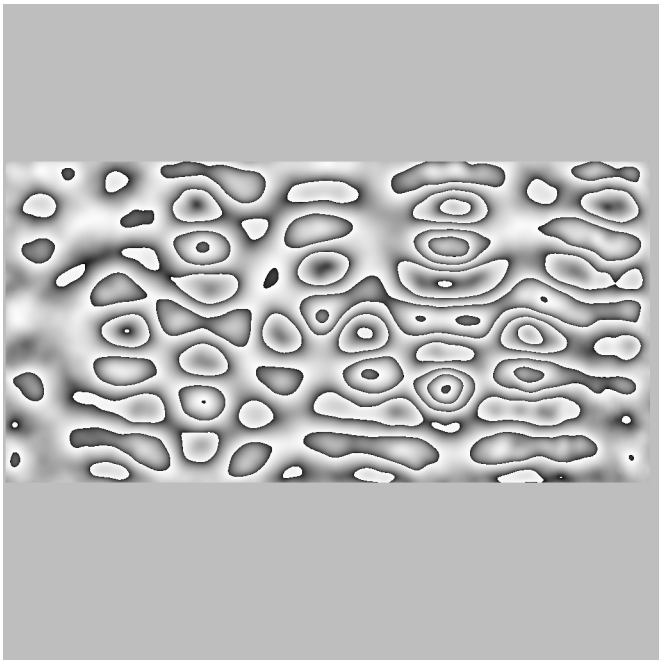


Figure 5 : Noise-free Test 3 phase

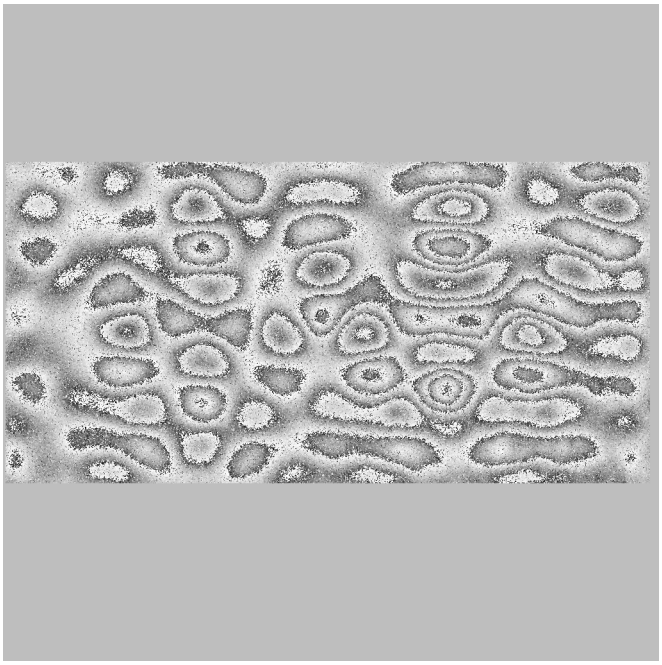


Figure 6 : Noisy Test 3 phase

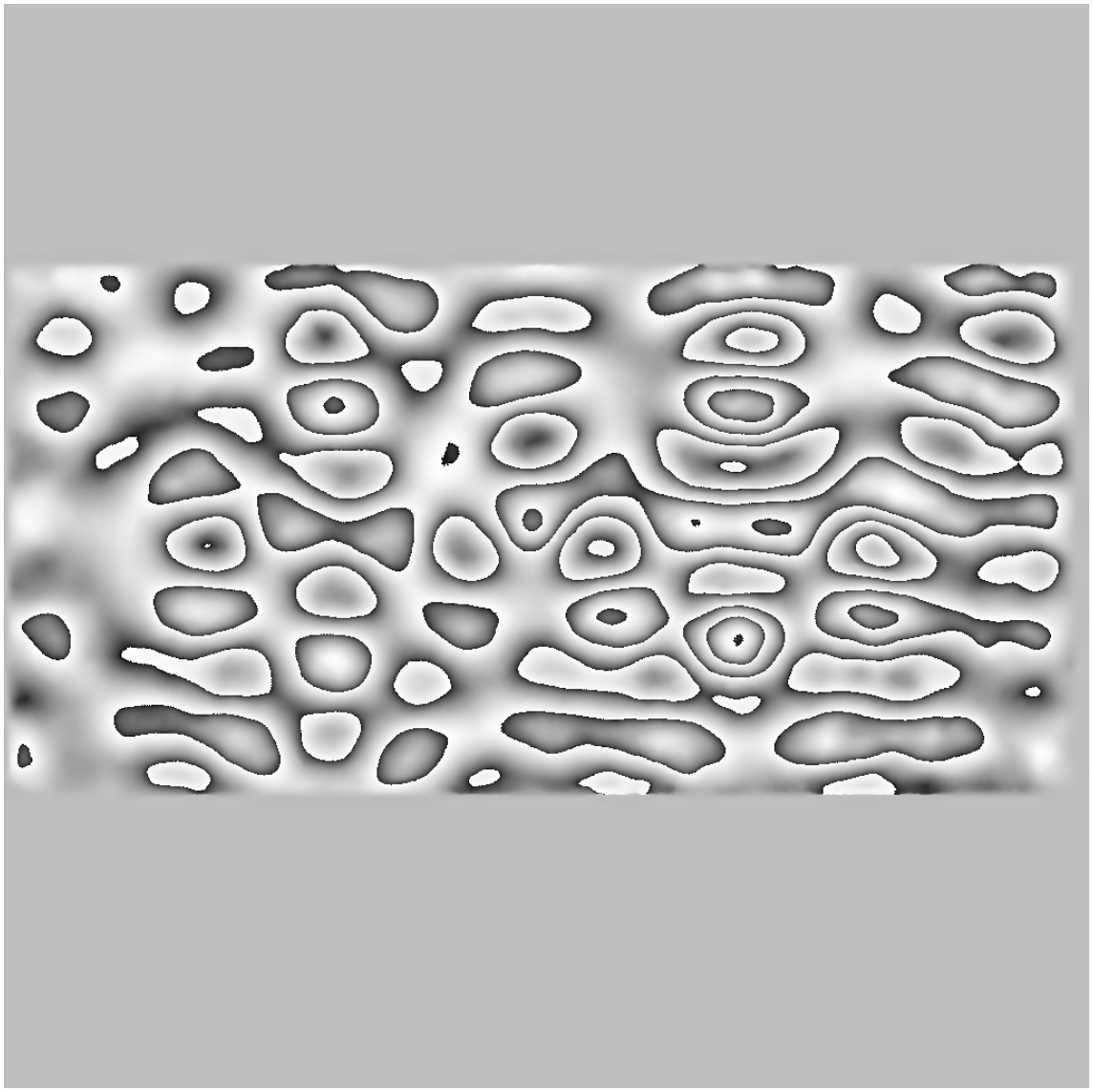


Figure 7 : De-noisedTest 3 phase

1.3. NATURAL database

This database is generally used for natural gray level image Gaussian de-noising. It consists of 400 images of size 180×180. The RGB images are available on the link [2]. Noisy images are obtained by adding Gaussian noise with different SNR values (over $13\ dB$) directly to the clean images.

2 - http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/BSR/BSR_bsds500.tgz

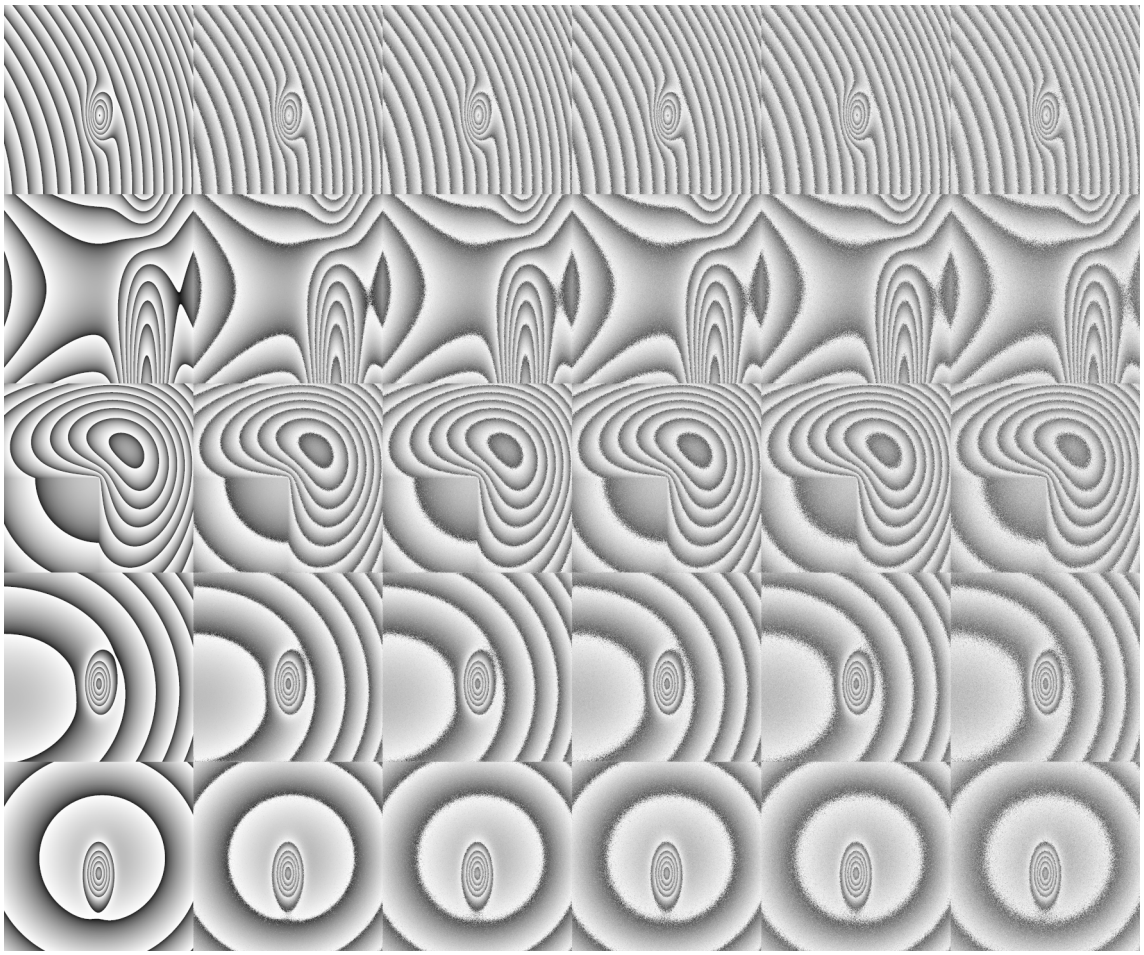


Figure 2 : HOLODEEP training phase images: 5 patterns (in lines) with simulated speckle noise with five values of Δ (in columns)

2. Baseline approaches

In this section, baseline results from the state of the art are presented in Table 4. Phase error in radians was obtained from HOLODEEP benchmark database and DATAEVAL images.

Following the protocol described in [15] [Quantitative appraisal for noise reduction in digital holographic phase imaging], three algorithms from signal processing have been tested: WFT2F, BM3D and DtDWT.

A first CNN named DL-3 [25] [An iterative scheme based on deep learning combined with input noise estimator for phase data processing in digital holographic interferometry] have been trained on HOLODEEP images with small speckle noise, $i.e. \Delta\lambda = 0$.

Validation results are given in terms of average $\Delta\Phi$ over the whole HOLODEEP validation database (i.e. 25 images of size 1034×1024), and test results are given on the three images of DATEVAL and in average. All these results are summarized in Table 1 in terms of phase error.

2.1. Signal processing approaches for speckle de-noising

Following the protocol described in [14] [Quantitative appraisal for noise reduction in digital holographic phase imaging], three algorithms from signal processing were tested: WFT2F, BM3D and DtDWT.

