UCLA
UCLA Electronic Theses and Dissertations

Title
Task-specific programming of optical diffraction using deep learning-designed surfaces

Permalink
https://escholarship.org/uc/item/98b4k7nt

Author
Rahman, Md Sadman Sakib

Publication Date
2024

Supplemental Material
https://escholarship.org/uc/item/98b4k7nt#supplemental

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

UNIVERSITY OF CALIFORNIA

Los Angeles

Task-specific programming of optical diffraction using deep learning-designed surfaces

A dissertation submitted in partial satisfaction of the requirements for the degree Doctor of Philosophy in Electrical and Computer Engineering

by

Md Sadman Sakib Rahman

2024

© Copyright by

Md Sadman Sakib Rahman

2024

ABSTRACT OF THE DISSERTATION

Task-specific programming of optical diffraction

using deep learning-designed surfaces

by

Md Sadman Sakib Rahman

Doctor of Philosophy in Electrical and Computer Engineering

University of California, Los Angeles, 2024

Professor Aydogan Ozcan, Chair

In recent years, the perceived stagnation in the growth of electronic computing has fueled the search for alternative computing platforms. At the same time, the rise and success of deep neural networks have sparked renewed interest in neuromorphic computing for machine learning. Among these emerging platforms, optics stands out as a promising candidate for energy-efficient and low-latency computing. While much of the focus in optical computing has been on integrated optics, the recent success of diffractive deep neural networks or diffractive networks has also revived interest in diffraction-based free-space computing. Diffractive networks exploit programmed diffraction of light by spatially engineered surfaces for passive all-optical information processing at the speed of light propagation. These surfaces are digitally optimized through deep learning, enabling the input wave's diffraction to be tailored for a specific task by leveraging data that encapsulates the input-output relationship. Once the digital optimization is complete, these

surfaces are fabricated and integrated into an all-optical ‘computer’, driven by passive light-matter interactions. Diffractive networks enable all-optical classification of input objects and support a diverse range of computational imaging tasks. This thesis presents methods to enhance the inference capabilities of diffractive networks and explores novel applications of this optical computing framework. First, the accuracy of diffractive systems for object classification is improved through ensemble learning, which involves combining the outputs of multiple diffractive networks. Novel strategies to diversify and assort the ensemble members are introduced, leading to significant performance gains. Then, time-lapse image classification is explored to improve the accuracy of standalone diffractive networks. By leveraging the natural jitter of the input object relative to the diffractive network, this method achieves, with a single diffractive network, accuracy comparable to that of an ensemble of 30 time-static networks. Next, the application of diffractive networks for all-optical hologram processing is demonstrated. This approach bypasses high-latency and power-intensive digital processing for removing twin-image artifacts, enabling passive inline hologram reconstruction at the speed of light propagation. Finally, a novel optical communication scheme is introduced, which utilizes electronic encoding and diffractive decoding to transfer images around arbitrarily shaped opaque occlusions of zero light transmittance. The joint training of the electronic encoder and the diffractive decoder allows efficient encoding of information in the transmitted wavefront, enabling the joint system to evade occlusions and transfer images even when the direct rays between the transmitter and the receiver are blocked. In summary, this thesis makes significant progress in the application of diffractive networks, while also laying the groundwork for future advancements in this field.

The dissertation of Md Sadman Sakib Rahman is approved.

Achuta Kadambi

Mona Jarrahi

Sam Emaminejad

Aydogan Ozcan, Committee Chair

University of California, Los Angeles

2024

Table of contents

Chapter 1 Introduction 1

1.1 Background 1

1.2 Contribution of this thesis 5

1.3 Organization of this dissertation 6

Chapter 2 Ensemble Learning of Diffractive Optical Networks 8

2.1 Introduction 9

2.2 Results 12

2.3 Discussion 31

2.4 Materials and Methods 37

2.4.1 Implementation of D^2NNs 37

2.4.2 Feature engineering of diffractive networks 38

2.4.3 Training details 39

2.3.4 Ensemble pruning 42

Chapter 3 Time-lapse Image Classification using a Diffractive Network 46

3.1 Introduction 47

3.2 Results 50

3.3 Discussion 62

3.4 Materials and Methods 77

3.4.1 Forward model 77

3.4.2 Numerical implementation 79

3.4.3 Training 80

3.4.4 Ensemble Learning of Time-Lapse Diffractive Networks 82

Chapter 4 Computer-free, All-optical Reconstruction of Holograms using Diffractive Networks 83

4.1 Introduction 84

4.2 Results 88

4.3 Discussion 100

4.4 Methods 103

4.4.1 Numerical Model of Diffractive Networks 103

4.4.2 Network architecture 105

4.4.3 Training 105

Chapter 5 Learning Diffractive Optical Communication Around Arbitrary Opaque Occlusions 109

5.1 Introduction 110

5.2 Results 112

5.2.1 Numerical analysis of diffractive optical communication around opaque occlusions 117

5.2.2 Impact of phase bit depth on performance 124

5.2.3 Impact of misalignments on performance 127

5.2.4 Output power efficiency 133

5.2.5 Occlusion shape 135

5.2.6 Experimental validation 137

5.2.7 Dynamic occlusions 140

5.3 Discussion 143
5.4 Methods 152
5.4.1 Model 152
5.4.2 Training 154
5.4.3 Experimental design 157
5.4.4 Terahertz experimental setup 158
5.4.5 Dynamic occlusion modeling 159
Chapter 6 Conclusion 160
References 162

List of Figures

Figure 1.1 Diffractive network. 3

Figure 2.1 Schematic diagram of the ensemble diffractive network system. 12

Figure 2.2 Inference accuracy of D^2NN ensembles as a function of N_{max} and N. 18

Figure 2.3 An ensemble of N=14 D^2NNs achieves a blind classification accuracy of 61.21% on CIFAR-10 test dataset. 21

Figure 2.4 An ensemble of N=12 D^2NNs achieves a blind classification accuracy of 60.29% on CIFAR-10 test dataset. 24

Figure 2.5 A D^2NN ensemble consisting of only phase-encoded-input networks (N=14) achieves a blind classification accuracy of 60.65% on CIFAR-10 test dataset. 26

Figure 2.6 A D^2NN ensemble consisting of only phase-encoded-input networks (N=12) achieves a blind classification accuracy of 60.43% on CIFAR-10 test dataset. 28

Figure 2.7 Flow chart of the ensemble pruning process. 41

Figure 3.1 Time-lapse image classification using a D^2NN. 50

Figure 3.2 Dependence of the time-lapse image classification D^2NN performance on the hyperparameters of the input plane. 52

Figure 3.3 Comparison between a time-static and a time-lapse diffractive image classifier. 55

Figure 3.4 Performance of a time-lapse diffractive network without trainable detector exponents (i.e., $\gamma_{c,\pm} = 1$). 57

Figure 3.5 Parameters of the diffractive models of Fig. 3.3 and Fig. 3.4. 58

Figure 3.6 Impact of the lateral shifts. 60

Figure 3.7 Example object test trajectories during time-lapse inference and the corresponding blind testing accuracies on CIFAR-10 test images for the N_{tr} = 25 time-lapse diffractive network of Fig. 3.6c. 64

Figure 3.8 Investigation of the individual class accuracies. 67

Figure 3.9 List of the architectures of the 20 time-lapse diffractive networks trained for ensemble learning. 71

Figure 3.10 The trained diffractive layers for the 20 time-lapse diffractive networks used for ensemble learning. 72

Figure 3.11 An ensemble of 20 time-lapse diffractive networks that attains 65.05% blind testing accuracy on CIFAR-10 test images. 73

Figure 3.12 An ensemble of 20 time-lapse diffractive networks that attains 64.84% blind testing accuracy on CIFAR-10 test images without utilizing trainable class-specific weights for ensemble class-scores. 74

Figure 3.13 An ensemble of 10 time-lapse diffractive networks that attains 64.21% blind testing accuracy on CIFAR-10 test images utilizing N=11 object displacements..... 75

Figure 3.14 An ensemble of 10 time-lapse diffractive networks that attains 63.81% blind testing accuracy on CIFAR-10 test images utilizing N=8 object displacements and no trainable class-specific weights for ensemble class-score. 76

Figure 4.1 Hologram recording and reconstruction scheme for the proposed all-optical holographic reconstruction method..... 87

Figure 4.2 Performance of all-optical hologram reconstruction using diffractive networks for handwritten digits and letters from the English alphabet (a) and fashion products (b). 90

Figure 4.3 All-optical hologram reconstructions by a diffractive network for different object scaling factors. 92
Figure 4.4 All-optical hologram reconstructions by a diffractive network for different rotation angles of the recorded holograms. 92
Figure 4.5 Quantification of all-optical holographic image reconstruction resolution. 94
Figure 4.6 Robustness of all-optical diffractive network reconstructions against the hologram recording distance variations. 96
Figure 4.7 Diffraction efficiency improvement of all-optical holographic reconstructions performed by diffractive networks. 99
Figure 4.8 Examples of training images. 107
Figure 5.1 Schematic of the optical communication framework around fully opaque occlusions using electronic encoding and diffractive all-optical decoding. 114
Figure 5.2 The critical occlusion width w_c. 117
Figure 5.3 Generalization of trained encoder-decoder pairs to previously unseen handwritten digit objects. 119
Figure 5.4 Quantification of the performance of encoder-decoder pairs with different numbers of decoder layers (L) trained for increasing occlusion widths (w_o) in terms of (a) PSNR and (b) SSIM between the diffractive decoder outputs and the ground-truth messages. 121
Figure 5.5 Same as Fig. 5.3a, except that these results reflect external generalizations on object types different from those used during the training. 122
Figure 5.6 Output resolution of diffractive decoders corresponding to $L = 1$, $L = 3$, and $L = 5$ designs trained for different occlusion widths (w_o). 123

Figure 5.7 Effect of the phase bit depth of the encoded object and the diffractive layer features on the performance of trained encoder-decoder pairs..... 126
Figure 5.8 Impact of lateral random misalignments of the layers of the diffractive decoder on the performance of our framework for communication around opaque occlusions. 128
Figure 5.9 Impact of axial random misalignments of the layers of the diffractive decoder on the performance of our framework for communication around opaque occlusions. 130
Figure 5.10 Impact of lateral random displacements of the opaque occlusion on the performance of our framework for communication around opaque occlusions. 131
Figure 5.11 Impact of axial random displacements of the opaque occlusion on the performance of our framework for communication around opaque occlusions..... 132
Figure 5.12 Output power efficiency of the electronic encoding-diffractive decoding scheme for optical communication around fully opaque occlusions..... 134
Figure 5.13 Performance of encoder-decoder pairs trained for different opaque occlusion shapes. 136
Figure 5.14 Average SSIM values of the diffractive decoder outputs for the four designs of Fig. 5.13..... 137
Figure 5.15 Experimental results with an $L = 1$ design for an occlusion width of $w_o = 32\lambda \approx 0.5w_c$ operating at a wavelength of $\lambda = 0.75\text{mm}$..... 139
Figure 5.16 10 Experimental results for communication around randomly changing opaque occlusions using the same electronic encoder and diffractive decoder; $\lambda = 0.75\text{mm}$. 142
Figure 5.17 Transfer learning of the CNN encoder at the transmitter, while the diffractive decoder at the receiver remains unchanged. 145

Figure 5.18 Performance of the electronic encoder and diffractive decoder framework for communication around opaque occlusions for large values of the axial distance d_{tl} between the transmitter and the receiver apertures. 148
Figure 5.19 Depiction of alternative schemes to diffractive all-optical decoding for communication around opaque occlusions. 150
Figure 5.20 Comparative performance analysis of different schemes illustrated in Fig. 5.19. .. 151
Figure 5.21 Examples of the custom-prepared training images. 156

List of Tables

Table 2.1: The initial network pool containing 1252 individually trained D^2NNs. 15

Table 2.2: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with a maximum allowed ensemble size of $N_{max}=14$. 20

Table 2.3: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with only phase encoded input D^2NNs and a maximum allowed ensemble size of $N_{max}=14$. 25

Table 2.4: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with a maximum allowed ensemble size of $N_{max}=14$, where in the pruning process trainable weights are also assigned to the individual detector signals and are *simultaneously* optimized with the class score weights (referred to as the weight assignment scheme (1)). 30

Table 2.5: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with a maximum allowed ensemble size of $N_{max}=14$, where in the pruning process trainable weights are also assigned to the detector signals and are *alternatively* optimized with the class score weights (referred to as the weight assignment scheme (2)). 31

Table 3.1: Dependence of the performance of time-static diffractive networks on the input-aperture size. 54

Table 3.2: Dependence of the performance of time-static and time-lapse diffractive networks on the trainability of $\gamma c, \pm$. 56

ACKNOWLEDGEMENTS

First and foremost, I want to express my deepest gratitude to Prof. Aydogan Ozcan. I consider myself incredibly fortunate to have had the opportunity to work under his guidance throughout my PhD journey. His positive energy, unique vision, creativity, and exceptional communication skills have profoundly influenced and inspired my research. At the same time, his unwavering support and mentorship have been invaluable for me.

I am also grateful to Prof. Mona Jarrahi for the opportunity to collaborate across disciplines and for serving on my committee. Additionally, I would like to extend my heartfelt thanks to my doctoral committee members, Prof. Achuta Kadambi and Prof. Sam Emaminejad, for their generous support and encouragement throughout my PhD.

It has been a privilege to be part of the Ozcan lab, and I feel fortunate to have collaborated with and learned from a cohort of brilliant minds: Derek Tseng, Dr. Yair Rivenson, Dr. Muhammed Veli, Dr. Deniz Mengu, Dr. Yi Luo, Dr. Jingxi Li, Dr. Bijie Bai, Dr. Onur Kulce, Dr. Jingtian Hu, Dr. Hatice Ceylan Koydemir, Dr. Aniruddha Ray, Dr. Hyou-arm Joung, Dr. Zach Ballard, Dr. Calvin Brown, Dr. Hongda Wang, Dr. Tairan Liu, Dr. Kevin de Haan, Dr. Vishal Srivastava, Dr. Nir Pillar, Dr. Gyeo-Re Han, Dr. Merve Eryilmaz, Osama Hassan, Serhat Sahin, Artem Goncharov, Yijie Zhang, Çağatay Işıl, Luzhe Huang, Yuzhu Li, Yilin Luo, Xilin Yang, Yuhang Li, Hanlong Chen, Che-Yung Shen, Aras Fırat Ünal, Şahan Yoruç Selçuk, Yiyang Wu, Yuntian Wang, Guangdong Ma, Kun Liao and Emir Arda Deger. I am also grateful to my collaborators in Professor Jarrahi's group: Dr. Xurong Li, Yifan Zhao, and Tianyi Gan. Although I have made every effort to be comprehensive, it is impossible to include every individual who has assisted me. I apologize if I have inadvertently left anyone out. Any omission reflects a lapse in my memory, not a lack of appreciation for their support.

Finally, I want to extend my deepest thanks to my family for their steadfast support throughout this journey. Their boundless love and encouragement have been the foundation of my strength. Their sacrifices for me have played a crucial role in this accomplishment, making it as much a testament to their support as it is to my own efforts.

VITA

Md Sadman Sakib Rahman received his B.Sc. in 2014, and M.Sc. in 2017, both in Electrical and Electronic Engineering from Bangladesh University of Engineering and Technology, Dhaka. In 2018, he joined Prof. Aydogan Ozcan's Bio- and Nano-Photonics Group at the Electrical and Computer Engineering Department, UCLA, to pursue his Ph.D. His research interest lies broadly in the intersection of deep learning and optics, and currently in the use of deep learning for design of diffractive optical information processors.

SELECTED PUBLICATIONS

1. G. Ma, C.-Y. Shen, J. Li, L. Huang, C. Isil, F. O. Ardic, X. Yang, Y. Li, Y. Wang, **M. S. S. Rahman**, and A. Ozcan, "Unidirectional imaging with partially coherent light," *in submission* (2024).
2. **M. S. S. Rahman** and A. Ozcan, "Integration of Programmable Diffraction with Digital Neural Networks," ACS Photonics 11, 2906–2922 (2024).
3. J. Hu, K. Liao, N. U. Dinç, C. Gigli, B. Bai, T. Gan, X. Li, H. Chen, X. Yang, Y. Li, Ç. Işıl, **M. S. S. Rahman**, J. Li, X. Hu, M. Jarrahi, D. Psaltis, and A. Ozcan, "Subwavelength imaging using a solid-immersion diffractive optical processor," eLight 4, 8 (2024).
4. X. Yang, **M. S. S. Rahman**, B. Bai, J. Li, and A. Ozcan, "Complex-valued universal linear transformations and image encryption using spatially incoherent diffractive networks," Adv. Photonics Nexus 3, 016010 (2024).
5. **M. S. S. Rahman**, T. Gan, E. A. Deger, Ç. Işıl, M. Jarrahi, and A. Ozcan, "Learning diffractive optical communication around arbitrary opaque occlusions," Nat. Commun. 14, 6830 (2023).

6. **M. S. S. Rahman**, X. Yang, J. Li, B. Bai, and A. Ozcan, "Universal linear intensity transformations using spatially incoherent diffractive processors," Light Sci. Appl. 12, 195 (2023).
7. **M. S. S. Rahman** and A. Ozcan, "Time-Lapse Image Classification Using a Diffractive Neural Network," Adv. Intell. Syst. 5, 2200387 (2023).
8. D. Mengu, **M. S. S. Rahman**, Y. Luo, J. Li, O. Kulce, and A. Ozcan, "At the intersection of optics and deep learning: statistical inference, computing, and inverse design," Adv. Opt. Photonics 14, 209–290 (2022).
9. **M. S. S. Rahman** and A. Ozcan, "Computer-Free, All-Optical Reconstruction of Holograms Using Diffractive Networks," ACS Photonics 8, 3375–3384 (2021).
10. **M. S. S. Rahman**, J. Li, D. Mengu, Y. Rivenson, and A. Ozcan, "Ensemble learning of diffractive optical networks," Light Sci. Appl. 10, 14 (2021).

Chapter 1 Introduction

This dissertation outlines our contributions to the development of diffractive networks as a flexible, all-optical platform for information processing. In this introductory chapter, we start with a concise overview of diffractive optical networks, highlighting their fundamental principles, distinctive features and capabilities. In addition, we provide a brief account of existing demonstrations and applications of this framework. Finally, we present the key contributions of this dissertation and outline the structure of the subsequent chapters.

1.1 Background

The interplay of deep learning and optics is revolutionizing both fields, creating synergies that enhance the capabilities of each [1–3]. Deep learning, known for its robust pattern recognition and data analysis algorithms, is increasingly integrated into optical systems to enhance performance, automate design processes, and unlock new functionalities. For example, deep learning techniques are used to design optical components, such as lenses, waveguides, metasurfaces, nanophotonic structures and so on, with remarkable performance, enabling customized light manipulation for specific tasks [4–9]. Conversely, optics provides innovative hardware platforms for deep learning, where a network of optical components utilize light for computation, offering faster processing speeds and reduced energy consumption compared to traditional electronic systems [10–14]. In free-space optical computing, light beams are manipulated using optical components such as lenses, mirrors, beam splitters, and diffractive elements. These components can be arranged to perform various operations, such as Fourier transforms, convolution, and matrix multiplication, which are fundamental to many computational tasks [15]. The key advantage of free-space optical computing

is its ability to perform these operations in parallel, allowing for high-throughput data processing [16]. The principle behind free-space optical computing is to manipulate or engineer the diffraction of light, largely using diffractive components such as diffractive optical elements (DOEs) and metasurfaces [17]. These components are designed with micro or nanoscale patterns, which interact with light to control its propagation, focusing, and filtering. By adjusting the geometry, spacing, and shape of these patterns, engineers can tailor how light is diffracted—altering its direction, phase, intensity and polarization [18–23]. Advanced fabrication methods, including photolithography, 3D printing, and laser writing, enable the creation of these highly precise structures [24,25]. These innovations push the boundaries of optical systems, improving imaging resolution, enhancing data transmission, and enabling novel functionalities [26–28]. Advanced computational methods like Fourier optics and optimization algorithms are used to design diffractive patterns for deterministic tasks; however, they may fall short in addressing nondeterministic tasks, such as all-optical statistical inference.

Diffractive deep neural networks (D^2NNs) [29], also known as diffractive optical networks or diffractive networks, harness the power of data-driven deep learning for task-specific programming of optical diffraction, even when the task is nondeterministic. A diffractive optical network is formed by a series of engineered surfaces, each comprising spatially varying features of half a wavelength or larger, that link the input and output fields-of-view (FOVs) of an optical system. The forward model of a diffractive optical network device can be expressed mathematically as a sequence of complex-valued matrix operators that transform an input field into an output field. These operators are optimized using deep learning to transform a set of complex input fields (representing the input of the task) at the network's input aperture into corresponding ground truth output fields (representing the output of the task) at the output aperture.

The physical realization of this transformation occurs through the interaction of the input light with the carefully designed diffractive surfaces, as well as through free-space propagation within the network. To optimize the spatial features within these diffractive surfaces or layers, data-driven supervised learning techniques, such as error backpropagation and stochastic gradient descent [30], are employed on a digital model of the network. Once the training process is complete, which is a one-time task, these surfaces are fabricated and assembled into the physical network that performs the desired computation. The computation is performed entirely optically at the speed of light propagation through a compact optical volume, typically spanning less than a few hundreds of the operation wavelength in the axial direction [1].

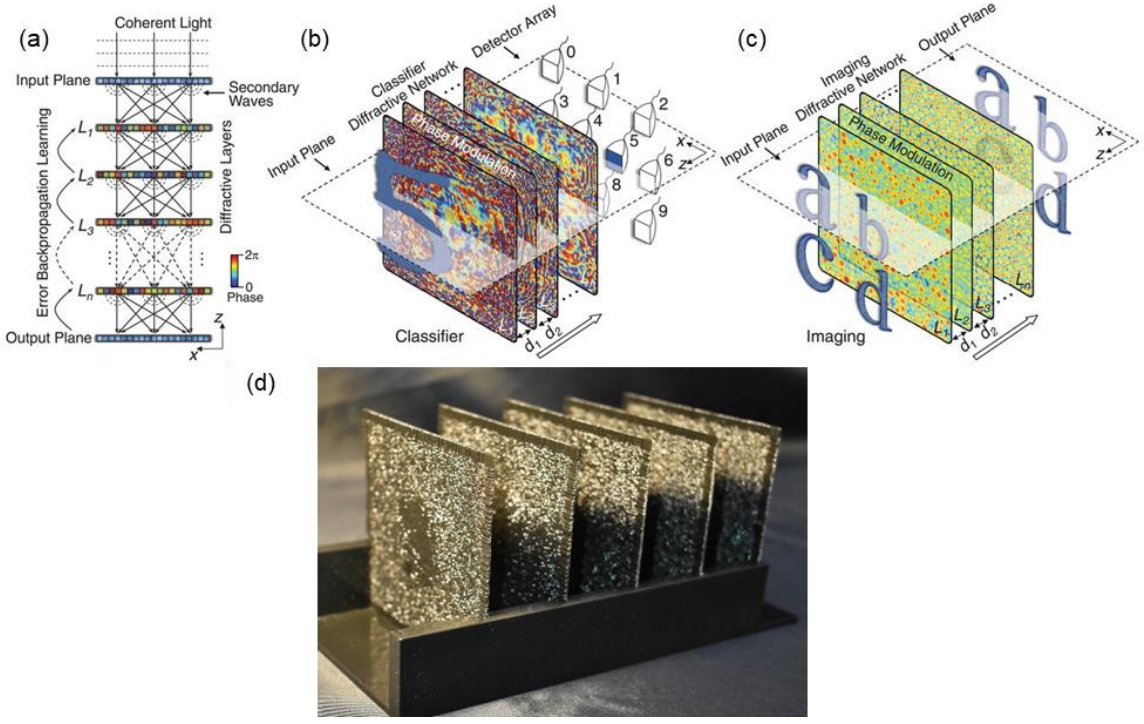


Figure 1.1 Diffractive network. (a) Schematic representation of a diffractive network, which comprises multiple spatially engineered transmissive surfaces. Using deep learning, the local transmission coefficients of each surface can be optimized to collectively perform a desired transformation of light field between the input and the output planes (apertures). Following this

one-time digital optimization, the diffractive surfaces are fabricated and assembled to perform the learned transformation all-optically at the speed of light propagation through passive interaction of light with these surfaces. (b) Cartoon of a diffractive network for all-optical classification of handwritten digits (c) Cartoon of a diffractive network for diffraction-limited imaging. (d) A classifier diffractive network for terahertz operation fabricated using 3D-printing. Reprinted from Ref. [29].

By cascading multiple surfaces in a deeper architecture, diffractive network achieve enhanced processing capabilities, improved inference accuracy, and higher output diffraction efficiency compared to a single surface with the same number of degrees of freedom [31–33]. Despite relying on linear optical materials, diffractive networks can approximate nonlinear transformations [34–37] and have been successfully applied to various tasks, including all-optical statistical inference, image encryption, phase recovery, phase conjugation, and computational imaging and sensing without a digital computer [38,39,31,40–43,34,44–47,35,48–51,36,52,17]. With sufficient degrees of freedom (in the form of independent optimizable features), diffractive networks can offer universal linear processing of both spatially coherent and incoherent light, while also supporting broadband operation and wavelength multiplexing [33,53,54,46,55–57]. Moreover, even with isotropic materials forming the constituent surfaces, diffractive networks equipped with a static grid of polarizer array can be optimized for universal polarization transformations, accurately approximating spatially encoded polarization scattering matrices at the diffraction limit of light [58]. In short, diffractive networks can access and process the input amplitude, phase, polarization and spectral information directly, without necessitating any specialized hardware or preprocessing (e.g., for phase retrieval).

1.2 Contribution of this thesis

This thesis makes significant contributions in advancing the utility of diffractive networks. It explores strategies to improve the performance of diffractive networks on optical inference tasks while also showcasing novel applications of these networks. For example, it addresses the performance issue of diffractive networks on the classification of challenging datasets such as CIFAR-10 beyond simpler objects like hand-written digits and fashion products. It introduces ensemble learning to significantly improve performance on such challenging tasks by utilizing a collection of networks that are trained diversely and assorted using a novel algorithm. Following this, the thesis goes on to explore time-lapse image classification to achieve ensemble-level performance using a single diffractive network. This research also paves the way for all-optical processing of spatiotemporal information with diffractive networks. Furthermore, the thesis reports the application of diffractive networks to twin-image free reconstruction of inline holograms, a problem conventionally tackled using digital holography. Such all-optical removal of twin-image artifact with diffractive networks avoids high-latency, power-consuming digital processing and comes with additional flexibility such as extended depth-of-field at the hologram recording distance, tuning of diffraction efficiency and so on. Lastly, the thesis presents the utilization of diffractive networks as decoders in tandem with digital encoders in a hybrid architecture for transferring images around arbitrarily shaped zero-transmittance occlusions. These contributions collectively propel diffractive networks as a highly flexible and efficient platform for diverse applications, while also providing a solid groundwork for future breakthroughs in free-space diffraction-based optical computing and information processing.

1.3 Organization of this dissertation

This dissertation is organized as follows:

Chapter 2 delves into enhancing the performance of diffractive networks at optical inference through feature engineering and ensemble learning. It discusses the improvement of image classification accuracy by assembling a large number of D^2NNs, diversely trained with various passive input filters and object encoding. Then a novel pruning algorithm is used to limit the number of D^2NNs in the optimized ensemble. This approach led to blind testing accuracies of 61.14±0.23% and 62.13±0.05% on grayscale CIFAR-10 images with ensembles of 14 and 30 D^2NNs, respectively, marking over a 16% improvement compared to the average performance of individual D^2NNs, and representing the highest accuracy achieved to date by time-static diffractive optical networks on CIFAR-10. The potential of the framework and challenges to its practical implementation are also discussed.

Chapter 3 introduces a novel 'time-lapse' image classification method using a stand-alone diffractive optical network, which greatly improves its inference and generalization capabilities. In this approach, the object and the diffractive network move laterally, relative to each other, during the detector integration time, enhancing the information available to the network. Inspired by pixel super-resolution techniques that use object shifts to boost image resolution, the exploitation of similar displacements for time-lapse classification achieves a record 62.03% accuracy on grayscale CIFAR-10 images with a single diffractive optical network. This advancement not only enhances the performance of diffractive networks but also opens new possibilities for all-optical processing of spatiotemporal data.

Chapter 4 constitutes a numerical exploration of twin-image free all-optical reconstruction of inline holograms with diffractive networks. A passive diffractive network that axially spans a few hundred wavelengths allows reconstruction of images from intensity holograms without the need for digital computation or external power. Such all-optical twin-image removal generalizes well to new examples, reconstructs images at the speed of light propagation, and improves both diffraction efficiency and depth-of-field at the hologram recording distance. This demonstration bears promise for significant developments in holographic imaging and display, offering instant, computer-free image reconstruction.

Chapter 5 presents a novel method for direct optical communication around zero-transmittance occlusions, combining electronic encoding at the transmitter and all-optical diffractive decoding at the receiver. In this approach, an electronic neural network encodes the message to bypass the opaque obstacle. The encoded optical message is transmitted and scattered by the occlusion, which is then all-optically decoded by a jointly trained diffractive network. The all-optical decoder reconstructs the encoded information passively at the speed of light propagation within a thin diffractive volume. The chapter goes on to discuss experimental demonstration and address issues such as misalignment-resilience, power-efficiency, and adaptability to dynamic changes in occlusion size or shape without retraining. This technique is versatile and applicable across various electromagnetic spectrum regions, making it suitable for high-data-rate free-space communication in scenarios where direct communication channels are obstructed.

Finally, Chapter 6 concludes the dissertation by summarizing the key contributions and offering perspectives.

Chapter 2 Ensemble Learning of Diffractive Optical Networks

In this chapter, I introduce ensemble learning and feature engineering to enhance the performance of diffractive optical network-based systems. Parts of this chapter have been published in the peer-reviewed research article "M. S. S. Rahman, J. Li, D. Mengu, Y. Rivenson, and A. Ozcan, 'Ensemble learning of diffractive optical networks,' Light: Science & Applications **10**, 14 (2021)," (Licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/).

A plethora of research advances have emerged in the fields of optics and photonics that benefit from harnessing the power of machine learning. Specifically, there has been a revival of interest in optical computing hardware, due to its potential advantages for machine learning tasks in terms of parallelization, power efficiency and computation speed. Diffractive Deep Neural Networks (D^2NNs) form such an optical computing framework, which benefits from deep learning-based design of successive diffractive layers to all-optically process information as the input light diffracts through these passive layers. D^2NNs have demonstrated success in various tasks, including e.g., object classification, spectral-encoding of information, optical pulse shaping and imaging, among others. Here, we significantly improve the inference performance of diffractive optical networks using feature engineering and ensemble learning. After independently training a total of 1252 D^2NNs that were diversely engineered with a variety of passive input filters, we applied a pruning algorithm to select an optimized ensemble of D^2NNs that collectively improve their image classification accuracy. Through this pruning, we numerically demonstrated that ensembles of N=14 and N=30 D^2NNs achieve blind testing accuracies of 61.14±0.23% and 62.13±0.05%, respectively, on the classification of CIFAR-10 test images, providing an inference improvement of >16% compared to the average performance of the individual D^2NNs within each

ensemble. These results constitute the highest inference accuracies achieved to date by any diffractive optical neural network design on the same dataset and might provide a significant leapfrog to extend the application space of diffractive optical image classification and machine vision systems.

2.1 Introduction

Recent years have witnessed the emergence of deep learning [30], which has facilitated powerful solutions to an array of intricate problems in artificial intelligence, including e.g., image classification [59,60], object detection [61], natural language processing [62], speech processing [63], bioinformatics [64], optical microscopy [65,66], holography [67–69], sensing [70] and many more [71]. Deep learning has become particularly popular because of the recent advances in the development of dedicated computing hardware and the availability of large amounts of data for training of deep neural networks. Algorithms such as stochastic gradient descent and error back propagation enable deep neural networks to learn the mapping between an input and the target output distribution by processing a large number of examples. Motivated by this major success enabled by deep learning, there has also been a revival of interest in optical computing [72,11,73,38,74–83], which has some important and appealing features such as e.g., (1) parallelism provided by optics/photonics systems, (2) potentially improved power efficiency through e.g., passive and/or low-loss optical interactions, and (3) minimal latency.

As a recent example of an entirely passive optical computing system, Diffractive Deep Neural Networks (D^2NN) [38,78,80,39,53,84,54,85,32] have been demonstrated to perform all-optical inference and image classification through the modulation of input optical waves by successive

diffractive surfaces that are trained through deep learning methods, e.g., stochastic gradient-descent and error-backpropagation. Earlier generations of these diffractive networks achieved >98% blind testing accuracies in classification of handwritten digits (MNIST) that are encoded in the amplitude or phase channels of the input optical fields, and were experimentally demonstrated using terahertz wavelengths along with 3D-printing of the resulting diffractive layers/surfaces that form a physical network. In a D^2NN that is fabricated with linear materials, where nonlinear optical processes including surface nonlinearities are negligible, the only form of nonlinearity in the forward optical model occurs at the opto-electronic detector plane. Without the use of any non-linear activation function, D^2NN framework still exhibits depth feature as its statistical inference and generalization capabilities improve with additional diffractive layers, which was demonstrated both empirically [80,86] and theoretically [32]. The same diffractive processing framework of D^2NNs has also been utilized to design deterministic optical components for e.g., ultra-short pulse shaping, spectral filtering and wavelength division multiplexing [53,54].

To further improve the inference capabilities of optical computing hardware, coupling of diffractive optical systems with jointly-trained electronic neural networks that form opto-electronic hybrid systems has also been reported [74,39,80], where the front-end is optical/diffractive and the back-end is all-electronic. Despite all this progress, there is still significant room for further improvements in diffractive processing of optical information. Here we demonstrate major advances in the optical inference and generalization capabilities of D^2NN framework by feature engineering and ensemble learning over multiple independently trained diffractive networks, where we exploit parallel processing of optical information. To create this advancement, first we focused on diversifying the base D^2NN models by manipulating their

training inputs by means of spatial feature engineering. In this approach, the input fields are filtered either in the object space or in the Fourier space by introducing an assortment of curated passive filters, before the diffractive networks (see Fig. 2.1). Following the individual training of 1252 uniquely different D^2NNs with various features, we used an iterative pruning strategy to obtain ensembles of D^2NNs that work in parallel to improve the final classification accuracy by combining the decisions of the individual diffractive classifiers. Based on this feature engineering and iterative pruning strategy, we numerically achieved blind testing accuracies of 61.14±0.23% and 62.13±0.05% (referring to the mean ± standard deviation, which were calculated using 3 independent runs) on the classification of CIFAR-10 [87] test images with ensemble sizes of N=14 and N=30 D^2NNs, respectively. Stated differently, 14 D^2NNs (30 D^2NNs) selected through this pruning approach work in parallel to collectively reach 61.14±0.23% (62.13±0.05%) optical inference accuracy for CIFAR-10 test dataset, which provides an improvement of >16% over the average classification accuracy of the individual D^2NNs within each ensemble, demonstrating the "*wisdom of the crowd*". This image classification performance is the highest achieved to date by any diffractive optical network design, applied on the same dataset. We believe that this significantly improved inference and generalization performance provided by feature engineering and ensemble learning of D^2NNs marks a major step forward to open up new avenues for optics-based computation, machine learning and machine vision related systems, benefiting from parallelism of optical systems.

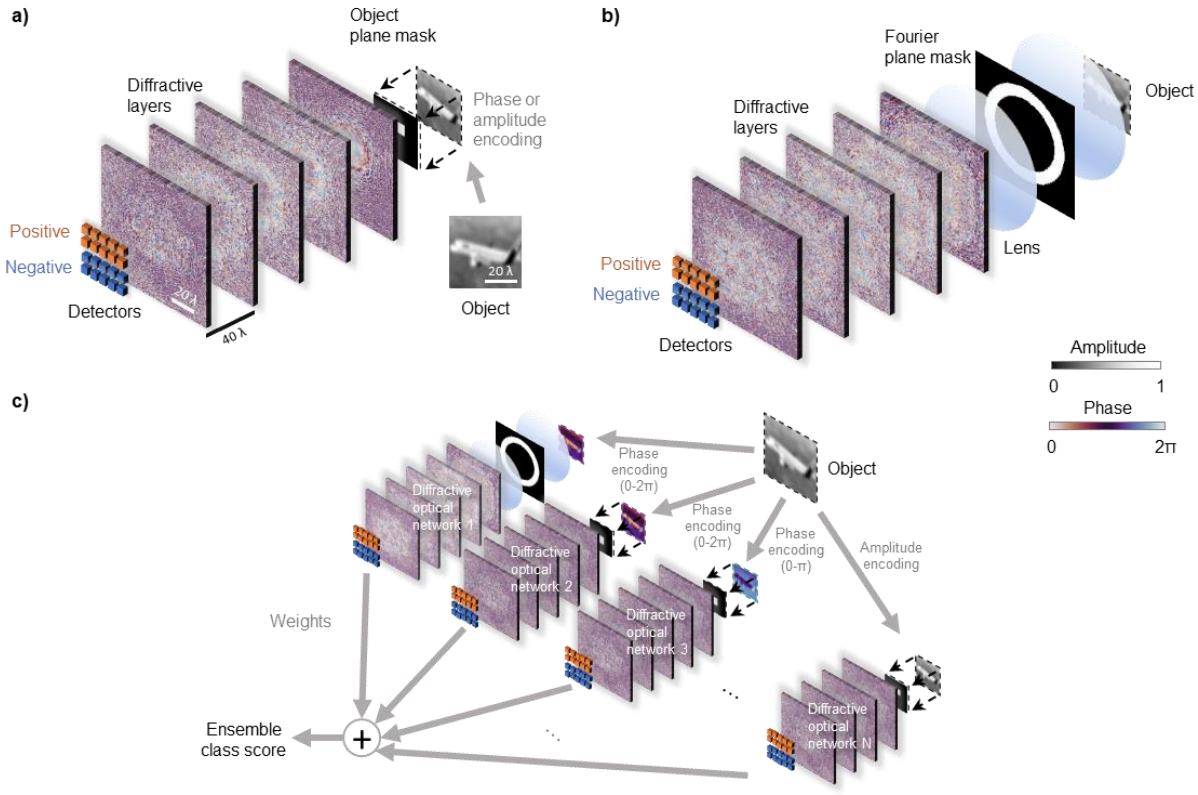


Figure 2.1 Schematic diagram of the ensemble diffractive network system. (a) Example of a D^2NN using a feature engineered input, where an input mask with a passive transmission window opened at a certain position is employed against the object plane. An object from the CIFAR-10 image dataset is shown as an example and is encoded either in the amplitude channel or in the phase channel of the input plane of the diffractive network. (b) Same as in (a), but uses a passive input mask placed on the Fourier plane of a 4-f system; here a bandpass filter is shown as an example. (c) An ensemble D^2NN system, formed by N different feature engineered D^2NNs, is shown where each diffractive network of the ensemble takes the form of (a) or (b). The final ensemble class score is computed through a weighted summation of the differential detector signals obtained from the individual diffractive networks. Through feature engineering and ensemble learning, we achieved blind inference accuracies of 62.13±0.05%, 61.14±0.23% and 60.35±0.39% on CIFAR-10 test image dataset using N=30, N=14 and N=12 D^2NNs, respectively. The standard deviations are calculated through 3 repeats using the same hyperparameters.

2.2 Results

Ensemble learning refers to improving the inference capability of a system by training multiple models instead of a single model, and combining the predictions of the constituent models (known

as base models, base learners or inducers). It is also possible to learn how to combine the decisions of the base learners, which is known as meta-learning [88] (learning from learners). Ensemble learning is beneficial for several reasons [89]; if the size of the training data is small, the base learners are prone to overfitting and as a result suffer from poor generalization to unseen data. Combining multiple base learners helps to ameliorate this problem. Also, by combining different models, the hypothesis space can be extended and the probability of getting stuck in a local minimum is reduced. An important aspect to consider when generating ensembles is the diversity of the learned base models [89]. The learned models should be diverse enough to ensure that different models learn from different attributes of the data, such that through their "*collective wisdom*" the ensemble of these models can eliminate the implicit variance of the constituent models and substantially improve the collective inference performance. One approach to enrich the diversity of the base models is to manipulate the training data used to train different classifiers, making them learn different features of the input space in each trained model. On top of the training of these unique and independent classifiers, pruning methods that aim at finding small sized ensembles, while also achieving a competitive inference performance are also very important [89].

Based on these considerations, Figs. 2.1a and 2.1b depict the two types of D^2NNs [39] (base learners) that we have selected to constitute our ensemble diffractive system. The difference between these two types lies in the placement of the input mask (passive) used to filter out different spatial features of the object field to variegate the information fed to the base D^2NN classifiers. In the structure of Fig. 2.1a, the input filter is placed on the object plane, whereas the structure of Fig. 2.1b uses an input filter on the Fourier plane of a 4-f system placed before the D^2NN. Further heterogeneity is introduced by diversifying the input filter profiles for both types depicted in Figs.