The results are given in terms of the standard deviation $\Delta\phi$ of the phase error e_{ij} defined in Eq. 1, where N is the total number of pixels and $e_{ij} = \phi_{denoised}(i,j) - \phi_{noisefree}(i,j)$ is the difference between the de-noised phase $\phi_{denoised}$ and the noise-free phase $\phi_{noisefree}$ $at\ pixel(i,j)$,

$$\Delta\phi^2 = \frac{1}{N}\sum_{i,j}\left(e_{ij} - m_e\right)^2, \qquad (1)$$

(1)

where m_e is the average of $e(i,j)$ over the set of pixels. Note that since $\phi_{denoised}$ and $\phi_{noisefree}$ are calculated modulo 2π, there difference e_{ij} has also to be computed modulo 2\pi according to $e_{ij} = \arg[\exp(i\, e_{ij})]$.

Baseline results are given in terms of the average of $\Delta\phi$ over the whole HOLODEEP database (i.e. 25 images sized 1024×1024), and with the 3 images of the DATAEVAL database. The results for the phase error $\Delta\phi$ are summarized in Table 1.

The iteration number corresponds to how many times the noisy image have been processed by the de-noiser. From Table 1 can be observed that only one iteration is required with WFT2F to get the best error at $\Delta\phi = 0.026\ rad$ with HOLODEEP, because WTF2F uses a threshold on the decomposition 2D waveforms, and the process ends after one iteration. Even with 3 iterations, the two other methods only reach $\Delta\phi = 0.046\ rad$ (DtDWT) and $\Delta\phi = 0.068\ rad$ (BM3D), thus confirming the best performance for WFT2F.

Method	# iter	HOLODEEP	DATAEVAL		
		25 Images	Test1	Test2	Test3
WFT2F	1	0.026	**0.044**	0.164	0.105
DtDWT	1-3	0.046	0.078	0.519	0.214
BM3D	1-3	0.068	0.113	0.580	**0.094**
ine DL-3	1	0.041	0.107	0.585	0.105
DL-3	3	0.031	0.078	0.559	0.077

Table 1. Baseline standard deviation of the phase errors (Df in rad) obtained on the 25 images from the HOLODEEP database (in average) and individual images from DATAEVAL. Iter is the number of times the image passes through the de-noiser

2.2. Deep learning approach for speckle de-noising

a) Data augmentation

Since the training database might be not sufficiently extended, signal processing is used to increase it. For each original phase image, its cosine and sine versions ($\times 2$) are considered together with their transposed and phase shifted version ($\pi/4$ phase shift). This operation helps in increasing by 8 the number of original images.

b) Baseline implementation

The starting network considered in this section is the one proposed [11] [Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising], called DnCNN.

It includes 59 layers organized upon 6 residual blocks (ResBlocks) with ReLu activation function, one input layer, one output layer. The residual network principle implies that the network behaves as a noise estimator.

A first input layer (3×3 convolutional layer and rectified linear units ReLU), **16** intermediate convolutional blocks (ConvBlocks : $3 \times 3 \times 64$ convolutional layer, batch normalization and ReLU), and one output layer ($3 \times 3 \times 64$ convolutional layer) which is used to reconstruct the output noise. The de-noised image is the subtraction of the noisy image and the ouput noise. Different sizes for the number of layer D were trained.

The network has been previously trained on natural images corrupted with gaussian noise, then trained on holographic phase images.

The loss function is a L2 loss between the reference and the predicted pixel values.

The pre-trained network has been downloaded directly from Matlab Deep Learning : *https://www.mathworks.com/help/images/ref/dncnnlayers.html*[3].

Parameters of the training process are summarized in Table 2.

	DnCNN [11]	DL-3 [20]	DL-Py		
original size	180 × 180	1024 × 1024	1024 × 1024		
patch size	50 × 50	50 × 50	50 × 50		
batch size	128	128	128		
learning rate	0.1 to 0.001	0.0006	0.001 ; 0.0005		
# epochs	50	1920	<200		
noise type	Gaussian	Gauss+speckle	speckle		
noise	σ ϵ [0; 55], μ = 0	Δ = 0	Δ = 0	Δ = 0 – 1,5	Δ = 0 -2,5
SNR (dB) range	>13	7.32 – 11.46	7.32 – 11.46	5.08 – 11.46	3.10 -11.46
# train images	400	5 × 8 = 40	5 × 8= 40	5 × 3 × 8 =120	5 × 5 × 8 = 200
# patches	128 × 3000 = 384K	384 × 40 = 15.3k	384 × 40 = 15.3k	384 × 120 =46.1k	384 × 200 = 76.8k

Table 2. Parameters used to train the networks. Δ lies for the simulated speckle noise

DnCNN network[11] [Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising] was pre-trained with 400 grey natural images sized 80×80 from the NATURAL database and optimized with Adam algorithm. The blind Gaussian de-noiser was trained with a large set of noise levels, ($\sigma \in [0; 55]$ is the standard deviation of the additional noise, mean is 0) and a patch size of 50×50. **In the end,** $128 \times 3\,000$ patches were cropped to train the model.

128 being the number of patches per batch used in the implementation (batch size).

DL-3[20] [Computational de-noising based on deep learning for phase data in digital holographic interferometry] uses a pre-trained network : (*https://www.mathworks.com/help/images/ref/dncnnlayers.html*[4]), which is then fine-tuned with data coming from the 5 fringe patterns, and a noise level fixed to 2 pixels per speckle grain in the simulator ($\Delta = 0$). The model was optimized using the stochastic gradient descent (SGD) algorithm. This situation corresponds to realistic digital on-line holographic recording conditions. Each phase image is then augmented 8 times, thus a total of 40 images sized 1024×1024 are used to adapt the model.

c) Baseline results

The results on the validation database are given in table 3 . They are given in terms of average $\Delta\Phi$ over the whole validation database, i.e. 25 images of size 1034×1024.

Method	**# iter** 25 images	**HOLODEEP #25**
DL-3	0	
DL-3	1	.0376
DL-3	2	.0308
DL-3	3	.0292
DL-3	4	.0292

Table 3. Results in Δ φ (rad), obtained with 4 iterations with DL-3 model adapted to fringe pattern Speckle noise. Results from (Montresor, 2020)

The results obtained with DL-3 are reported in Table 1 The aforementioned deep learning model is compared to the signal processing approaches.