2.1a and 2.1b (see Table 2.1). For example, input filters with transmissive windows of different shapes (rectangular, Gaussian, Hamming, Hanning) and different locations are used at the object plane. The input filters used at the Fourier plane also vary in terms of their pass/stop bands (see the Materials and Methods section for more details). In designing the object plane filters, we used windows of various shapes, sizes and locations to help the individual D^2NNs independently learn the object features at different spatial positions and windows of the input plane. Similar considerations were also made during the design of the Fourier plane filters. Although a filtering operation at the Fourier plane can be represented by an equivalent convolution on the object plane, the two types of input filters that we used serve different purposes. The spatial domain filters provide attention (similar to the attention mechanism used in deep learning [90]) to spatial features and regions of interest at the input plane, while the Fourier plane filters provide different engineered point spread functions and convolution operations that are uniformly applied over the entire sample field-of-view; in this sense these two sets of filters complement each other for a desired inference task.

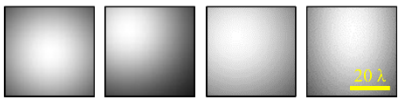
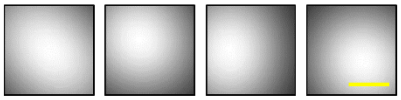
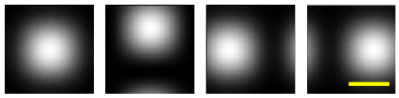
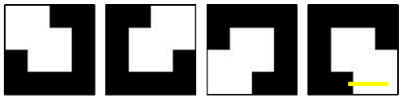
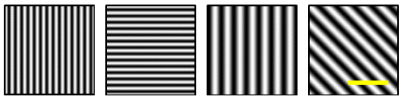
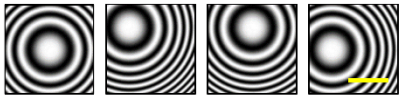
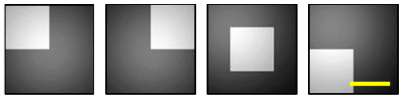
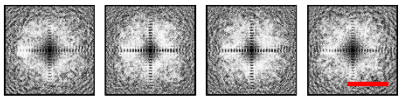
Type	Index	Description	Number of trained base D^2NNs	Example of masks
Masks placed on the object plane	1)	2D Gaussian function defined with variable shapes and spatial x-y positions	310	
	2)	Multiple superposed 2D Gaussian function defined with variable spatial x-y positions	50	
	3)	2D Hamming/Hanning function defined with variable spatial x-y positions	50	
	4)	Square windows with different sizes at variable spatial x-y positions	234	
	5)	Multiple square windows at variable spatial x-y positions	50	
	6)	Patch shaped windows rotated at variable orientations	40	
	7)	Circular windows at variable spatial x-y positions	45	
	8)	Sinusoidal gratings with variable periods and orientations	48	
	9)	Fresnel zone plates with variable x-y spatial positions	45	
	10)	Superposition of Gaussian functions and square windows at variable spatial x-y positions	60	
Masks placed on the Fourier plane	11)	Combination of circular/annular transmissive/non-transmissive zones serving as a low pass, high pass or multi-band pass filter	300	
	12)	A single trainable transmissive layer	20	
			Total = 1252	Mask transmittance 0 1

Table 2.1: The initial network pool containing 1252 individually trained D^2NNs.
For each type of the input mask design, a brief description is given, and the number of trained base D^2NN classifiers and some examples are presented.

To further improve the diversity of the base models, the input object information is encoded into either the phase channel with 4 different dynamic ranges, or the amplitude channel of the illumination field. Using all of these different hyperparameter choices and their combinations, 1252 unique D^2NN classifiers were trained to form the initial pool of our networks. 340 of these networks had the input object information encoded in the amplitude channel, while 912 of them had phase encoded inputs. 276 of the amplitude encoded D^2NNs had an input filter located on the object plane and 64 had an input filter located on the Fourier plane. 656 of the phase-encoded-input networks had a filter on the object plane and 256 had a filter on the Fourier plane. For these 1252 unique D^2NN classifiers, each diffractive network subsequently acts on the filtered version of the input image, and therefore the trained diffractive layers of each base D^2NN directly act on the space domain information (*not* the frequency or Fourier domain).

The preparation of this initial set of 1252 unique D^2NNs was followed by iterative pruning, with the aim of obtaining ensembles of significantly reduced size, i.e., with much smaller number of D^2NNs (base models) in the ensemble. Ensemble pruning was performed by assigning weights to each class score provided by the individual D^2NN classifiers and defining the ensemble class score as a weighted sum of the individual class scores. At each iteration of the ensemble pruning, the weights were optimized through the gradient descent and error backpropagation method to minimize the softmax-cross-entropy (SCE) loss between the predicted ensemble class scores and their one-hot labeled ground truth, followed by choosing the set of weights giving the highest accuracy (see the Materials and Methods section). Then, the 'significance' of the individual D^2NNs in a given state of the ensemble was quantified and ranked by the absolute summation (i.e., L1 norm) of their weights, based on which a certain fraction of the networks was then eliminated from

the ensemble due to their relatively minor contributions. In addition to this greedy search, a periodic *random* elimination of the individual classifiers from the ensemble was also used in the pruning process, so that the solution space could be expanded (see the Materials and Methods section for details).

Based on this pruning process, the iterative search algorithm resulted in a sequence of D^2NN ensembles with gradually decreasing sizes. To select the final ensemble with a desired size (i.e., the number of unique networks), we set a maximum limit on the ensemble size (referred to as the 'maximum allowed ensemble size', i.e., N_{max}), and searched for the D^2NN ensemble that achieves the best performance in terms of inference accuracy on the *validatio*n dataset (i.e., the test dataset was never used during the pruning phase). As we followed this procedure for different values of the pruning hyperparameters, D^2NN ensembles with different sizes and blind testing accuracies were created; we repeated our search 3 times for each set of hyperparameters, which helped us quantify the mean and standard deviation of the inference accuracy for the resulting D^2NN ensembles. We repeated the pruning process 3 times for each set of hyperparameters and reported the mean and standard deviation over these repeats, in the form of mean ± standard deviation. Based on these analyses, Fig. 2.2a reveals that as the maximum allowed ensemble size (N_{max}) gets larger, the blind testing accuracies increase; Fig. 2.2b shows a similar trend reporting the blind testing accuracies as a function of N, i.e., the number of D^2NNs in the ensemble that is selected. Figure 2.2c further reports the relationship between N and N_{max} during the pruning process, which indicates that on average these two quantities vary linearly (with a slope of ~1).

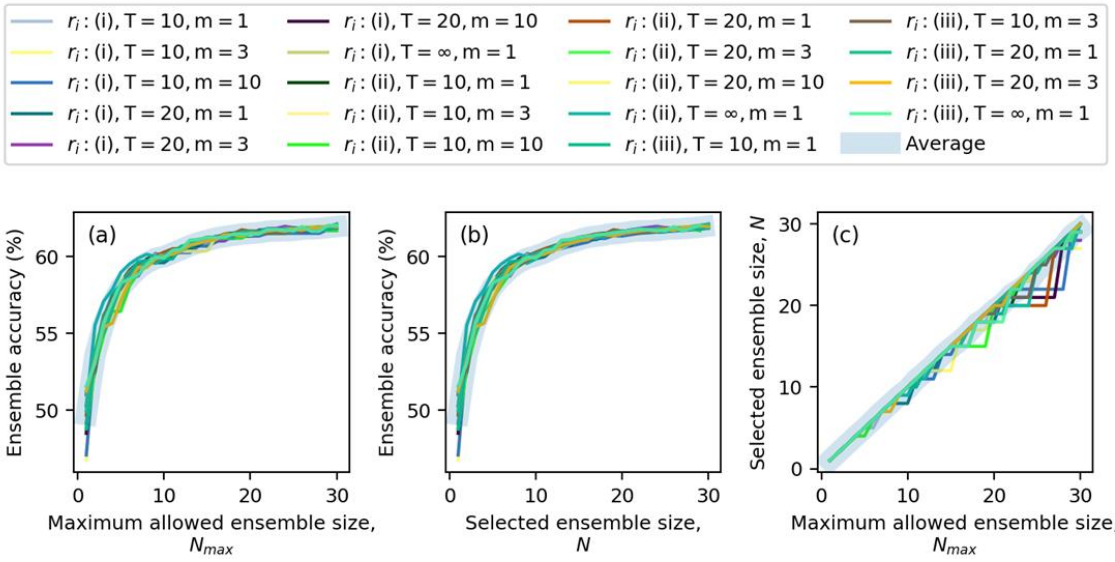


Figure 2.2 Inference accuracy of D^{2}NN ensembles as a function of N_{max} and N. (a) Variation of the blind testing accuracy as a function of the maximum allowed ensemble size (N_{max}) during the pruning; (b) Variation of the blind testing accuracy as a function of the selected ensemble size (N); (c) Relationship between N_{max} and N. The symbols in the legend denote different pruning hyperparameters used in our ensemble selection process; also see Fig. 2.7.

While the results reported in Figs. 2.2a and 2.2b demonstrate the significant gains achieved through the ensemble learning of diffractive networks, they also highlight a diminishing return on the blind inference accuracy of the ensemble with increasing number of D^{2}NNs selected. For example, with ensemble sizes of N=14 and N=30 D^{2}NNs, we achieved blind inference, image classification accuracies of 61.14±0.23% and 62.13±0.05%, respectively, on CIFAR-10 test dataset. Increasing the ensemble size to e.g., N=77 D^{2}NNs resulted in a classification accuracy of 62.56% on the same test dataset. Because of this diminishing return achieved by larger ensemble sizes, we further focused on the case of N_{max}=14 to better explore a sweet-spot: Table 2.2 reports the blind testing accuracies (mean ± standard deviation) achieved for different pruning hyperparameters for a maximum allowed ensemble size of 14. These results summarized in Table 2.2 reveal that, although non-intuitive, the periodic *random* elimination of diffractive models

during the pruning process results in better classification accuracies, compared to pruning with no random model elimination; see the columns in Table 2.2 with $T = \infty$, where T refers to the interval between periodic random elimination of D^2NN models. In Table 2.2, the best average blind testing accuracy (61.14±0.23%) that was achieved for N_{max}=14 is highlighted with a green box. For 3 individual repeats of the pruning process using the same hyperparameters, the classification accuracies achieved by the resulting 14 D^2NNs were 60.88%, 61.33% and 61.21%. Figure 2.3 further presents a detailed analysis of the latter N=14 ensemble that achieved a blind testing accuracy of 61.21%, which is the median among the 3 repeats. Six of the selected base D^2NN classifiers have input filters on the object plane, while the remaining eight have input filters on the Fourier plane (Fig. 2.3a). Figure 2.3b also shows the magnitudes of the class specific weights, optimized for the base classifiers of this N=14 ensemble. Even if these optimized weights are ignored, and made all to be equal to 1, the same diffractive ensemble of 14 D^2NNs achieves a similar inference accuracy of 61.08%, a small reduction from 61.21%.

Accuracy (%) / Number of networks / Accuracy per network(%)	T	10			20			∞		
	r_i	(i)	(ii)	(iii)	(i)	(ii)	(iii)	(i)	(ii)	(iii)
m	1	60.813 ± 0.131	60.873 ± 0.091	60.993 ± 0.195	60.857 ± 0.329	61.090 ± 0.251	61.000 ± 0.277	61.120	60.830	61.120
		14	14	14	14	14	14	14	14	14
		4.344 ±009	4.348 ±0.006	4.357 ±0.014	4.347 ±0.023	4.364 ±0.018	4.357 ±0.020	4.366	4.345	4.366
	3	61.010 ± 0.262	60.903 ± 0.261	61.140 ± 0.233	61.060 ± 0.421	60.847 ± 0.142	60.887 ± 0.130	61.120	60.830	61.120
		14	14	14	14	14	14	14	14	14
		4.358 ±0.019	4.350 ±0.019	4.367 ±0.017	4.361 ±0.030	4.346 ±0.010	4.349 ±0.009	4.366	4.345	4.366
	10	60.693 ± 0.035	61.033 ± 0.152		60.743 ± 0.299	60.353 ± 0.389		61.120	60.830	61.120
		14	14		14	12		14	14	14
		4.335 ±0.003	4.360 ±0.011		4.339 ±0.021	5.029 ±0.032		4.366	4.345	4.366

(i)	(ii)	(iii)
$r_i = 0.98$	$r_i = 0.98 + (0.9 - 0.98)e^{-i/2}$	$r_i = \begin{cases} 0.9, i < 20 \\ 0.95, 20 \le i < 40 \\ 0.98, i \ge 40 \end{cases}$

Table 2.2: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with a maximum allowed ensemble size of N_{max}=14.
For the classification accuracies that are reported, the average and the standard deviation values result from 3 independent repeats of the pruning process using the same hyperparameters. The lower table describes the schemes used for r_i denoted by (i), (ii) and (iii). The green box highlights the ensemble achieving the best average blind testing accuracy (N=14), and the red box highlights the ensemble achieving the best average blind testing accuracy *per network* (N=12). See Fig. 2.7 for definition of the pruning hyperparameters r_i, m and T.

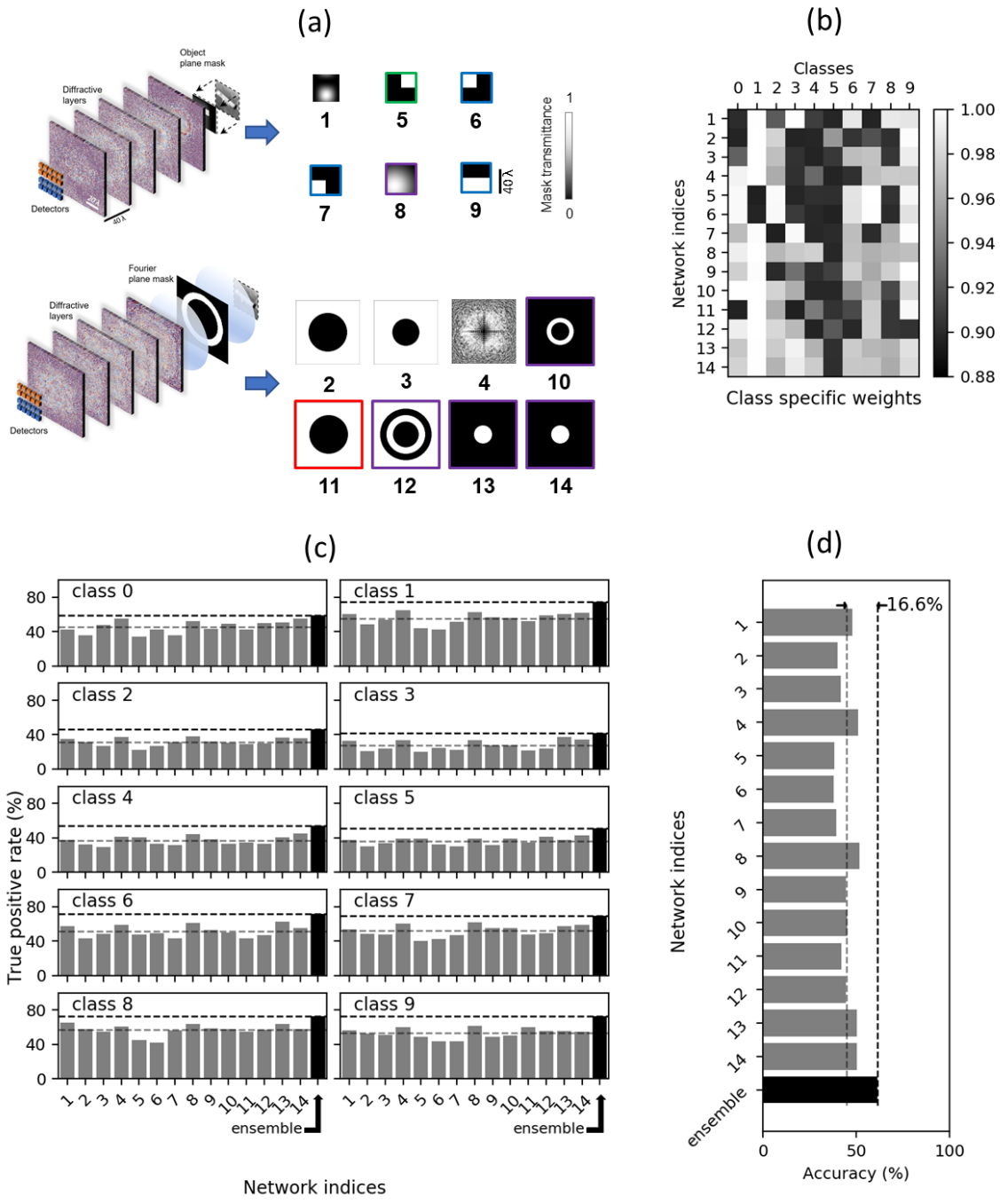


Figure 2.3 An ensemble of N=14 D^{2}NNs achieves a blind classification accuracy of 61.21% on CIFAR-10 test dataset. Input filters/masks used before each one of the D^{2}NNs that form the ensemble. For D^{2}NNs 1, 5-9: the input filters are on the object plane. For the remaining D^{2}NNs 2-4, 10-14: the input filters are on the Fourier plane. The input filters corresponding to the networks with phase encoded inputs are enclosed within a border/frame (5-14), while the inputs of the diffractive networks 1-4 are amplitude encoded. The dynamic range of the input phase encoding

is represented by the border color; red: 0-π/2, green: 0-π, blue: 0-3π/2, purple: 0-2π. (b) Class specific weights for each D^2NN of the ensemble. If one ignores these class specific weights and replaces them with all ones, the blind inference accuracy slightly decreases to 61.08%, from 61.21%. (c) True positive rates of the individual diffractive networks, compared against their ensemble for different classes. (d) Test accuracy of the individual networks compared against their ensemble. The dotted lines show the classification performance improvement (~16.6%) achieved by the diffractive ensemble over the mean performance of the individual D^2NNs. Three repeats with the same hyperparameters resulted in a blind classification accuracy of 61.14±0.23%, where 61.21% represents the median, detailed in this figure.

In addition to these, Fig. 2.3c also shows the true positive rates for each class, corresponding to the individual members of N=14 D^2NNs as well as the ensemble. The improvements in the true positive rates of the ensemble over the mean performance of the individual classifiers for different data classes lie between 13.47% (for class 0) and 19.98% (for class 6). Figure 2.3d further presents a comparison of the classification accuracies of the individual diffractive classifiers compared against their ensemble. Through these comparative analyses reported in Figs. 2.3c and 2.3d, it is evident that the performance of the ensemble is significantly better than any individual diffractive network of the ensemble, demonstrating the "wisdom of the crowd" achieved through our pruning process.

In Table 2.2, we also report another metric, i.e., 'the accuracy per network', which is the average accuracy divided by the number of networks in the ensemble, to reveal the performance efficiency of ensembles that achieve at least 60% average blind testing accuracy for CIFAR-10 test dataset. The best performance achieved in Table 2.2 based on this metric is highlighted with a red box: N=12 unique D^2NNs selected by the pruning process with N_{max}=14 achieved a blind testing accuracy of 60.35±0.39%, where the accuracy values for the individual 3 repeats were 60.77%, 60.00% and 60.29%. Details of the latter ensemble with a blind testing accuracy of 60.29%

(which is the median among the 3 repeats) can be found in Fig. 2.4, revealing the selected input filters and the class specific weights of the resulting 12 D^2NNs of this ensemble.

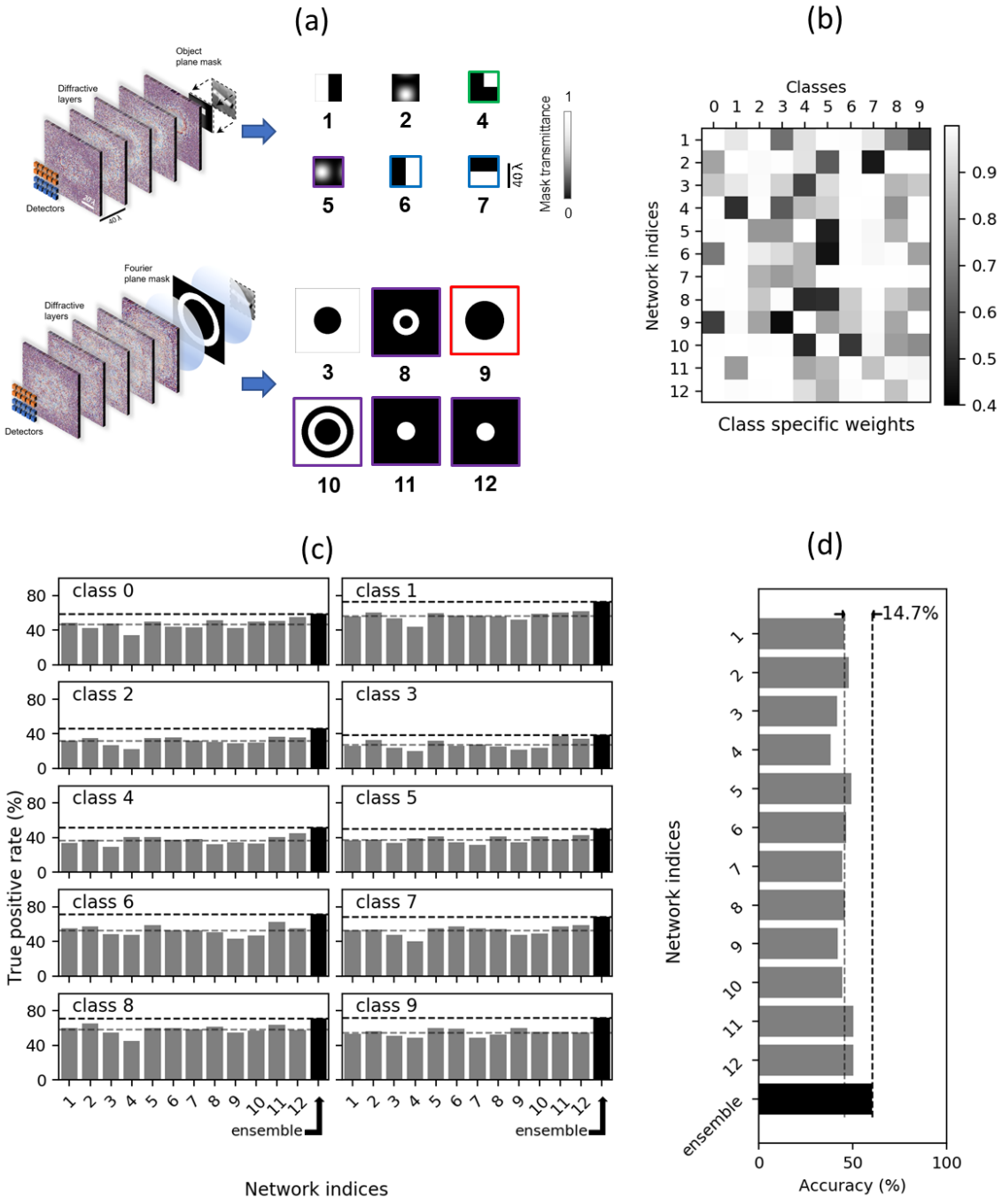


Figure 2.4 An ensemble of N=12 D^2NNs achieves a blind classification accuracy of 60.29% on CIFAR-10 test dataset. (a) Input filters/masks used before each one of the D^2NNs that form the ensemble. For D^2NNs 1, 2, 4-7: the input filters are on the object plane. For the remaining D^2NNs 3, 8-12: the input filters are on the Fourier plane. The input filters corresponding to the networks with phase encoded input are enclosed within a border/frame (4-12), while the inputs of the diffractive networks 1-3 are amplitude encoded. The dynamic range of the input phase encoding is represented by the border color; red: 0-$\pi/2$, green: 0-π, blue: 0-$3\pi/2$, purple: 0-2π. (b) Class specific weights for each D^2NN of the ensemble. (c) True positive rates of the individual networks, compared against their ensemble for different classes. (d) Test accuracy of the individual networks compared against their ensemble. The dotted lines show the classification performance improvement (~14.7%) achieved by the diffractive ensemble over the mean performance of the individual D^2NNs. Three repeats with the same hyperparameters resulted in a blind classification accuracy of 60.35±0.39%, where 60.29% represents the median.

Our results reveal that encoding the input object information in the amplitude channel of some of the base D^2NNs and in the phase channel of the other D^2NNs helps to diversify the ensemble. Table 2.3 further confirms this by reporting the blind testing accuracies achieved when the initial ensemble consists of *only* the 912 D^2NNs whose input is encoded in the phase channel. A direct comparison of Table 2.2 and Table 2.3 reveals that including both types of input encoding (phase and amplitude) within the ensemble helps improve the inference accuracy. Using *only* phase encoding for the input of D^2NNs, the best average blind testing accuracy achieved using N_{max}=14 was 60.74±0.17% with an ensemble size of N=14. The detailed description of the *median* of these D^2NN ensembles with a classification test accuracy of 60.65% is provided in Fig. 2.5. Fig. 2.6 also shows the details of another phase-only input encoding ensemble with N=12 D^2NNs, achieving a blind testing accuracy of 60.43%.

Accuracy (%)	T	10			20			∞		
Number of networks										
Accuracy per network(%)	r_i	(i)	(ii)	(iii)	(i)	(ii)	(iii)	(i)	(ii)	(iii)
m	1	60.437 ±0.067	60.433 ±0.045	60.513 ±0.121	60.740 ±0.165	60.560 ±0.182	60.410 ±0.177	60.360	60.360	60.570
		14	14	14	14	14	14	14	14	14
		4.317 ±0.005	4.317 ±0.003	4.322 ±0.009	4.339 ±0.012	4.326 ±0.013	4.315 ±0.013	4.311	4.311	4.326
	3	60.700 ±0.149	60.667 ±0.163	60.630 ±0.328	60.430 ±0.286	60.627 ±0.410	60.663 ±0.076	60.360	60.360	60.570
		14	14	13	13	14	14	14	14	14
		4.336 ±0.011	4.333 ±0.012	4.664 ±0.025	4.648 ±0.022	4.330 ±0.029	4.333 ±0.005	4.311	4.311	4.326
	10	60.413 ±0.096	60.700 ±0.159		60.620 ±0.046	60.553 ±0.163		60.360	60.360	60.570
		12	14		14	14		14	14	14
		5.034 ±0.008	4.336 ±0.011		4.330 ±0.003	4.325 ±0.012		4.311	4.311	4.326

Table 2.3: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with only phase encoded input D^2NNs and a maximum allowed ensemble size of N_{max}=14.

For the classification accuracies that are reported, the average and the standard deviation values result from 3 independent repeats of the pruning process using the same hyperparameters. The green box highlights the D^2NN ensemble achieving the best average blind testing accuracy (N=14), and the red box highlights the D^2NN ensemble achieving the best average blind testing accuracy *per network* (N=12).

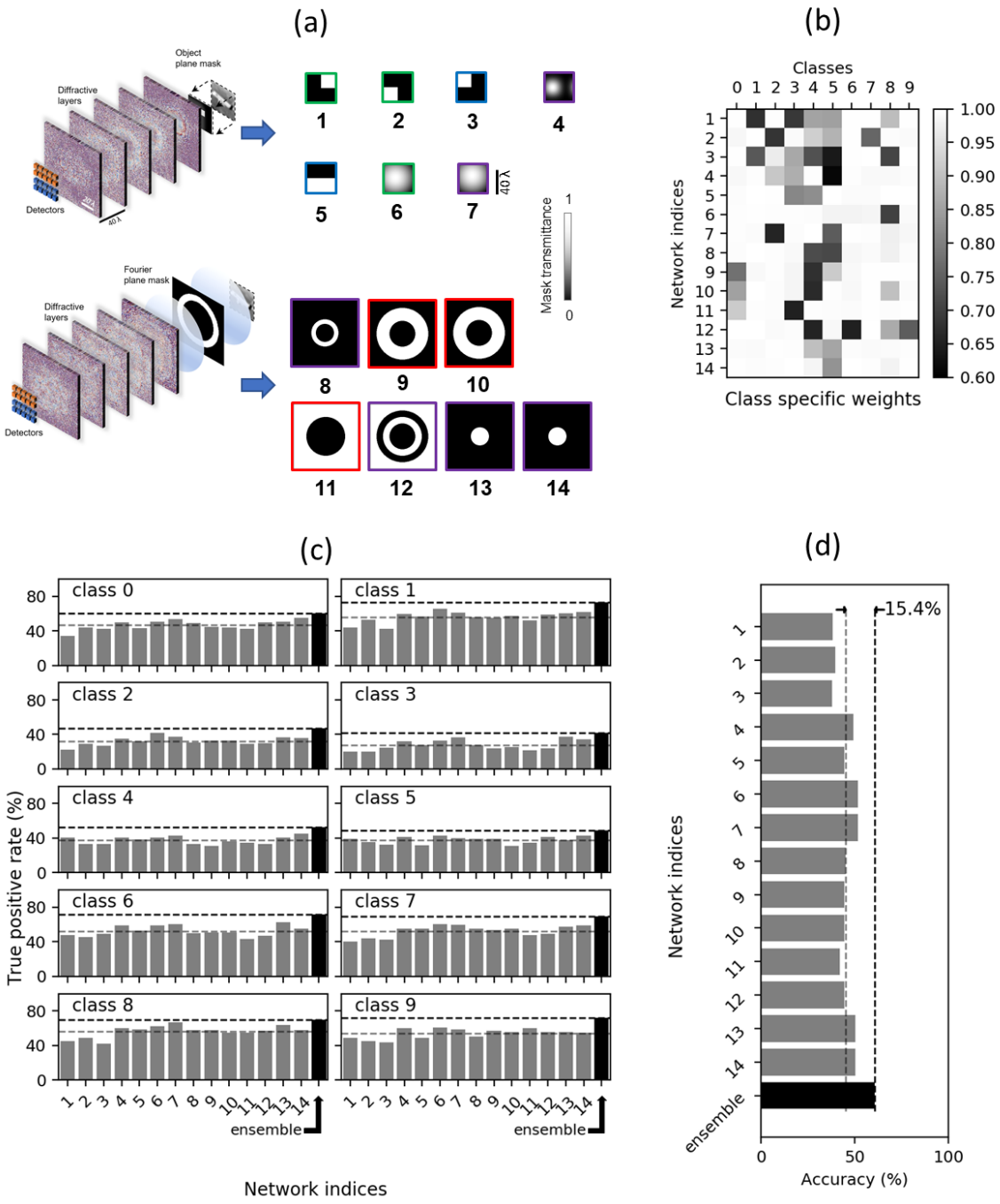


Figure 2.5 A D^{2}NN ensemble consisting of only phase-encoded-input networks (N=14) achieves a blind classification accuracy of 60.65% on CIFAR-10 test dataset. (a) Input filters/masks used before each one of the D^{2}NNs that form the ensemble. For D^{2}NNs1-7: the input filters are on the object plane. For the remaining D^{2}NNs 8-14: the input filters are on the Fourier plane. The dynamic range of the input phase encoding is represented by the border color; red: 0-π/2, green: 0-π, blue: 0-3π/2, purple: 0-2π. (b) Class specific weights for each D^{2}NN of the

ensemble. (c) True positive rates of the individual networks, compared against their ensemble for different classes. (d) Test accuracy of the individual networks compared against their ensemble. The dotted lines show the classification performance improvement (~15.4%) achieved by the diffractive ensemble over the mean performance of the individual D^2NNs. Three repeats with the same hyperparameters resulted in a blind classification accuracy of 60.74±0.17%, where 60.65% represents the median.

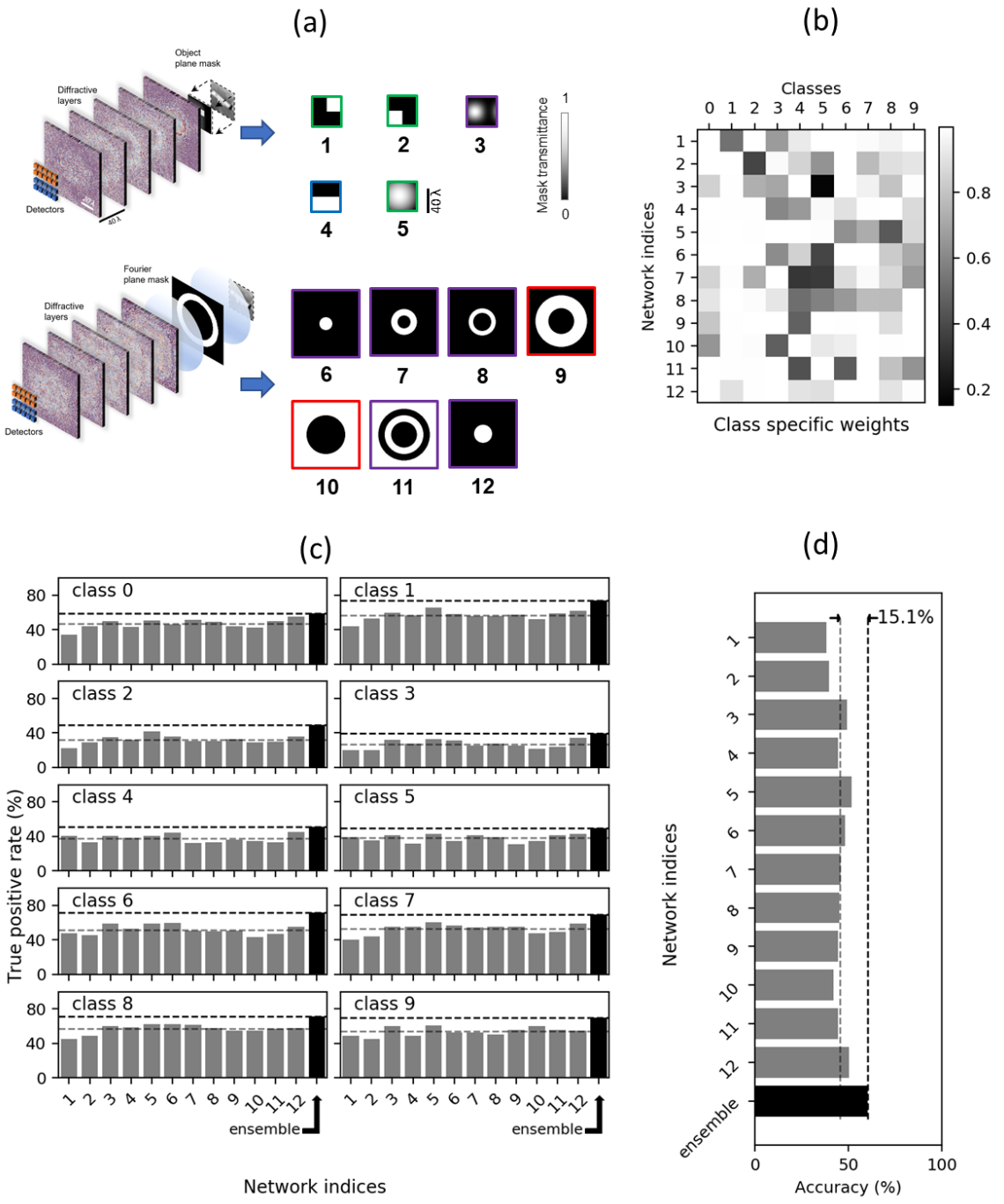


Figure 2.6 A D^{2}NN ensemble consisting of only phase-encoded-input networks (N=12) achieves a blind classification accuracy of 60.43% on CIFAR-10 test dataset. (a) Input filters/masks used before each one of the constituent D^{2}NNs that form the ensemble. For D^{2}NNs 1-5: the input filters are on the object plane. For the remaining D^{2}NNs 6-12: the input filters are on the Fourier plane. The dynamic range of the input phase encoding is represented by the border/frame color; red: 0-π/2, green: 0-π, blue: 0-3π/2, purple: 0-2π. (b) Class specific weights

for each D^2NN of the ensemble. (c) True positive rates of the individual networks, compared against their ensemble for different classes. (d) Test accuracy of the individual networks compared against their ensemble. The dotted lines show the classification performance improvement (~15.1%) achieved by the diffractive ensemble over the mean performance of the individual D^2NNs. Three repeats with the same hyperparameters resulted in a blind classification accuracy of 60.41±0.10%, where 60.43% represents the median.

Furthermore, it is noteworthy that the top 10 D^2NNs (in terms of their individual blind testing accuracies) within the initial pool of 1252 networks were **not** selected in any of the D^2NN ensembles of Figs. 2.3, 2.4, 2.5 and 2.6. *This corroborates our conjecture that the individual performance of a base model might not be indicative of its performance within an ensemble*. In fact, several of the base D^2NNs selected in the ensembles of Figs. 2.3, 2.4, 2.5 and 2.6 had blind testing accuracies <40%, whereas the blind testing accuracies of the best models (not chosen in any of the ensembles) were >50%.

So far, the pruning strategy that we have investigated is based on assigning weights to each differential class score of the individual D^2NNs. Based on a differential detection scheme [39], these class scores are computed through the normalized difference of the signals from the class detector pairs. To further explore if this weight assignment can be improved, we also considered a more general case, where the trainable weights are assigned not only to the class scores, but also to each of the detectors, representing a much broader scheme compared to the differential balanced detection [39]. We optimized this augmented set of weights in two different schemes: (1) the detector signal weights are optimized simultaneously with the class score weights in each iteration of the pruning process, and (2) the detector signal weights and the class score weights are alternatively optimized in different iterations (see the Materials and Methods section for details). The results of these alternative pruning strategies are shown in Tables 2.4 and 2.5. With N_{max} = 14,

the best testing accuracy reported using the assignment scheme (1) was found to be 61.02%, while using the assignment scheme (2), we achieved a testing accuracy of 61.35%. Compared to the previous classification accuracy (61.14%) achieved using class score weights only, these new results present very similar performance. This comparative analysis further confirms our previous observation that although the weights are vital for ensemble pruning, their ultimate effect on the inference accuracy is not significant.

Accuracy (%) / Number of networks / Accuracy per network(%)	T	10			20			∞		
	r_i	(i)	(ii)	(iii)	(i)	(ii)	(iii)	(i)	(ii)	(iii)
m	1	60.520 ±0.135	60.757 ±0.118	60.770 ±0.175	60.780 ±0.105	60.877 ±0.232	60.967 ±0.061	60.770	60.850	60.770
		14	14	14	14	14	14	14	14	14
		4.323 0.010	4.340 0.008	4.341 0.013	4.341 ±0.008	4.348 ±0.017	4.355 ±0.004	4.341	4.346	4.341
	3	61.007 ±0.182	60.617 ±0.075	60.833 ±0.201	60.933 ±0.224	60.510 ±0.101	60.913 ±0.108	60.770	60.850	60.770
		14	14	14	14	12	14	14	14	14
		4.358 ±0.013	4.330 ±0.005	4.345 ±0.014	4.352 ±0.016	5.042 ±0.008	4.351 ±0.008	4.341	4.346	4.341
	10	60.927 ±0.206	60.820 ±0.242		61.023 ±0.289	60.607 ±0.222		60.770	60.850	60.770
		14	14		14	12		14	14	14
		4.352 ±0.015	4.344 ±0.017		4.359 ±0.021	5.051 ±0.019		4.341	4.346	4.341

Table 2.4: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with a maximum allowed ensemble size of N_{max}=14, where in the pruning process trainable weights are also assigned to the individual detector signals and are *simultaneously* optimized with the class score weights (referred to as the weight assignment scheme (1)).

For the classification accuracies that are reported, the average and the standard deviation values result from 3 independent repeats of the pruning process using the same hyperparameters. The green box highlights the ensemble achieving the best average blind testing accuracy (N=14), and the red box highlights the ensemble achieving the best average blind testing accuracy *per network* (N=12).

Accuracy (%) / Number of networks / Accuracy per network(%)	T	10			20			∞		
	r_i	(i)	(ii)	(iii)	(i)	(ii)	(iii)	(i)	(ii)	(iii)
m	1	60.920 ±0.166	60.913 ±0.179	60.953 ±0.289	60.937 ±0.131	61.103 ±0.263	60.883 ±0.091	60.610	61.270	61.350
		14	14	14	14	14	14	13	14	14
		4.351 ±0.012	4.351 ±0.013	4.354 ±0.021	4.353 ±0.009	4.365 ±0.019	4.349 ±0.006	4.662	4.376	4.382
	3	60.710 ±0.529	60.793 ±0.215	60.973 ±0.085	61.010 ±0.195	60.923 ±0.158	61.033 ±0.137	60.610	61.270	61.350
		14	14	14	14	14	14	13	14	14
		4.336 ±0.038	4.342 ±0.015	4.355 ±0.006	4.358 ±0.014	4.352 ±0.011	4.360 ±0.010	4.662	4.376	4.382
	10	60.767 ±0.121	60.940 ±0.193		61.227 ±0.244	60.177 ±0.734		60.610	61.270	61.350
		14	14		14	12		13	14	14
		4.340 ±0.009	4.353 ±0.014		4.373 ±0.017	5.015 ±0.061		4.662	4.376	4.382

Table 2.5: Comparison of blind testing accuracy results achieved under different pruning hyperparameters, with a maximum allowed ensemble size of N_{max}=14, where in the pruning process trainable weights are also assigned to the detector signals and are *alternatively* optimized with the class score weights (referred to as the weight assignment scheme (2)).
For the classification accuracies that are reported, the average and the standard deviation values result from 3 independent repeats of the pruning process using the same hyperparameters. The green box highlights the D^2NN ensemble achieving the best average blind testing accuracy (N=14), and the red box highlights the D^2NN ensemble achieving the best average blind testing accuracy *per network* (N=12).

2.3 Discussion

Although forming an ensemble of separately trained D^2NNs ensues a major improvement in the classification and generalization performance of diffractive networks, further improvements might be possible to reduce the performance gap with respect to the state-of-the-art electronic neural networks. The classification accuracies of widely known all-electronic classifiers on grayscale CIFAR-10 test image dataset can be summarized as [39]: Support Vector Machine (SVM) [91] 37.13%, LeNet [92] 66.43%, AlexNet[2] 72.64%, ResNet[3] 87.54%. While the blind testing accuracy of an ensemble of N=30 unique diffractive optical networks (62.13±0.05%)

comes close to the performance of LeNet, which was the first demonstration of a convolutional neural network (CNN), there is still a large performance gap with respect to the state-of-the-art CNNs, and this suggests that there might be more room for improvements, especially through a wider span of input feature engineering within larger pools of D^2NNs, forming a much richer and more diverse initial condition for iterative pruning.

The presented improvement in the classification performance of D^2NNs obtained with feature engineering and ensemble learning does not come free of cost. Due to the multiple optical paths as part of this framework, the number of diffractive layers and the opto-electronic detectors to be fabricated and used increases in proportion to the number of networks (N) used in the final ensemble, which results in an increased complexity for the optical network set-up. The required training time also rises up significantly because of the need for a large number of individual networks in the initial pool, which in our case was 1252 individual D^2NNs. However, this training process is a one-time effort, and the inference time or latency remains the same by virtue of the parallel processing capability of the diffractive optical system; stated differently, the information processing occurs through diffraction of light within each D^2NN of the ensemble, and because all of the individual diffractive networks of an ensemble are passive devices that work in parallel, we do not expect a slowdown in speed of inference. Also, the detection circuitry complexity of the diffractive optics-based solutions is still minimal compared to its electronic counterparts, and the hardware complexity of D^2NN ensembles can be reduced even further by using an additive sum of the individual class scores instead of the weighted sum, at the cost of a very small sacrifice in the inference accuracy. For example, for the ensemble of D^2NNs depicted in Fig. 2.3, if a simple additive sum of the individual class scores is used instead of the optimized class-specific weights,

the blind classification accuracy reduces only slightly from 61.21% to 61.08%. This suggests that a further reduction in the hardware complexity is attainable with a very small sacrifice in the inference accuracy by discarding the specific weights of the class scores. However, these weights still play a very significant role in the pruning process as they help in our selection of the diffractive models to be retained in each iteration during the ensemble pruning by measuring/quantifying the significance of the individual networks in an ensemble (see the Materials and Methods section). Some of the drawbacks associated with the relatively increased size and complexity of the optical hardware should also become less restrictive since the advances in integrated photonics and fabrication technologies have led to continuous miniaturization of opto-electronic devices [93]. The physical dimensions of an individual D^2NN model with a fixed number of diffractive layers are dictated by the illumination wavelength. For example, the longitudinal dimension of the D^2NN designs used in our models is ~240 λ (which refers to the distance between the input and the output planes), and the lateral dimension is ~100 λ (which refers to the width of each diffractive layer). Using state-of the art fabrication technologies, it is possible to create diffractive structures with a feature size of a few hundreds of nanometers [94,95], potentially extending the application of diffractive systems to e.g., the visible spectrum. On top of these 3D nanofabrication technologies based on multiphoton polymerization, multilayer photolithographic methods may also be useful for the fabrication of diffractive systems [96]. Nanoimprinting, roll-to-roll patterning might also prove to be cheaper alternative to these expensive fabrication techniques for this purpose [97,98]. Realization of D^2NNs in the visible spectrum would also significantly reduce the overall size of the ensemble. On the other hand, such D^2NNs operating at visible wavelengths would also present 3D alignment challenges, requiring high-resolution structuring of free-space diffractive layers, which need to be precisely aligned with respect to each other. Recent work on the design of

misalignment resilient[31] D^2NN models could be rather useful for practical implementations of such diffractive systems operating at visible wavelengths.

Furthermore, while miniaturization of D^2NN fabrication is certainly possible for operation in visible wavelength, the fabrication of miniaturized optical components such as filters and lenses might seem challenging, given the bulkiness of the traditional optical components. However, recent research in metasurface-based optics [99,100] has enabled remarkable miniaturization in fabrication of optical components and accordingly should come in handy in the realization of miniaturized D^2NN ensembles.

In addition to the issues of hardware complexity and size, to maintain a desired signal-to-noise (SNR) ratio at the output detectors, the optical input (illumination) power of the system needs to be increased in proportion to the ensemble size. However, due to the availability of various high-power laser sources, this higher demand for increased illumination power of the system should not be a significant obstacle for its operation. While the use of high-power lasers might not offer a cost-effective solution, all-optical object detection and classification applications that require extremely fast inference on the spot (e.g., for threat detection) might still justify their use. Also, since D^2NNs are inherently passive, the availability of low-loss materials for the fabrication of diffractive layers might lead to power-efficient diffractive networks, partially offsetting the high-power illumination requirement. Furthermore, given the fact that broadband diffractive networks have already been reported to process pulsed optical inputs, [53,54,85] the utilization of pulsed lasers, such as the ones that are widely used in e.g., telecommunications and microscopy applications, might help to provide sufficient SNR at each detector plane of the ensemble. Another potential solution to reduce the input power requirement could be to time-gate the illumination

signals to different diffractive networks at the cost of some increase in the inference time, by illuminating each individual D^2NN of the ensemble sequentially, i.e., one by one.

The passive nature of a physically fabricated D^2NN model, while an advantage in terms of power requirements, is also a disadvantage as it creates limitations for e.g., dynamically changing datasets. Incorporating dynamic spatial light modulators (SLMs) to implement the diffractive layers would augment the D^2NN framework to become reconfigurable, at the cost of additional hardware complexity and power. Furthermore, diffractive networks have been shown to benefit from transfer learning, where an already fabricated D^2NN can be improved in performance by inserting additional, new diffractive layers or replacing some of the existing diffractive layers with newly trained ones [38,54] benefiting from the modularity of a D^2NN design.

Another limitation of our current method is the computing time necessary for preparing the initial ensemble. In this paper we prepared in total 1252 D^2NNs, which resulted in a huge computational burden and long training time even though we used multiple GPUs to train all the models in parallel. However, significant reduction in training time might be possible through further optimizing the numerical implementation of the models or appropriately reducing the numerical precision, e.g. by using hybrid precision of different bit depths. On the other hand, since our study of the optimized ensembles after pruning revealed that many of the class of filters (see Table 2.1) were rarely represented in the final ensembles, there is also a potentiality to reduce the demand on the total number of networks to be trained by achieving better understanding of the filter space and designing the filters more delicately, such as extracting and harnessing the principle components of the given dataset.

Finally, diffractive networks reported in this work utilize coherent illumination and operate at a single illumination wavelength. Recent studies have reported diffractive networks that can process a continuum of wavelengths, [53,54,85] which lends itself to the possibility of multiplexing the object information at different wavelength channels of the illumination. The inference accuracy of an ensemble diffractive model might benefit from this wavelength diversity by utilizing diffractive networks that process specific color channels (e.g., red, green and blue), either jointly or individually. These are promising research directions for future D^2NN ensemble designs that might further enhance their blind inference performance.

In summary, we significantly improved the statistical inference and generalization performance of D^2NNs using feature engineering and ensemble learning. We independently trained a total of 1252 unique D^2NNs that were diversely engineered with various passive input filters. Using a pruning algorithm, we searched through these 1252 D^2NNs to select an ensemble that collectively improves the image classification accuracy of the optical network. Our results revealed that ensembles of N=14 and N=30 D^2NNs achieve blind testing accuracies of 61.14±0.23% and 62.13±0.05%, respectively, on the classification of CIFAR-10 test images, which constitute the highest inference accuracies achieved to date by any diffractive optical neural network design applied on this dataset. The versatility of D^2NN framework stems from its applicability to different parts of the electromagnetic spectrum and the availability of miscellaneous fabrication techniques such as 3D printing and lithography. Together with further advances in the miniaturization and fabrication of optical systems, the presented results and the underlying platform might be utilized in a variety of applications, for e.g. ultra-fast object classification, diffraction-based optical computing hardware, and computational imaging tasks.

2.4 Materials and Methods

2.4.1 Implementation of D^2NNs

As the basic building block of our diffractive ensemble, all the individual D^2NN base classifiers presented in this paper consist of 5 successive diffractive layers, which modulate the phase of the incidence optical field and are connected to each other by free space propagation in air. The propagation model we used was formulated based on the Rayleigh-Sommerfeld diffraction equation [38,80], assuming that each diffractive feature (or "neuron") on the diffractive layers serves as a source of modulated secondary waves, which jointly form the propagated wave field. The presented results and analyses of this manuscript are broadly applicable to any part of the electromagnetic spectrum as long as the diffractive features and the physical dimensions are accordingly scaled with respect to the wavelength of light. Using a coherent illumination wavelength of λ, for all the diffractive network designs, the size of each neuron and the axial distance between two successive diffractive layers were set to be ~0.5 λ and 40 λ, respectively, which guarantees an adequate diffraction cone for each neuron to optically communicate with all the neurons of the consecutive layer and enables the diffractive optical network to be "fully-connected". Each photodetector at the output plane of a D^2NN is assumed to be a square, with a width of 6.4 λ. Since we employed here a differential detection scheme [39], the detectors were divided into two groups: positive detectors and negative detectors, and were collectively used to compute the differential class scores for network *k*, i.e., z_{ck} through the following equation:

$$z_{ck} = \frac{z_{ck}^{+} - z_{ck}^{-}}{z_{ck}^{+} + z_{ck}^{-}} \tag{2.1}$$

where z_{ck}^{+} and z_{ck}^{-} denote the optical signal of the positive and the negative detectors for class *c*, respectively. Since the dataset used in this paper, i.e., the CIFAR-10 dataset, has 10 classes and

a pair of positive and negative detectors constitutes the score for each class, there are in total 20 detectors on the detector plane of a single D^2NN. An empirical factor of $K=0.1$, also termed as the "temperature" coefficient in machine learning literature [101], was a non-trainable hyperparameter that we utilized to achieve more efficient convergence during the training phase by dividing Eq. (2.1) by K. In addition, the input object was encoded either in the amplitude or in the phase channel of the input illumination, which is assumed to be a uniform plane wave generated by a coherent source. The phase encoding of the input objects took values from either of the following four intervals: 0-0.5π, 0-π, 0-1.5π or 0-2π.

2.4.2 Feature engineering of diffractive networks

We used two types of feature engineered diffractive networks: one employed an input filter placed on/against the object plane that filters the spatial signals directly, while the other one used an input filter placed on the Fourier plane of a 4-f system to filter certain spatial frequency components of the object. Unless the filters are specifically mentioned to be trainable, these input filter designs were pre-defined, keeping the transmittance of their pixels *constant* during the training of the diffractive networks (see e.g., Table 2.1). For all the feature engineered D^2NN classifiers, each diffractive network subsequently acts on the filtered input image, directly processing the input information on the spatial domain, *not* the frequency or Fourier domain.

The object plane filters are designed to be of the same size as the object, containing transmissive patterns, the amplitude distribution of which takes one of the following forms: 1) 2D Gaussian functions defined with variable shapes and center positions; 2) multiple superposed 2D Gaussian functions defined with variable center positions; 3) 2D Hamming/Hanning functions defined with variable center positions; 4) square windows with different sizes at variable center

positions; 5) multiple square windows at variable center positions; 6) patch-shaped windows rotated at variable angles; 7) circular windows at variable center positions; 8) sinusoidal gratings with variable periods and orientations; 9) Fresnel zone plates with variable x-y spatial positions; and 10) superpositions of Gaussian functions and square windows at variable spatial x-y positions.

For the second type of D^2NNs with a Fourier plane input filter, using the same Rayleigh-Sommerfeld diffraction equation mentioned above, we numerically implemented a 4-f system with two lenses; the first lens transforms the object information from the spatial domain to the frequency domain and the second one does the opposite. On the Fourier plane that is 2f away from the object plane, a single amplitude-only input filter, designed in one of the following forms is employed: 1) various combinations of circular/annular passbands, which are defined through specifying a series of equally spaced concentric ring-like areas, such that it can serve as a low/high pass, single-band pass or multi-band pass filter, or 2) a single *trainable* layer enabling the system to learn an input spatial frequency filter on its own. On the output image plane of the 4-f system that is 4f away from the object plane, a square aperture is placed with the same size as the object or 1.5 times the size of the object, before feeding the resulting complex-valued field into the diffractive network. In the numerical implementation, the lens has a focal length f of ~145.6 λ and a diameter of 104 λ.

For each type of the input filter design, the number of trained base D^2NNs and some input filter examples can be found in Table 2.1.

2.4.3 Training details

All the D^2NNs and their weighted ensembles in this paper were numerically implemented and trained using Python (v3.6.5) and TensorFlow (v1.15.0, Google Inc.). An Adam optimizer [102] with default parameters (predefined in TensorFlow) was used to calculate the back-propagated

gradients during the training of the individual optical models and the ensemble weights. The learning rate, starting from an initial value of 0.001, was set to decay at the rate of 0.7 every 8 epochs. The publicly available CIFAR-10 dataset consists of 50,000 training images and 10,000 test images [87]. The training images were split into sets of 45,000 and 5,000 images, for training and validation, respectively. All the blind testing accuracies reported in this paper (individual D^2NN and ensemble models) were evaluated on the 10,000 test images, which were *never* used either during the training of the individual networks or during the optimization of the weights for the ensemble pruning (detailed in the following subsection). Since the images in the original CIFAR-10 dataset contain three color channels (red, green and blue) and monochromatic illumination is used in our diffractive optical network models, the built-in *rgb_to_grayscale* function in TensorFlow was applied to convert these color images to grayscale. Also, to enhance the generalization capability of the trained D^2NNs, we randomly flipped the images (left-to-right) with a probability of 0.5 while training. For training of the individual D^2NNs, we used a batch size of 8 and trained each model for 50 epochs using the training image set, and selected the best model based on the classification performance on the validation image set. The D^2NN loss function for a given network k used softmax-cross-entropy between the differential class scores z_{ck} and their one-hot labeled ground truth vector g:

$$D^2NN\ \text{Loss} = -E\left[\sum_{c=1}^{C} g_c\, log\left(\frac{exp(z_{ck})}{\sum_{c=1}^{C} exp(z_{ck})}\right)\right] \tag{2.2}$$

where $E[.]$ denotes the expectation over the training images in the current batch, C=10 denotes the total number of classes in the dataset, and g_c represents the c^{th} entry of the ground truth label vector g.

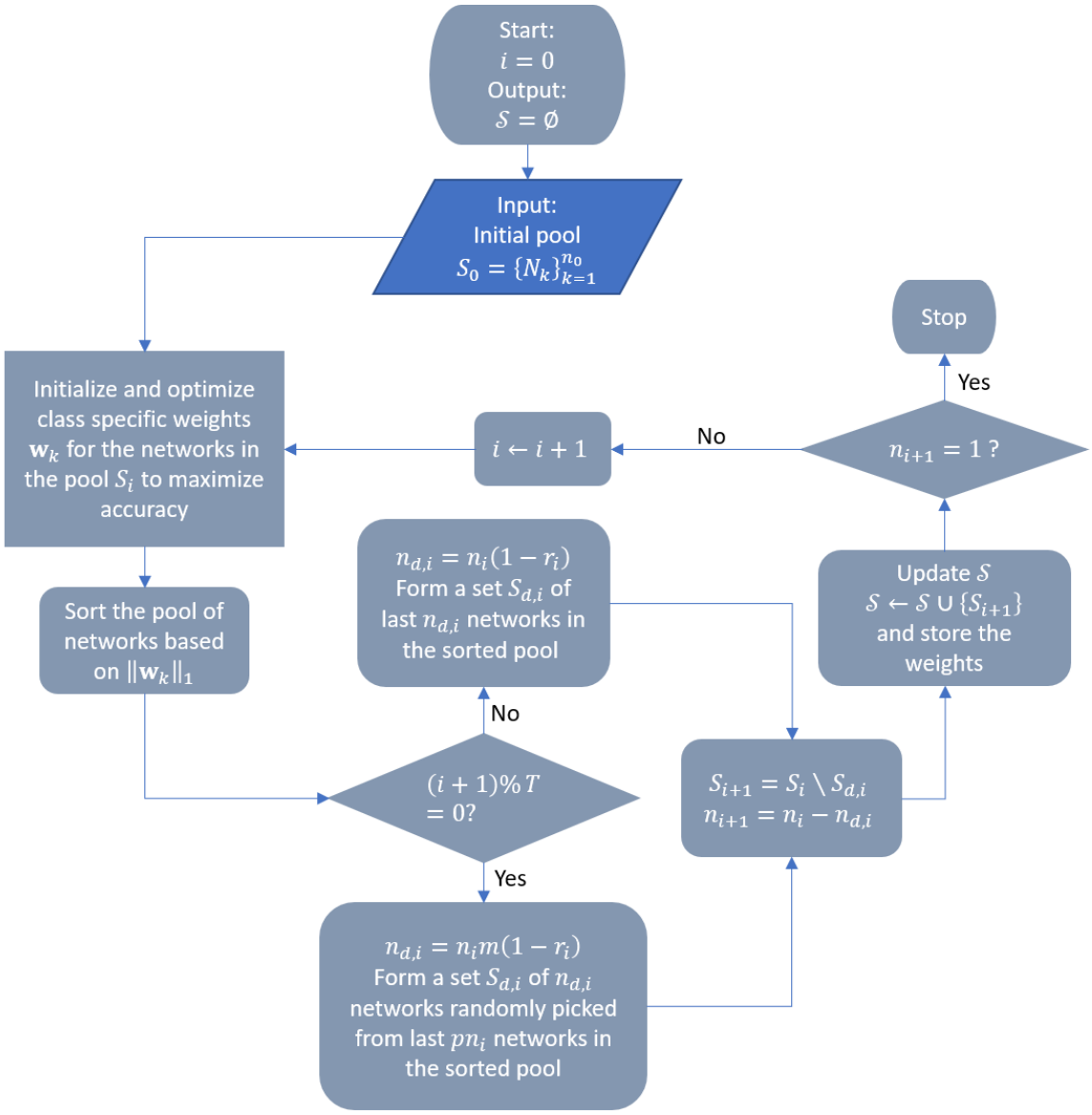


Figure 2.7 Flow chart of the ensemble pruning process. The meaning of the symbols is as follows: i: iteration number. $\mathcal{S}$: the set of ensembles, resulting after each iteration; S_i: ensemble on iteration i; n_i: the number of networks in the ensemble on iteration i; w_k: the weight vector for network k; T: the interval between random elimination of D^{2}NNs; $S_{d,i}$: the set of networks to eliminate from the ensemble on iteration i; $n_{d,i}$: the number of networks to eliminate from the ensemble on iteration i; r_i: the fraction of networks to retain on iteration i; m: the ratio of the number of randomly eliminated networks to the number of networks eliminated based on ranking; p: the fraction of the networks in the ensemble on which random elimination is applied. At the end

of the pruning process, $\mathcal{S}$ comprises a series of D^{2}NN ensembles (formed by S_i) of gradually decreasing size.

2.3.4 Ensemble pruning

The method we followed for ensemble pruning involved iterative elimination of the D^{2}NN members from the initial pool of 1252 unique networks based on a quantitative metric, which is indicative of an individual network's "significance" in the collective inference process. However, since a member's individual performance supremacy might not always translate to an improvement of the ensemble, during the iterative process we occasionally eliminated some members *randomly*. Ensemble pruning with intermittent random elimination of members was found to result in better performing ensembles compared to pruning without random elimination as detailed in the Results section and Table 2.2.

Our pruning method (see Fig. 2.7) was initiated with an ensemble that consisted of all the n_0 = 1252 individually trained D^{2}NN models. An ensemble class score z_c was defined as:

$$z_c = \sum_k w_{ck} z_{ck} \tag{2.3}$$

where z_{ck} is the score predicted for class c by the member/network k, see Eq. (2.1); and w_{ck} is the corresponding class-specific weight. The weight vectors $w_k = \{w_{ck}\}_{c=1}^{C}$, k = 1, 2, …, n_0, were optimized by minimizing the softmax cross-entropy loss of the class scores predicted by the ensemble of D^{2}NNs; C=10 denotes the total number of classes in our dataset. To reduce overfitting, the weights to the training data examples, an L2 loss term was also included in our pruning loss function:

$$\text{Pruning loss} = -E\left[\sum_{c=1}^{C} g_c \, log\left(\frac{exp(z_c)}{\sum_{c=1}^{C} exp(z_c)}\right)\right] + \alpha\left(\frac{1}{2}\sum_{k=1}^{n_0}\sum_{c=1}^{C} w_{ck}^2\right) \tag{2.4}$$

where α is set to be 0.001, $E[.]$ denotes the expectation over the image batch, and g_c represents the c^{th} entry of the ground truth label vector g. During the optimization of the ensemble, in each iteration of the back propagation, all the image samples in the validation set were fed into the ensemble model (i.e. the batch size equals to 5K); using training images for weight optimization during the ensemble pruning resulted in overfitting and therefore was not implemented. The class-specific weights were optimized using the gradient-descent algorithm (Adam [102]) for 10,000 steps. After optimizing the weights, the individual members/networks were ranked based on a quantitative metric. An intuitive choice for this metric could have been the individual prediction accuracy of each network. However, a better metric for measuring the significance of individual networks in an ensemble was found to be the L1 norm of the individual weight vectors optimized for validation accuracy. The superiority of the weight L1 norm as a metric was substantiated by the fact that it resulted in ensembles achieving much better blind testing accuracies, consistently. After ranking the members based on their weight vectors, a certain fraction of them was eliminated from the bottom (i.e., lowest ranked ones), and the procedure was repeated with the reduced ensemble until only one member was left in the ensemble. As mentioned earlier, after *T* iterations of the pruning, this member/network elimination was done *randomly* instead of the ranking based elimination. However, to avoid elimination of the members with the largest weights, random elimination was selected within a fraction *p* (which was 2/3 in our case) of the networks counted from the bottom. Once the pruning process was complete (see Fig. 2.7), a maximum allowed ensemble size (N_{max}) was set, and the ensemble with the best performance on the validation dataset

and satisfying the size limit was chosen. *The test image dataset was never used during the pruning process*.

To further explore different weight assignment schemes, we used a modified version of Eq. (2.3):

$$z_c = \sum_k w_{ck} \frac{w_{ck}^+ z_{ck}^+ - w_{ck}^- z_{ck}^-}{w_{ck}^+ z_{ck}^+ + w_{ck}^- z_{ck}^-} \tag{2.5}$$

where w_{ck}^+ and w_{ck}^- are the newly added weights assigned to each individual detector pair (positive and negative detector, respectively). Accordingly, the pruning loss defined in Eq. (2.4) was changed to be:

$$\begin{aligned} \text{Pruning loss} = -E\left[\sum_{c=1}^{C} g_c \, log\left(\frac{exp(z_c)}{\sum_{c=1}^{C} exp(z_c)}\right)\right] \\ +\alpha\left(\frac{1}{2}\sum_{k=1}^{n_0}\sum_{c=1}^{C} {w_{ck}}^2\right) \\ +\beta\left(\frac{1}{2}\sum_{k=1}^{n_0}\sum_{c=1}^{C} {w_{ck}^-}^2 + {w_{ck}^+}^2\right) \end{aligned} \tag{2.6}$$

where α and β are empirically set to be 0.001 each. During the pruning process, when the weight assignment scheme (1), described in the Results section, was used, all the weights w_{ck}, w_{ck}^+ and w_{ck}^- were *simultaneously* optimized for 10,000 iterations. In the weight assignment scheme (2), described in the Results section, the optimization of w_{ck} and (w_{ck}^+, w_{ck}^-) were performed alternatively; each time, one group of weights were optimized for 100 iterations, and in total 50

cycles were used to obtain an equivalent number of total iterations (10,000), the same as in scheme (1).

For all the training and optimization tasks detailed above, we used multiple desktop computers all with one or two GTX 1080 Ti graphical processing units (GPU, Nvidia Inc.), Intel® Core™ i7-8700 central processing unit (CPU, Intel Inc.) and 64 GB of RAM, running Windows 10 operating system (Microsoft Inc.). The typical required training time of one D^2NN model on a single GPU is ~3 hours. The time required for the iterative ensemble pruning process depends on the pruning hypermeters, varying between 0.75 to 7.5 hours.

Chapter 3 Time-lapse Image Classification using a Diffractive Network

In this chapter, I introduce time-lapse image classification to enhance the performance of standalone diffractive optical networks. Parts of this chapter have been published in the peer-reviewed research article "M. S. S. Rahman and A. Ozcan, 'Time-Lapse Image Classification Using a Diffractive Neural Network,' Advanced Intelligent Systems **5**, 2200387 (2023)," (Licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/).

Diffractive deep neural networks (D^2NNs) define an all-optical computing framework comprised of spatially engineered passive surfaces that collectively process optical input information by modulating the amplitude and/or the phase of the propagating light. Diffractive optical networks complete their computational tasks at the speed of light propagation through a thin diffractive volume, without any external computing power while exploiting the massive parallelism of optics. Diffractive networks were demonstrated to achieve all-optical classification of objects and perform universal linear transformations. Here we demonstrate, for the first time, a 'time-lapse' image classification scheme using a diffractive network, significantly advancing its classification accuracy and generalization performance on complex input objects by using the lateral movements of the input objects and/or the diffractive network, relative to each other. In a different context, such relative movements of the objects and/or the camera are routinely being used for image super-resolution applications; inspired by their success, we designed a time-lapse diffractive network to benefit from the complementary information content created by controlled or random lateral shifts. We numerically explored the design space and performance limits of time-lapse diffractive networks, revealing a blind testing accuracy of 62.03% on the optical classification of objects from the CIFAR-10 dataset. This constitutes the highest inference

accuracy achieved so far using a single diffractive network on the CIFAR-10 dataset. Time-lapse diffractive networks will be broadly useful for the spatio-temporal analysis of input signals using all-optical processors.

3.1 Introduction

Machine learning and artificial intelligence research has experienced rapid growth in the past two decades [103]. One of the core engines that has driven this growth is deep learning [30], permitting efficient and rapid training of deep artificial neural network models. The ability to train deep neural networks has revolutionized artificial intelligence, and electronics has been the undisputed platform of choice for implementing artificial neural networks. Specialized processing hardware such as Graphics Processing Units (GPUs) are widely used today for deep learning. However, these electronic processors are power-hungry and bulky, making researchers wary of the environmental impact of machine learning [104,105]. Therefore, there is strong interest in low-power and fast computing platforms for machine learning applications. Optical computing has been identified as a promising potential alternative for such purposes because of the large bandwidth, high speed, and massive parallelism of optics [106].

Diffractive deep neural networks (D^2NNs), also known as diffractive optical networks or diffractive networks, form a passive all-optical computing platform that exploits the diffraction of light waves to perform computation [38]. These diffractive networks are composed of several spatially engineered surfaces, separated by free-space. The diffractive features/elements of a layer, also termed ‘diffractive neurons’, locally modulate the amplitude and/or the phase of the light incident upon the layer. Successive modulation by and diffraction through the layers give rise to

an all-optical transformation between the input and the output fields-of-view at the speed of light propagation without any external power. The amplitude and/or the phase values of the diffractive neurons corresponding to a desired optical transformation or computational task are trained/learned through a digital computer using deep learning. Once the training is complete, the layers can be fabricated and assembled to form a 'physical' network that performs the desired computation in a passive manner and at the speed of light propagation. Diffractive networks can achieve universal linear transformations [32,33,107], and various applications using diffractive processors have been demonstrated such as object classification, pulse processing, imaging through random diffusers, hologram reconstruction, quantitative phase imaging, class-specific imaging, super-resolution image display, all-optical logic operations, beam shaping and orbital angular momentum mode processing, among others [39,53,78,80,108,109,42,54,110,111,34,45,112–120].

While diffractive networks have shown competitive performance on the classification of relatively simpler objects, for example, hand-written digits and fashion products [39], for more complex natural objects such as those from the CIFAR-10 dataset [87], their performance gap compared to the classification accuracy of electronic neural networks is still large [39,121]. Ensemble learning through multiple D^2NNs has been demonstrated to improve the inference and generalization of diffractive networks at the cost of reducing the compactness and simplicity of the optical hardware [121].

In this work, we demonstrate, for the first time, a 'time-lapse' image classification scheme with a stand-alone diffractive optical network that significantly enhances the inference and generalization performance of diffractive computing. In this scheme, the objects and/or the diffractive network laterally move relative to each other, either randomly or in a controlled manner,

during the detector integration time, enriching the information provided to the diffractive network. In a different context and application, lateral shifts of the object of interest relative to the imager have been routinely used for pixel super-resolution imaging, enhancing the resolution of the reconstructed images [122–125]. Inspired by the success of these pixel super-resolution approaches, here we use the controlled or random relative displacements between the input objects and the diffractive network for time-lapse image classification and report a numerical blind testing accuracy of 62.03% for the classification of grayscale CIFAR-10 images, which constitutes the highest classification accuracy for this dataset achieved so far using a single diffractive optical network. In addition to significantly advancing the inference and generalization performance of D^2NNs, these time-lapse diffractive networks can also find broader use in the all-optical processing of spatio-temporal information of a scene or object.

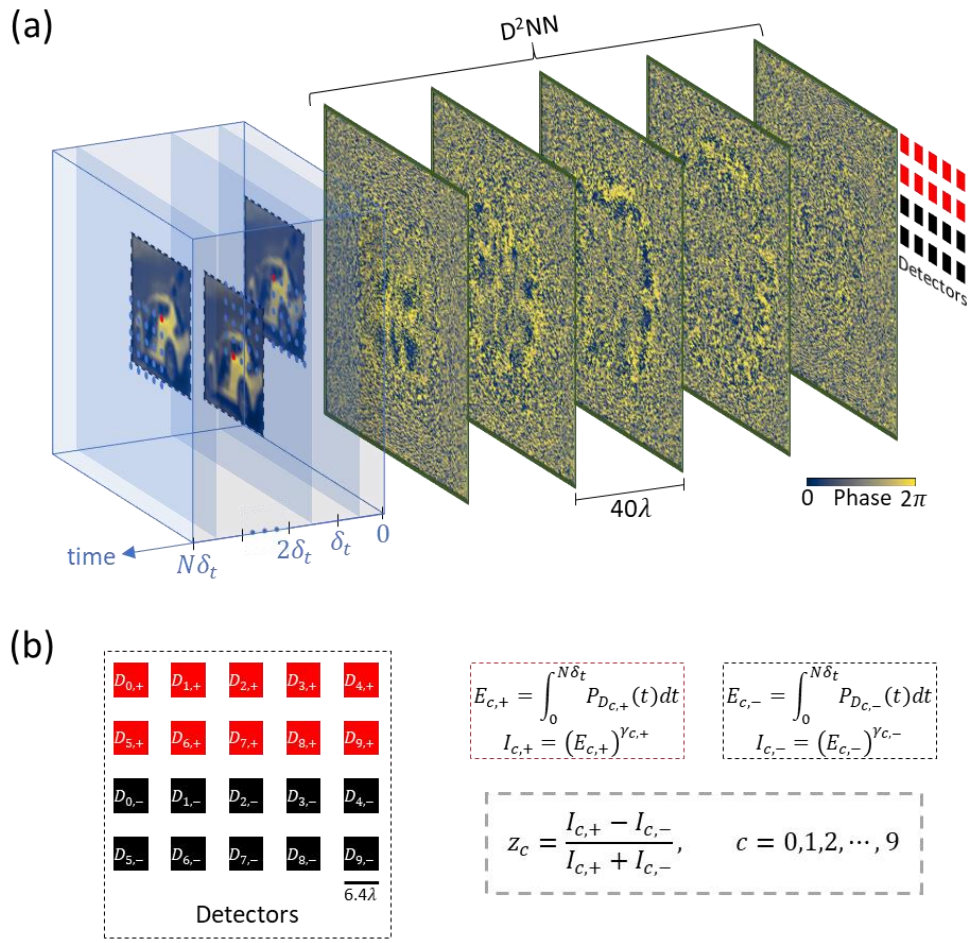


Figure 3.1 Time-lapse image classification using a D²NN. (a) A diffractive network with 5 phase-only diffractive layers followed by 20 detectors at the detector plane for differential image classification. The integration time of each detector is $N\delta_t$. During each one of the N intervals of δ_t duration, the center of the object is laterally displaced to a new point (red circle); these lateral displacements can be entirely random or follow a predefined grid (blue circles). (b) Labeling of the detectors where $D_{c,+}$ ($D_{c,-}$) denote the positive (negative) detectors assigned to class c. The differential class-scores z_c are used for the final classification decision based on the maximum score.

3.2 Results

The concept of time-lapse image classification with a diffractive network is illustrated in Fig. 3.1. A diffractive network comprising 5 phase-only diffractive layers, axially separated by 40λ, is

placed between the object plane and the detector plane. The detector plane includes 20 detectors [39]: 2 detectors for each class c of the CIFAR-10 dataset, i.e., a 'positive' detector $D_{c,+}$ and a 'negative' detector $D_{c,-}$. The integration time of the output detectors is assumed to be $N\delta_t$, where N is the number of lateral object shifts and each of the N individual shifts has an equal integration time of δ_t. Without changing our conclusions, in alternative implementations, the diffractive network can also laterally move relative to the static object, or both the object and the diffractive network can laterally move at the same time. Each detector $D_{c,\pm}$ is assigned an exponent $\gamma_{c,\pm}$ which operates on the integrated detector power to yield the detector signal $I_{c,\pm}$ (see Fig. 3.1 and the Methods section). We will report diffractive classification results under two different conditions: (1) the exponents are assumed to be trainable, and (2) non-trainable, fixed as $\gamma_{c,\pm} = 1$. The normalized differential class scores $z_c = \frac{I_{c,+} - I_{c,-}}{I_{c,+} + I_{c,-}}$ are calculated from these detector signals, and the prediction/inference is made in favor of the class receiving the highest differential optical score (see Fig. 3.1).

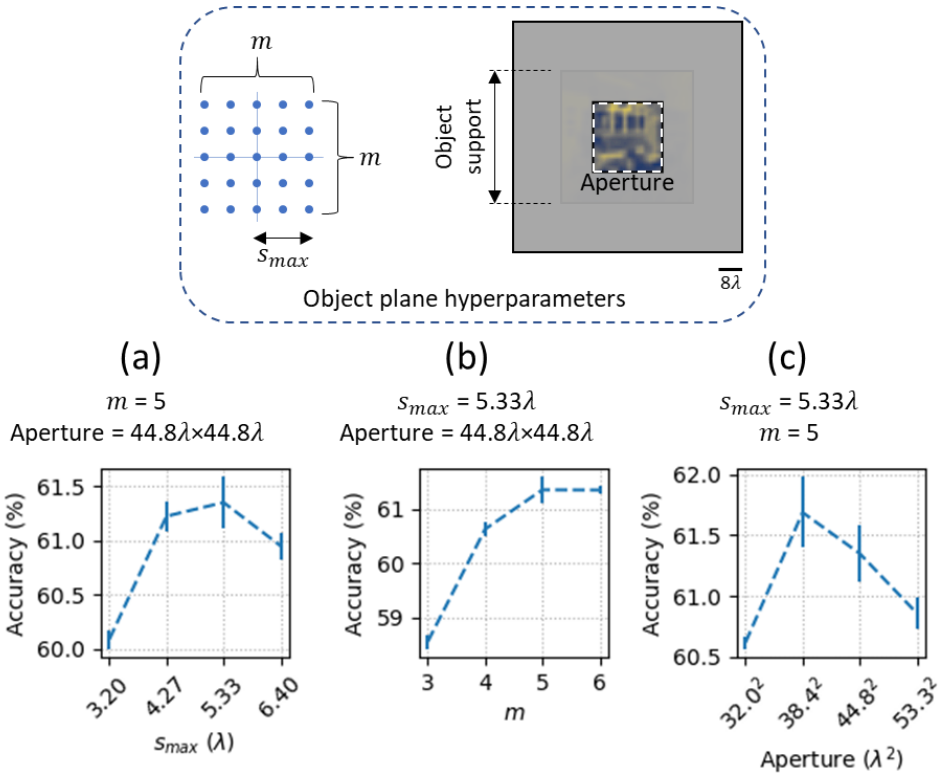


Figure 3.2 Dependence of the time-lapse image classification D^{2}NN performance on the hyperparameters of the input plane. Top left: Grid of points representing the lateral displacements of the input objects during the training, defined by the two hyperparameters, s_{max} and m. s_{max} represents the maximum displacement along either the vertical or the horizontal direction, whereas m^2 is the total number of grid points. Top right: the dashed white square represents the input aperture immediately following the object, the area of which is another hyperparameter. (a) Dependence of the blind testing accuracy on s_{max} with m and the input aperture kept constant. (b) Effect of m on the blind testing accuracy of the trained time-lapse diffractive classifiers as s_{max} and the input aperture are kept constant. (c) Dependence of the blind testing accuracy on the input aperture size while s_{max} and m are kept constant. For (a)-(c), the data points and the error bars represent the mean and the standard deviation values, respectively, calculated from three designs, which are obtained by training three different time-lapse D^{2}NN classifiers for the same set of hyperparameter values. The curves are linearly interpolated between the data points.

For all the D2NNs reported in this work, each trainable diffractive layer consists of 200×200 diffractive elements (diffractive neurons) of size $0.53\lambda \times 0.53\lambda$. The objects are assumed to be phase-only and the diffractive networks are trained using the grayscale CIFAR-10 dataset (refer to the Methods section for details). The hyperparameters that define the grid of lateral displacements

of the objects during the time-lapse image classification are s_{max} and m, where s_{max} is the maximum (relative) lateral displacement along x/y and m^2 refers to the total number of points on the grid, see Fig. 3.2. The size of the input aperture is another hyperparameter that affects the classification performance of the time-lapse diffractive networks. The impact of these hyperparameters, s_{max}, m and the input aperture size, on the performance of time-lapse diffractive classifiers is shown in Fig. 3.2. The classification performance is quantified by the blind testing accuracy of the networks on 10,000 previously unseen images belonging to the test set of the CIFAR-10 dataset. To obtain each data point in Fig. 3.2, we trained 3 different diffractive networks with the same hyperparameters and calculated the mean and standard deviation of blind testing accuracies of these 3 trained networks. We see from Fig. 3.2a that as s_{max} is increased from 3.20λ to 6.40λ (while keeping m = 5 and the aperture size = $44.8\lambda \times 44.8\lambda$ constant), the mean blind testing accuracy increases until $s_{max} = 5.33\lambda$, where it reaches its highest value of 61.35%. Beyond $s_{max} = 5.33\lambda$, the mean classification accuracy starts to decrease. In Fig. 3.2b, we set $s_{max} = 5.33\lambda$, aperture size = $44.8\lambda \times 44.8\lambda$ and vary m. As m is varied between 3 and 6, the mean accuracy increases rapidly from 58.56% to 61.35% until m = 5, beyond which the mean accuracy reaches a plateau. For Fig. 3.2c, we selected m = 5 and $s_{max} = 5.33\lambda$ (as optimized from Figs. 3.2a and 3.2b) and the width of the input aperture was varied between 32.0λ and 53.3λ. The highest mean accuracy (Fig. 3.2c) is observed for an input aperture size of $38.4\lambda \times 38.4\lambda$, which is smaller than the object support $44.8\lambda \times 44.8\lambda$. We compared this observation with its counterpart for time-static diffractive image classification (see Table 3.1), where the aperture size corresponding to the highest mean blind testing accuracy is larger than the object support. This comparison indicates that a time-lapse diffractive network prefers a relatively smaller input field-of-view compared to its time-static counterparts.

Aperture area (λ^2)	Accuracy (%)
38.4 × 38.4	50.80 ± 0.18
44.8 × 44.8	51.92 ± 0.29
51.2 × 51.2	52.76 ± 0.35
57.6 × 57.6	52.83 ± 0.13
64.0 × 64.0	52.76 ± 0.02
76.8 × 76.8	52.12 ± 0.17

Table 3.1: Dependence of the performance of time-static diffractive networks on the input-aperture size.

Next, we juxtapose a time-lapse image classification diffractive network with a time-static diffractive network; see Fig. 3.3. For this comparison, we chose the time-lapse diffractive network with the best individual blind testing accuracy (62.03%) among the networks constituting the results of Fig. 3.2 and the time-static diffractive network with the best individual blind testing accuracy (53.14%) among the networks constituting the results of Table 3.1. For the time-lapse image classification diffractive network, the hyperparameters corresponding to the highest individual accuracy were m = 5, $s_{max} = 5.33\lambda$ and input aperture size = $38.4\lambda \times 38.4\lambda$; while for the time-static network, the input aperture size corresponding to best individual accuracy was $51.2\lambda \times 51.2\lambda$. Another difference to be noted between the time-static and the time-lapse diffractive networks chosen for comparison in Fig. 3.3 is that for the time-static one, the detector exponents were not trainable, i.e., $\gamma_{c,\pm} = 1$, whereas the detector exponents were trainable for the time-lapse network. The reason for this selection is that, unlike the time-lapse diffractive networks, time-static diffractive networks showed overfitting when the detector exponents are trainable, leading to inferior generalization; see Table 3.2.