3 - https://www.mathworks.com/help/images/ref/dncnnlayers.html
4 - https://www.mathworks.com/help/images/ref/dncnnlayers.html

The results show that DL-3 model slightly under-performs WFT2F on HOLODEEP with 3 iterations, however the computation time is more interesting in the case of deep learning [25] [An iterative scheme based on deep learning combined with input noise estimator for phase data processing in digital holographic interferometry]. The addition of a noise estimator can further improve the performances.

To be comparable with the baseline of de-noising algorithms, only one iteration is taken into account in the following experiments.

From Table 1, with DL-3 and 3 iterations the results are in the range of those from DtDWT and better than BM3D for phase maps `Test1` and `Test2` (speckle size at 4 pixels per grain). DL-3 was trained with only speckle grain at size 2, so this shows that the neural network can generalize with phase maps which do not exactly correspond to the same trained speckle size.

Method	**# iter**	**HOLODEEP**		**DATAEVAL**			
		Time	25 images	DATA1	DATA20	VibMap	AVG
WFT2F*	1	241	.025	.0443	0.1643	.1051	.1046
DL-3	1						
DL-3	3	210	0.41	.0778	.5590	.0737	.2368

Table 4. Comparison of denoising performances(in Δ φ (rad)) between DL models and WFT2F

3. Experimental protocols

The global framework is presented in Figure 8. , where HOLODEEP database is used to train the networks. The evaluation metric is the phase error $\Delta\phi$ computed between the predicted noise-free image and the noise-free reference (refer to Eq. 1).

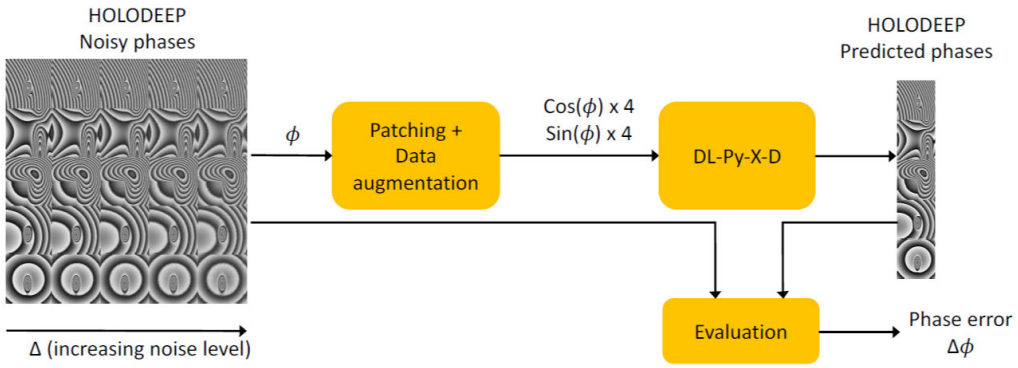


Figure 8 : Overall overview of the system training phase

3.1. Data pre-processing and implementation

The following experiments consider two independent parameters: the type of phase pattern (5 patterns in HOLODEEP database) and the level of speckle noise.

which is given in $\Delta\lambda$ (from 0 to 2.5). For each original image sized 1024×1024, candidate patches are extracted. These patches are sized 50×50 without any overlap. Between each patch (`stride = 50`) and all the image is considered (`step = 0`).

Consequently a total of 400 patches are extracted per image.

A random selection aims at extracting 384 patches per image. The seed is fixed once for all experiments in order to have reproducible patch selection. The whole patches are then shuffled in order to remove their dependency to a specific image.

More precisely, we generate a random permutation of all patches indexes, which allow us to keep the link between clean (target) and noisy (source) images.

Cosine and sine input patches are normalized between 0 and 1.

From this permutation, all the batches configured for network training are built and a final numpy matrix sized $(\#batches \times 50 \times 50 \times 1)$ is saved on the server for experiments

3.2. Network architecture

We treat Matlab matrix format in input, each of them is converted in numpy format.

```
def dncnn(input, D, is_training=True, output_channels=1): #D: number of
    ConvBlocks.
with tf.variable_scope('input'):
output = tf.layers.conv2d(input, 64, 3, padding='same', activation=tf.nn.relu)
for layers in range(2, D + 1):
with tf.variable_scope('block%d' % layers):
output = tf.layers.conv2d(output, 64, 3, padding='same', name='conv%d' % layers,
    use_bias=False)
output = tf.nn.relu(tf.layers.batch_normalization(output, training=is_training))
with tf.variable_scope('output'):
output = tf.layers.conv2d(output, output_channels, 3, padding='same')
return input-output
```

Figure 9 : Python code with tensorflow framework (as tf) which defines model architecture.

A Tensorflow implementation was used as starting point (*https://github.com/wbhu/DnCNN-tensorflow*[5]) and adapted with Matlab matrices as inputs (*https://git-lium.univ-lemans.fr/tahon/dncnn-tensorflow-holography/*[6]).

DL-Py is the Python implementation used in this paper. The architecture is described in Figure 9, where tf denotes the tensorflow library and D is the number of ConvBlocks.

During the training step, the convergence is very fast in the first 10 epochs, then the loss function decreases continuously and slowly.

The maximum number of epochs was fixed to 200 as the performances do not increase significantly with more epochs. However, due to cluster usage constraints, the training has to be stopped before the computing time overpasses a limit of 20 days. This limit has been set due to cluster usage. The number of epochs corresponding to the best phase error is included in Table 6.

The final model is the one that reaches the best results with the development set.

All models are trained on a cluster server with GPUs. Preliminary experiments have shown that training directly our network on normalized phase image (between $[-\pi;\pi]$) drastically decreased the performances as shown in Table 5

Input images used for training are phase data in Matlab format, both their cosine or sine transformed are computed together with their transposition, thus augmenting the database four times. Cosine and sine input images are thus between -1 and 1. They are all normalized in [0;1] to be coherent with the output sigmoid function.

5 - https://github.com/wbhu/DnCNN-tensorflow
6 - https://git-lium.univ-lemans.fr/tahon/dncnn-tensorflow-holography/

Check whether it is the nb of patches for training or the addition of noise in input that brings better performances.

Add more epochs.

3.3. Evaluation network depth and architecture

The network architecture slightly differs from the one proposed in the previous section. The model can be trained with different levels of noise (from $\Delta = 0$ *to* 2.5), different noise-free phase fringe patterns (from 1 to 5), and different depths i.e. different number of ConvBlocks ($D = 4$ or 16).