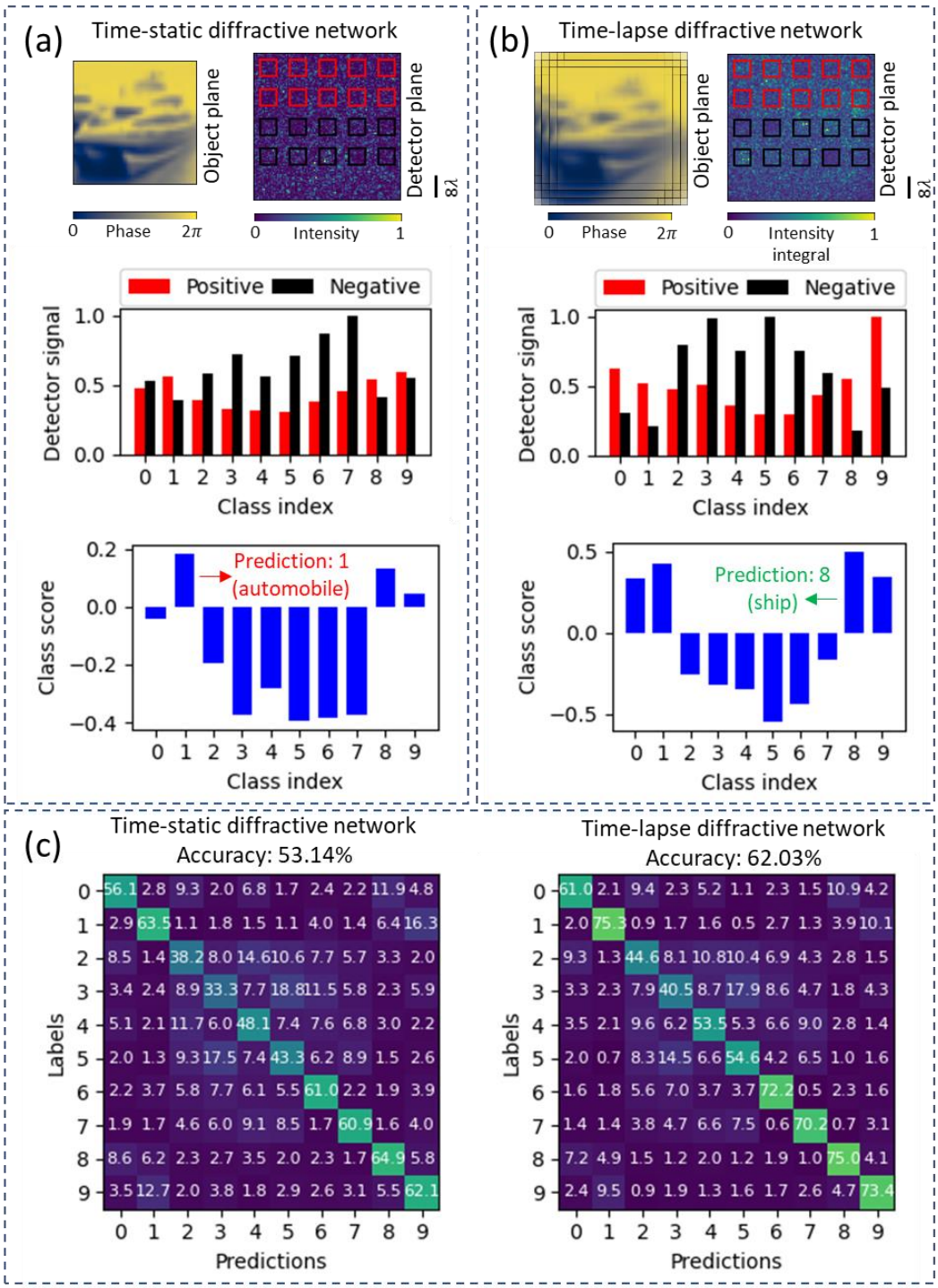


Figure 3.3 Comparison between a time-static and a time-lapse diffractive image classifier. (a) Detector plane intensity, detector signals, and class scores from the time-static network for an object (a ship) from the data class 8 of the CIFAR-10 dataset. Based on the class scores, the object is misclassified to be an automobile. (b) Integrated detector plane intensity, detector signals, and class scores from the time-lapse diffractive network for the same object, which is correctly

classified as a ship. (c) Confusion matrices for the two networks, evaluated by blind testing on 10000 CIFAR-10 test images. The overall accuracies of the networks are 53.14% and 62.03%, respectively. The training parameters of these two networks are shown in Figs. 3.5a and 3.5b.

	Time-static*	Time-lapse**
$\gamma_{c,\pm}$ (trainable)	50.47 ± 0.90	61.69 ± 0.36
$\gamma_{c,\pm} = 1$ (non-trainable)	52.83 ± 0.13	60.27 ± 0.10

*Hyperparameter: Aperture = 57.6λ × 57.6λ

**Hyperparameters: s_{max} = 5.33λ, m = 5, Aperture = 38.4λ × 38.4λ

Table 3.2: Dependence of the performance of time-static and time-lapse diffractive networks on the trainability of $\gamma_{c,\pm}$.

For an example object from the image class ‘ship’ (true label: 8), we show in Fig. 3.3a the detector plane intensity, detector signals and the class scores for the time-static network; similarly, in Fig. 3.3b we show the time-integral of the detector plane intensity, detector signals and the class scores for the time-lapse image classification network. While the time-static network misclassifies the object for an ‘automobile’, the time-lapse image classification diffractive network correctly predicts the object to be a ‘ship’ (predicted label: 8). We also show in Fig. 3.3c the confusion matrices calculated over 10,000 test images of the CIFAR-10 dataset: the time-lapse image classification diffractive network performs consistently better than the time-static one for all the CIFAR-10 data classes.

Note also that the time-lapse image classification diffractive network designed with non-trainable detector exponents (i.e., $\gamma_{c,\pm} = 1$) achieved a blind testing accuracy of 60.35% on the same grayscale CIFAR-10 test dataset (see Fig. 3.4), performing much better than the time-static one for all the CIFAR-10 data classes. The diffractive layers for all these networks are shown in Fig. 3.5c.

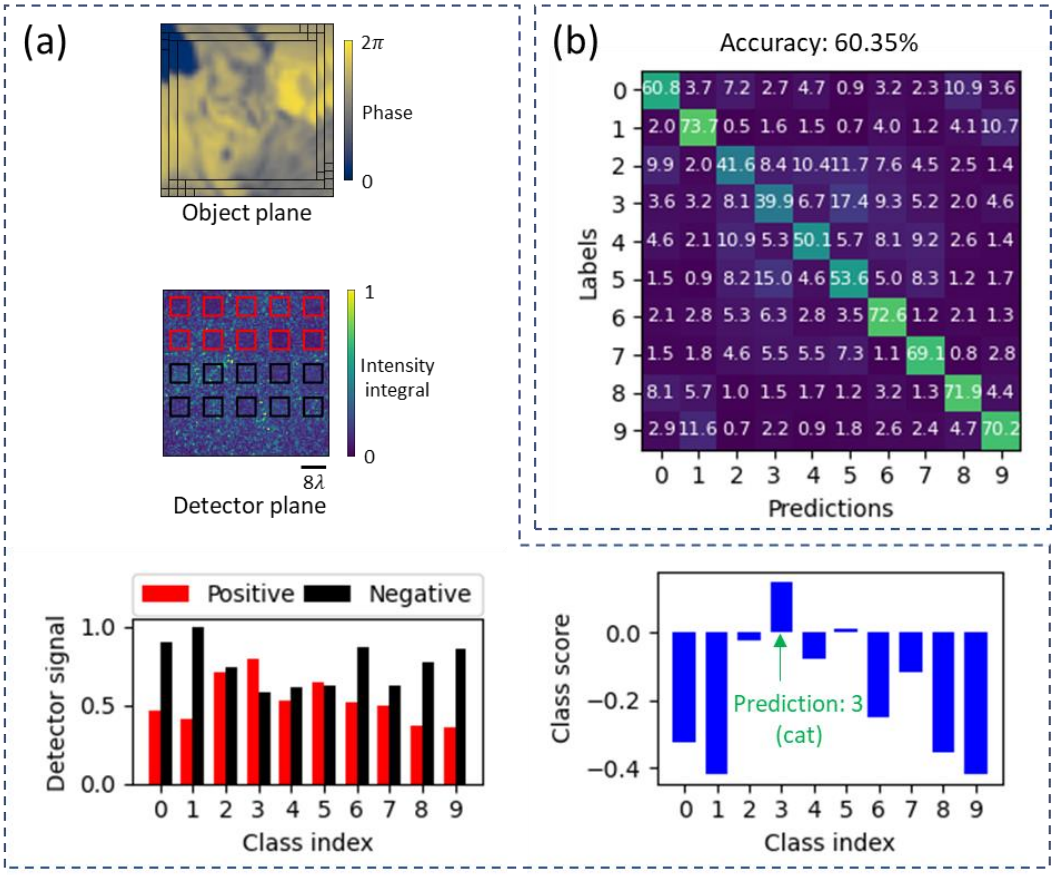


Figure 3.4 Performance of a time-lapse diffractive network without trainable detector exponents (i.e., $\gamma_{c,\pm} = 1$). (a) Integrated detector plane intensity, detector signals, and class scores from the network for an input object (a cat) from data class 3 of the CIFAR-10 dataset. Based on the class scores, the object is correctly classified. (b) Confusion matrix for the network, evaluated by blind testing on 10000 images from the CIFAR-10 test set. The overall accuracy of the network on CIFAR-10 test set is 60.35%. The phase profiles of the trained diffractive layers of this network are shown in Fig. 3.5c.

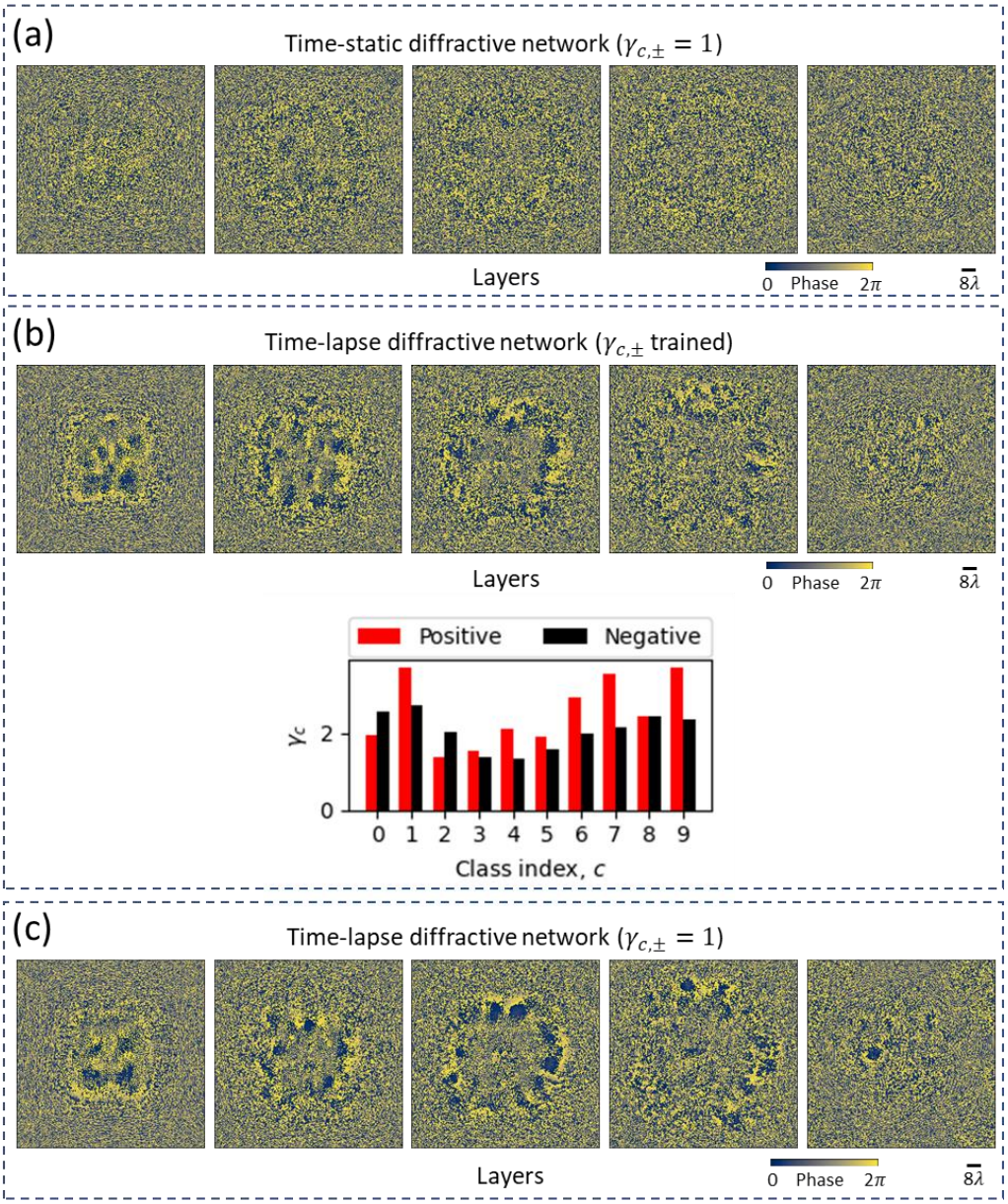


Figure 3.5 Parameters of the diffractive models of Fig. 3.3 and Fig. 3.4. (a) Trained diffractive layer phase values for the time-static D^2NN of Fig. 3.3a. (b) Trained diffractive layer phase values and detector exponents for the time-lapse D^2NN of Fig. 3.3b. (c) Trained diffractive layer phase values for the time-lapse D^2NN of Fig. 3.4, for which the detector exponents were not trainable (i.e., $\gamma_{c,\pm} = 1$).

During the training of the time-lapse diffractive networks, we followed a method similar to the 'dropout' method, which is used in deep learning to reduce overfitting and improve the

generalization of a trained model [126]. We defined a hyperparameter p which is the probability that a point on the object-plane grid is 'active' during training, i.e., the probability that the object is positioned at that lateral point during the signal integration at the detector. All the time-lapse networks described thus far were trained with p = 0.5. As we describe below, the resilience of the trained time-lapse image classification diffractive networks to deviations from the training settings can be improved by a proper choice of p, which is intuitively similar to the dropout strategy in deep learning literature.

Related to this hyperparameter p, next, we explored the impact of decreasing the number of lateral shifts, N, on the blind testing accuracy of time-lapse classifiers: see Fig. 3.6. The value for each data point in Fig. 3.6 represents the mean of the classification accuracies over 25 independent blind tests with the same N. For Fig. 3.6a, these N lateral displacements were restricted to coincide with the pre-determined training grid points, and for the case of N < m^2, $m^2 - N$ of the m^2 lateral shifts were randomly eliminated (not used). For Fig. 3.6b, however, the N lateral displacements were randomly selected without following the training grid points. As we can see in Fig. 3.6a, the blind testing accuracy decreases as N is decreased; however, the slope of this performance degradation varies depending on the training hyperparameter p. For example, in the case of the time-lapse image classification diffractive network shown in Fig. 3.3b, trained with p = 0.5 (green curve in Fig. 3.6a), the test accuracy drops from 62.03% to 60.69% and 59.37% as N decreases from 25 to 15 and 10, respectively. Compare this with the case of a time-lapse diffractive network trained with p = 1.0 (red curve in Fig. 3.6a), for which the classification accuracy is affected much more severely and decreases from 61.61% to 59.61% and 57.45% as N is decreased from 25 to 15 and 10, respectively. We see that networks trained with lower p values show less sensitivity to

decreasing N, which is further corroborated by the curves corresponding to two other time-lapse diffractive networks trained with p = 0.2 and p = 0.3.

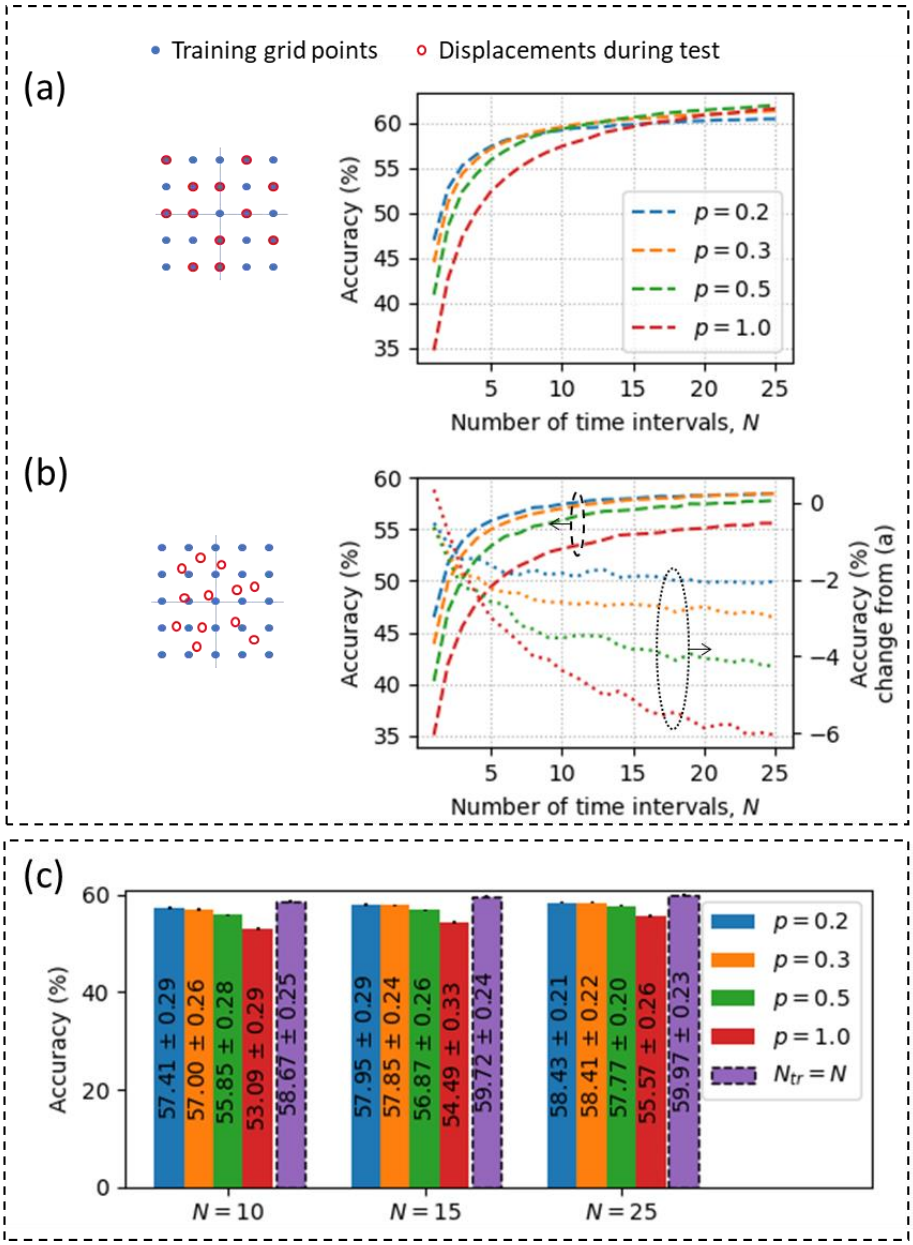


Figure 3.6 Impact of the lateral shifts. Impact of decreasing the number of lateral shifts (N) on the blind testing accuracy while confining the input object displacements to the training grid points; the time-lapse D^2NNs were trained with different p values. Diffractive networks trained with lower p values are less affected by a decrease in N. (b) Impact of *random off-grid* lateral displacements of input objects during the blind testing of time-lapse diffractive networks trained with displacements confined to a predefined grid. Time-lapse diffractive classifiers trained with

lower p values are more resilient to such deviation from the training settings. (c) Improvement of blind testing accuracies for a given N, by training the time-lapse diffractive network with $N_{tr} = N$ arbitrary/random displacements within the range $2s_{max}$×$2s_{max}$ instead of training with a set of fixed lateral displacements defined by a pre-determined lateral grid. For (a)-(c), the values (errors) corresponding to the data points represent the mean (standard deviation) values calculated through the blind testing of the same trained network 25 times, every time with N arbitrary lateral displacements of the input objects.

Another advantage of training with lower p values is decreased sensitivity to the exact object positions (see Fig. 3.6b). For Fig. 3.6b, we selected the N lateral displacements without following the training grid points, allowing the object to be displaced (during the time-lapse imaging process) to N arbitrary, randomly selected points within the area $2s_{max}$×$2s_{max}$. In general, for a given N, the blind testing accuracies corresponding to such arbitrary displacements (left y-axis of Fig. 3.6b) are lower than their counterparts for the on-grid displacements shown in Fig. 3.6a. However, the degradation in classification accuracy, which is shown on the right y-axis of Fig. 3.6b, is much smaller when p is lower. For example, at N = 25, the mean accuracy drop is ~2% for the diffractive network trained with p = 0.2, whereas the accuracy drop is ~6% for the p = 1.0 diffractive network.

The accuracy of time-lapse diffractive network-based image classifiers for arbitrary lateral displacements of the input objects can be improved by utilizing such random displacements of the objects during the training, rather than training with a pre-determined grid of lateral displacements. For this, the training hyperparameters p and m can be absorbed into a single hyperparameter N_{tr}, where N_{tr} refers to the number of arbitrary displacements within $2s_{max}$×$2s_{max}$. To demonstrate this, we trained three time-lapse diffractive networks with N_{tr} = 10, N_{tr} = 15 and N_{tr} = 25 and compared their accuracies for N = 10, N = 15, and N = 25 arbitrary displacements of the input objects, respectively, against the classification accuracies of the time-lapse diffractive networks reported in Figs. 3.6a and 3.6b. The result of this comparison is shown in Fig. 3.6c: for N = 10, N

= 15 and N = 25 arbitrary lateral displacements during the time-lapse imaging process, the mean blind testing accuracies of the corresponding $N_{tr} = N$ diffractive networks are 1.26%, 1.77%, and 1.54%, respectively, higher than the accuracies of the p = 0.2 time-lapse diffractive network. This generalization improvement and the inference accuracy increase are due to using arbitrary random lateral displacements of the input objects during the training process instead of blindly applying such random lateral shifts only during the testing phase.

3.3 Discussion

In previous work, we reported a significant improvement in diffractive network inference performance by ensemble learning and combining the output of several different diffractive networks. For example, mean blind testing accuracies of 61.14% and 62.13% on the CIFAR-10 test set were reported for ensembles of 14 and 30 different D2NNs, respectively [121]. However, the improvement with such a strategy is accompanied by a sacrifice in the compactness of the optical hardware and increased complexity in aligning several diffractive networks within the ensemble. Another shortcoming of ensemble learning of diffractive networks is the large training time. In our previous work, 1252 diffractive models were trained, and ensemble pruning was then performed to arrive at the final design [121]. Time-lapse diffractive network-based image classification provides blind testing accuracies comparable to ensemble learning with only a single trained diffractive network. For comparison, the time-lapse diffractive network of Fig. 3.3b gives 62.03% blind testing accuracy on CIFAR-10 test images. The trade-off for such an advantage is the increase in the imaging/classification time due to the lateral shifts of the objects. However, the alignment and synchronization requirements associated with diffractive network ensembles are

evaded. Also, the training of a time-lapse diffractive classifier takes ~20 hours on an NVIDIA GeForce RTX 3090 GPU (see the Methods section), which is orders of magnitude less than the time required to design an ensemble of diffractive networks working together.

Regarding the implementation of time-lapse diffractive network-based image classification, Spatial Light Modulators (SLMs) can be used to perform the lateral displacements of the input objects digitally if a digital representation of each object is available. In an alternative implementation, the diffractive layers and the detectors could be mounted on a movable stage to shift the entire system with respect to the object or input FOV. Perhaps, the simplest implementation of time-lapse diffractive network-based image classification would exploit the natural jitter or movement of the input objects during the integration time of the class detectors. As shown in Fig. 3.6c, ~60% blind testing accuracy on CIFAR-10 test images can be reached with arbitrary object displacements during the time-lapse inference.

A point worth emphasizing is that although the displacements in Fig. 3.6b (as well as in the other figures) are shown to be discrete and mostly disconnected to reflect our numerical analysis, the proposed scheme is also applicable for the continuous, connected motion of objects as it occurs in real life during time-lapse imaging. As long as the motion of the object covers the training lateral range $2s_{max}\times 2s_{max}$ within the detector integration time, we expect a performance improvement due to time-lapse inference over time-static inference. We corroborate this fact with additional numerical analysis, the results of which are presented in Fig. 3.7. In the simulations shown in Figs. 3.7a and 3.7b, we tested the already trained time-lapse diffractive network of Fig. 3.6(c) for different continuous object trajectories (with connected sets of random points), irregularly spanning an area of $2s_{max}\times 2s_{max}$ – matching the training scheme in terms of the maximum lateral displacements. For these irregular, continuous trajectories that fit within an area of

$\sim 2s_{max} \times 2s_{max}$ on the input plane, the average blind testing accuracy over 10,000 CIFAR-10 test images was 60.68% and 61.00% for Figs. 3.7a and 3.7b, respectively. In contrast, we obtained a decreased blind testing accuracy of 56.18% for the larger continuous/connected trajectory of Fig. 3.7c that spans an area larger than $2s_{max} \times 2s_{max}$; this reduced inference accuracy is due to the object displacements that lie outside the region for which the network was not trained (green dashed square in Fig. 3.7). The accuracy becomes 61% when only the portion of this trajectory within the $2s_{max} \times 2s_{max}$ area is used, as shown in Fig. 3.7b.

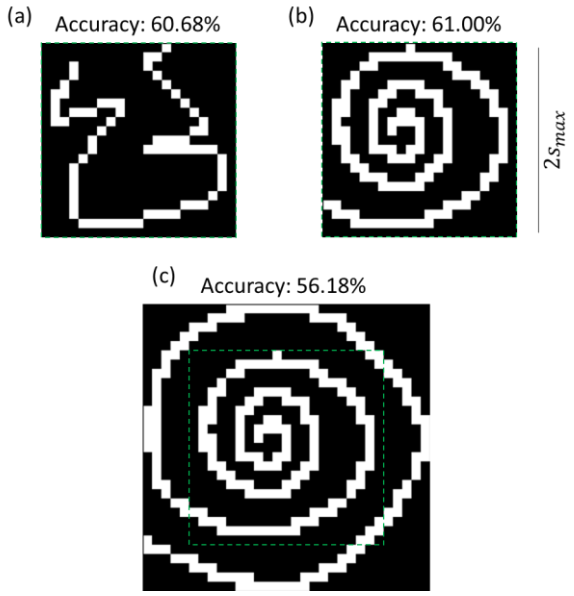


Figure 3.7 Example object test trajectories during time-lapse inference and the corresponding blind testing accuracies on CIFAR-10 test images for the N_{tr} = 25 time-lapse diffractive network of Fig. 3.6c. The dashed green square indicates the $2s_{max} \times 2s_{max}$ area over which the input objects were arbitrarily/randomly displaced during the training phase of the time-lapse diffractive network. For this network, s_{max}=5.33λ.

One of the challenges regarding the practical implementations of our proposed framework is the realization of the relative displacements between the objects and the input aperture of the diffractive network. The tolerance of the time-lapse diffractive network to imprecise displacements as well as deviations from the training conditions (such as the one shown in Fig. 3.7c) can be

improved by using 'vaccination' schemes [84], where such deviations are modeled as random noise during the training phase to make the diffractive network designs robust to them. However, as discussed earlier in relation to the results of Fig. 3.6c and Fig. 3.7, the proposed time-lapse inference scheme provides a significant improvement over time-static classification even for arbitrary/random displacements of the input objects (regardless of being continuous/connected or disconnected). The sensitivity of a time-lapse diffractive network to these random displacements between the object and the diffractive network input aperture can become an issue if the input objects move significantly outside of the lateral shift area used during the training, as shown in Fig. 3.7c. This can potentially be addressed by further increasing s_{max} in the training phase. Other practical concerns, such as fabrication errors and mechanical misalignments between the diffractive layers, could also be addressed similarly using a form of vaccination [84] during the training phase, which guides the evolution of the diffractive network layers to be resilient against such random imperfections; in general, a down-side of the vaccination strategy is a reduction in the independent degrees of freedom available within the diffractive network layers, which can be mitigated by, e.g., increasing the width of each diffractive layer or using a deeper network architecture with more layers. To overcome the detection noise associated with optoelectronic detectors at the output plane, the SNR of the time-lapse diffractive network can be improved by penalizing low output diffraction efficiencies during the training phase, which will be detailed next.

In terms of the photon budget, the output diffraction efficiency of the time-lapse diffractive image classifier of Fig. 3.3b is 0.6%, which refers to the ratio of the total incident energy on the output detectors with respect to the total energy at the input aperture. The output diffraction efficiency of a diffractive network can be significantly improved by introducing an additional loss term that penalizes low efficiency designs. This diffraction efficiency-related term can be added

to the total training loss function with a linear weight (η), which is a hyperparameter that can be tuned depending on the acceptable trade-off between the inference accuracy and the output diffraction efficiency (see the Methods section). To illustrate this trade-off, we trained two new time-lapse diffractive networks with η = 0.85 and η = 0.9, where the former resulted in a blind testing accuracy of 60.66% (CIFAR-10) and an output diffraction efficiency of 3.2%, whereas the latter, η = 0.9 design, resulted in a blind testing accuracy of 58.02% and an output diffraction efficiency of 32.7%.

We should also note that the inference accuracy of the time-lapse diffractive network of Fig. 3.3b is nonuniform across different classes, e.g., varying from 40.5% for class 3 to 75.3% for class 1. This inference nonuniformity from class to class can be reduced by using class-specific loss weights during the training phase. To demonstrate this, we calculated the mean loss (L_c) of the diffractive network of Fig. 3.3b for all the classes $c = 0, 1, \cdots, 9$. Then, we trained two new diffractive networks: one with $w_c \propto L_c$ and the other with $w_c \propto \exp(2L_c)$, where w_c refers to the class-specific weights used to calculate the training loss function. We show these class-specific loss weights and the resulting confusion matrices for these time-lapse diffractive networks in Fig. 3.8. For the time-lapse diffractive network trained with $w_c \propto L_c$, the average blind testing accuracy is 61.36% (for CIFAR-10) and the individual class accuracies range from 42.9% to 72.5%; on the other hand, for the time-lapse diffractive network trained with $w_c \propto \exp(2L_c)$, the average blind testing accuracy is 60.26% and the individual class accuracies range from 50.3% to 67.4%, see Fig. 3.8c. These results indicate that the statistical inference nonuniformity from class to class can be reduced, if desired, by using class-specific loss weights during the training phase with a small trade-off in the average classification accuracy of the time-lapse diffractive network.

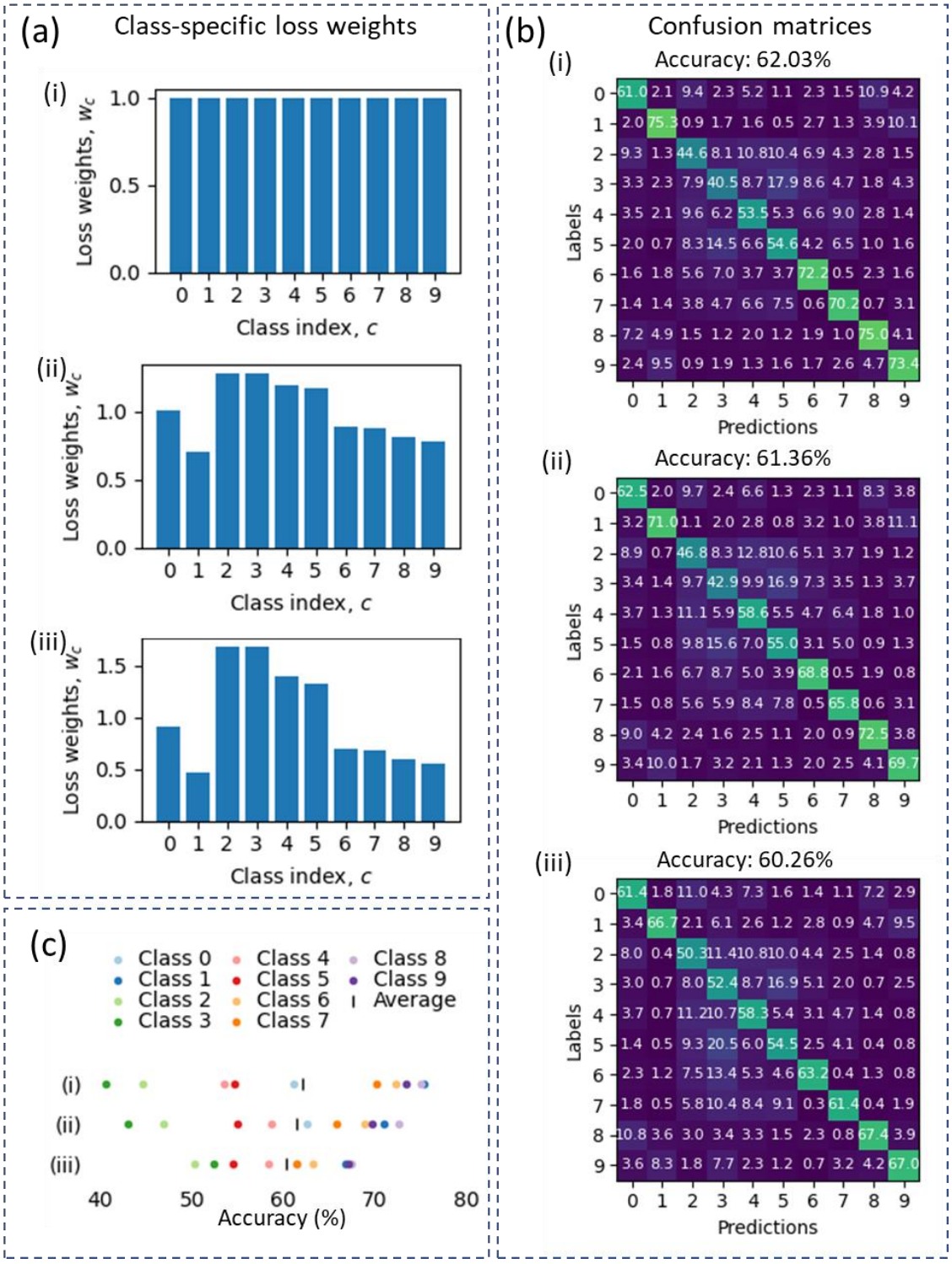


Figure 3.8 Investigation of the individual class accuracies. (a) Class-specific loss weights used for the loss calculation in the training of 3 different time-lapse diffractive networks: (i), (ii), (iii). (i) is the same as the time-lapse diffractive network of Fig. 3.3b, the training of which used equal

loss weights for all classes, i.e., $w_c = 1$. For (ii), $w_c \propto L_c$ while for (iii), $w_c \propto \exp(2L_c)$. L_c is the mean loss of network (i) over examples of class c. w_c is scaled such that $\sum_c w_c = 10$. (b) Confusion matrices for the networks (i)-(iii). (c) Comparison of the time-lapse diffractive networks (i), (ii) and (iii) in terms of individual class accuracies and average classification accuracy. The spread of the individual class accuracy values is reduced for networks (ii) and (iii) while the average accuracies remain greater than 60% for CIFAR-10 test images.

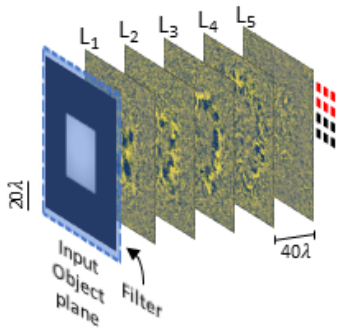
Although a time-lapse diffractive network has access to enriched information regarding an input sample due to lateral shifts of its field-of-view, it still needs a rich training dataset for a successful design and generalization. Stated differently, time-lapse diffractive network training with a certain number (N) of lateral displacements of the objects cannot compensate for a reduction of the training dataset size since the unique spatial features of each image of the training dataset help the diffractive networks' generalization. For example, training a time-lapse diffractive network with m = 2 (i.e., N = 4 lateral displacements) using a reduced dataset that has 4–fold smaller number of examples achieved a lower blind testing accuracy of 51.78% for CIFAR-10. As another example, training a time-lapse diffractive network with m = 5 (i.e., N = 25 lateral displacements) using a reduced dataset that has 25–fold smaller number of examples achieved an even lower blind testing accuracy of 44.62%. These indicate that the generalization improvement of time-lapse diffractive networks needs diverse training datasets and time-lapse imaging alone cannot overcome poor generalization due to reductions in the size of the training dataset.

The use of a pair of detectors for each data class in a differential detection scheme was shown to improve the inference performance of time-static diffractive networks [39]. This differential detection scheme helps to virtually circumvent the nonnegativity constraint on the class-scores, imposed by optoelectronic detectors which are sensitive only to the intensity of light. This inference performance improvement due to the differential detection scheme is also translated to time-lapse diffractive networks. For example, we trained, from scratch, a time-lapse diffractive

network with the same hyperparameters as those of the network shown in Fig. 3.3b, but this time used the sum of the optical signals from the two detectors assigned to each class as the class-score, instead of using their difference signal. The blind testing accuracy with this non-differential detection scheme was 61.02% compared to the 62.03% blind testing accuracy of the differential scheme.

While time-lapse image classification significantly boosts the inference accuracy of a single D2NN on the classification of complex objects, there remains plenty of room for improvement to potentially close the large performance gap with their electronic counterparts, convolutional deep neural networks [121]. One possible avenue for such an improvement could be the incorporation of ensemble learning with time-lapse image classification, where the outputs of diversely trained time-lapse D2NNs could be combined for further improvement in generalization and statistical inference. To numerically demonstrate that ensemble learning can improve the inference of time-lapse networks, we trained 20 different time-lapse image classifiers, all with m = 5 and s_{max} = 5.33λ. To diversify the individual diffractive classifiers that are trained, we used two different strategies [121] as illustrated in Fig. 3.9; first, we used different encodings, i.e., amplitude encoding or phase encoding of the object information at the input plane; second, we used fixed (non-trainable) amplitude-only filters with different spatial profiles, either immediately following the object or at the Fourier plane of a 4-f system following the object [121]. The trained diffractive layers for these 20 time-lapse image classification networks are shown in Fig. 3.10. The blind testing accuracies of these individual time-lapse diffractive networks range from 48.18% to 60.79%. In Figs. 3.11-3.14, we present 4 ensembles with constituents selected from these 20 individual time-lapse diffractive networks. For example, Fig. 3.11 shows an ensemble of M = 20 time-lapse diffractive networks that attain 65.05% blind testing accuracy on CIFAR-10 test images

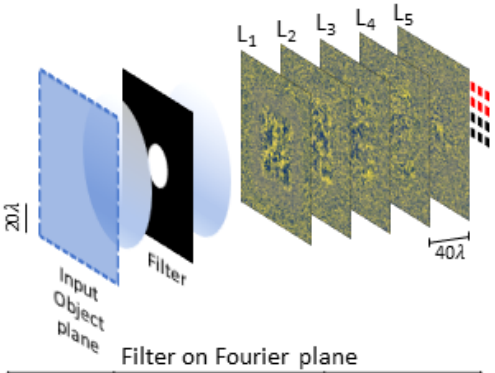
utilizing N = 25 lateral object displacements; as another example, a smaller ensemble with M = 10 time-lapse diffractive networks attained 64.21% blind testing accuracy utilizing N = 11 lateral displacements (Fig. 3.13). Two other ensembles, with M = 20 (N = 25) and M = 10 (N = 8), are also reported in Figs. 3.12 and 3.14, respectively, with similar performances. Further optimization of the ensemble hyperparameters and the encoding/filtering strategies of the input object information could further improve the inference accuracy of time-lapse diffractive network ensembles.



Filter adjacent to object plane

Network Index	Object modulation (Modulation depth)	Filter transmittance
1	Phase (2π)	
2	Phase ($3\pi/2$)	
3	Amplitude	
4	Amplitude	
5	Phase (2π)	
6	Phase (π)	
7	Phase (π)	
8	Amplitude	
9	Phase ($3\pi/2$)	
10	Phase (2π)	
11	Phase ($3\pi/2$)	

20λ



Filter on Fourier plane

Network Index	Object modulation (Modulation depth)	Filter transmittance
12	Phase (2π)	
13	Phase ($\pi/2$)	
14	Phase (2π)	
15	Amplitude	
16	Phase (2π)	
17	Amplitude	
18	Phase (2π)	
19	Phase (2π)	
20	Phase (2π)	

50λ

0 Amplitude 1

Figure 3.9 List of the architectures of the 20 time-lapse diffractive networks trained for ensemble learning. For networks 1-11, an amplitude-only filter is placed adjacent to the object plane, whereas for networks 12-20, an amplitude-only filter is placed at the Fourier plane of a 4-f

system following the object plane. The corresponding object encoding schemes and the filter transmittance profiles are presented in the 2nd and 3rd columns of the tables.