The following experiments intend to evaluate the influence of these factors on the de-noising performances of the deep learning models.

The number of data and parameters used for training and evaluating the DL-Py networks are given in Table 2. The learning rate is set to $LR = 0.001$, as it has been shown that this parameter has a large impact on the training duration and the results, with an Adam optimizer.

Depth of the network: Because of the high specificity of phase images, the goal is to ensure that the network will not overfit to the training data. To do so, two different networks are trained, one with the original 16 ConvBlocks, the other with only 4 ConvBlocks. With the choice of 4 ConvBlocks as small model, training can be carried out rapidly while maintaining a certain level of complexity.

Noise level for training: Additionally, the network is supposed to be able to de-noise images that have a wide range of noise levels. Therefore, including various level of noise in the training data could help the network to do it. To do so, 3 networks are trained on different noise ranges.

Learning rate: By choice, only one hyper parameter is considered, that is the learning rate. This choice is motivated by the comparison with previous work. The training is done with $LR = 0.0005$ (as previous work) or $LR = 0.001$, as it has been shown that this parameter has a large impact on the training duration.

Method	Train noise	HOLODEEP – Eval noise Δ λ				
	Δ λ	0	1	1,5	2	2,5	AVG
cos/sin (best 100 epochs)	0	0.06	0.08	0.10	0.12	0.15	0.10
phase (best 100 epochs)	0	0.21	0.25	0.28	0.32	0.36	0.28
cos/sin (best 100 epochs)	0, 1, 1.5	0.04	0.05	0.06	0.07	0.09	0.06
phase (best 92 epochs)	0, 1, 1.5	0.17	0.19	0.21	0.23	0.26	0.21
original size	1024 × 1024						
patch size	50 × 50						
# patches per image	384						
batch size	128						
# epochs	< 200						
learning rate	0.001 – 0.0005						
train noise Δ λ	0	0 – 1,5	0 – 2,5				
# train original images	4	12	20				
# train images + augment.	16	48	80				
# batches	60	80	300				
# patches	7\,680	23\,040	38\,400				
# eval images	5 patterns × 5 noise levels						
# test images	**data1**	**data20**	**VibMap**				

Table 5. Parameters used to train our DL-Py models.

For the rest of the evaluations, only normalized cosine/sine transformed are used to train the networks. As the addition of phase data corrupted with speckle noise seem to return good performances, we will keep different levels of noise in our training data set.

3.4. Over-fitting and influence of the number of layers of the CNN.

The architecture was first designed for natural images denoising, where the diversity of the image contours is quite high. In the task of phase images denoising, the task is probably less diverse. We can also note that the amount of natural images is almost 20 times over the

amount of patches used in the first experiments in Table 2 realized on phase images corrupted with speckle noise. The small amont of data for training and the reduced diversity of the processed images may conduct to very specialized trained networks. If the model is too specialized for processing images closed to the ones in the training set, results on such images may be very impressive, while results on unseen images will drastically drop in performance.

We hypothesized that reducing the number of layers of the network could help not to over-fit the data, and help in the training of less specialized models. The reference network contains a first convolutional layer (3×3 and rectified linear units ReLU), $D = 16$intermediate convolutional layers ($3 \times 3 \times 64$ + batch normalization and ReLU), and a last convolutional layer ($3 \times 3 \times 64$ is used to reconstruct the output noise. We decided to train different size for the number of layer D.

3.5. Evaluation of a pre-trained network

In a second step, estimation of how better the network pre-trained on natural images with additional Gaussian noise would be. Then, adapt it to holographic phase images, or to direct use of a network trained entirely with holographic phase images.

Indeed, phase images are very specific (fringe patterns) in comparison to natural images, and speckle noise is different from Gaussian noise, mainly because it depends on the frequency and can not be modeled with additional noise.

In this section, we present results obtained while pretraining the network on natural images, then fine-tuning it on holographic phase images.

3.6. Protocole

400 images of the NATURAL database are used to pre-train the network with the best architecture obtained in the previous section, *i.e.* 4 ConvBlocks (see section "Results and discussion"). Once the network is pre-trained, a second fine-tuning stage is carried out using holographic images following the aforementioned protocol. The DL-nat-pt model corresponds to the model trained with natural images during 75 epochs which seems reasonable regarding the 50 epochs used to train the original DnCNN [8] [Computational image speckle suppression using block matching and machine learning]. Without fine-tuning, this model reaches $\Delta\phi = 0.380\ rad$ with the development set, which is not suitable at all for holographic images. The fine-tuning results are presented in the next section.

However, its adaptation to holographic images improves significantly the performances of this preliminary model. From the Table 6 we can also see that, for similar epochs, the performances reached by the pre-trained models are comparable to the one obtained without pre-training.

4. Results and discussion

4.1. Network depth and architecture

The results obtained with HOLODEEP are summarized in Table 6. To help the reader, model names explicit the different parameters : DL-Py-X-D-z with X being the maximum Δ in the training data, D stands for the depth of the model ($D = 4$ or $D = 16$) and the optional z indicates if the model has been previously trained on natural images (pt) or not.

When the training noise is $\Delta = 0$, the best results are obtained with a complex network (DL-Py-0-16,$\Delta\phi = 0.057\ rad$). However, overall, the best results are obtained with only 4 ConvBlocks and a large range of training noise (DL-Py-2.5-4, $\Delta\phi = 0.035\ rad$).

We conclude that the complexity of the network is not beneficial when the model is learnt on small noises only, probably because the network do not generalize well to other levels of noise.

Introducing noise level diversity allows reducing drastically the average phase error for all configurations. Especially the best configuration ($D = 4$ ConvBlocks) lowers $\Delta\phi$ from $0.058\ rad$ ($\Delta = 0$) to $0.035\ rad$ ($\Delta = 0 - 2.5$). This suggests that a reduced network trained with a large diversity is probably more able to generalize than a deep network trained with very few data. One point remains uncertain, we are not sure whether the improvement observed on de-noising is due to the diversity of noise, or to the larger amount of data used to train the network.

The advantage of using a smaller number of layers is that the computation time is more than twice lower.

Almost 1 day with $D = 4$ and 1,5 days with $D = 16$.