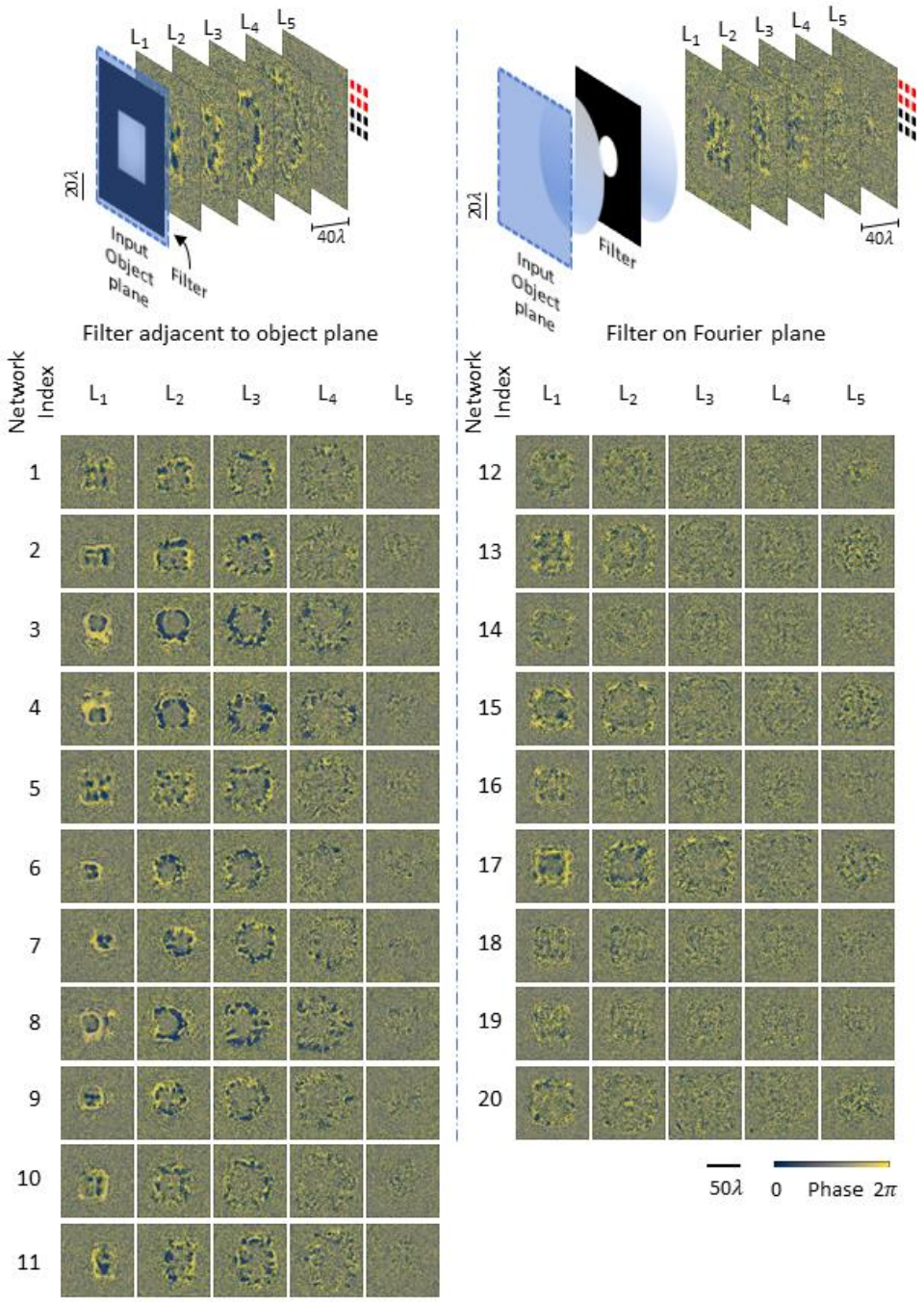


Figure 3.10 The trained diffractive layers for the 20 time-lapse diffractive networks used for ensemble learning. These networks were trained separately.

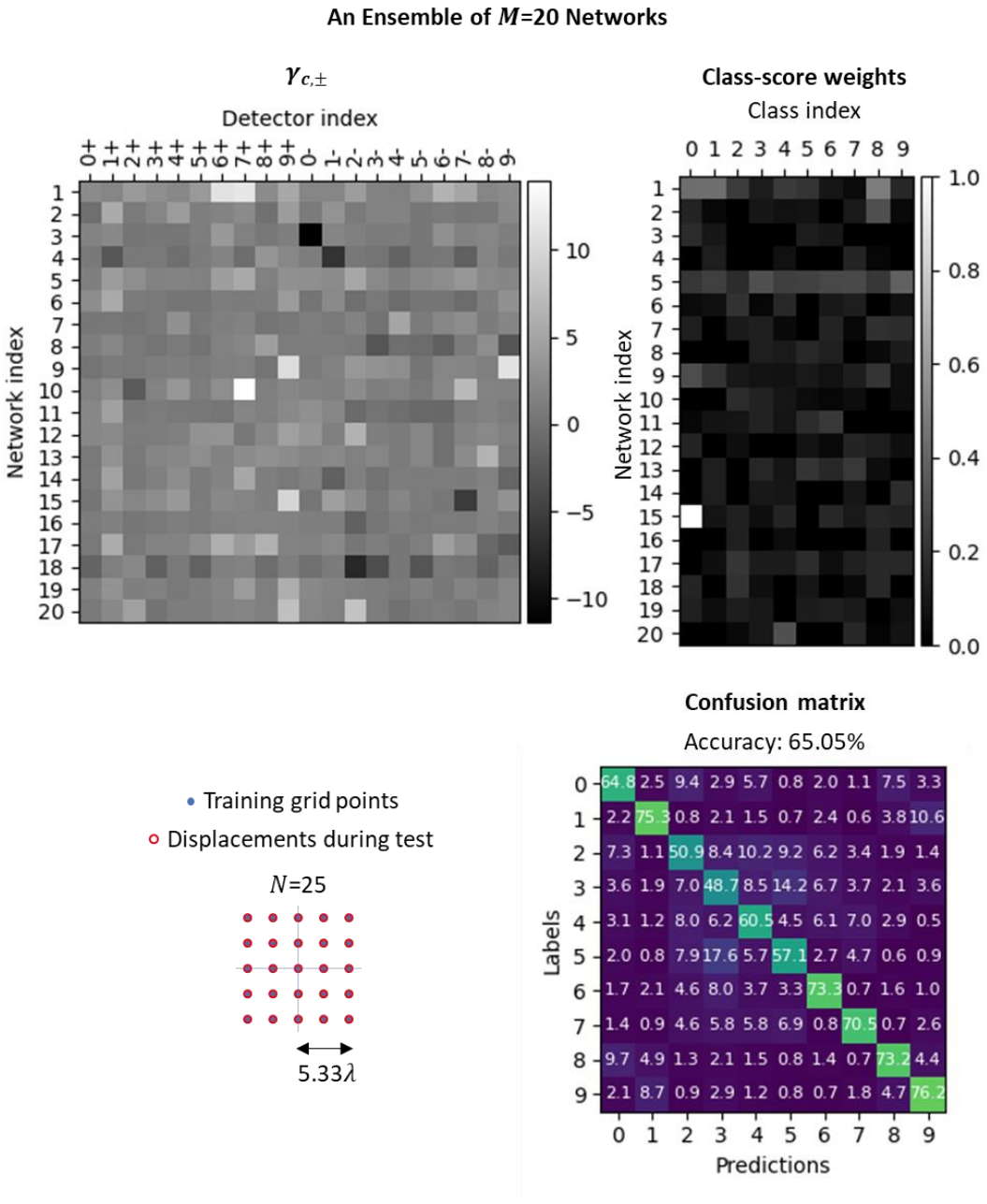


Figure 3.11 An ensemble of 20 time-lapse diffractive networks that attains 65.05% blind testing accuracy on CIFAR-10 test images. Top-left: Trained detector exponents for the time-lapse diffractive networks in the ensemble. Top-right: Trained class-specific weights for each network of the ensemble that are used to calculate the ensemble class-score from the individual network class-scores. Bottom: Confusion matrix of the ensemble with 65.05% blind testing accuracy utilizing N=25 object displacements.

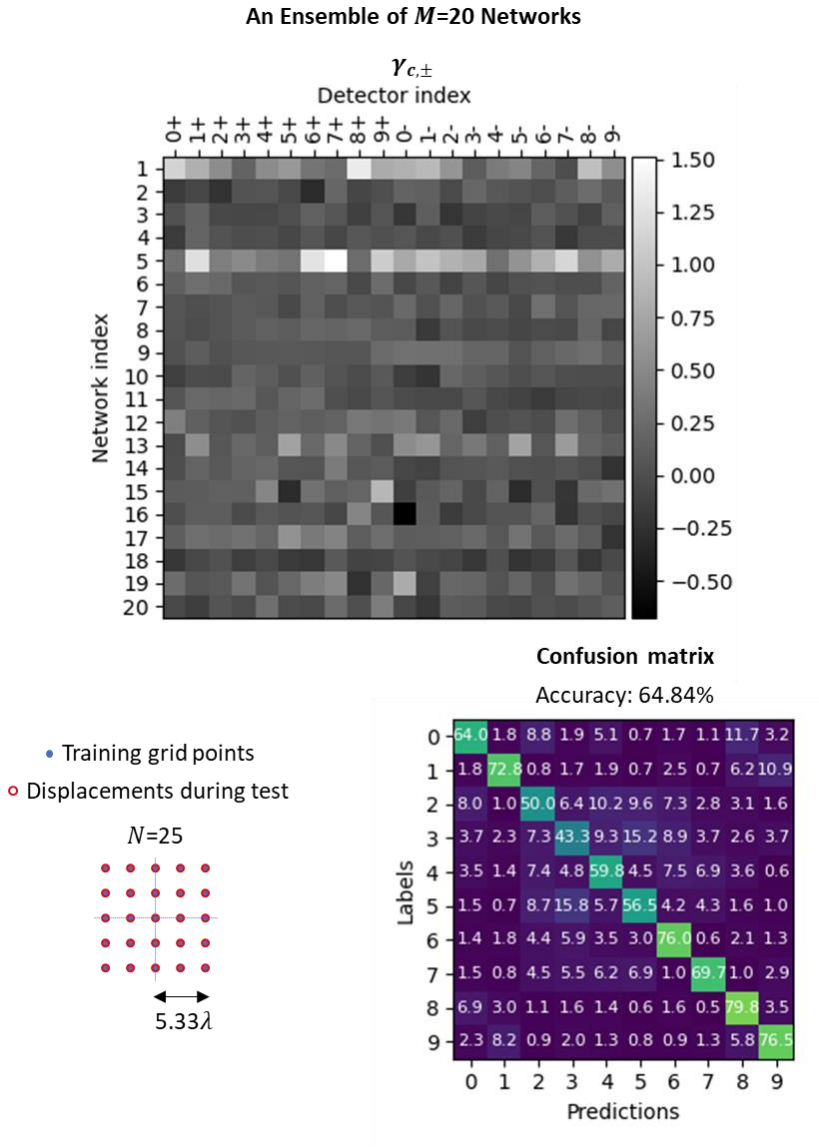


Figure 3.12 An ensemble of 20 time-lapse diffractive networks that attains 64.84% blind testing accuracy on CIFAR-10 test images without utilizing trainable class-specific weights for ensemble class-scores. The class-scores from the individual networks are simply added to calculate the ensemble class-score, i.e., $\theta_{c,\mu}$=1. Top: Trained detector exponents for the time-lapse diffractive networks in the ensemble. Bottom: Confusion matrix of the ensemble with 64.84% blind testing accuracy utilizing N=25 object displacements.

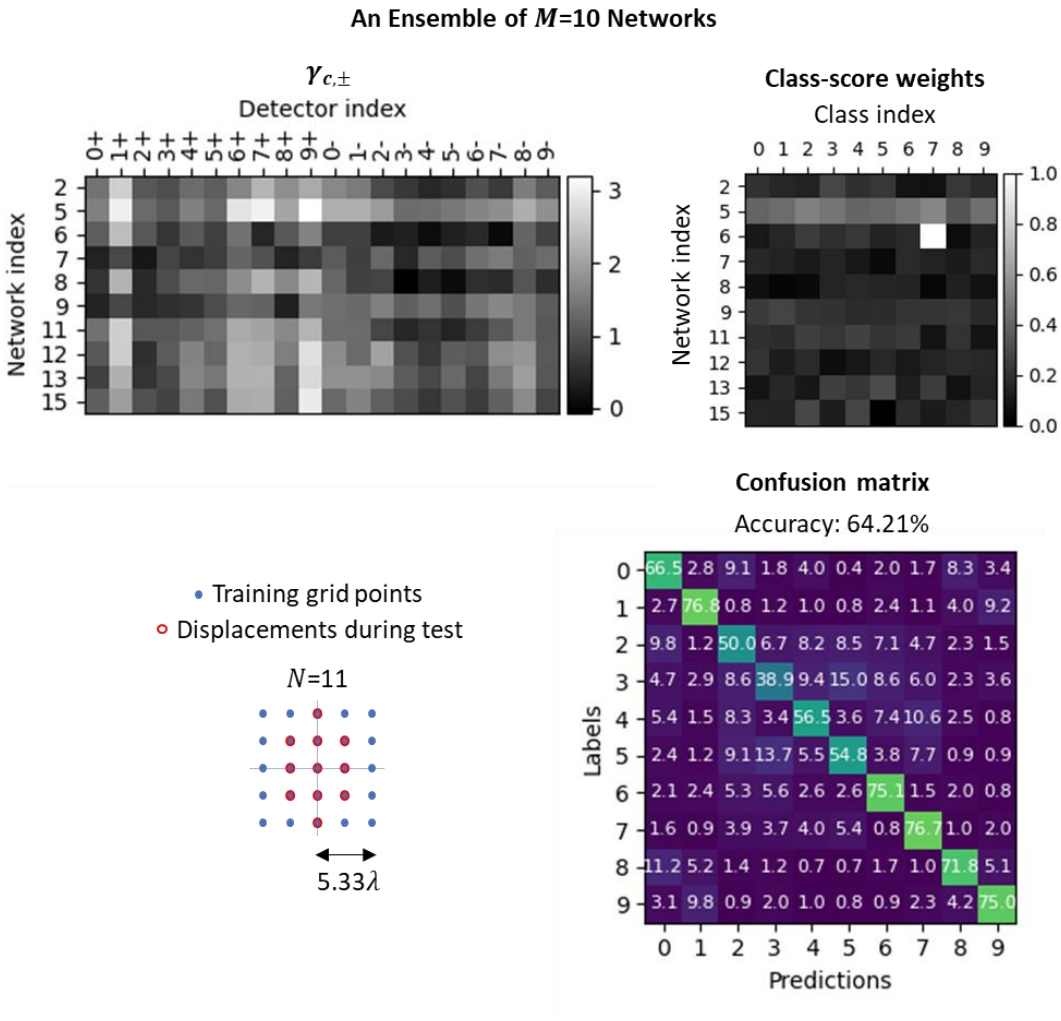


Figure 3.13 An ensemble of 10 time-lapse diffractive networks that attains 64.21% blind testing accuracy on CIFAR-10 test images utilizing N=11 object displacements. Top-left: Trained detector exponents for the time-lapse diffractive networks in the ensemble. Top-right: Trained class-specific weights for each network of the ensemble that are used to calculate the ensemble class-score from the individual network class-scores. Bottom: Confusion matrix of the ensemble with 64.21% blind testing accuracy utilizing N=11 object displacements.

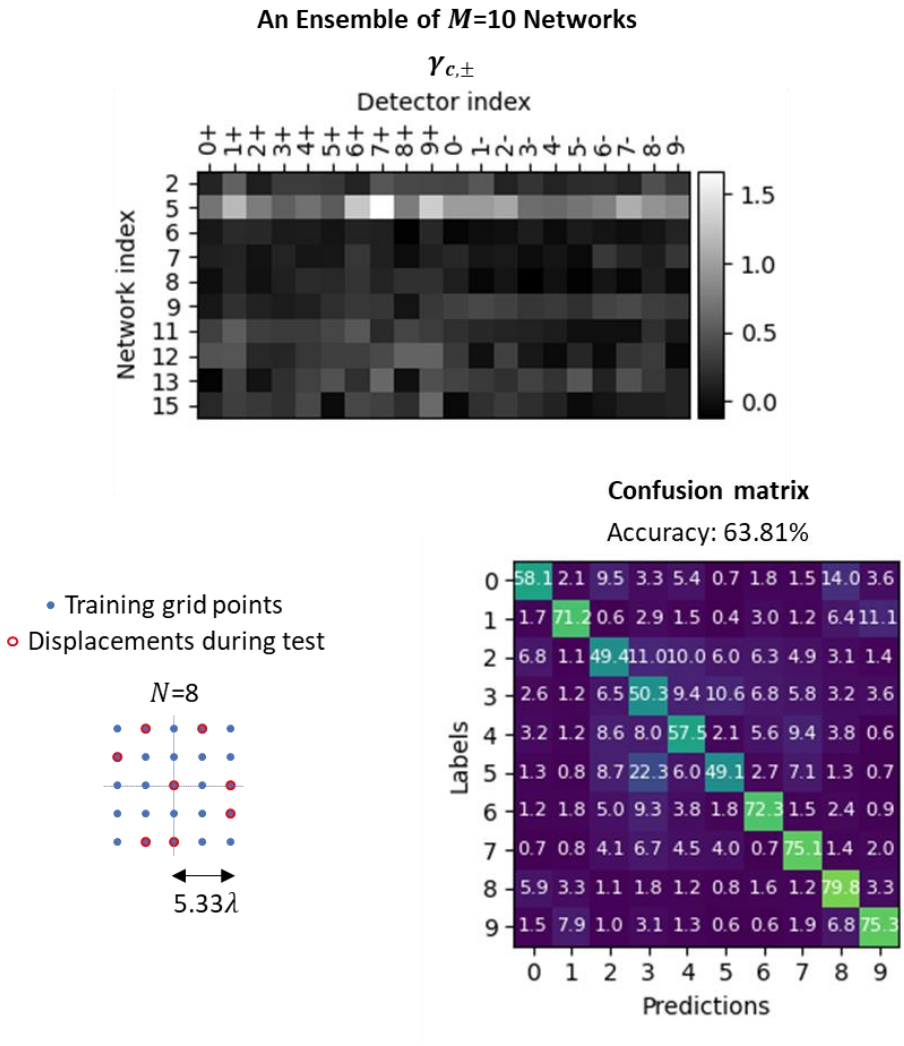


Figure 3.14 An ensemble of 10 time-lapse diffractive networks that attains 63.81% blind testing accuracy on CIFAR-10 test images utilizing N=8 object displacements and no trainable class-specific weights for ensemble class-score. The class-scores from the individual networks are simply added to calculate the ensemble class-score, i.e., $\theta_{c,\mu}$=1. Top: Trained detector exponents for the time-lapse diffractive networks in the ensemble. Bottom: Confusion matrix of the ensemble with 63.81% blind testing accuracy utilizing N=8 object displacements.

Moreover, in the same way that the time-lapse scheme utilizes the complementary information resulting from the input objects that are laterally shifted, other attributes of light such as polarization or wavelength could also be utilized to further improve the inference performance of time-lapse diffractive networks [107,127]. For example, time-lapse diffractive networks can be

trained to work with RGB images instead of grayscale images to benefit from the complementary information carried by different color channels. The incorporation of optical nonlinearities between the diffractive layers of D2NNs could also extend their approximation capability and consequently improve their statistical inference; for further details, see the Supplementary Information of Ref. [38], where the impact of optical nonlinearities within a D2NN architecture was discussed for the first time. All of these constitute possible future directions to explore for further decreasing the performance gap between electronic deep neural networks and D2NNs.

In summary, we reported a time-lapse diffractive network-based image classification scheme for significantly improving the performance of D2NN classifiers with only a single trained diffractive network. The presented time-lapse diffractive network scheme could be vital for realizing compact, low-cost and passive optical processors for all-optical spatio-temporal analysis of information.

3.4 Methods

3.4.1 Forward model

The propagation of coherent light across K+2 parallel planes defined by the input (object) plane, K successive diffractive layers, and the output (detector) plane is modeled using the Rayleigh-Sommerfeld theory of scalar diffraction [15], according to which the propagation of a complex wave $U(x, y)$ through a distance z in free-space is described by a linear shift-invariant system with an impulse response defined as follows:

$$h(x, y; z) = \frac{z}{r^2}\left(\frac{1}{2\pi r} + \frac{1}{j\lambda}\right)\exp\left(j\frac{2\pi r}{\lambda}\right) \tag{3.1}$$

where λ is the illumination wavelength, $r = \sqrt{x^2 + y^2 + z^2}$ and $j = \sqrt{-1}$. Upon propagation through the free-space separating layer l-1 and layer l, the complex field is modulated by the spatially varying complex transmittance $t_l(x, y)$ of layer l, i.e.:

$$U(x, y; z_l) = t_l(x, y) \iint h(x - x', y - y'; z_l - z_{l-1}) U(x', y'; z_{l-1}) dx' dy' \tag{3.2}$$

$$t_l(x, y) = a_l(x, y) \exp\big(j\varphi_l(x, y)\big) \tag{3.3}$$

Here, z_l is the axial coordinate of the l-th plane, and $l = 1, \cdots, K$, whereas $a_l(x, y)$ and $\varphi_l(x, y)$ are the amplitude and the phase of the complex field transmittance $t_l(x, y)$. For the phase-only diffractive networks reported in this work, $a_l(x, y)$ is assumed to be 1.

In a differential classification scheme, each of the 10 classes of the CIFAR-10 dataset is assigned to two detectors: a virtual positive detector and a virtual negative detector. $D_{c,+}$ ($D_{c,-}$) denotes the active area of the positive (negative) detector assigned to class c, $c = 0,1, \cdots, 9$. For the time-static diffractive networks, the detector signals $I_{c,\pm}$, based on which the class scores are computed, are proportional to the detector powers $P_{c,\pm}$, where

$$P_{c,\pm} = \iint_{D_{c,\pm}} |U(x, y; z_{K+1})|^2 dx dy \tag{3.4}$$

For the time-lapse diffractive network, light is assumed to be integrated at the detectors over N intervals of δ_t duration each. The object function (O) during the n-th interval can be expressed as:

$$O_n(x, y) = O(x - x_n, y - y_n) \tag{3.5}$$

where $n = 1, \cdots, N$, $(x_n, y_n) \in X \times Y$, $X = Y = linspace(-s_{max}, s_{max}, m)$ and $linspace(-s_{max}, s_{max}, m)$ denotes the set of m linearly spaced values between $-s_{max}$ and s_{max}.

Accordingly, the optoelectronic signals at the detectors are proportional to the integrated photon signals

$$E_{c,\pm} = \alpha \int_0^{N\delta_t} P_{D_{c,\pm}}(t)dt \tag{3.6}$$

Here, α is an optoelectronic detector-specific constant, and we assume that the propagation delay of light between the object plane and the detector plane is negligible compared to δ_t.

The detectors are assigned the exponents $\gamma_{c,\pm}$, which operate on the optoelectronic signals $E_{c,\pm}$ (after $E_{c,\pm}$ are normalized to have a maximum value of 1) and generate the detector signals $I_{c,\pm}$:

$$I_{c,\pm} = \left(E_{c,\pm}\right)^{\gamma_{c,\pm}} \tag{3.7}$$

Finally, the differential class scores are calculated as:

$$z_c = \frac{I_{c,+} - I_{c,-}}{I_{c,+} + I_{c,-}} \tag{3.8}$$

and the prediction for the object class is defined to be $\arg\max_c z_c$.

3.4.2 Numerical implementation

When numerically modeling light propagation through the diffractive networks, the grid spacing along the transverse directions (x and y) was chosen to be $\sim 0.53\lambda$. The Rayleigh-Sommerfeld convolution integrals were computed using the Angular Spectrum Method [15] based on the Fast Fourier Transform (FFT). For all the results presented in this paper, the diffractive networks consisted of 5 phase-only diffractive layers, axially separated by 40λ. Each layer comprised 200×200 diffractive features/neurons, the phases of which were trainable. The (physical) size of each diffractive neuron was assumed to be $\sim 0.53\lambda \times 0.53\lambda$.

The RGB images in the CIFAR-10 dataset were converted to grayscale to represent the input objects illuminated by a monochromatic and spatially-coherent wave. The objects were resized to span an area of $44.8\lambda \times 44.8\lambda$. The object information was assumed to be encoded in the phase channel of the input light, i.e., within the input field of view,

$$U(x, y; z_0) = \exp\big(j2\pi O(x, y)\big) \tag{3.9}$$

where $O(x, y)$ is the object function, with its values normalized to lie between 0 and 1. On the output plane, the active area of each detector was assumed to be $6.4\lambda \times 6.4\lambda$, and the spacing between the detectors was ~4.27λ along both x and y directions (see Fig. 3.1).

3.4.3 Training

The diffractive networks were trained using the cross-entropy loss function together with a loss term penalizing low diffraction efficiency at the output detectors. The differential class-scores $\{z_c\}_{c=0}^{9}$ were converted to probabilities $\{q_c\}_{c=0}^{9}$ over the data classes using the softmax function, i.e.,

$$q_c = \frac{\exp(\beta z_c)}{\sum_i \exp(\beta z_i)} \tag{3.10}$$

where $\beta = 10$ was used. The training loss function was defined as:

$$\mathcal{L} = \mathcal{L}_{CE} + \eta \mathcal{L}_{eff} \tag{3.11}$$

$$\mathcal{L}_{CE} = -\sum_{c=0}^{9} \delta_{ck} \log q_c \tag{3.12}$$

$$\mathcal{L}_{eff} = 1 - \frac{E_D}{E_{illum}} \tag{3.13}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, k is the (true) label, and δ_{ck} is the Kronecker delta function, i.e., $\delta_{ck} = 1$ if $c = k$ and 0 otherwise. $\mathcal{L}_{eff}$ is introduced to penalize low diffraction efficiency at the output detectors, where $E_D = \sum_{c=0}^{9}\left(E_{c,+} + E_{c,-}\right)$ is the total incident energy on the 20 detectors and E_{illum} is the total energy of the illumination beam at the input aperture. η is a training hyperparameter, which is used to fine-tune the output diffraction efficiency of the resulting diffractive networks.

The trainable parameters of the model were trained by minimizing the loss $\mathcal{L}$ using the Adaptive Momentum ('Adam') stochastic gradient descent algorithm [102]. The forward model was implemented using the open-source deep learning library TensorFlow [128]. The automatic differentiation functionality of TensorFlow was exploited to facilitate the gradient computations for optimization. A batch size of 8 was used to implement the stochastic gradient descent. The built-in TensorFlow implementation of Adam optimizer was used with the default values except for the learning rate, which had an initial value of 0.001 and was reduced by a factor of 0.7 every 8 epochs.

All the networks were trained for 100 epochs using 45000 images from the training set of the CIFAR-10 dataset. The remaining 5000 images of the CIFAR-10 training set were left out for validation, i.e., after every epoch, the accuracy of the model on these 5000 images was evaluated. The model state at the end of the epoch for which the validation accuracy was maximum was ultimately used for blind testing.

The training time of the time-lapse diffractive networks depended upon the hyperparameters m and p. For m = 5 and p = 0.5, the training took ~20 hours on an NVIDIA GeForce RTX 3090 GPU in a machine running on Windows 10.

3.4.4 Ensemble Learning of Time-Lapse Diffractive Networks

The training of diffractive network ensembles was performed in 2 phases. In the first phase, the individual time-lapse diffractive networks (diffractive layers and the detector exponents $\gamma_{c,\pm}$) were trained separately. In the second phase, the resulting diffractive networks were assigned class-specific weights, and an ensemble class-score $z_{c,ensemble}$ was defined as the weighted sum of the individual time-lapse diffractive network class-scores, i.e.,

$$z_{c,ensemble} = \sum_{\mu=1}^{M} \theta_{c,\mu} z_{c,\mu} \tag{3.14}$$

where M is the number of time-lapse diffractive networks in the ensemble, $z_{c,\mu}$ is the score for class c from network μ and $\theta_{c,\mu}$ is the corresponding weight. The cross-entropy loss function based on the ensemble class-score $z_{c,ensemble}$ was minimized for ensemble training. In addition to the class-specific weights $\theta_{c,\mu}$, the detector exponents $\gamma_{c,\pm}$ corresponding to every network were further trained in the second phase. For training ensembles which do not utilize trainable class-specific weights (see Figs. 3.12 and 3.14), $\theta_{c,\mu}$ were set to 1.0 and only the detector exponents $\gamma_{c,\pm}$ of each time-lapse diffractive network were further trained in the second phase.

To design diffractive network ensembles that operate with a smaller number of displacements, i.e., $N < m^2$, in the second phase, a trainable weight with a value between 0.0 and 1.0 was attached to each of the m^2 lateral displacements. After the training phase, a predefined threshold was applied to binarize these trained weights. Only the lateral displacements corresponding to the non-zero weights after this binarization step were retained. The 10 constituent diffractive networks of the ensembles shown in Figs. 3.13 and 3.14 were selected from the original 20 time-lapse diffractive networks (see Figs. 3.9 and 3.10) using a greedy search.

Chapter 4 Computer-free, All-optical Reconstruction of Holograms using Diffractive Networks

In this chapter, I introduce all-optical twin image-free reconstruction of inline holograms using diffractive networks. Parts of this chapter have been published in the peer-reviewed research article "M. S. S. Rahman and A. Ozcan, 'Computer-Free, All-Optical Reconstruction of Holograms Using Diffractive Networks,' ACS Photonics **8**, 3375–3384 (2021)," (Reprinted (adapted) with permission from ACS Photonics 2021, 8, 11, 3375–3384. Copyright 2021 American Chemical Society).

Reconstruction of in-line holograms of unknown objects in general suffers from twin-image artifacts due to the appearance of an out-of-focus image overlapping with the desired image to be reconstructed. Computer-based iterative phase retrieval algorithms and learning-based methods have been used for the suppression of such image artifacts in digital holography. Here we present a conceptual framework and the corresponding numerical simulation results for an all-optical hologram reconstruction method that can instantly retrieve the image of an unknown object from its in-line hologram and eliminate twin-image artifacts without using a digital processor or a computer. Multiple transmissive diffractive layers are trained using deep learning so that the diffracted light from an arbitrary input hologram is processed all-optically, through light-matter interaction, to reconstruct the image of an unknown object at the speed of light propagation and without the need for any external power. This passive all-optical processor composed of spatially engineered transmissive layers forms a diffractive network, which successfully generalizes to reconstruct in-line holograms of unknown, new objects and exhibits improved diffraction efficiency as well as extended depth-of-field at the hologram recording distance. This all-optical

hologram processor and the underlying design framework can find numerous applications in coherent imaging and holographic display-related applications owing to its major advantages in terms of image reconstruction speed and computer-free operation.

4.1 Introduction

Holography is a widely used technique with a myriad of applications in e.g., computational imaging, displays, interferometry and data storage [129]. What distinguishes holography from other optical methods is its ability to record and reconstruct both the intensity and the phase of the object field of interest. For its recording, the object wave is made to interfere with a reference wave to generate an intensity pattern, i.e., the hologram, encoding both the amplitude and the phase information of the object wave. Holographic reconstruction broadly refers to the retrieval of the object information from the recorded hologram intensity.

In the original hologram recording scheme demonstrated by Gabor, the so-called 'in-line holography', the reference wave and the object wave co-propagate along the same axis [130]. In its analog implementation, the recorded hologram (e.g., a photographic plate or its digital copy) can be illuminated with a reference wave and the object field can be partially recovered. Reconstruction of in-line holograms in general suffers from a spatial artifact known as the twin-image, which is due to the appearance of an out-of-focus image overlapping with the desired image of the object, degrading the reconstruction quality. To get rid of the twin-image problem, Leith and Upatniek proposed an alternative hologram recording geometry, known as the 'off-axis holography' [131]. In this scheme, a small angle is introduced between the reference wave and the object wave, resulting in a spatial separation of the twin-image from the desired image of the

unknown object during the reconstruction process. However, in addition to the relatively increased complexity of the experimental setup, the achievable space-bandwidth product with off-axis holography is smaller. In fact, the simplicity and the experimental robustness of the recording geometry of in-line holograms make it ideal for various field-based measurements and applications that require compact and cost-effective imaging and sensing systems [132].

Powered by modern computers, the emergence of digital holography enabled the numerical reconstruction of holograms, exploiting the availability of phase retrieval algorithms for suppressing twin-image artifacts [133]. Various phase-retrieval algorithms have been reported over the last several decades to recover the missing phase information and reconstruct an image of the specimen from one or more in-line holograms using a computer [134–140]. As an alternative approach, deep learning-based hologram reconstruction methods have also been demonstrated, performing phase recovery and twin-image elimination using trained deep neural networks [141–150]. Some of these earlier neural networks devised to blindly reconstruct an arbitrary hologram were trained with examples of holograms (input to the network) and the corresponding object fields (target ground truth) [141]. Although the data generation and the training process is time consuming (taking e.g., 12-24 h, depending on the availability of graphics processing units, GPUs), this is still a one-time effort, and once the neural network has been trained, it can be deployed to blindly reconstruct an input hologram of an unknown, new object, in a single feed-forward through the network, without any iterative phase retrieval or optimization steps. This constitutes one of the important advantages of deep learning-based hologram reconstruction methods, in addition to enabling some other unique imaging capabilities such as e.g., extended depth-of-field [151] and virtual image transformations [152,153].

All of this earlier body of work is based on digital processing of the acquired holograms through a computer in order to reconstruct the images of unknown objects. In this work, we present an all-optical hologram reconstruction method that processes the diffracted wave from a holographic recording through a series of transmissive diffractive layers/surfaces that collectively project an image of the unknown object at an output plane. These diffractive layers are designed through deep learning to specifically reconstruct in-line holograms, and form a passive optical network [38,53,80,121,32,54,85] that can execute a desired task between an input and output plane without an external power source, except for the illumination light at a wavelength of λ. The input optical field of a diffractive network is transformed via light-matter interactions and diffraction through the spatially engineered layers to produce the target field at its output plane, dictated by the inference task that is statistically learned. In addition to computing the desired output field at the speed of light propagation (between the input and output planes), this diffractive network architecture also exploits the large connectivity and parallelism of free-space optical diffraction and the advantages of layer-by-layer optical processing [32].

By training five successive diffractive layers using deep learning, we report the numerical design of an all-optical processor, in the form of a passive diffractive network extending only $\sim 225\lambda$ in the axial direction, that can reconstruct the image of an unknown object from its intensity hologram without the need for any digital computation or an external power source. We numerically show that the designed diffractive networks can generalize very well to unseen examples and accurately reconstruct their images at the speed of light propagation, also exhibiting an enhanced diffraction efficiency as well as an improved depth-of-field at the hologram recording distance. We believe that this all-optical hologram processor will find various applications in

holographic imaging and display related applications, especially benefiting from its computer-free and instantaneous image reconstruction capability.

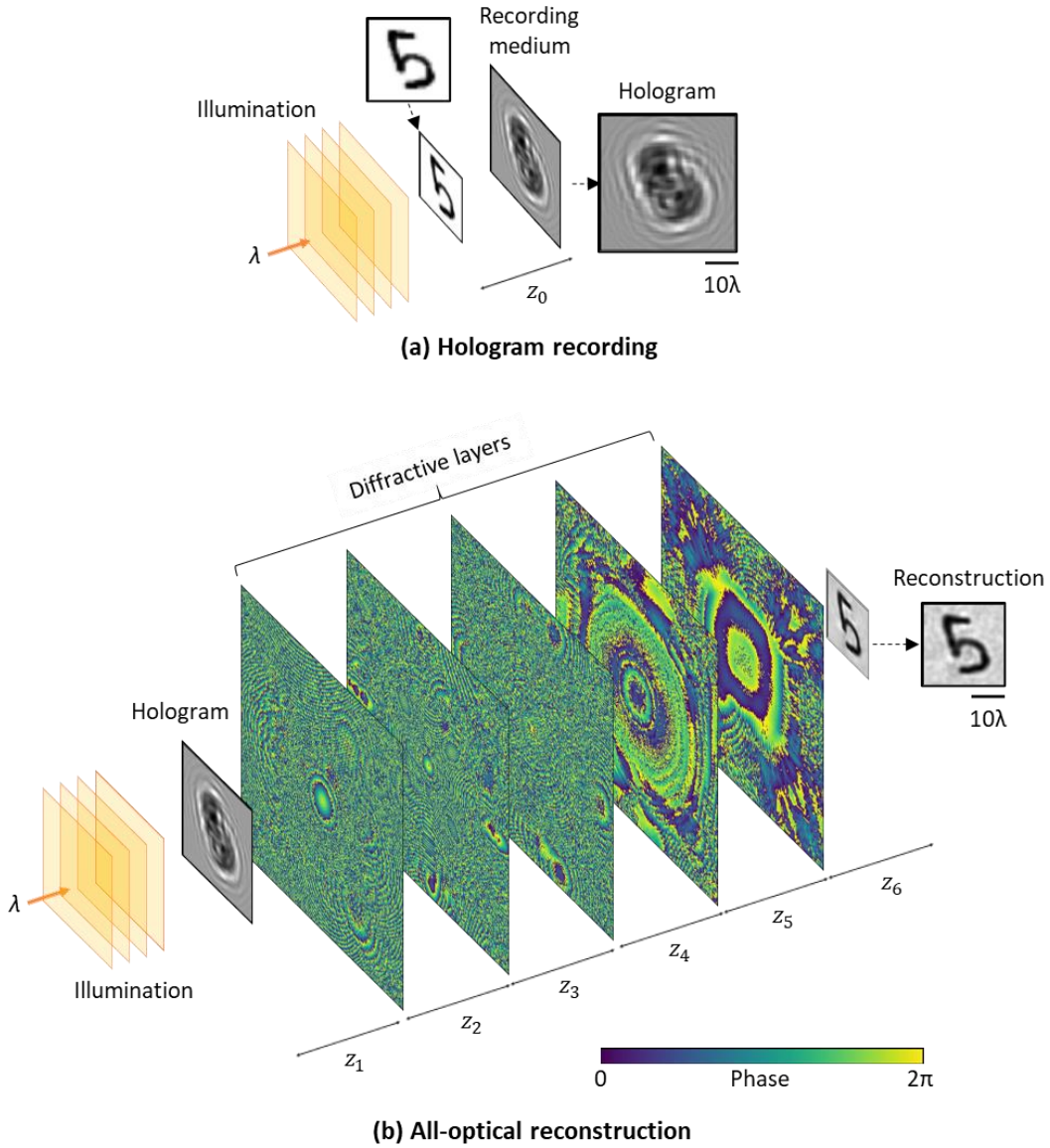


Figure 4.1 Hologram recording and reconstruction scheme for the proposed all-optical holographic reconstruction method. (a) For hologram recording, the object is illuminated by a plane wave, and the resulting diffraction pattern is recorded, forming the hologram. (b) For reconstruction, the hologram is illuminated with a plane wave, and the diffractive network all-optically computes a twin-image-free reconstruction at the output plane.

4.2 Results

A schematic of our all-optical hologram reconstruction framework is shown in Fig. 4.1a. The hologram recording process is the same as for in-line holography, i.e., an unknown object is illuminated with a coherent plane wave of wavelength λ, and the recording medium, e.g., an opto-electronic image sensor-array or a photographic emulsion, is placed at a distance z_0 from the object plane; in this work, we used $z_0 = 30\lambda$. This choice of z_0 is especially relevant for on-chip holographic microscopy [154,155] applications, where the twin-image artifact is strong. The component of the wave scattered by the object interferes with the unscattered, directly transmitted wave, producing an in-line hologram. During the all-optical reconstruction through a diffractive network (Fig. 4.1b), the hologram recording medium or its digital copy is assumed to be illuminated with a plane wave of the same wavelength. In this embodiment, a set of spatially engineered, transmissive diffractive layers is placed between the input hologram and the output (i.e., the reconstruction or observation) plane, which all-optically computes a twin-image-free reconstruction of the original object at the network output (see Fig. 4.1b). The axial distances between successive transverse planes i.e., the input hologram plane, diffractive network layers and the output/reconstruction plane, are denoted by z_i, $i = 1, \ldots, N+1$, where $N$is the number of layers in the diffractive network; in this work, we used $N = 5$ and $z_i = 37.5\lambda$. The size of the unknown object to be reconstructed is assumed to be 25λ, the lateral extent of the recorded hologram is 42λ, and the width of each diffractive layer is assumed to be 100λ.

Based on these hyper-parameters, we trained an all-optical network with N=5 diffractive layers, where each layer had a total of 200 × 200 trainable parameters. Each one of these parameters corresponds to the phase value of a diffractive feature (neuron) over an area of $\lambda/2 \times \lambda/2$.

Therefore, using deep learning we optimized a total of 0.2 million independent phase values, spread across N=5 transmissive diffractive layers forming an optical network (Fig. 4.1b). For our training image data, we used the 'MNIST' handwritten digit dataset [92], which was augmented with an additional custom-built image dataset of generic shapes, e.g., patches, circles, straight-line and circular-line gratings etc. (see Fig. 4.8 for examples of these additional training images). The total number of images in the training dataset was 110,000 (55,000 from the MNIST training set, 55,000 from the custom prepared image dataset); none of these images in the training dataset were used in the blind testing of the final diffractive network models. Other details of the network training process, including the loss function, optimization algorithm and training times, are reported in the Methods section.