		Trained on HOLODEEP		Pre-Trained
Δ (#patch)	***D***	**16**	**4**	**4**
0 (15.3k)	model	DL-Py-0-16	DL-Py-0-4	DL-Py-0-4-pt
	BestEpoch/Max	195/200	200/200	190/200
	$\Delta\phi$	**0.057**	0.058	0.055
ine 0–1.5 (46.1k)	model	DL-Py-1.5-16	DL-Py-1.5-4	DL-Py-1.5-4-pt
	BestEpoch/Max	70/70	140/150	85/95
	$\Delta\phi$	0.042	**0.040**	0.045
ine 0–2.5 (76.8k)	model	DL-Py-2.5-16	DL-Py-2.5-4	DL-Py-2.5-4-pt
	BestEpoch/Max	40/50	90/95	50/55
	$\Delta\phi$	0.038	**0.035**	**0.048**

Table 6. Phase errors (Δ φ in rad), obtained with one iteration on HOLODEEP. Best configurations in bold font. Three training sets are used, each corresponding to a larger diversity of noise, the number of patches used to train the model in each case is given. The model names are given for each configuration. The best epoch is given relatively to the total number of epochs used to train the model.

Investigation of the results according to speckle noise level in the HOLODEEP images confirms that the higher the noise level, the higher is the error in the restored phase map.

Table 6 details values obtained during evaluation on HOLODEEP, according to their level of noise (parameter Δ) with the three best models DL-Py-0-4 (train noise level $\Delta = 0$), DL-Py-1.5-4 (train noise level $\Delta = 0 - 1.5$) and DL-Py-2.5-4 (train noise level $\Delta = 0 - 2.5$).

As aforementioned DL-Py-2.5-4 is better in average than DL-Py-0-4 on HOLODEEP. But the additional experiments show that this performance improvement is significantly more important on images with high noise level (-49% of relative reduction with $\Delta = 2.5$) than with images with low noise (-31% with $\Delta = 0$). These results underline the relevance of introducing a large diversity of patterns and noise levels during the training step if the application images to be processed also have high noise levels.

4.2. Pre-training

In order to evaluate the network, we use DATAEVAL database which contains 3 images than have not been seen during the training and validation stages. There is a small difference between the results presented in Table 1 for the development set (25 images) and the ones presented in Table 6. This is due to the fact, that in Table 6 the denoising is applied using the model extracted at the best epoch, while in Table 1 the denoising is applied using the model extracted at the last epoch. With the current framework, we are not able to selected one specific checkpoint. This the reason why, we would like to move to PyTorch framework in the future.

Table 6 shows that the pre-trained model outperforms the initial models only when a small level of noise ($\Delta = 0$) is used for fine-tuning. This leads to the conclusion that pre-training the network on natural images helps to compensate for the lack of diversity in the specific training data and also the relatively small amount of training data. This results confirms the advantage of using pre-trained models when the amount of specific target data is low [12] [On the use of self-supervised pre-trained acoustic and linguistic features for continuous speech emotion recognition].

Two hypothesis may explain the poor performances reached by the pre-trained model.

NATURAL and HOLODEEP databases differ in many points: additive Gaussian vs. multiplicative speckle noise and natural vs. wrapped phase images. Such data difference could explain the poor performances obtained with pre-training: training a network with phase images using an initialization obtained on NATURAL database does not seem worthy in the present case.

Therefore, training a network with phase data corrupted with speckle noise requires to be deeper investigated.

The second hypothesis concerns the performance of the model trained on NATURAL data. Due to cluster usage constraints, the total number of epochs to train this model is 75 epochs. It aims at getting a performing model on natural images. However, this number is higher than the 50 epochs used to train the original DnCNN model mentioned in [11] [Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising]) and the model might be too specific for natural images. As such models require a lot of resources to be trained, we did not have the opportunity to train on a higher number of epochs. However, it is worth considering this aspect.

a) Effect of iterations

In the field of speckle image denoising, images are usually decoded more than one time. Therefore, the noisy image is passed several times through the network as shown in Table 1 where 6 iterations have been used to denoise test images with BM3D and DtDWT algorithms.

Method	# iter	HOLODEEP	DATAEVAL		
		25 images	Data1	Data20	VibMap
WFT2F	1	.025	**.044**	.164	.105
DL-3	1	.041	.107	.585	.077
	3	.031	.078	.559	.074
DL-Py-1.5-4	1	.043	.092	.592	.105
	3	.045	.065	.589	.118
DL-Py-2.5-4	1	.036	.072	.591	.108
	3	.042	**.057**	.591	.121

Table 7. Results in Δ φ (rad), obtained with different methods on phase images

Δ λ – train	pre -trained	No	Yes
—	Model (epochs)		DL-nat-pt (75)
	Δ λ-AVG		.380
0	Model (epochs)	DL-Py2 (200)	DL-Py2-pt (240)
	Δ λ-AVG	.058	**.055**
0 – 1.5	Model (epochs)	DL-Py6 (140)	DL-Py6-pt (102)
	Δ λ-AVG	**.040**	.045
0 – 2.5	Model (epochs)	DL-Py10 (90)	DL-Py10-pt (36)
	Δ λ-AVG	**.035**	.048

Table 8. Results in Δ φ (rad), obtained with only one iteration on the 25 images from the HOLODEEP database. Results for pre-trained models and trained models obtained on the best epoch.

4.3. Evaluation on target images

Method	HOLODEEP	DATAEVAL		
	25 Images	Test1	Test2	Test3
WFT2F	0.026	0.044	0.163	0.105
DL-3	0.041	0.107	0.585	0.105
DL-Py-0-4	0.058	0.142	0.629	0.117
DL-Py-0-4-pt	0.055	0.146	0.629	0.105
DL-Py-1.5-4	0.040	0.095	0.593	0.103
DL-Py-1.5-4-pt	0.045	0.112	0.609	0.111
DL-Py-2.5-4	0.035	0.072	0.597	0.109
DL-Py-2.5-4-pt	0.048	0.097	0.660	0.134

Table 9. Δ φ (rad) obtained on HOLODEEP database (in average) and individual images from DATAEVAL with one iteration. Best epochs for pre-trained and trained models on HOLODEEP validation database

Table 9 summarizes the performances obtained with development and validation images.

DL-Py-2.5-4 performs better on the training data HOLODEEP ($\Delta\phi = 0.035\ rad$) and on `Test1` ($\Delta\phi = 0.072\ rad$).

However, the performance is degraded when testing with `Test2`which has a high level of noise and with `Test3`, which is the phase image from vibration experiments. No clear answer can be given here. DL-Py-2.5-4 model is trained on a large number of data and noise, thus it should be able to deal with high level of noise.

However, by construction of HOLODEEP database, there are few redundancies in the phase images, and `Test1` appears relatively similar to those in HOLODEEP, while `Test2`and `Test3` are not.

Therefore the model might not be able to generalize easily to unseen images.

Another hypothesis is that the structure of the model implies additive noise, what could be relevant for small SNR, but not for high SNR where speckle noise is clearly multiplicative.

One can notice that while the model trained on phase images with a large diversity of noise (DL-Py-2.5-4) reaches the best performance on the development set ($\Delta\phi = 0.035\ rad$), it does not generalize well on unseen images from DATAEVAL. `test1` image is very similar to those contained in HOLODEEP.

The model that best generalize on the `test2` and `Test3` is the one trained on a medium range of speckle noise (DL-Py-1.5-4). This model is even able to outperform the baseline WFT2F on the experimental vibration map `Test3` phase image. Figure 10 (cf. Figure 10 : Noise-free (left), noisy (middle) and de-noised (right) phase images from DATAEVAL. De-noising is performed with DL-Py-1.5-4 mode. p 24) shows how these images from DATAEVAL are de-noised by the best model. Therefore, the proposed networks are able to reach interesting performances in comparison to WFT2F, especially for some specific experimental images. These networks have the advantage of being faster to train than DL-3 network as they only contains 4 ConvBlocks.

Regarding pre-trained models, it seems that they are not able to generalize on unseen images except DL-Py-0-4-pt which gets $\Delta\phi = 0.105\ rad$ with `Test3`. Additional experiments show that models trained with more epochs can improve the performances on `Test1`, but degrade on `Test2` and `Test3`.

III.Conclusion

This paper discusses on holographic phase images de-noising and presents an alternative approach that is specific for speckle noise. The results show that a pre-trained model is not useful except if the amount and diversity of simulated data are low. In this case, the pre-training compensate the lack of data. Experiments also demonstrate that the use of very deep networks is not necessary, and that the use of 4 ConvBlocks yields reliable performances in comparison to WFT2F. Reduced networks have also the advantage of being faster to train.

This study also addresses the issue of the generalization power of the networks. It appears that WFT2F remains the best algorithms for phase images with high level of noise (Test2). However, the best model is able to outperform the baseline of WFT2F with experimental data (Test3).

The poor performance of DL-Py models with phase image with a high level of noise may be related from the additive hypothesis implemented in the network itself. A multiplicative model will be investigated in the future.

Further work intends to improve speckle de-noising by combining the advantages of the two approaches following preliminary works on the addition of a noise estimator [25] [An iterative scheme based on deep learning combined with input noise estimator for phase data processing in digital holographic interferometry].

Other data augmentation functions will be implemented in order to increase the amount of training data.

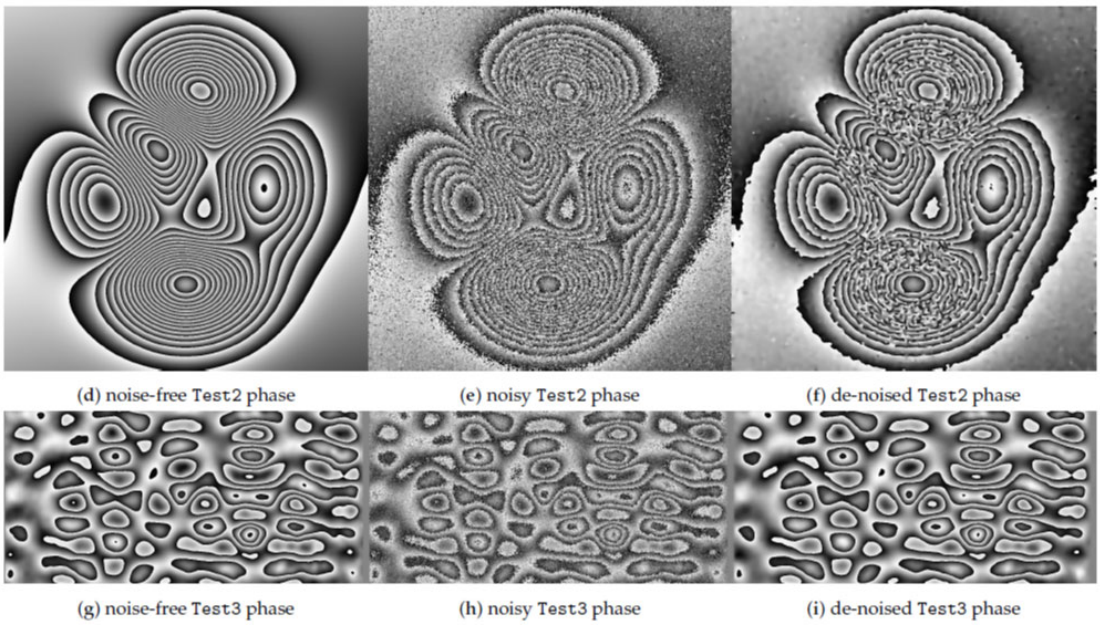
In addition, the construction of a new database with an increased diversity of fringe images would be of interest to train the networks with a high diversity of patterns.

IV.Acknowledgment

We thank the LIUM for the access to the cluster.
The research work has no external funding.
We thank LIUM for authorization access to the GPU cluster.
The authors declare no conflict of interest.

Ressources annexes

- Figure 10 : Noise-free (left), noisy (middle) and de-noised (right) phase images from DATAEVAL. De-noising is performed with DL-Py-1.5-4 mode.