Once the diffractive network was trained for all-optical reconstruction of input holograms, it was blindly tested using numerically computed in-line holograms of new, unknown objects. Fig. 4.2 reports numerical examples of input holograms and the corresponding all-optical reconstructions obtained at the output of the diffractive network, which reveal that twin-image related artifacts that normally appear in free-space propagation of the input hologram are eliminated at the observation/output plane of the diffractive network, as desired. Also note that the images of the letters in the English alphabet were not part of our training image dataset, and therefore the success in the all-optical reconstruction of "UCLA" text from its hologram (see Fig. 4.2a) indicates that the trained diffractive network is capable of generalizing to distributions different from the distribution of the objects used during the training phase.

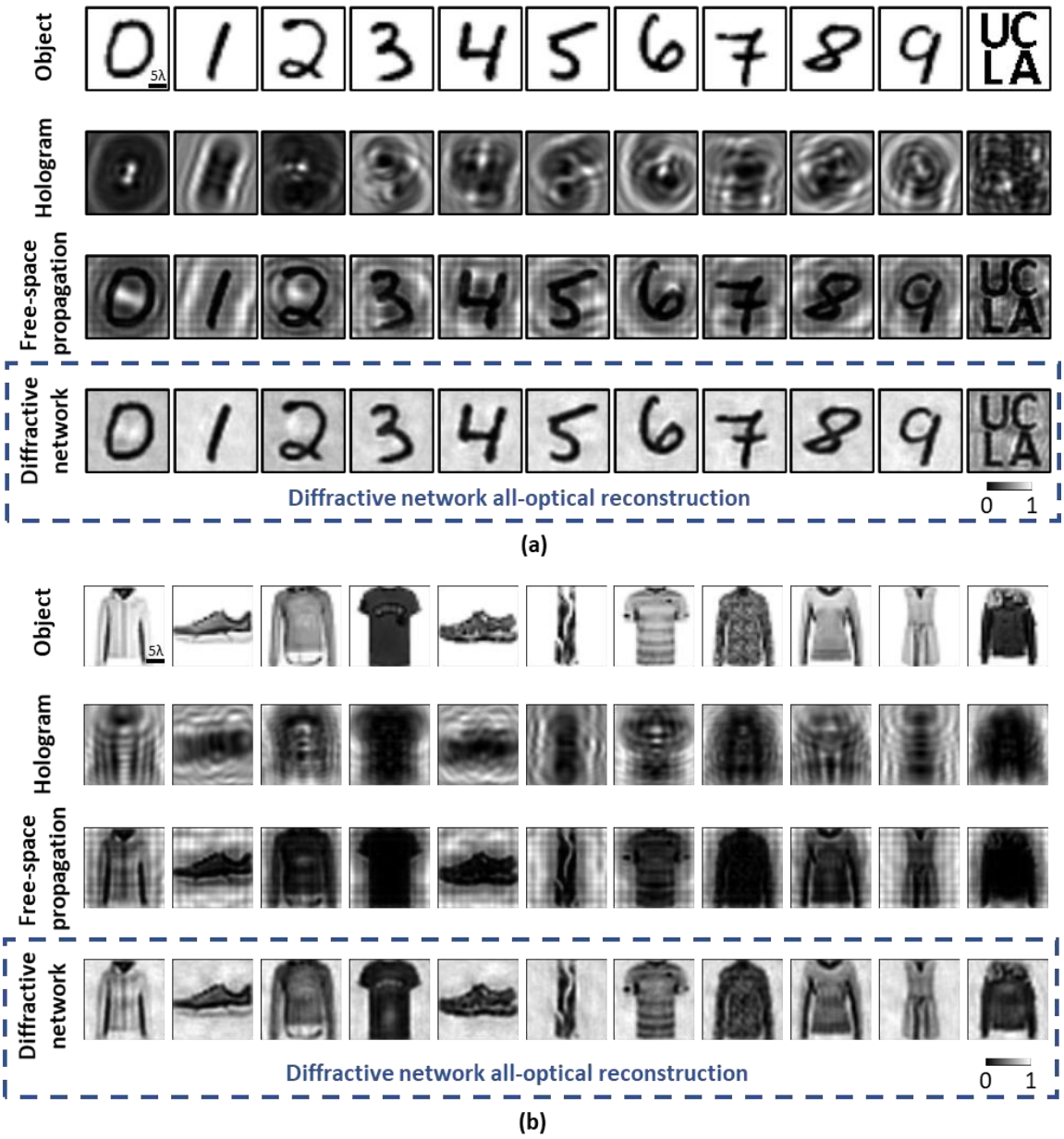


Figure 4.2 Performance of all-optical hologram reconstruction using diffractive networks for handwritten digits and letters from the English alphabet (a) and fashion products (b). In both (a) and (b), the first two rows depict the target object amplitudes, and the corresponding recorded in-line holograms, respectively. The axial distance between the object and the recording plane is assumed to be 30λ. The reconstructed object intensities by free-space propagation are shown in row 3, whereas those by the designed diffractive network are shown in row 4.

To further demonstrate the generalizability of the trained diffractive network to grayscale and low contrast objects, we blindly tested the performance of the same diffractive network on objects selected from the test set of 'Fashion-MNIST' dataset [156], the results of which are reported in Fig. 4.2b. As is evident from the figure, the diffractive network successfully reconstructed these grayscale and lower contrast fashion objects, even though these types of objects were not presented to the diffractive network during its training. Various spatial features and details that are lost in free-space propagation of the input holograms are preserved in the reconstructions of the trained diffractive network, as illustrated in Fig. 4.2b.

Apart from successful generalization to object distributions very different from its training dataset (e.g., fashion products vs. handwritten digits), diffractive network-based hologram reconstructions are also invariant to object scaling: as shown in Fig. 4.3, the diffractive network's hologram reconstructions remain successful under scaling (expansion or shrinkage) of the objects beyond their typical size used during the training. Furthermore, the diffractive network also exhibits invariance to angular rotations of the recorded holograms during the reconstruction phase and can successfully reconstruct randomly rotated holograms as shown in Fig. 4.4.

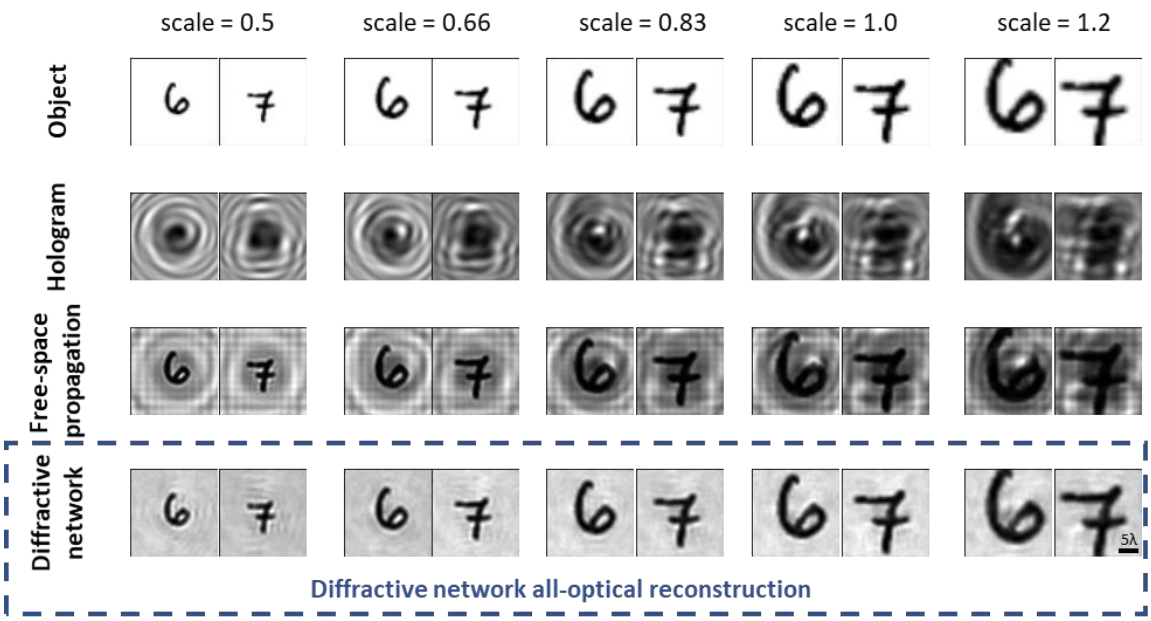


Figure 4.3 All-optical hologram reconstructions by a diffractive network for different object scaling factors.

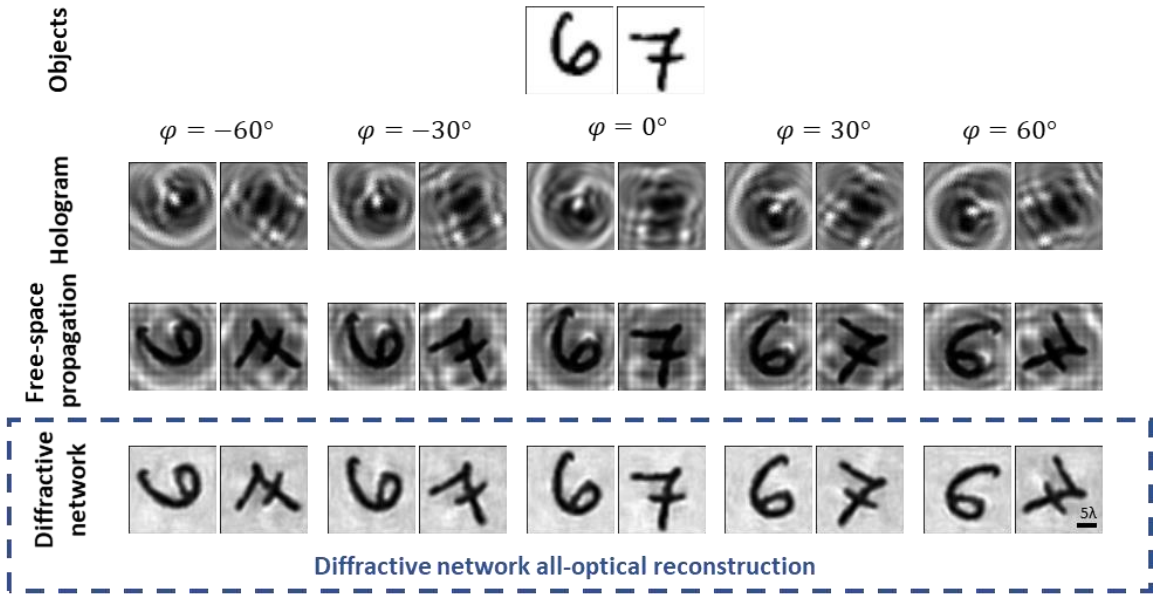


Figure 4.4 All-optical hologram reconstructions by a diffractive network for different rotation angles of the recorded holograms.

Next, we explored through numerical simulations the capability of the same diffractive network to all-optically reconstruct and resolve closely spaced objects, composed of two parallel lines with gradually increasing separation; see Fig. 4.5. For line separations that are equal to or larger than λ, the diffractive network succeeded in reconstructing the in-line holograms of these test objects and faithfully resolved the two lines from each other as shown in Fig. 4.5. Also notice that for line separations that are larger than 3.5λ, the free-space propagation of the input holograms presents stronger spurious/artificial drops around the center of the lines due to the strong twin-image artifacts; this causes the appearance of artificial lines that are comparable to the signal strength of the true object lines. Such spatial artifacts due to twin-images have been successfully suppressed at the output plane of the diffractive network as shown in Fig. 4.5.

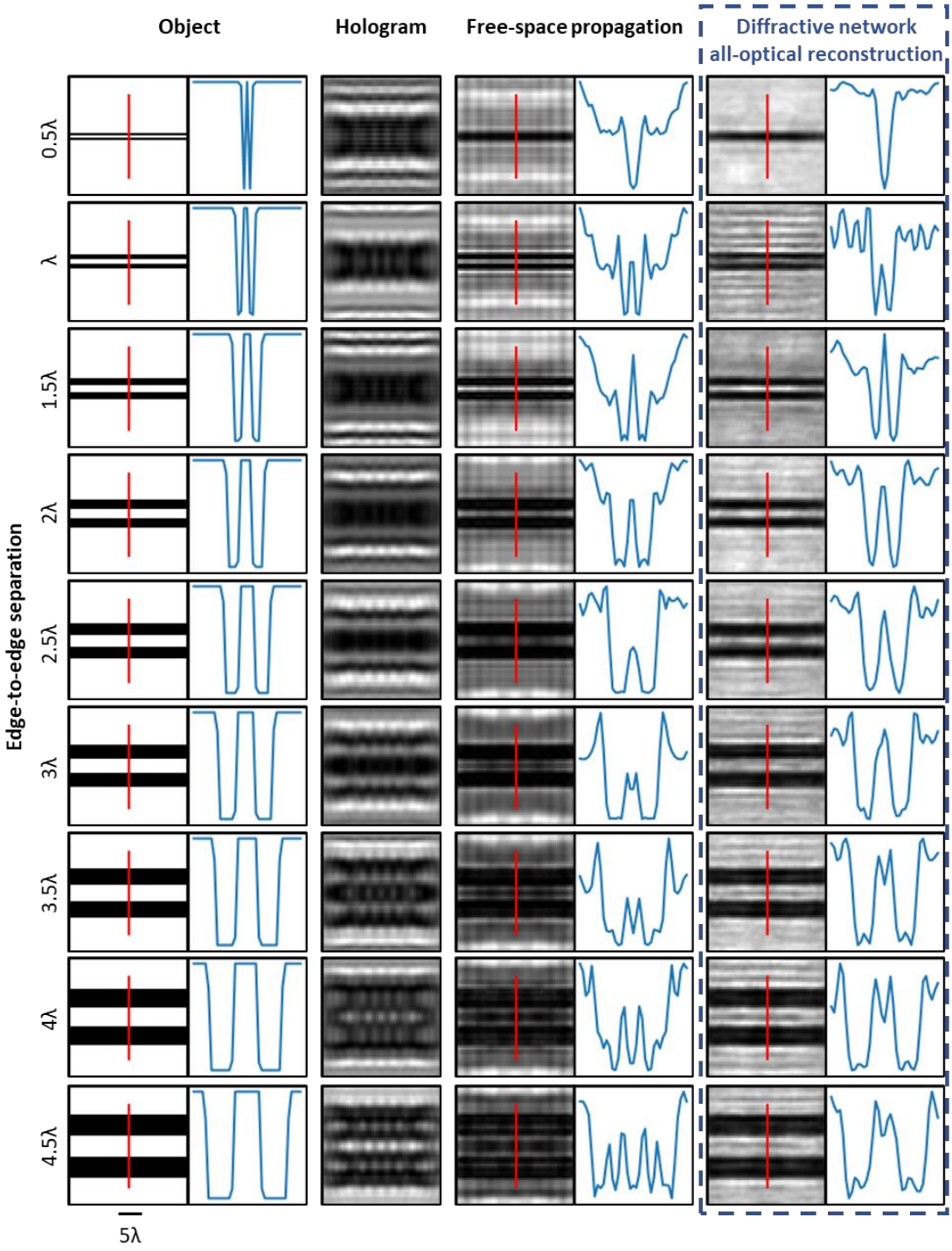


Figure 4.5 Quantification of all-optical holographic image reconstruction resolution. The edge-to-edge separation refers to the separation from the inner edge of one line to the inner edge of the other. Intensity variations along the red lines are shown on the accompanying blue curves.

In addition to all-optical reconstruction of in-line holograms and the removal of twin-image artifacts, the presented diffractive network-based holographic image reconstruction framework can provide further advantages such as being more robust to uncontrolled changes in z_0, covering an extended depth-of-field in the hologram recording distance. To demonstrate this, we numerically explored the performance of all-optical reconstructions achieved by the same diffractive network design shown in Fig. 4.1b when the hologram recording distance $z_0 = 30\lambda$ was changed from its nominal value, covering an axial range of $z_0 = 24\lambda: 36\lambda$; see Fig. 4.6. In this case, however, the diffractive network did not know the exact hologram recording distance that was used and therefore its design did not change compared to earlier results so that we could truly test the robustness of its all-optical reconstructions against changes in the hologram recording distance. To quantify the all-optical image reconstruction performance under this change, we used 10,000 test images of the MNIST dataset to calculate the structural similarity index measure (SSIM) and the peak signal-to-noise ratio (PSNR) values for the reconstructed images at the diffractive network's output as shown in Fig. 4.6a. Fig. 4.6b also presents numerical simulations for visual comparison of the resulting all-optical reconstructions for different hologram recording distances, revealing the robustness of the diffractive network's reconstructions even though the hologram recording distance was significantly changed (±20%) from its nominal value which was used during the training of the diffractive network.

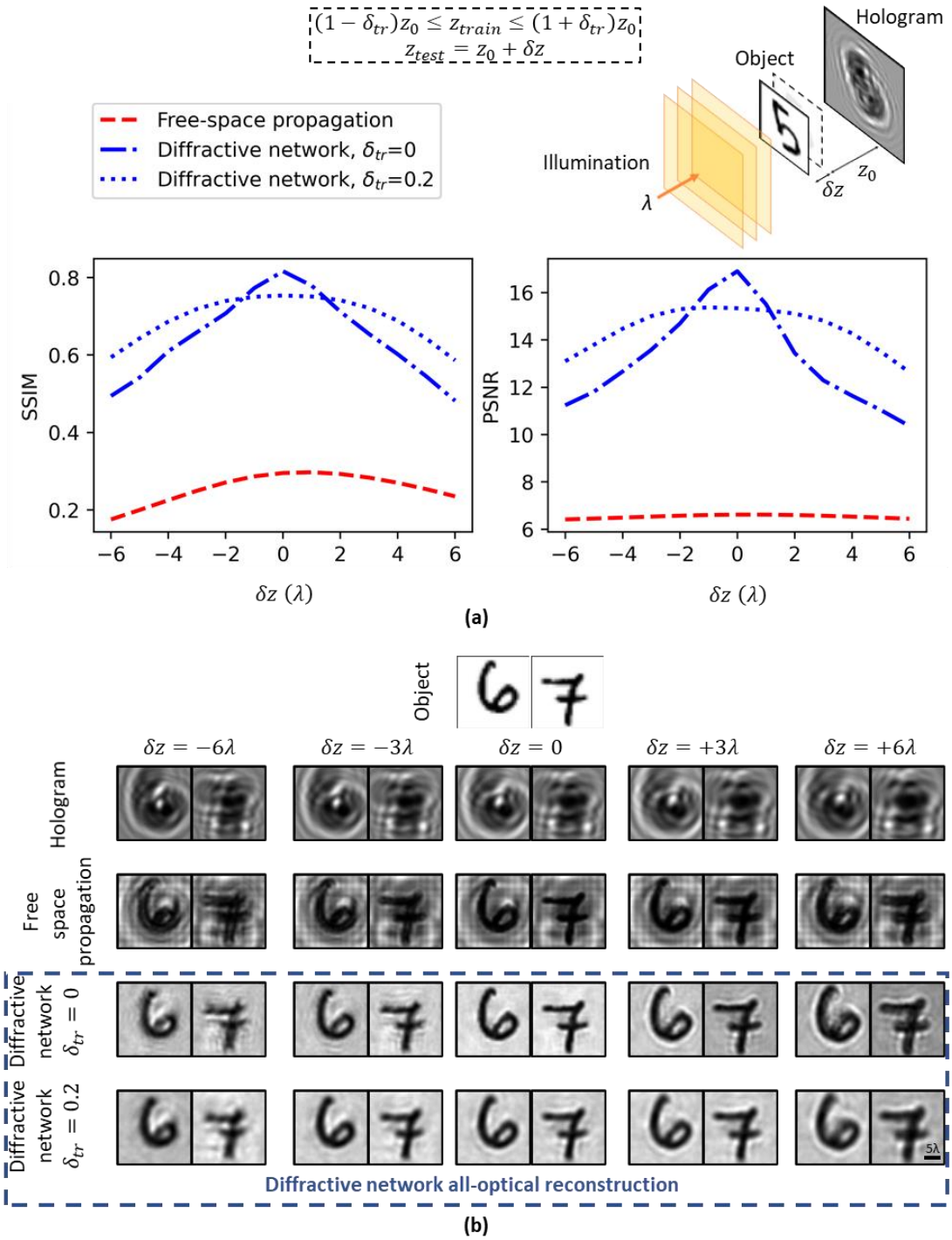


Figure 4.6 Robustness of all-optical diffractive network reconstructions against the hologram recording distance variations. (a) Quantitative comparison of the robustness of the diffractive network reconstructions and free-space propagation-based reconstructions to deviations in the hologram recording axial distance. SSIM and PSNR of the diffractive network results remain acceptable over a wide range of hologram recording distance variations. This robustness can be further improved by incorporating such random deviations in the training phase, as illustrated by

the curves corresponding to $\delta_{tr} = 0.2$. These metrics (SSIM and PSNR) were computed over the 10,000 test images of MNIST dataset. (b) Visual comparison of the reconstructed images.

In fact, the all-optical hologram reconstruction performance of the diffractive network design can be further improved by incorporating the extended depth-of-field in hologram recording distance during its training process so that the hologram reconstruction at the output plane becomes even less sensitive to the exact value of the recording distance. To demonstrate this capability, we trained another diffractive network from scratch, where the hologram recording distance during the training process was treated as a random variable, $z_{train} \sim Uniform\big((1-\delta_{tr})z_0, (1+\delta_{tr})z_0\big)$, where $\delta_{tr} = 0.2$ was used (see the Methods section for details). The numerical simulation results presented in Fig. 4.6a reveal that the resulting new diffractive network design based on this training strategy achieves all-optical reconstruction of input holograms with relatively flat SSIM and PSNR curves, demonstrating its inference success over a large range of hologram recording distances. Fig. 4.6b also supports the same conclusion by reporting the reconstructed holographic images at the output plane of the diffractive network for $\delta_{tr} = 0.2$.

A practical consideration for the use of this presented all-optical hologram reconstruction approach would be the diffraction efficiency of the optical design. The average diffraction efficiencies of the diffractive networks trained with $\delta_{tr} = 0$ and $\delta_{tr} = 0.2$ are found to be ~0.31% and ~0.58% respectively, whereas the average diffraction efficiency for free-space propagation-based reconstruction is ~8.61%. Here, the diffraction efficiency is defined as the ratio of the power at the output reconstruction field-of-view to the input power illuminating the hologram area; the above reported average diffraction efficiency values were calculated over 10,000 test images of the MNIST dataset. In Fig. 4.7, we numerically show that the diffraction efficiency of the hologram reconstruction diffractive network can be significantly improved by adding an efficiency-related

penalty term in the training loss function (see Methods section). By properly adjusting the relative weight (η) of this additional loss term to 0.5, a diffractive network with an output diffraction efficiency of ~23.64% was designed; its all-optical reconstruction performance is reported in Fig. 4.7a. Furthermore, Fig. 4.7b shows the diffraction efficiency-SSIM trade-off that can be achieved by changing η. For example, by further increasing η to 3.0, a hologram reconstruction diffractive network with an average diffraction efficiency of ~26.2% can be designed with a minor sacrifice in the image reconstruction quality. Note here that the optical power diffracted beyond the input aperture of a diffractive layer is blocked by opaque regions that surround each diffractive layer, and these blocking regions also account for the reduced output diffraction efficiency. For further clarification, the numerical aperture of a diffractive layer in our design is ~0.8, corresponding to a diffraction cone half-angle of ~53°; only the optical power that is diffracted within this diffraction cone can be accepted and processed by the following diffractive layer.

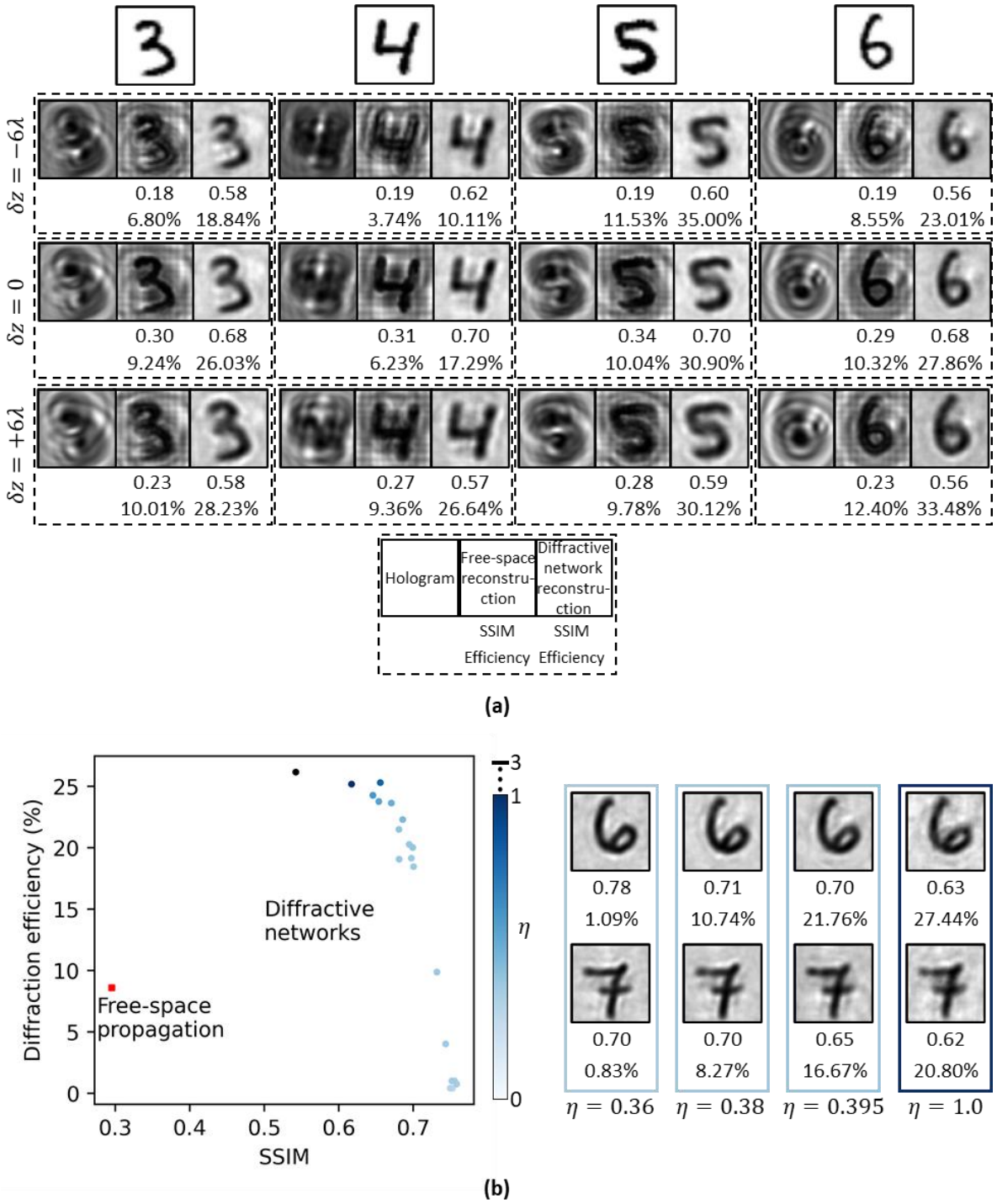


Figure 4.7 Diffraction efficiency improvement of all-optical holographic reconstructions performed by diffractive networks. (a) Examples of all-optical holographic reconstructions by a high-efficiency diffractive network (with training hyperparameters: $\delta_{tr} = 0.2, \eta = 0.5$). For each panel in (a) corresponding to a recording δz, left: hologram, center: free-space propagation-based reconstruction, right: diffractive network reconstruction. (**b**) SSIM vs. diffraction efficiency

tradeoff achieved by tuning η in the loss function. The two numbers below the reconstructed images correspond to the SSIM and the diffraction efficiency (%) of each reconstruction.

4.3 Discussion

Unlike traditional phase recovery and twin-image elimination methods that are based on algorithms implemented in digital computers, this work introduces a passive all-optical processor, composed of deep learning-designed diffractive layers, that can collectively reconstruct a hologram at its output plane without any digital computation or an external power source except for the illumination light. In addition to its passive nature, the hologram reconstruction process occurs almost instantaneously, i.e., at the speed of light propagation through a compact diffractive network that extends only ~225λ in the axial direction (from the input plane to the output).

Despite its major advantages, there are also some limitations of the presented framework. If the hologram recording distance changes drastically compared to the training z_0 value, we might need to fabricate a new set of diffractive layers trained by using the new hologram recording distance. However, our results in Fig. 4.6 confirmed that the all-optical reconstruction quality of a diffractive network remains very good over a relatively broad range of hologram recording distances, even though the network was trained with a fixed z_0 value; in fact, this depth range can be further improved by increasing δ_{tr} in the training process. Another limitation of the presented method is the required 3D alignment and fabrication precision of the diffractive layers. Previous work [157,158] on diffractive networks for all-optical image classification related tasks revealed that one can vaccinate the diffractive network design and make it more robust in its statistical inference accuracy by incorporating potential misalignments, fabrication imperfections and input object distortions as part of the optical forward model as random variables, similar to the strategy

that we used earlier for extending the range of hologram recording distances. Furthermore, the performance of diffractive network-based hologram reconstructions with respect to other practical parameters such as e.g., output diffraction efficiency can be fine-tuned and optimized by adjusting the training loss function as illustrated in Fig. 4.7, presenting a powerful design flexibility.

Although our results report the reconstruction of only the object intensity, the presented training process can also be adapted for the reconstruction of the complex field of an object by properly adjusting the training loss function to include terms related to the phase and amplitude of the sample. Similarly, an extension of the presented framework to 3D holographic image reconstruction could be possible through proper changes to the loss function; such a training process for 3D imaging would be computationally more intensive since structural errors from different planes of the reconstruction volume would have to be back-propagated through the diffractive network.

For practical adoption of this framework and its experimental uses, the resolution and precision of the fabrication method that is employed to create the diffractive layers are important. State-of-the-art additive nano-manufacturing technologies [24] could facilitate the realization of diffractive networks for operation in the visible and infrared wavelengths. In our simulations, the diffractive feature/neuron size was taken as $\lambda/2$, where λ is the illumination wavelength. With this selection of neuron size, all the propagating modes in free-space are taken into account, making our forward model accurate, except for multiple reflections between/within the diffractive layers that are ignored. In earlier work that experimentally demonstrated various diffractive networks designed for different tasks [38,53,54,85], the strong agreement between our numerical forward model predictions and the experimental results indicates that such secondary reflections are indeed much weaker compared to the directly transmitted, diffracting light and therefore can be ignored without

introducing a significant error in our image reconstruction or inference task. Along with the spatial resolution and precision of the used fabrication method, the performance of diffractive networks would also be affected by potential misalignments during the assembly of the diffractive layers. Earlier work introduced various "vaccination" strategies to design misalignment tolerant diffractive networks that incorporate stochastic lateral and axial misalignments of successive layers as part of the training process to build resilience toward such imperfections, while maintaining the desired inference performance at the network output [157,158].

In general, the range of object sizes supported by the presented all-optical hologram reconstruction framework is directly related to the diffractive layer size, which in turn is limited by the available fabrication technology. Different lithography methods and additive manufacturing techniques [24,159–161] can be used to fabricate diffractive layers for all-optical processing/reconstruction of e.g., microscopic objects that cover a field-of-view of ≥ 1 mm^2. To compare the speed of all-optical hologram reconstruction with state-of-the-art digital hologram reconstruction methods, an inference time of e.g., > 3 s was reported for holographic reconstruction of microscopic objects (e.g., cells and tissue samples) over a field-of-view of 1 mm^2 using a deep neural network running on a graphics processing unit (NVIDIA GTX 1080) [141]; in comparison the time required for all-optical reconstruction of a hologram using light propagation through a diffractive network that extends $\sim 225\lambda$ in the axial direction would be on the order of ~1 ps.

In conclusion, we introduced a framework for a passive all-optical processor designed using deep learning, that is capable of instantaneously reconstructing artifact-free images of unknown objects from their in-line holograms, without any digital computation. This all-optical computing framework that is composed of successive transmissive diffractive layers generalizes very well to unseen objects, and exhibits image quality, depth-of-field and diffraction efficiency advantages

compared to free-space propagation of a recorded hologram. This all-optical processor and the underlying deep learning-based design framework can find numerous applications in holographic imaging and display applications due to its reconstruction speed (driven by the propagation of light) and passive nature (operating without the need for a digital computer or an external power source, except for the illumination light).

4.4 Methods

4.4.1 Numerical Model of Diffractive Networks

In developing the mathematical model of a diffractive network, we index the transverse planes of the system by the letter l. For a diffractive network with N diffractive layers between the input and the output planes, $l = 0,1,2,\dots,N,N+1$, where $l = 0$ represents the input hologram plane and $l = N + 1$ represents the output (image reconstruction) plane. Furthermore, we denote the complex amplitude of the optical wave before and after the optical modulation imposed by element/pixel $m$of a diffractive layer l by $u_m^l$and v_m^l, respectively. To clarify, we use $u$to denote the field before modulation and v to denote the field after modulation by a diffractive layer, while the index in the superscript is used to represent the corresponding diffractive layer and the subscript represents the individual diffractive features of that layer. Then, following the Rayleigh-Sommerfeld formulation of diffraction [15], we can write:

$$u_k^l = \sum_m \frac{z_l}{\left(r_{mk}^l\right)^2}\left(\frac{1}{2\pi r_{mk}^l} + \frac{1}{j\lambda}\right) exp\left(j\frac{2\pi r_{mk}^l}{\lambda}\right) v_m^{l-1} \tag{4.1}$$

Here λ is the wavelength of the optical wave and $r_{mk}^{l} = \sqrt{(x_k - x_m)^2 + (y_k - y_m)^2 + z_l^2}$; z_l is the axial distance between layers $l-1$ and l; (x_m, y_m) and (x_k, y_k) are the transverse coordinates of feature $m$of layer $l-1$and feature k of layer l, respectively; v^0 and u^{L+1} are the optical fields at the input and the output fields-of-view of the diffractive network, respectively.

If the complex-valued transmittance of pixel k of layer l is denoted with t_k^l, then assuming the diffractive layers to be thin, we can write $v_k^l = u_k^l \cdot t_k^l$, which can be rewritten as:

$$v_k^l = \left(\sum_m w_{km}^l v_m^{l-1} \right) \cdot t_k^l \tag{4.2}$$

$$w_{km}^l = \frac{z_l}{\left(r_{mk}^l\right)^2} \left(\frac{1}{2\pi r_{mk}^l} + \frac{1}{j\lambda} \right) exp\left(j \frac{2\pi r_{mk}^l}{\lambda} \right) \tag{4.3}$$

Eq. (4.2) resembles a feedforward neural network, where w_{km}^l and t_k^l are the analogues of weights and (multiplicative) biases, respectively. Note that we use the terms diffractive feature and neuron interchangeably in the context of diffractive networks. The multiplicative biases t_k^l of the neurons are trainable since they are determined by the physical parameters of the diffractive surfaces. For example, for thin diffractive layers, the phase φ_k^l of the complex transmittance t_k^l of a diffractive layer neuron is related to its height h_k^l by the equation $\varphi_k^l = \frac{2\pi(n-1)h_k^l}{\lambda}$, where $n$is the refractive index of the diffractive layer material (which is surrounded by air).

Given a training dataset of input-target pairs (x, y) specific to a desired task, the input x can be mapped to the diffractive network input optical field v^0, whereas the diffractive network output optical field u^{L+1} is mapped to $\hat{y}$, an estimate of the target y. The amplitude and/or the phase of the complex amplitude transmittance values t_k^l can be optimized by minimizing an error function

between the target and the estimate, i.e., $L(y, \hat{y})$, which is achieved using standard deep learning techniques such as stochastic gradient descent and error backpropagation.

4.4.2 Network architecture

The hyperparameters related to the diffractive network architecture were set empirically. All the reported diffractive networks have 5 phase-only modulation/diffractive layers, and the transverse dimension of the layers are $100\lambda \times 100\lambda$. The pixel/neuron size is set as $0.5\lambda \times 0.5\lambda$, and the number of trainable parameters (phase values) in each diffractive network was 5×200×200 = 0.2 million. The axial/longitudinal distance between the layers is set as 37.5λ. The input and output planes of the diffractive network were also assumed to be 37.5λ away from the first and the last layers, respectively.

In computing the holograms, the object aperture (transverse dimension) was assumed to be $25\lambda \times 25\lambda$and the recording plane was assumed to be 30λaway from the object plane. The lateral extent of the recorded holograms was limited to $42\lambda \times 42\lambda$. All the simulations for this work were done at $\lambda = 600$ nm. The results, however, are invariant to wavelength given that all the dimensions are scaled proportional to λ.

4.4.3 Training

For training diffractive networks to all-optically perform holographic reconstruction, we computed the in-line holograms corresponding to the training objects by numerically simulating the propagation of a plane wave scattered by the object from the object plane to the hologram recording plane. Assuming that the hologram amplitude transmittance is proportional to the recorded intensity, the input optical field v^0 was set to the normalized hologram intensity. The

reconstructed object pattern was defined as the intensity of the output optical field u^{L+1}, i.e., $\hat{y} = |u^{L+1}|^2$. We defined the training loss function (to be minimized) as:

$$L = L_{pixel} + 1000L_{fourier} + \eta L_{efficiency} \tag{4.4}$$

$$L_{pixel} = \frac{1}{N_p}\sum_{p=1}^{N_p}\left|y_p - \hat{y}_p\right| \tag{4.5}$$

$$L_{fourier} = \frac{1}{N_p}\sum_{p=1}^{N_p}\left|F\{y\}_p - F\{\hat{y}\}_p\right|^2 \tag{4.6}$$

$$L_{efficiency} = 1 - \frac{P_{out}}{P_{illum}} \tag{4.7}$$

Here, y refers to the target (object image), and$F\{\cdot\}$denotes the Fourier transform operation, P_{illum} is the power of the wave illuminating the hologram and P_{out} is the power over the region of interest (i.e., the reconstruction field-of-view) at the output plane. The coefficient η of the diffraction efficiency-related loss term ($L_{efficiency}$) is a training hyperparameter which can be used to control the diffraction efficiency of the resulting optical network (see Fig. 4.7). We used Adam optimization algorithm for minimizing the loss function [102].

For designing the extended depth-of-field diffractive networks, we treated the distance between the object and the hologram recording plane during the training phase as a random variable z_{train}, drawn from a uniform distribution centered around the nominal recording distance z_0 , i.e., $z_{train} \sim Uniform\big((1-\delta_{tr})z_0, (1+\delta_{tr})z_0\big)$ where δ_{tr} quantifies the width of the distribution. During blind testing phase, we defined $\delta z = z_{test} - z_0$ to be the deviation of the recording distance from the nominal distance.

For training the diffractive models, we used the MNIST handwritten digit dataset, augmented with an additional custom-built dataset of generic shapes, e.g., patches, circles, straight-line and

circular-line gratings etc. A few examples of the images from this custom-built dataset can be found in Fig. 4.8. The number of images in the training dataset was 110,000 (55,000 from the MNIST training set and 55,000 from the custom prepared image dataset). None of the images used for evaluating the model performances reported in this work were in the training dataset.

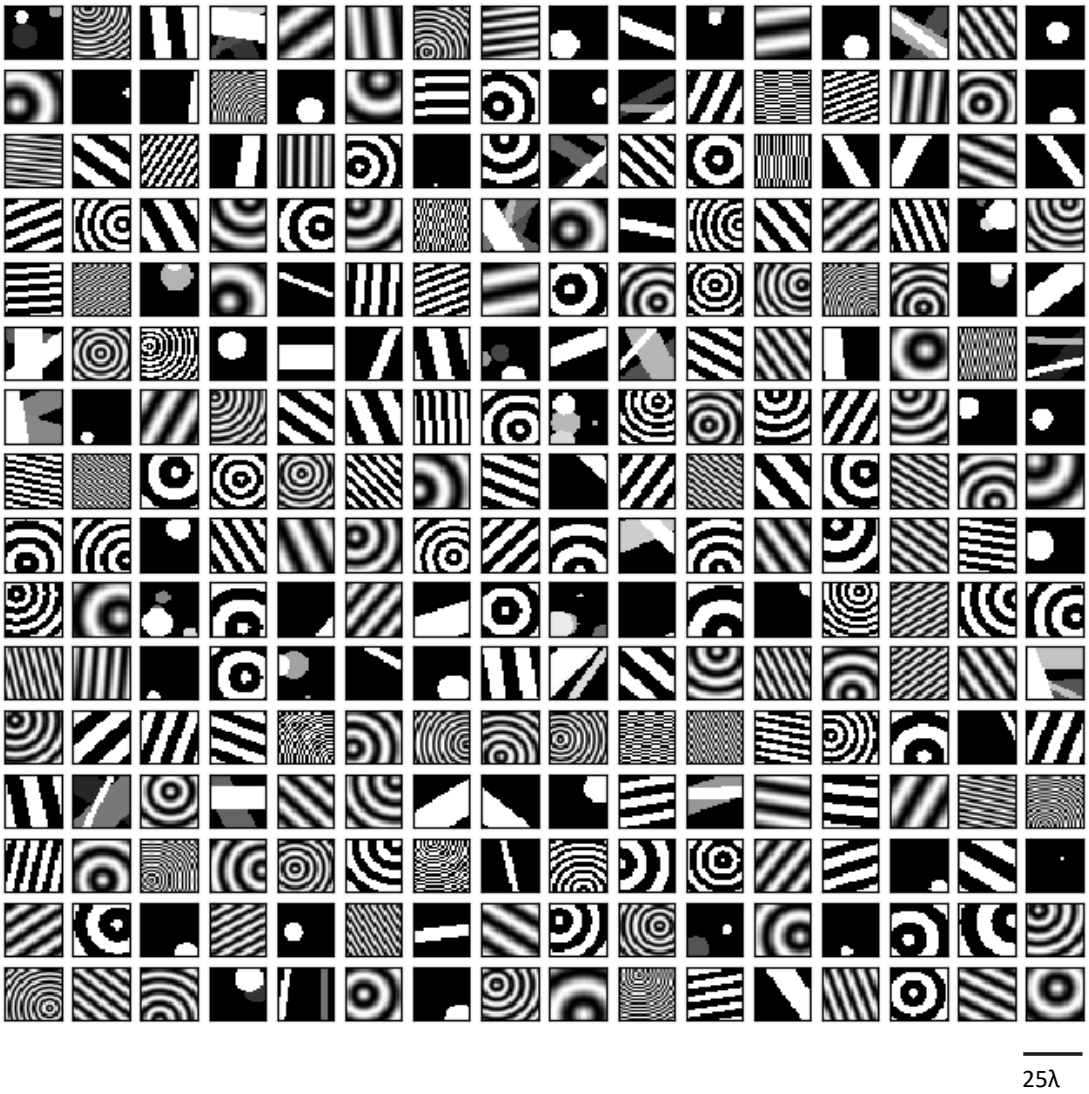


Figure 4.8 Examples of training images. Some examples of the images used to augment the MNIST dataset for the purpose of training the diffractive networks for all-optical holographic reconstruction. 55,000 such images were used together with 55,000 images in the MNIST training set, i.e., the total size of the augmented training image set was 110,000. The remaining 5,000

images in the MNIST training set were set aside for validation. None of the images used in blind testing for evaluating the reported models were in the augmented training set or the validation set.