Bibliographie

[Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising] K. Zhang and W. Zuo and Y. Chen and D. Meng and L. Zhang , *Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising, IEEE Transactions on Image Processing*, 2017

[Computational de-noising based on deep learning for phase data in digital holographic interferometry] Montresor,S. and Tahon,M. and Laurent,A. and Picart,P., *Computational de-noising based on deep learning for phase data in digital holographic interferometry, APL Photonics*, 2020

[Deep residual learning for image recognition] K. He and X. Zhang and S. Ren and J. Sun, *Deep residual learning for image recognition, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016

[Depth selection for deep ReLU nets in feature extraction and generalization] Han, Zhi and Yu, Siquan and Lin, Shao-Bo and Zhou, Ding-Xuan, *Depth selection for deep ReLU nets in feature extraction and generalization, IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 2020

[Digital Holography] P. Picart and J.C. Li,John Wiley \& Sons, Ltd, *Digital holography*, 2012

[Generative adversarial nets] Goodfellow, I. J. and Pouget-Abadie, J. and Mirza, M. and Xu, B. and Warde-Farley, D. and Ozair, S. and Courville, A. and Bengio, Y, *Generative adversarial nets, Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS) - Volume 2*, MIT Press, 2014

[High-speed holographic metrology: principle, limitations, and application to vibroacoustics of structures] Poittevin, Julien and Gautier, François and Pézerat, Charles and Picart, Pascal, *High-speed holographic metrology: principle, limitations, and application to vibroacoustics of structures,Optical Engineering*, International Society for Optics and Photonics, 2016

[Improvement of accuracy in digital holography by use of multiple holograms] Baumbach, Torsten and Kolenovic, Ervin and Kebbel, Volker and Jüptner, Werner, *Improvement of accuracy in digital holography by use of multiple holograms, Applied Optics*, 2006.

[Natural image denoising with convolutional networks] Jain, V. and Seung, S., *Natural image denoising with convolutional networks, Advances in Neural Information Processing Systems*, D. Koller and D. Schuurmans and Y. Bengio and L. Bottou, 2008

[On the use of deep learning for computational imaging] Barbastathis, G. and Ozcan, A. and Situ, G., *On the use of deep learning for computational imaging, Optica*, 2019

[On the use of self-supervised pre-trained acoustic and linguistic features for continuous speech emotion recognition] Macary, M.; Tahon, M.; Estève, Y.; Rousseau, A., *On the use of self-supervised pre-trained acoustic and linguistic features for continuous speech emotion recognition in 2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021

[The effectiveness of data augmentation in image classification using deep learning] Luis Perez and Jason Wang, *The effectiveness of data augmentation in image classification using deep learning, arXiv:1712.04621*, 2017

[Two-dimensional phase unwrapping: theory, algorithms, and software] Ghiglia, Dennis C. and Pritt, Mark D, *Two-dimensional phase unwrapping: theory, algorithms, and software*, Wiley, 1998

[Very deep convolutional networks for large-scale image recognition] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition, 3rd International Conference on Learning Representations, ICLR 2015*, 2015

[Visualization of travelling waves propagating in a plate equipped with {2D ABH} using wide-field holographic vibrometry] Lagny, L. and Secail-Geraud, M. and Le Meur, J. and Montresor, S. and Heggarty, K. and Pezerat, C. and Picart, P.,*Visualization of travelling waves propagating in a plate equipped with {2D ABH} using wide-field holographic vibrometry, Journal of Sound and Vibration*, 2019

[Windowed Fourier transform for fringe pattern analysis: theoretical analyse] Q. Kemao and H. Wang and W. Gao, *Windowed Fourier transform for fringe pattern analysis: theoretical analyse, Applied Optics*, OSA, 2008

Webographie

[An iterative scheme based on deep learning combined with input noise estimator for phase data processing in digital holographic interferometry] S. Montresor and M. Tahon and A. Laurent and P. Picart, *An iterative scheme based on deep learning combined with input noise estimator for phase data processing in digital holographic interferometry, Imaging and Applied Optics Congress*, Optical Society of America, 2020,

[Computational image speckle suppression using block matching and machine learning] T. Zeng and H. K.-H. So and E. Y. Lam, *Computational image speckle suppression using block matching and machine learning, Applied Optics*, OSA, 2019,

[Deep Learning] I. Goodfellow and Y. Bengio and A. Courville, *Deep Learning*, MIT Press, 2016,

[Error analysis for noise reduction in 3D deformation measurement with digital color holography] S. Montresor and P. Picart and O. Sakharuk and L. Muravsky, *Error analysis for noise reduction in 3D deformation measurement with digital color holography, Journal of the Optical Society of America B*, OSA, 2017,

[Experimental and theoretical investigation of the pixel saturation effect in digital holography] Picart, Pascal and Tankam, Patrice and Song, Qinghe, *Experimental and theoretical investigation of the pixel saturation effect in digital holography, Journal of the Optical Society of America A*, 2011

[Quality assessment of combined quantization-shot-noise-induced decorrelation noise in high-speed digital holographic metrology] Poittevin, Julien and Picart, Pascal and Gautier, François and Pezerat, Charles, *Quality assessment of combined quantization-shot-noise-induced decorrelation noise in high-speed digital holographic metrology, Optical Express*, Optical Society of America, 2015.

[Quantitative appraisal for noise reduction in digital holographic phase imaging] S. Montrésor and P. Picart, *Quantitative appraisal for noise reduction in digital holographic phase imaging, Optics Express*, OSA, 2016

[Refocus criterion based on maximization of the coherence factor in digital three-wavelength holographic interferometry] Picart, Pascal and Montresor, Silvio and Sakharuk, Oleksandr and Muravsky, Leonid, *Refocus criterion based on maximization of the coherence factor in digital three-wavelength holographic interferometry, Optics Letters*, Optical Society of America, 2017

[Speckle noise reduction for digital holographic images using multi-scale convolutional neural networks] W. Jeon and W. Jeong and K. Son and H. Yang, *Speckle noise reduction for digital holographic images using multi-scale convolutional neural networks, Opt. Lett.*, OSA, 2018,

[Speckle noise reduction in optical coherence tomography images based on edge-sensitive cGAN] Y. Ma and X. Chen and W. Zhu and X. Cheng and D. Xiang and F. Sh, *Speckle noise reduction in optical coherence tomography images based on edge-sensitive cGAN, Biomedical Optics Express*, OSA, 2018,