The model was implemented and trained using TensorFlow [128]. The Rayleigh-Sommerfeld diffraction integral was computed using the Angular Spectrum method [15]. We used the native TensorFlow implementation of Adam optimization algorithm with the default hyperparameter values, i.e., 0.001 for the learning rate, and 0.9 and 0.999 for the exponential decay rates for the first and the second moments of the gradients, respectively. The models were trained for 50 epochs with a mini-batch size of 4. The training of a model typically took ~8 hours on a GTX 1080 Ti GPU (Nvidia Inc.) on a machine running Windows 10 operating system (Microsoft Inc.).

Chapter 5 Learning Diffractive Optical Communication Around Arbitrary Opaque Occlusions

In this chapter, I introduce a novel scheme based on digital encoding and diffractive all-optical decoding for free-space optical communication around arbitrarily shaped opaque occlusions. Parts of this chapter have been published in the peer-reviewed research article "M. S. S. Rahman, T. Gan, E. A. Deger, Ç. Işıl, M. Jarrahi, and A. Ozcan, 'Learning diffractive optical communication around arbitrary opaque occlusions,' Nature Communications **14**, 6830 (2023)," (Licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/).

Free-space optical systems are emerging for high data rate communication and transfer of information in indoor and outdoor settings. However, free-space optical communication becomes challenging when an occlusion blocks the light path. Here, we demonstrate, for the first time, a direct communication scheme, passing optical information around a fully opaque, arbitrarily shaped obstacle that partially or entirely occludes the transmitter's field-of-view. In this scheme, an electronic neural network encoder and a diffractive optical network decoder are jointly trained using deep learning to transfer the optical information or message of interest around the opaque occlusion of an arbitrary shape. The diffractive decoder comprises successive spatially engineered passive surfaces that process optical information through light-matter interactions. Following its training, the encoder-decoder pair can communicate any arbitrary optical information around opaque occlusions, where information decoding occurs at the speed of light propagation. For occlusions that change as a function of time, the encoder neural network can be retrained to successfully communicate with the existing diffractive decoder, without changing the physical layer(s) already deployed. In an alternative design strategy, the encoder-decoder pair can be trained

to become resilient against various unknown changes in the occlusion shape and size. We also validate this framework experimentally in the terahertz spectrum using a 3D-printed diffractive decoder to communicate around a fully opaque occlusion. Scalable for operation in any wavelength regime, this scheme could be particularly useful in emerging high data-rate free-space communication systems.

5.1 Introduction

Traditionally radio frequency (RF) and microwave have dominated the area of wireless communication. To meet the growing need for faster data transfer rates, RF systems employ increasingly complex coding, multiple antennas, and higher carrier frequencies [162]. For example, by utilizing higher frequency bands, 6^{th} generation (6G) technology is predicted to provide 100 to 1000 times faster speed than 5^{th} generation (5G) systems deployed for wireless communication [163]. With ever-increasing data rates, maintaining the performance of these schemes will become more challenging. One possible solution is to shift to shorter wavelengths, such as the ultraviolet (UV), visible or infrared (IR) regions of the electromagnetic spectrum, which provide much wider bandwidths compared to radio waves or microwaves [162,164–166]. However, free-space optical communication becomes challenging when opaque occlusions block the light path. Non-line-of-sight (NLOS) communication, which exploits diffusely reflected waves from a nearby scattering medium, has been used as a way around the occlusion problem [167–171]. However, the adaptability of these solutions to emerging optical communication techniques for channel capacity expansion faces challenges since even weak turbulence can cause a significant loss of information [171]. Furthermore, the low power efficiency arising from the weak scattering

or diffuse reflection is another limitation of NLOS communication. Other NLOS systems, e.g., for imaging around corners, also exist [172–185]; these approaches, however, involve relatively slow and power-consuming digital methods for image reconstruction. Alternative methods have been developed for image transmission through thick (but transmitting) occlusions, including e.g., holography [186–188], adaptive wavefront control [189–191], and others [192,193]. However, many of these techniques also involve digital reconstruction of the information, often requiring iterative algorithms. Moreover, most of these are applicable for multiple-scattering media, and do not address situations where the light path is either partially or entirely obstructed by opaque occlusions *with zero light transmittance*.

Here we demonstrate a novel scheme for directly communicating optical information of interest around zero-transmittance occlusions using electronic encoding at the transmitter and all-optical diffractive decoding at the receiver. In our scheme, an electronic neural network, trained in unison with an all-optical diffractive decoder, encodes the message of interest to effectively bypass the opaque occlusion and be decoded at the receiver by an all-optical decoder, using passive diffraction through thin structured layers. This all-optical decoding is performed on the encoded wavefront that carries the optical information or the message of interest, after its obstruction by an arbitrarily shaped opaque occlusion. The diffractive decoder processes the secondary waves scattered through the edges of the opaque occlusion using a passive, smart material comprised of successive spatially engineered surfaces, [38] and performs the reconstruction of the hidden information at the speed of light propagation through a thin diffractive volume that axially spans $< 100\times\lambda$, where λ is the wavelength of the illumination light.

We show that this combination of electronic encoding and all-optical decoding is capable of direct optical communication between the transmitter and the receiver even when the opaque

occlusion body entirely blocks the transmitter's field-of-view (FOV). We also report an experimental demonstration of this scheme using a 3D-printed diffractive decoder that operates at the terahertz spectrum. Furthermore, we demonstrate that this scheme could be configured to be misalignment-resilient as well as highly power efficient, reaching diffraction efficiencies of >50% at its output. In the case of opaque occlusions that change their size over time, we also report that the encoder neural network could be retrained to successfully communicate with an existing diffractive decoder, without changing its physical structure that is already deployed. We also show that our encoder/decoder framework can be jointly trained to be resilient against unknown, random dynamic changes in the occlusion size and/or shape, without the need to retrain the encoder or the decoder. This makes the presented concept highly dynamic and easy to adapt to external and uncontrolled/unknown changes that might happen between the transmitter and receiver apertures. This framework can be extended for operation at different parts of the electromagnetic spectrum, and would find applications in emerging high-data-rate free-space communication technologies, under scenarios where different undesired structures occlude the direct channel of communication between the transmitter and the receiver.

5.2 Results

A schematic depicting the optical communication scheme around an opaque occlusion with zero light transmittance is shown in Fig. 5.1a. The message to be transmitted, e.g., the image of an object, is fed to an electronic/digital neural network, which outputs a phase-encoded optical representation of the message. This code is imparted onto the phase of a plane-wave illumination, which is transmitted toward the decoder through an aperture that is partially or entirely blocked by

an opaque occlusion. The scattered waves from the edges of the opaque occlusion travel toward the receiver aperture as secondary waves, where a diffractive decoder all-optically decodes the received light to directly reproduce the message/object at its output FOV. This decoding operation is completed as the light propagates through the thin decoder layers. For this collaborative encoding-decoding scheme, the electronic encoder neural network and the diffractive decoder are jointly trained in a data-driven manner for effective optical communication, bypassing the fully opaque occlusion positioned between the transmitter aperture and the receiver.

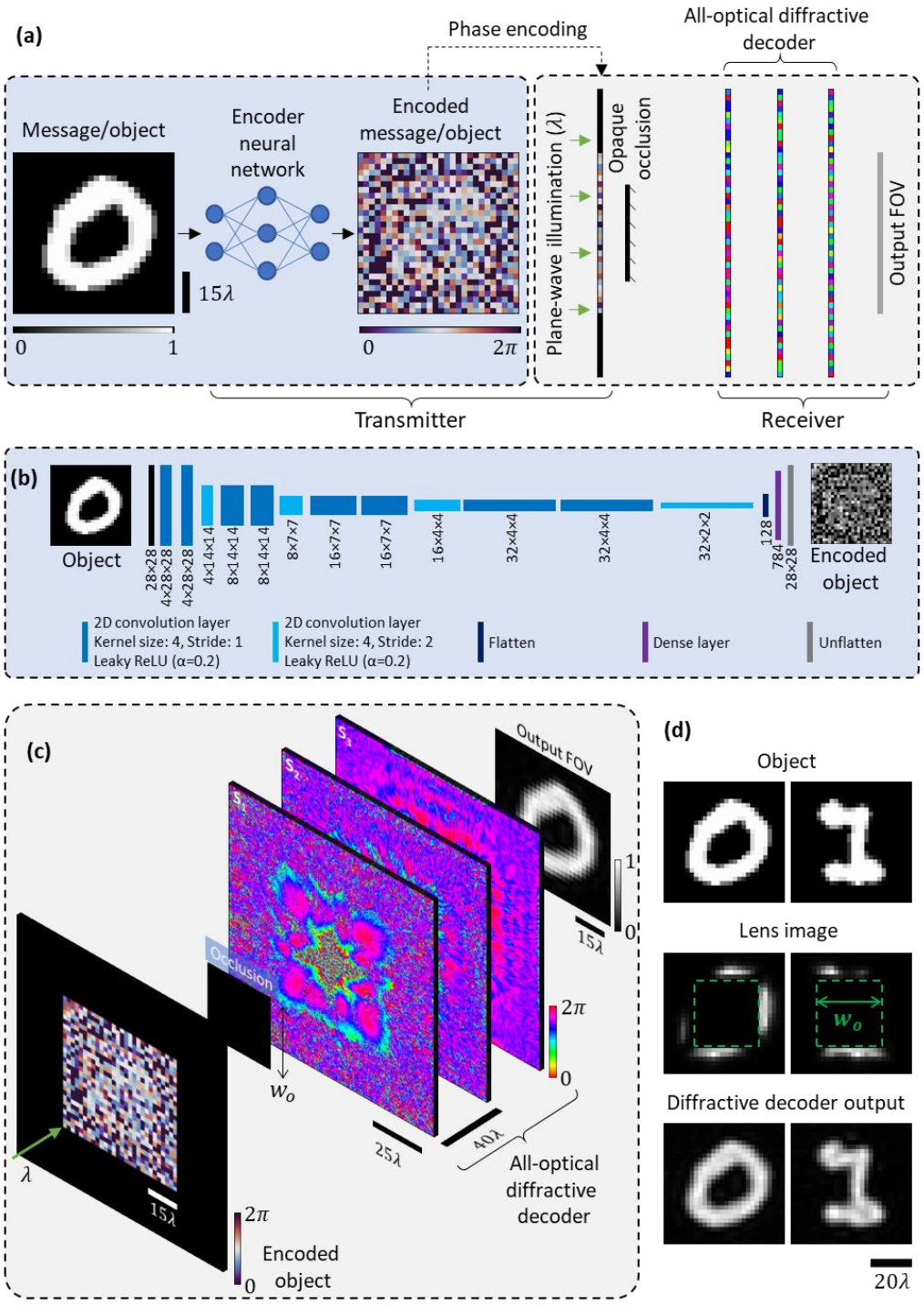


Figure 5.1 Schematic of the optical communication framework around fully opaque occlusions using electronic encoding and diffractive all-optical decoding. (a) An electronic neural network encoder and an all-optical diffractive decoder are trained jointly for communicating around an opaque occlusion. For a message/object to be transmitted, the electronic encoder outputs a coded 2D phase pattern, which is imparted onto a plane wave at the transmitter aperture. The

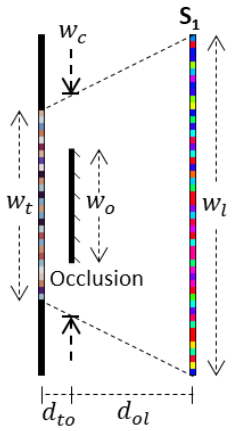
phase-encoded wave, after being obstructed and scattered by the fully opaque occlusion, travels to the receiver, where the diffractive decoder all-optically processes the encoded information to reproduce the message on its output FOV. (b) The architecture used for the convolutional neural network (CNN) encoder throughout this work. (c) Visualization of different processes, such as the obstruction of the transmitted phase-encoded wave by the occlusion of width w_o and the subsequent all-optical decoding performed by the diffractive decoder. The diffractive decoder comprises L surfaces $(S_1, \cdots, S_L)$ with phase-only diffractive features. In this figure, $L = 3$ is illustrated as an example. (d) Comparison of the encoding-decoding scheme against conventional lens-based imaging.

Figures 5.1b and 5.1c provide a deeper look into the encoder and the decoder architectures used in this work. As shown in Fig. 5.1b, the convolutional neural network (CNN) encoder is composed of several convolution layers, followed by a dense layer representing the encoded output. This dense layer output is rearranged into a 2D-array corresponding to the spatial grid that maps the phase-encoded transmitter aperture. We assumed that both the desired messages and the phase codes to be transmitted comprise 28×28 pixels unless otherwise stated. The architecture of the encoder remains the same across all the designs reported in this paper. The architecture of the diffractive decoder, which decodes the transmitted and obstructed phase-encoded waves, is shown in Fig. 5.1c. This figure shows a diffractive decoder comprising $L = 3$ spatially engineered surfaces/layers (i.e., S_1, S_2 and S_3); however, in this work, we also report results for designs comprising diffractive decoders with $L = 1$ and $L = 5$ layers, used for comparison. Together with the encoder CNN parameters, the spatial features of the diffractive surfaces of the all-optical decoder are optimized to decode the encoded and blocked/obscured wavefront. In this work, we consider *phase-only* diffractive features, i.e., only the phase values of the features at each diffractive surface are trainable (see the 'Methods' section for details). Figure 5.1 also compares the performance of the presented electronic encoding and diffractive decoding scheme to that of a lens-based camera. As shown in Fig. 5.1d, the lens images reveal significant loss of information

caused by the opaque occlusion in a standard camera system, showcasing the scale of the problem that is addressed through our proposed approach.

For all the models reported in this work, the data-driven joint training of the electronic encoder CNN and the diffractive decoder was accomplished by minimizing a structural loss function defined between the object (ground-truth message) and the diffractive decoder output, using 55,000 images of handwritten digits from the MNIST [194] training dataset, augmented by 55,000 additional custom-generated images (see the 'Methods' section as well as Fig. 5.21 for details). All our results come from blind testing with objects/messages never used during training.

To bring more insights into the occlusion width w_o, we define the critical width w_c as the minimum width of the occlusion at which no direct ray can reach the receiver aperture from the transmitter aperture; see Fig. 5.2. In addition to the widths of the transmitter (w_t) and the receiver (w_l) apertures, this critical occlusion width w_c is also a function of the ratio of the distances of the transmitter and the receiver from the occlusion, i.e., d_{to} and d_{ol}, respectively; it can be written as $w_c = w_t(1 + d_{to}/d_{ol})^{-1} + w_l(1 + d_{ol}/d_{to})^{-1}$ as detailed in Fig. 5.2. For all our simulations, $w_t \approx 59.73\lambda$, $w_l \approx 106.67\lambda$ and $d_{to}/d_{ol} = 1/8$ were used, resulting in $w_c \approx 64.95\lambda$. In our analyses and figures, we report the occlusion width w_o as a fraction of w_c, where in some cases $w_o > w_c$, i.e., no direct ray reaches the receiver aperture from the transmitter aperture.



w_t: Transmitter aperture width
w_o: Occlusion width
w_l: Receiver aperture (diffractive layer-1) width
d_{to}: Distance from transmitter to occlusion
d_{ol}: Distance from occlusion to diffractive layer-1

w_c: Minimum occlusion width for which no light ray from the transmitter aperture reaches the diffractive decoder directly

$$w_c = w_t \frac{d_{ol}}{d_{to} + d_{ol}} + w_l \frac{d_{to}}{d_{to} + d_{ol}}$$

Figure 5.2 The critical occlusion width w_c. For $w_o \geq w_c$, there is no direct path of light rays for communication between the transmitter and the receiver apertures.

5.2.1 Numerical analysis of diffractive optical communication around opaque occlusions

First, we compare, for various levels of opaque occlusions, the performance of trained encoder-decoder pairs with different diffractive decoder architectures in terms of the number of diffractive surfaces employed. Specifically, for each of the occlusion width values, i.e., $w_o = 32.0\lambda \approx 0.5w_c$, $w_o = 53.3\lambda \approx 0.8w_c$ and $w_o = 74.7\lambda \approx 1.15w_c$, we designed three encoder-decoder pairs, with $L = 1$, $L = 3$, and $L = 5$ diffractive layers within the decoders, and compared the performance of these designs for new handwritten digits in Fig. 5.3a. This blind testing refers to 'internal generalization' because even though these particular test objects were never used in training, they are from the same dataset. As shown in Fig. 5.3a, even $L = 1$ designs can faithfully decode the message for optical communication around these various levels of occlusions. Furthermore, as the number of layers in the decoder increases to $L = 3$ or $L = 5$, the quality of the output also gets better. While the performance of the $L = 1$ design deteriorates slightly as w_o increases, the $L = 3$

and $L = 5$ designs do not show any appreciable degradation in qualitative performance for such bigger occlusions. Note that for an occlusion size of $w_o = 74.7\lambda \approx 1.15w_c$, none of the ballistic photons can reach the receiver aperture since the opaque occlusion completely blocks any direct light ray from the encoding transmitter aperture. Nonetheless, the scattering from the occlusion edges suffices for the encoder-decoder pair to communicate faithfully.

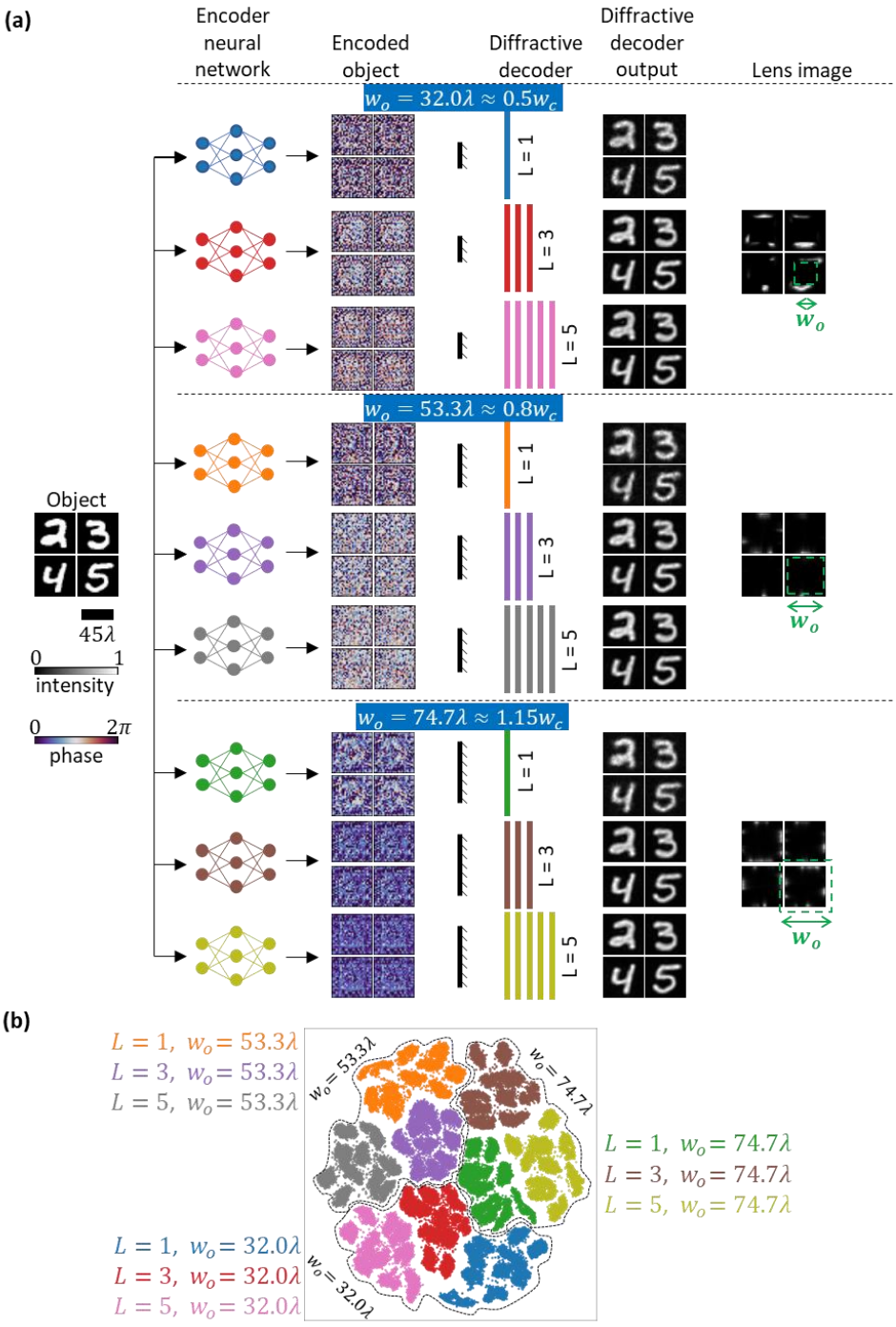


Figure 5.3 Generalization of trained encoder-decoder pairs to previously unseen handwritten digit objects. (a) For different values of the occlusion width w_o, the performances of trained encoder-decoder pairs with different numbers of decoder layers (L) are depicted for comparison. (b) t-SNE-based visualization of the electronic encoder outputs for the nine different designs of Fig. 5.3a.

The learned encoder phase representations of the objects by different designs of Fig. 5.3a look completely random to the human eye. To gain more insights into the generalization of these designs, we performed dimensionality reduction analysis on these encoded phase patterns representing the input objects [195]. For this analysis, we prepared a dataset of size 9×10,000=90,000 comprising the encoded phase objects corresponding to previously unseen 10,000 MNIST test images, for each one of these 9 designs shown in Fig. 5.3a. Subsequently, we applied an unsupervised dimensionality reduction algorithm, t-distributed stochastic neighbor embedding (t-SNE) [196], to learn a 2D manifold of these encoded phase patterns for all the encoder/decoder designs. A scatterplot of the projections of these encoded phase patterns on the learned manifold is presented in Fig. 5.3b. The clustering of these projections into 9 subgroups corresponding to the 9 different designs with unique (w_o, L) attests to the generalization of these designs, indicating that the learned object representations in the phase space are specific to each architecture rather than being random. Figure 5.3b also shows the formation of three superclusters for each design corresponding to the three different occlusion sizes.

To supplement the results of Fig. 5.3, we also quantified the performance of different encoder-decoder pairs designed for increasing occlusion widths (w_o), in terms of peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) [197] averaged over 10,000 handwritten digits from the MNIST test set (never used before); see Figs. 5.4a and 5.4b, respectively. With increasing w_o, we see a larger decrease in the performance of $L = 1$ designs compared to $L = 3$ and $L = 5$ designs. Interestingly, there is a slight improvement in the performance of $L = 1$ and $L = 3$ decoders as w_o surpasses $w_t = 59.73\lambda$ (the transmitter aperture width); this improved level of performance is retained for $w_o > w_t$, the cause of which will be discussed later in our Discussion section.

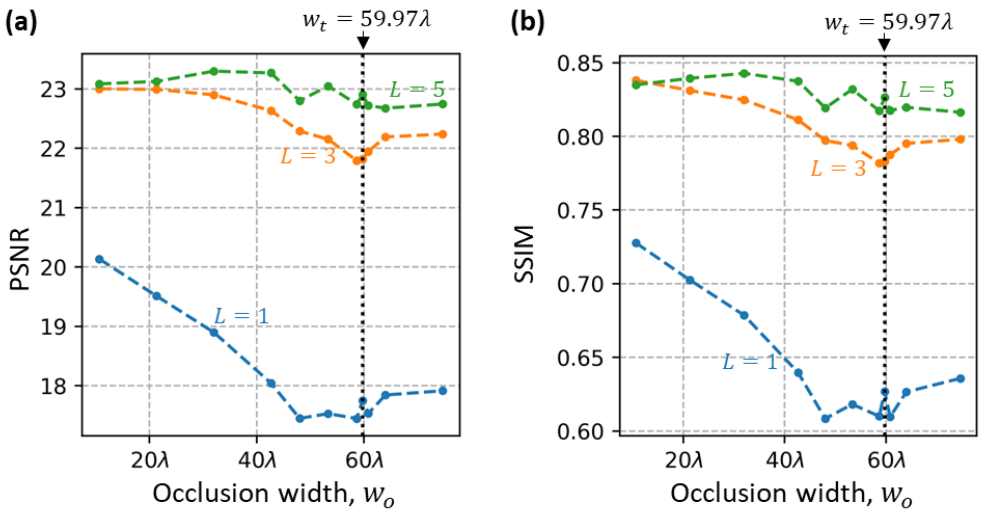


Figure 5.4 Quantification of the performance of encoder-decoder pairs with different numbers of decoder layers (L) trained for increasing occlusion widths (w_o) in terms of (a) PSNR and (b) SSIM between the diffractive decoder outputs and the ground-truth messages. The PSNR and SSIM values are calculated by averaging over 10,000 MNIST test images. w_t refers to the width of the transmitter aperture.

Next, for the same designs reported in Fig. 5.3, we explored the *external* generalization of these encoder-decoder pairs by testing their performance on types of objects that were not represented in the training set; see Fig. 5.5. For this analysis, we randomly chose two images of fashion products from the Fashion-MNIST [156] test set (top) and two additional images from the CIFAR-10 [87] test set (bottom). As shown in Fig. 5.5, our encoder-decoder designs show excellent generalization to these completely different object types. Although the decoder outputs of the $L = 1$ decoder designs for $w_o = 53.3\lambda \approx 0.8w_c$ and $w_o = 74.7\lambda \approx 1.15w_c$ are slightly degraded, the objects are still recognizable at the output plane even for the complete blockage of the transmitter aperture by the occlusion.

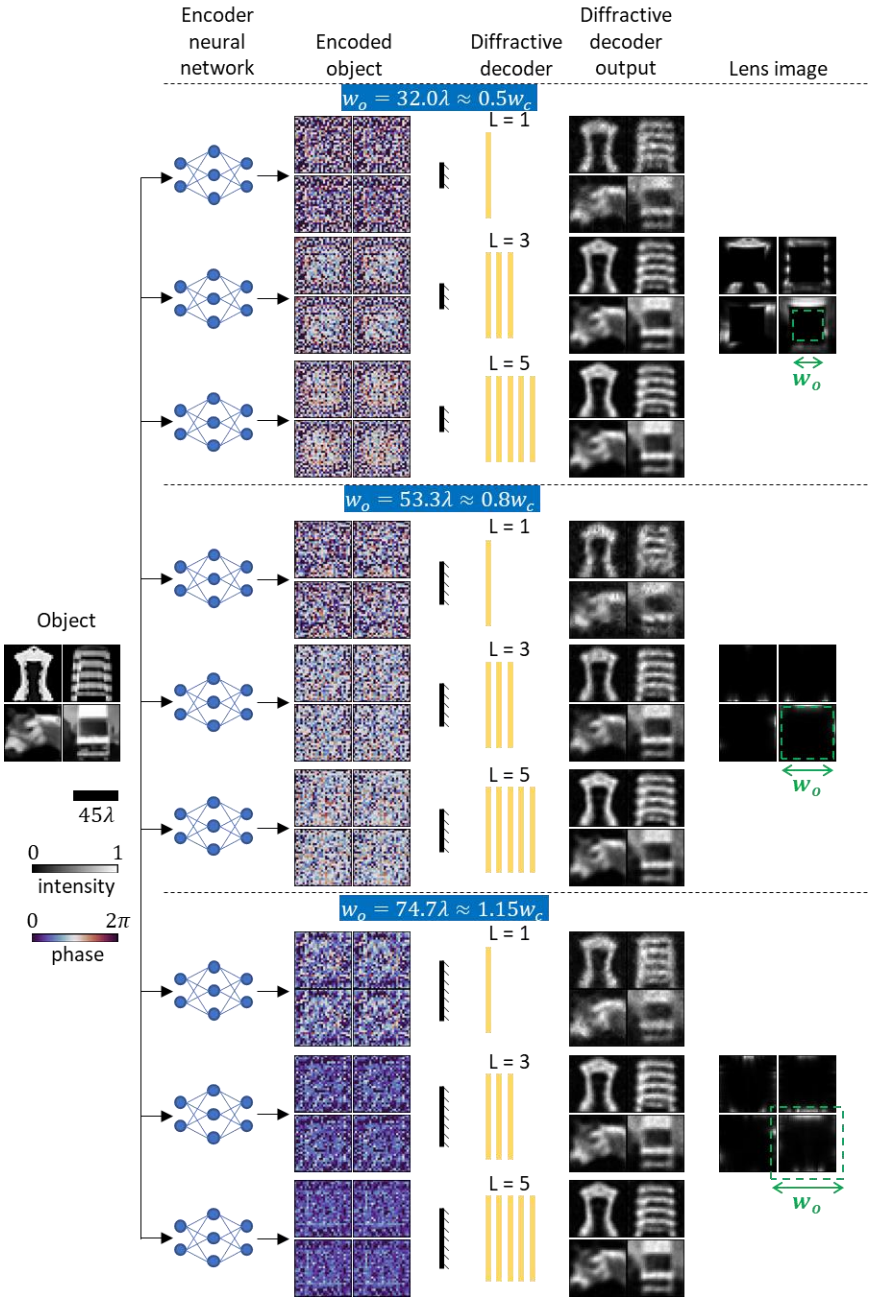


Figure 5.5 Same as Fig. 5.3a, except that these results reflect external generalizations on object types different from those used during the training.

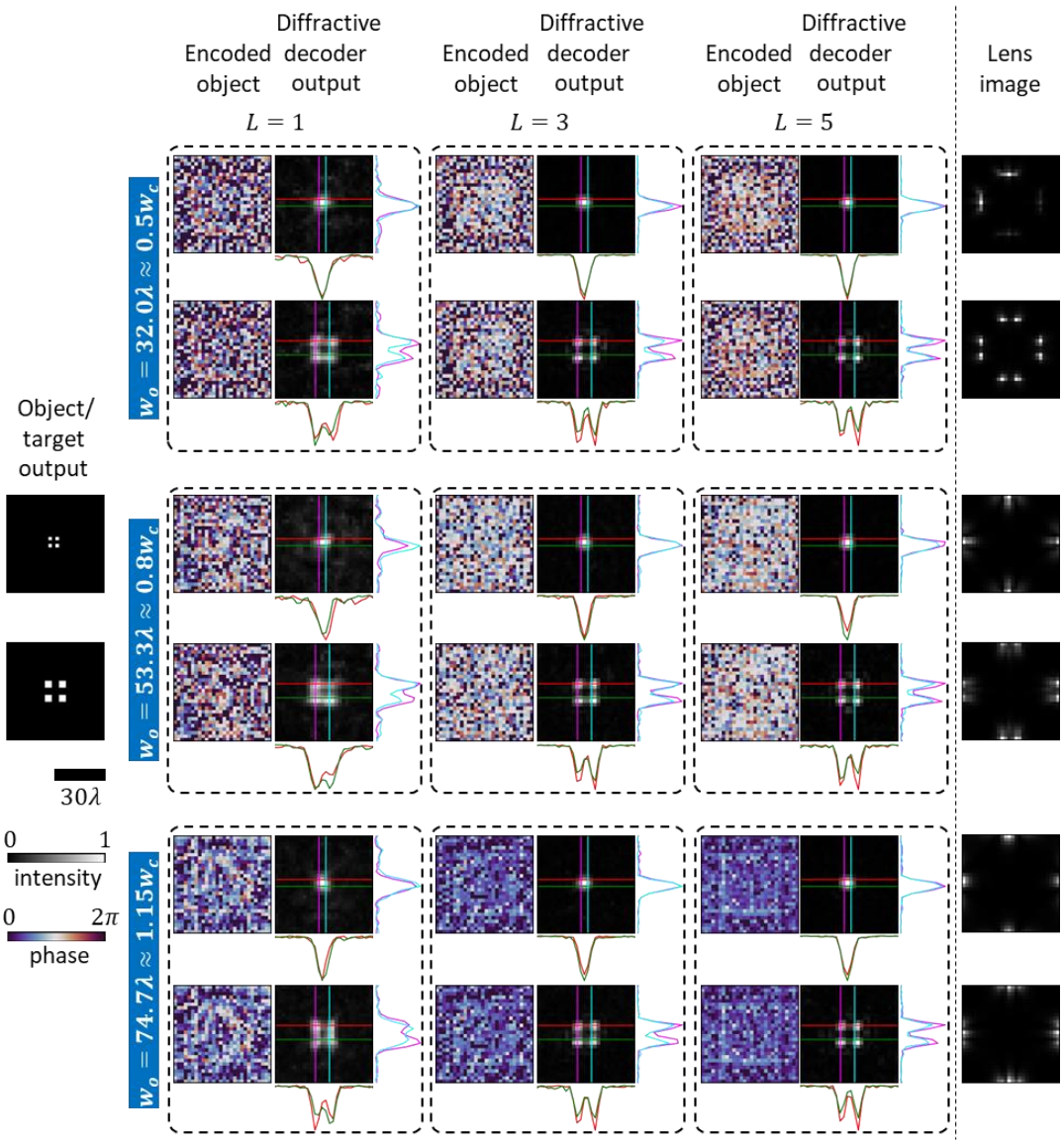


Figure 5.6 Output resolution of diffractive decoders corresponding to $L = 1$, $L = 3$, and $L = 5$ designs trained for different occlusion widths (w_o). As for the objects, the vertical/horizontal separation between the inner edges of the dots is 2.12λ for the test pattern on the top and 4.24λ for the one below. The diffractive decoder outputs are accompanied by cross-sections taken along the color-coded vertical/horizontal lines.

We also investigated the ability of these designs to resolve closely separated features in their outputs. For this purpose, we transmitted test patterns consisting of four closely spaced dots, and the corresponding diffractive decoder outputs are shown in Fig. 5.6. For the top (bottom) pattern,

the vertical/horizontal separation between the inner edges of the dots is 2.12λ (4.24λ). None of the designs could resolve the dots separated by 2.12λ; however, the dots separated by 4.24λ were resolved by all the encoder-decoder designs with good contrast, as can be seen from the cross-sections accompanying the output images in Fig. 5.6. It is to be noted that this resolution limit of 4.24λ is due to the output pixel size, which was set as 2.12λ in our simulations. The effective resolution of our encoder-decoder system can be further improved within the diffraction limit of light by using higher-resolution objects and a smaller pixel size during the training.

5.2.2 Impact of phase bit depth on performance

Here, we study the effect of a finite bit-depth b_q phase quantization of the encoder plane as well as the diffractive layers. For the results presented so far, we did not assume either to be quantized, i.e., an infinite bit-depth of phase quantization was assumed. For the $w_o = 32.0\lambda \approx 0.5w_c$, $L = 3$ design (trained assuming an infinite bit-depth $b_{q,tr} = \infty$), the first row of Fig. 5.7a shows the impact of quantizing the encoded phase patterns as well as the diffractive layer phase values with a finite bit-depth $b_{q,te}$. This represents an "attack" on the design since the encoder CNN and the diffractive decoder were trained without such a phase bit-depth restriction; stated differently, they were trained with $b_{q,tr} = \infty$ and are now tested with finite levels of $b_{q,te}$. For the $b_{q,tr} = \infty$ designs, the output quality remains unaffected for $b_{q,te} = 8$; however, there is considerable degradation under $b_{q,te} = 4$, and we face complete failure with $b_{q,te} = 3$ and $b_{q,te} = 2$. However, this sharp performance degradation with decreasing $b_{q,te}$ can be amended by considering the finite bit-depth during training. To showcase this, we trained two additional designs with $w_o = 32.0\lambda$ and $L = 3$ assuming finite bit-depths of $b_{q,tr} = 4$ and $b_{q,tr} = 3$; their

blind testing performance with decreasing $b_{q,te}$ is reported in the second and third rows of Fig. 5.7a, respectively. Both of these designs show robustness against bit-depth reduction up to $b_{q,te} = 3$ (i.e., 8-level phase quantization at the encoder and decoder layers). However, even with $b_{q,te} = 2$ (only 4-level phase quantization), the outputs are still recognizable as shown in Fig. 5.7. We also quantified the performance (PSNR and SSIM) of these three designs ($b_{q,tr} = \infty$ $b_{q,tr} = 4$, $b_{q,tr} = 3$) for different $b_{q,te}$ levels; see Figs. 5.7b and 5.7c. These quantitative comparisons restate the same conclusion: training with a lower $b_{q,tr}$ results in robust encoder-decoder designs that preserve their optical communication quality despite a reduction in the bit-depth $b_{q,te}$, albeit with a relatively small sacrifice in the output performance.

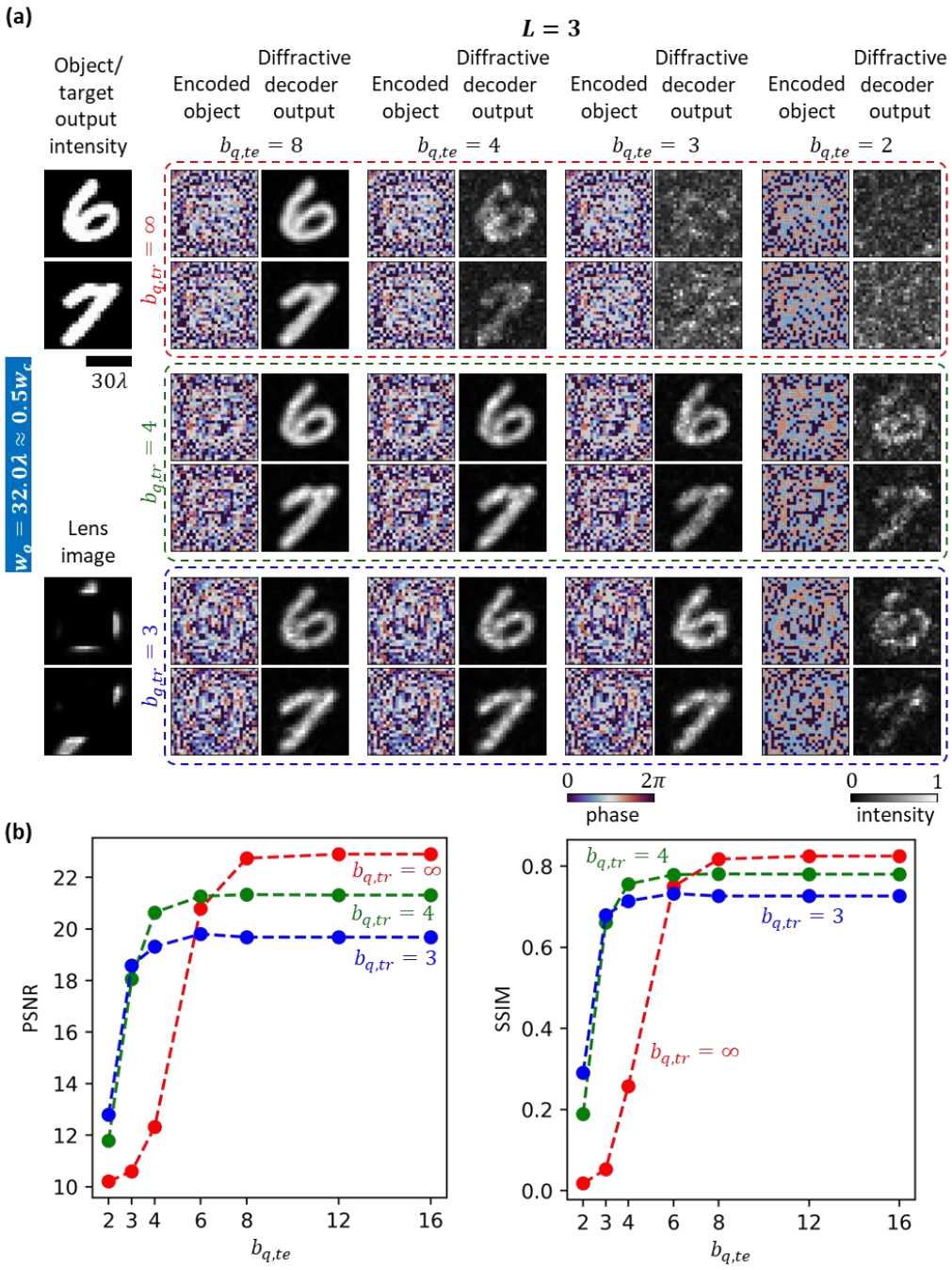


Figure 5.7 Effect of the phase bit depth of the encoded object and the diffractive layer features on the performance of trained encoder-decoder pairs. (a) Qualitative performance of the designs, which are trained assuming a certain phase quantization bit depth $b_{q,tr}$, reported as a function of the bit depth used during testing $b_{q,te}$. (b) For different $b_{q,tr}$, PSNR and SSIM values are plotted as a function of $b_{q,te}$. The PSNR and SSIM values are evaluated by averaging the results of 10,000 test images from the MNIST dataset.

5.2.3 Impact of misalignments on performance

Next, we focus on the effect of physical misalignments on the performance of our framework for communication around opaque occlusions. First, we explore the effect of random misalignments of the physical layers of the diffractive decoder. For this analysis, we model the misalignments of the layers using random variables $\delta_{x,l} \sim Uniform(-\delta_{lat}, \delta_{lat})$, $\delta_{y,l} \sim Uniform(-\delta_{lat}, \delta_{lat})$ and $\delta_{z,l} \sim Uniform(-\delta_{ax}, \delta_{ax})$ where $\delta_{x,l}$, $\delta_{y,l}$ and $\delta_{z,l}$ denote the displacements of the diffractive layer l from its nominal position along x, y and z directions, respectively; $l = 1, \cdots, L$. δ_{lat} and δ_{ax} are the parameters quantifying the degree of the lateral and axial random misalignments, respectively. In Fig. 5.8, we present the effect of only the lateral random misalignments of the diffractive layers ($\delta_{lat} \geq 0$) assuming no axial misalignment ($\delta_{ax} = 0$). These results reveal that the design trained without taking such random lateral misalignments of the layers into consideration ($\delta_{lat,tr} = 0$) fails to successfully communicate through an opaque occlusion when tested with various levels of random lateral misalignments, i.e., $\delta_{lat,te} > 0$. This sensitivity to random physical misalignments can be improved by taking such misalignments into account during the design phase by training with $\delta_{lat,tr} > 0$. We can see from the same Fig. 5.8 that the performance of the $\delta_{lat,tr} = 4\lambda$ design remains decent up to $\delta_{lat,te} = 8\lambda$, whereas for the $\delta_{lat,tr} = 8\lambda$ design, there is no perceptible degradation in the performance as $\delta_{lat,te}$ goes from 0 to 8λ. Fig. 5.8b further reports, as a function of $\delta_{lat,te}$, the average PSNR and average SSIM values for these designs trained with different $\delta_{lat,tr}$, showing that the resilience of encoder-decoder designs against random lateral misalignments can be significantly improved by training with suitably chosen $\delta_{lat,tr}$, with a modest trade-off in communication performance.

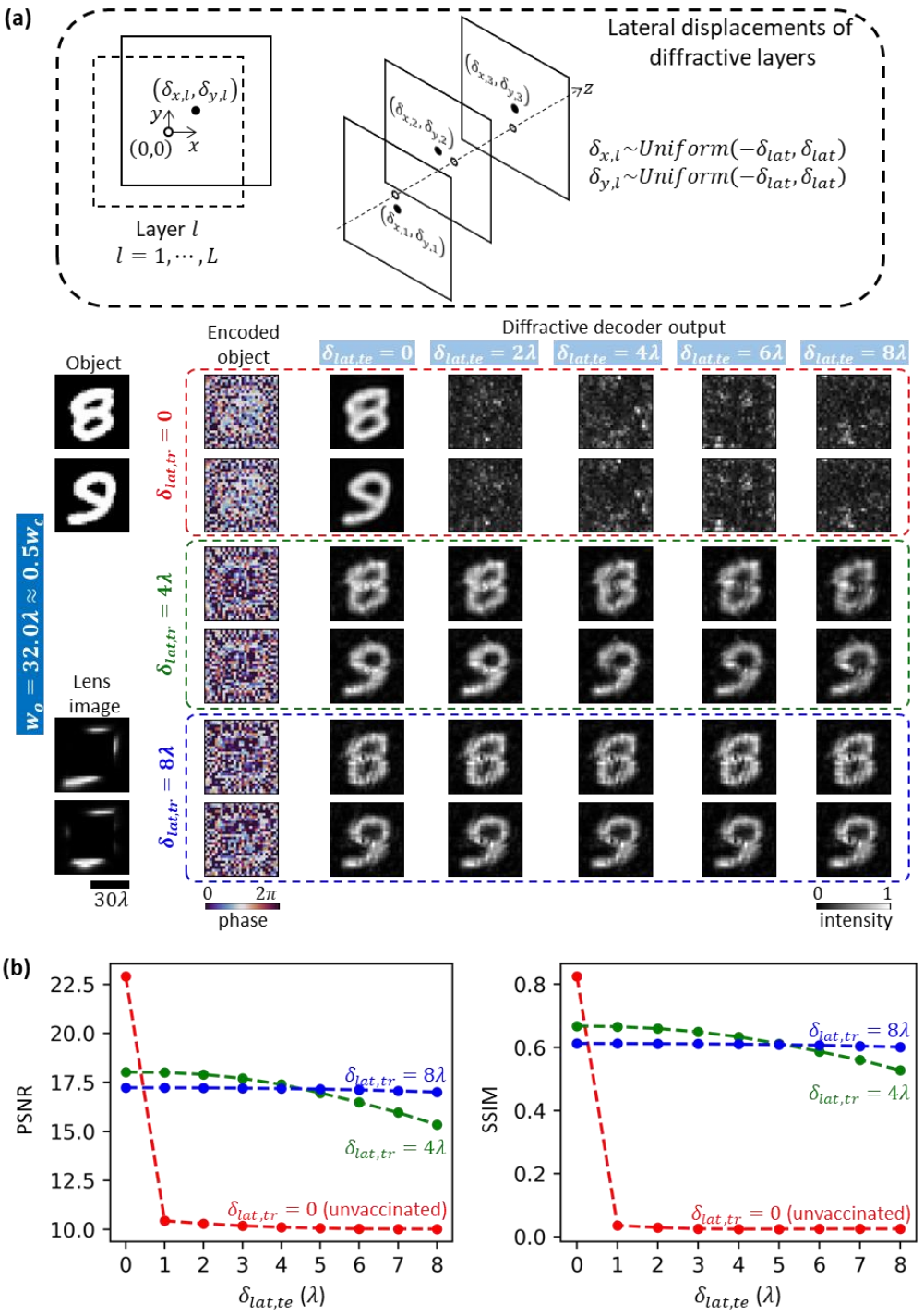


Figure 5.8 Impact of lateral random misalignments of the layers of the diffractive decoder on the performance of our framework for communication around opaque occlusions. (a) Examples of the diffractive decoder outputs at different levels of lateral random misalignments $\delta_{lat,te}$ during the testing for three designs trained with different levels of "vaccination" against such random misalignments ($\delta_{lat,tr} = 0$, $\delta_{lat,tr} = 4\lambda$, and $\delta_{lat,tr} = 8\lambda$). (b) Average PSNR and

SSIM values of the diffractive decoder outputs as a function of $\delta_{lat,te}$ for the designs corresponding to $\delta_{lat,tr} = 0$, $\delta_{lat,tr} = 4\lambda$, and $\delta_{lat,tr} = 8\lambda$.

The same conclusion also holds for axial random misalignments, as shown in Fig. 5.9. It is to be noted that as the resilience to large random axial misalignments (e.g., $\delta_{ax,te} = 8\lambda$) is attained by training with $\delta_{ax,tr} > 0$, the decrease in performance for no misalignments ($\delta_{ax,te} = 0$) is virtually negligible, which is highly desired. Following a similar strategy, our jointly trained encoder-decoder pair designs can also be made resilient to lateral and axial random displacements of the opaque occlusion as illustrated in Figs. 5.10 and 5.11.

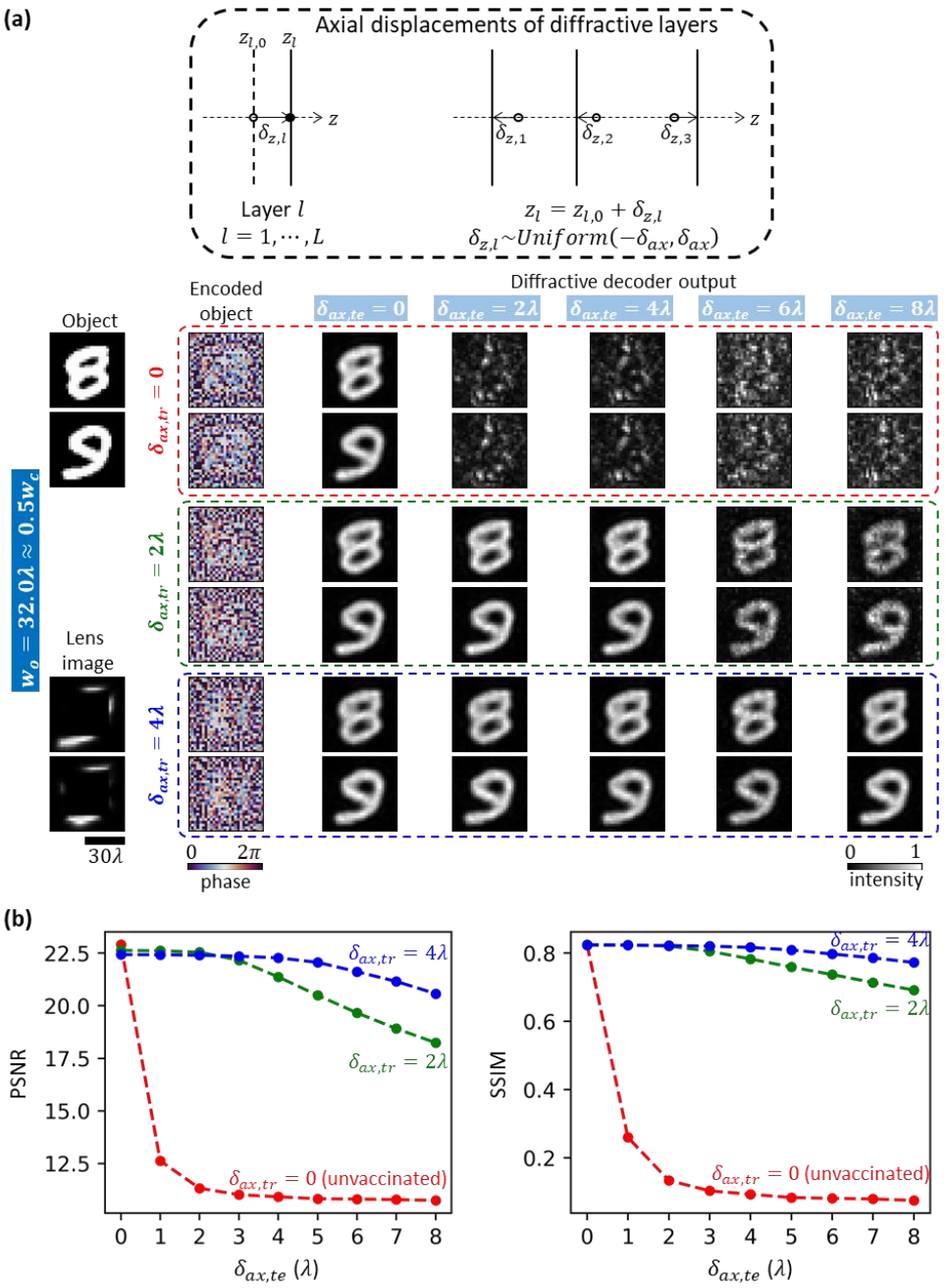


Figure 5.9 Impact of axial random misalignments of the layers of the diffractive decoder on the performance of our framework for communication around opaque occlusions. (a) Examples of the diffractive decoder outputs at different levels of axial random misalignments $\delta_{ax,te}$ during the testing for three designs trained with different levels of "vaccination" against such random misalignments ($\delta_{ax,tr} = 0$, $\delta_{ax,tr} = 2\lambda$, and $\delta_{ax,tr} = 4\lambda$). (b) Average PSNR and SSIM values of the diffractive decoder outputs as a function of $\delta_{ax,te}$ for the designs corresponding to $\delta_{ax,tr} = 0$, $\delta_{ax,tr} = 2\lambda$, and $\delta_{ax,tr} = 4\lambda$.

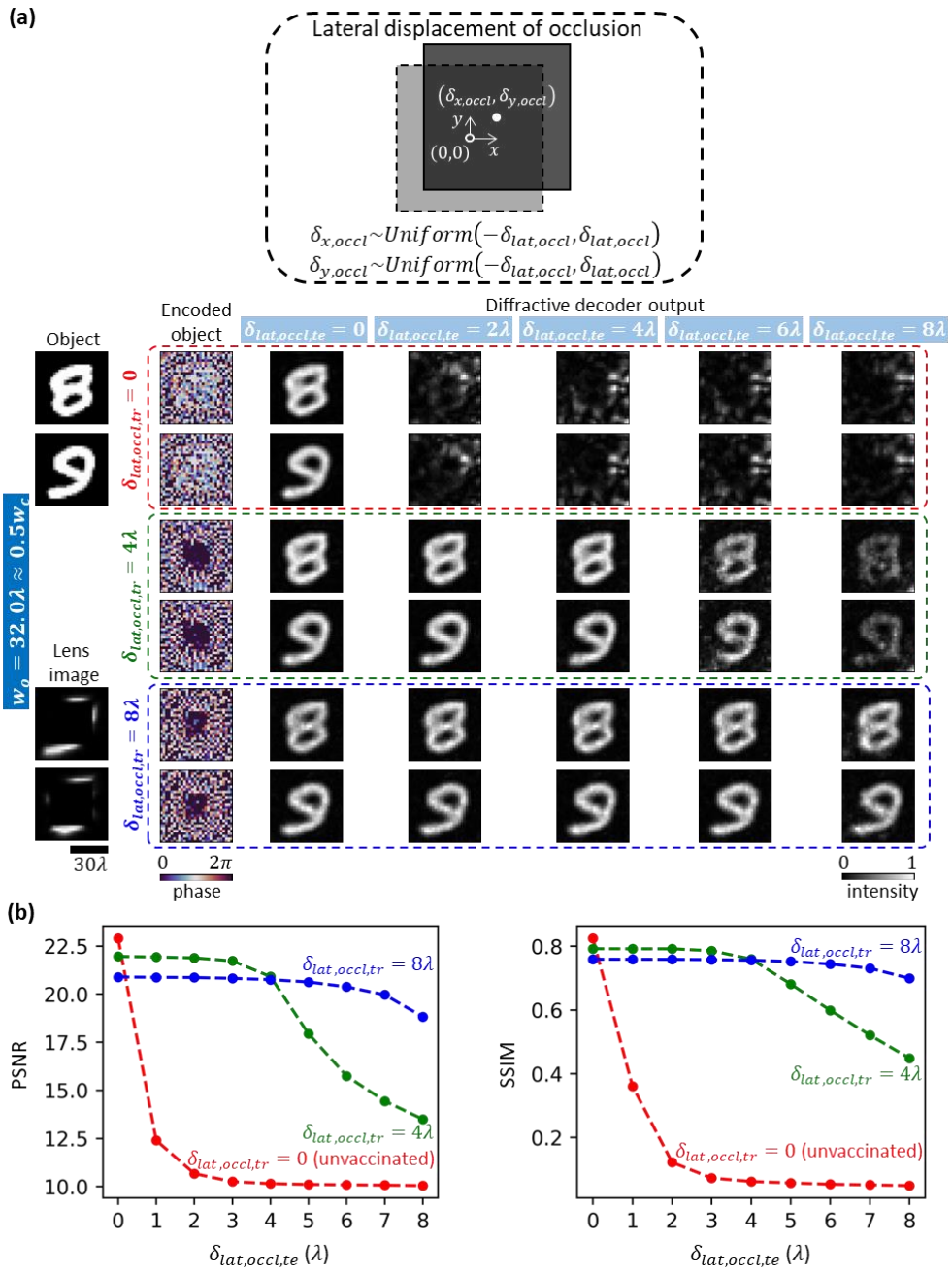


Figure 5.10 Impact of lateral random displacements of the opaque occlusion on the performance of our framework for communication around opaque occlusions. (a) Examples of the diffractive decoder outputs at different levels of lateral random displacements of the occlusion $\delta_{lat,occl,te}$ during the testing for three designs trained with different levels of "vaccination" against such random displacements of the occlusion ($\delta_{lat,occl,tr} = 0$, $\delta_{lat,occl,tr} = 4\lambda$, and $\delta_{lat,occl,tr} = 8\lambda$). (b) Average PSNR and SSIM values of the diffractive decoder outputs

as a function of $\delta_{lat,occl,te}$ for the designs corresponding to $\delta_{lat,occl,tr} = 0$, $\delta_{lat,occl,tr} = 4\lambda$, and $\delta_{lat,occl,tr} = 8\lambda$.

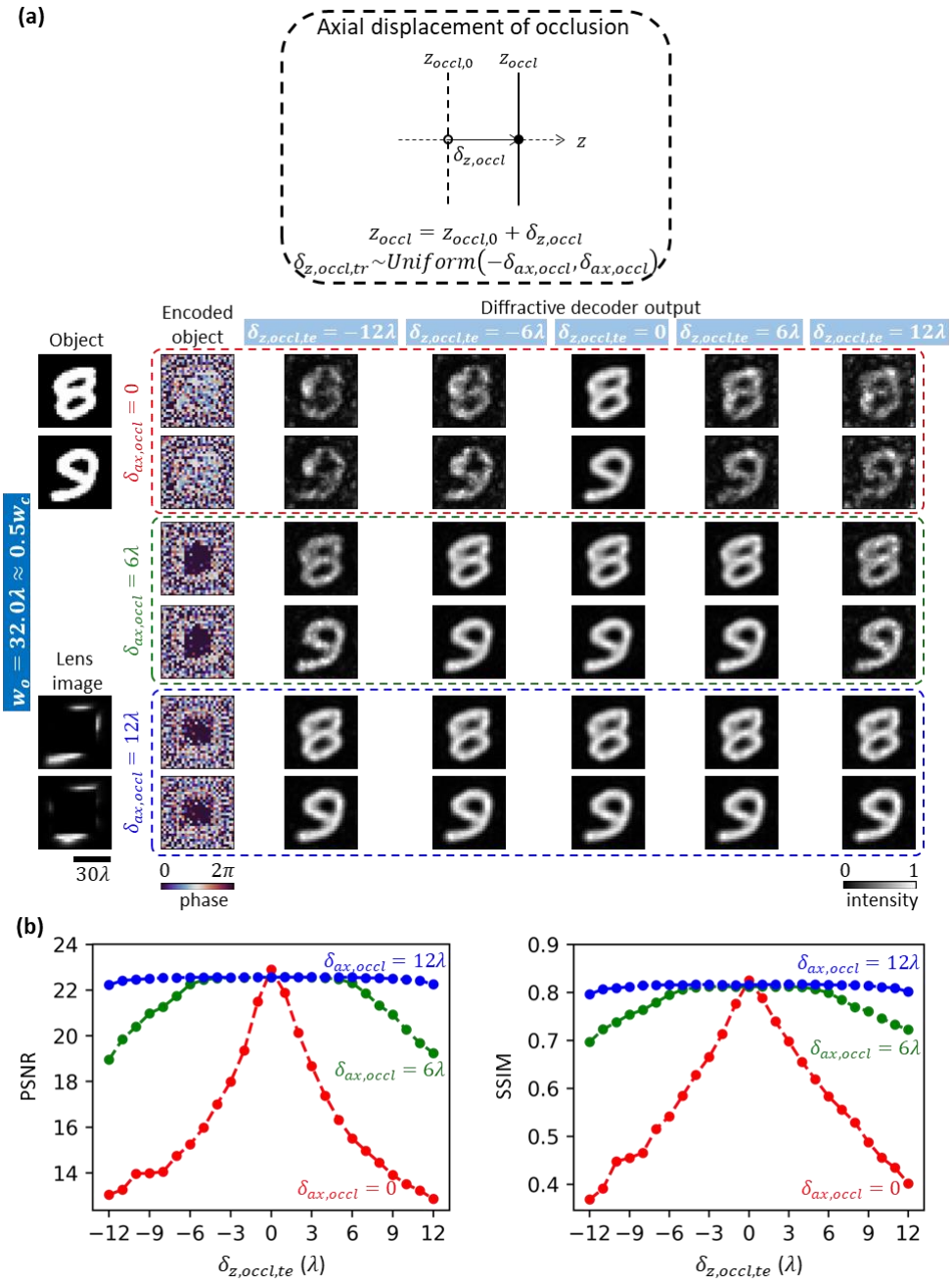


Figure 5.11 Impact of axial random displacements of the opaque occlusion on the performance of our framework for communication around opaque occlusions. (a) Examples of the diffractive decoder outputs at different axial displacements of the occlusion $\delta_{z,occl,te}$ during the testing for three designs trained with different levels of "vaccination" against random axial

displacements of the occlusion ($\delta_{ax,occl} = 0$, $\delta_{ax,occl} = 6\lambda$, and $\delta_{ax,occl} = 12\lambda$). (b) Average PSNR and SSIM values of diffractive decoder outputs as a function of $\delta_{z,occl,te}$, for the designs corresponding to $\delta_{ax,occl} = 0$, $\delta_{ax,occl} = 6\lambda$, and $\delta_{ax,occl} = 12\lambda$.

5.2.4 Output power efficiency

Next, we investigate the power efficiency of the optical communication scheme around opaque occlusions using jointly trained electronic encoder-diffractive decoder pairs. For this analysis, we defined the diffraction efficiency (DE) as the ratio of the optical power at the output FOV to the optical power departing the transmitter aperture. In Fig. 5.12a, we plot the diffraction efficiency of the same designs shown in Fig. 5.4, as a function of the occlusion size. These values are calculated by averaging over 10,000 MNIST test images. These results reveal that the diffraction efficiency decreases monotonically with increasing occlusion width, as expected. Moreover, the diffraction efficiencies are relatively low, i.e., below or around 1%, even for small occlusions. However, this issue of low diffraction efficiency can be addressed in the design stage by adding to the training loss function an additional loss term that penalizes low diffraction efficiency (see Eq. (5.4) the 'Methods' section). Figure 5.12b depicts the improvement of diffraction efficiency resulting from increasing the weight (η) of this additive loss term during the training stage. For example, the $\eta = 0.02$ and $\eta = 0.1$ designs yield an average diffraction efficiency of 27.43% and 52.52%, respectively, while still being able to resolve various features of the target images as shown in Fig. 5.12c. This additive loss weight η therefore provides a powerful mechanism for improving the output diffraction efficiency significantly with a relatively small sacrifice in the image quality as exemplified in Figs. 5.12b and 5.12c.

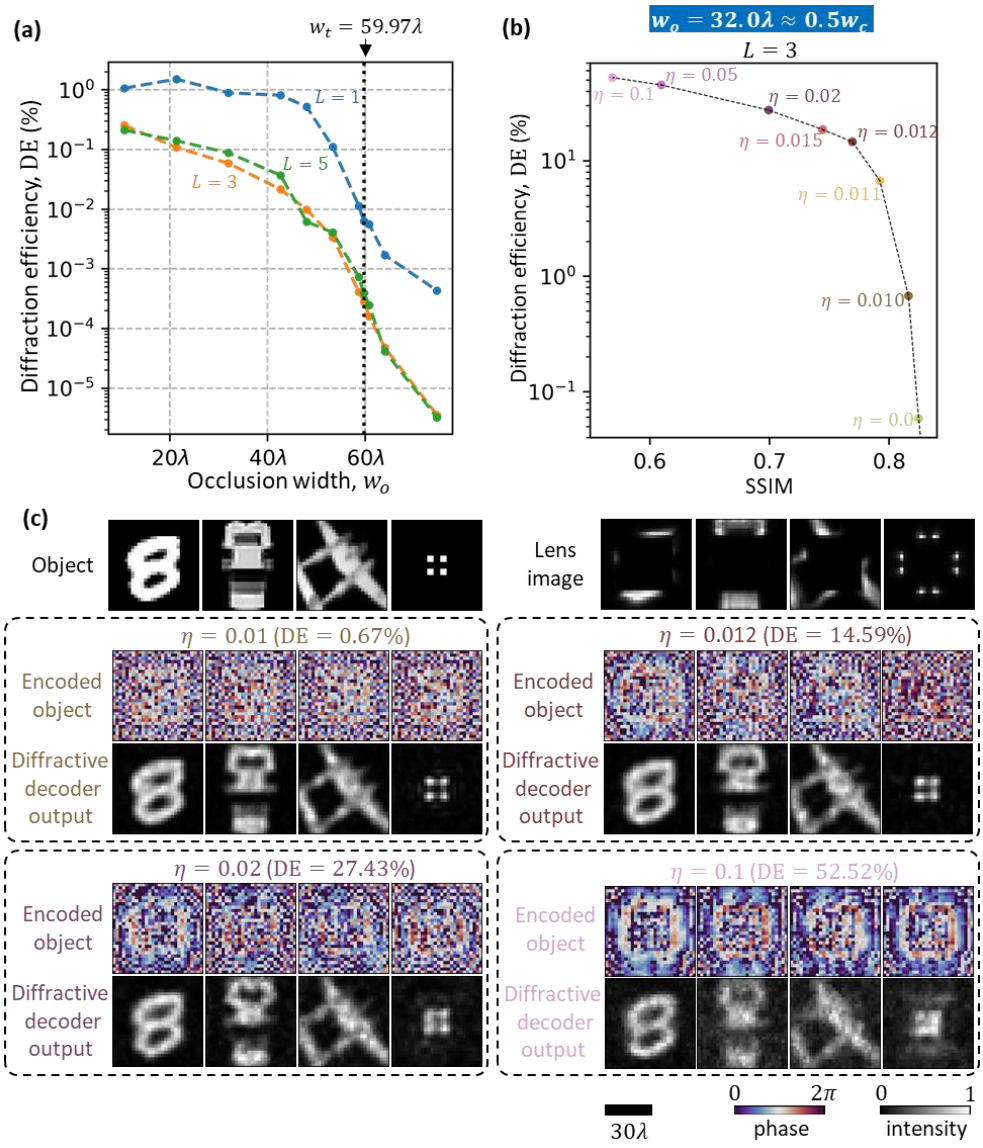


Figure 5.12 Output power efficiency of the electronic encoding-diffractive decoding scheme for optical communication around fully opaque occlusions. (a) Diffraction efficiency (DE) of the same designs shown in Fig. 5.4. (b) The trade-off between DE and SSIM achieved by varying the training hyperparameter η, i.e., the weight of an additive loss term used for penalizing low-efficiency designs. For these designs, $w_o = 32\lambda$ and $L = 3$ were used. The DE and SSIM values are calculated by averaging over 10,000 MNIST test images. (c) The performance of some of the designs shown in (b), trained with different η values.

5.2.5 Occlusion shape

So far, we have considered square-shaped opaque occlusions placed symmetrically around the optical axis. However, our proposed encoder-decoder approach is not limited to square-shaped occlusions and, in fact, can be used to communicate around any arbitrary occlusion shape. In Fig. 5.13, we show the performance comparison of four different trained encoder-decoder pairs for four different occlusion shapes, where the areas of the opaque occlusions were kept approximately the same. We can see that the shape of the occlusion does not have any perceptible effect on the output image quality. We also plot the average SSIM values calculated for these four models over 10,000 MNIST test images (internal generalization) as well as 10,000 Fashion-MNIST test images (external generalization) in Fig. 5.14, which further confirm the success of our approach for different occlusion structures, including randomly shaped occlusions.

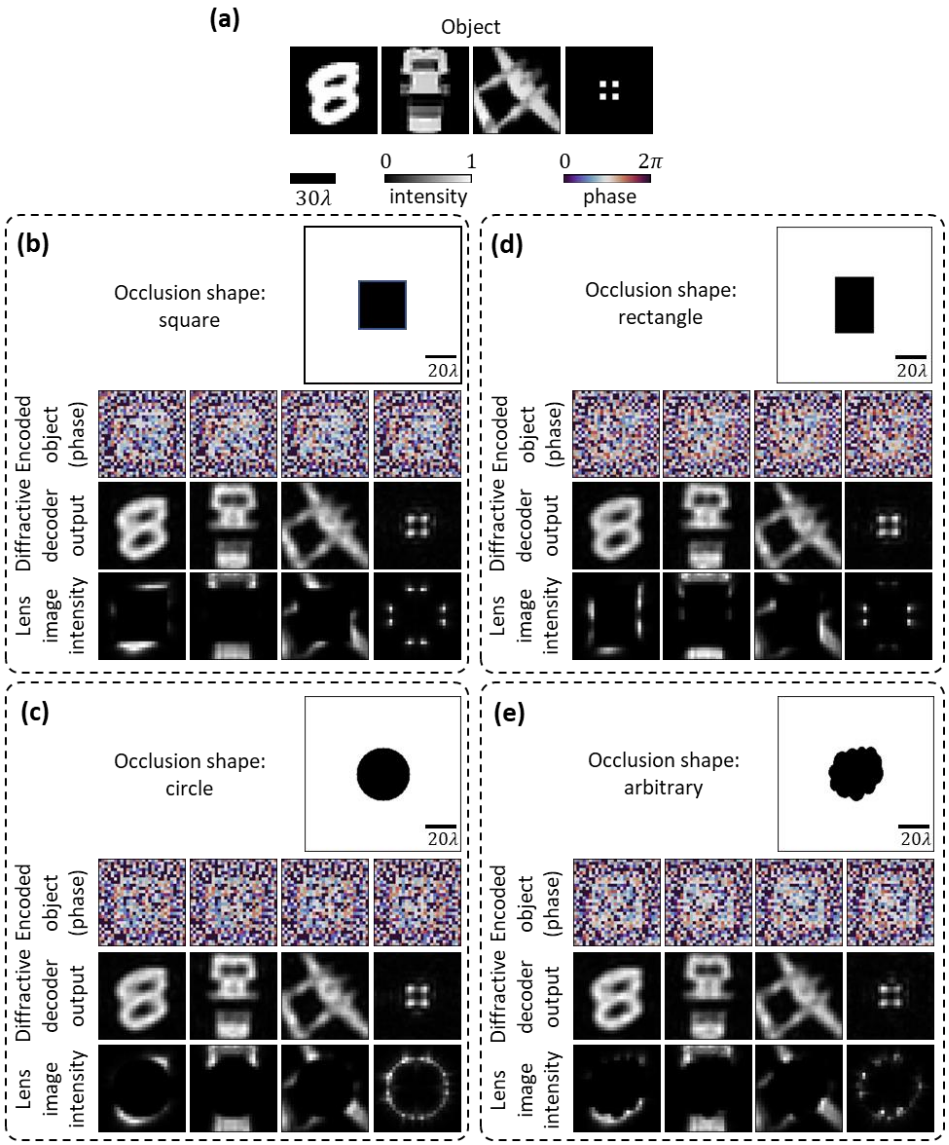


Figure 5.13 Performance of encoder-decoder pairs trained for different opaque occlusion shapes. The performances of four designs trained for different occlusion shapes, i.e., a square, a circle, a rectangle, and an arbitrary shape, are shown. The areas of these fully opaque occlusions are approximately equal.

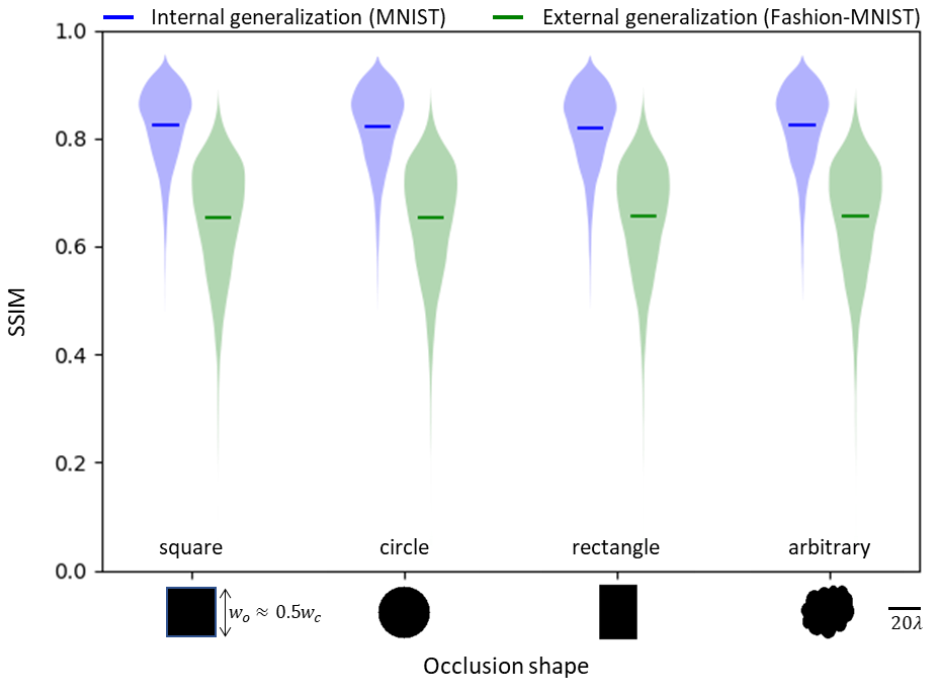


Figure 5.14 Average SSIM values of the diffractive decoder outputs for the four designs of Fig. 5.13. The averages are calculated over 10,000 test images from the MNIST dataset (internal generalization) and 10,000 test images from the Fashion-MNIST dataset (external generalization).

5.2.6 Experimental validation

We experimentally validated the electronic encoding-diffractive decoding scheme for communication around opaque occlusion in the terahertz (THz) part of the spectrum ($\lambda = 0.75\text{mm}$) using a 3D-printed single-layer ($L = 1$) diffractive decoder (see the 'Methods' section for details). We depict the setup used for this experimental validation in Fig. 5.15a. Figures 5.15b and 5.15c show the 3D printed components used to implement the encoded (phase) patterns, the opaque occlusion, and the diffractive decoder layer. Shown in Fig. 5.15c, the width of the transmitter aperture (dashed red square) housing the encoded phase patterns was selected as $w_t \approx 59.73\lambda$,

whereas the width of the opaque occlusion (dashed green square) was $w_o \approx 32.0\lambda$ and the diffractive decoder layer (dashed blue square) width was selected as $w_l \approx 106.67\lambda$. The axial distances between the encoded object and the occlusion, between the occlusion and the diffractive layer, and the diffractive layer and the output FOV were $\sim 13.33\lambda$, $\sim 106.67\lambda$, and $\sim 40\lambda$, respectively. In Fig. 5.15d, we show the input objects/messages, the simulated lens images, and the simulated and experimental diffractive decoder output images for ten different handwritten digits randomly chosen from the test dataset. Our experimental results reveal that the CNN-based phase encoding followed by diffractive decoding resulted in successful communication of the intended objects/messages around the opaque occlusion (see the bottom row of Fig. 5.15d).

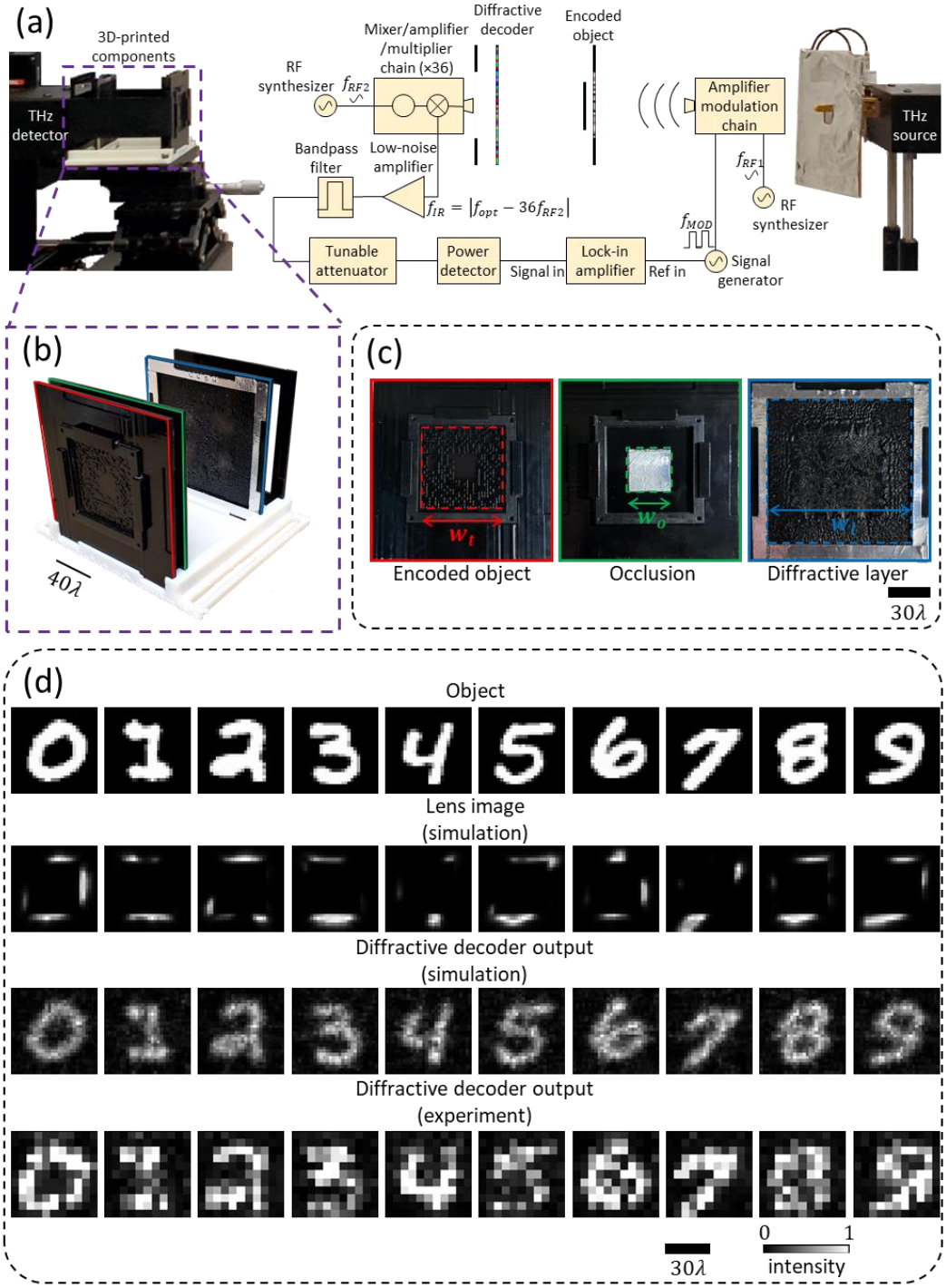


Figure 5.15 Experimental results with an $L = 1$ design for an occlusion width of $w_o = 32\lambda \approx 0.5w_c$ operating at a wavelength of $\lambda = 0.75$mm. (a) The terahertz setup comprising the source and the detector, together with the 3D-printed components used as the encoded phase objects, the occlusion, and the diffractive layer. (b) Assembly of the encoded phase objects, the occlusion, the diffractive layer, and the output aperture using a 3D-printed holder. (c) The encoded phase object

(one example), the occlusion, and the diffractive layer are shown separately, housed inside the supporting frames. (d) Experimental diffractive decoder outputs (bottom row) for ten handwritten digit objects (top row), together with the corresponding simulated lens images (second row) and the diffractive decoder outputs (third row).

5.2.7 Dynamic occlusions

So far, we have analyzed our framework for static occlusions that do not change over time. Here, we demonstrate the adaptability of our framework to situations where the occlusion shape/size can randomly change over time without our knowledge. In other words, we design encoder-decoder pairs which can communicate around opaque occlusions of varying unknown shapes, without any change to the encoder or the decoder. In Supplementary Video 5.1, we present our analysis depicting the performance of an $L = 3$ design as the shape of the opaque occlusion randomly changes within the occlusion plane. For training and testing of this design, the occlusion shape was parameterized by $r_{max} \approx 17.6\lambda$, where the radii of the partial circles comprising the occlusions were randomly drawn from the distribution $Uniform(9r_{max}/11\,, r_{max})$; see the Methods section for details. As shown in Supplementary Video 5.1, the same electronic encoder and diffractive decoder successfully communicate the desired images of the objects even if the occlusion changes randomly. In Supplementary Videos 5.2 and 5.3, we show two additional $L = 3$ designs with $r_{max} \approx 29.3\lambda$ and $r_{max} \approx 41.1\lambda$, respectively, showcasing a decent performance despite a significant loss of information caused by the different opaque occlusions of varying random and unknown shapes.

We also experimentally demonstrated robust communication around opaque occlusions of varying, random shapes ($r_{max} \approx 18.1\lambda$) at $\lambda = 0.75\text{mm}$, using a fixed encoder-decoder pair with $L = 1$. The results of these experiments are presented in Fig. 5.16, where all the desired objects of

interest were successfully communicated around two different occlusions by the same encoder-decoder design comprising a single-diffractive layer.

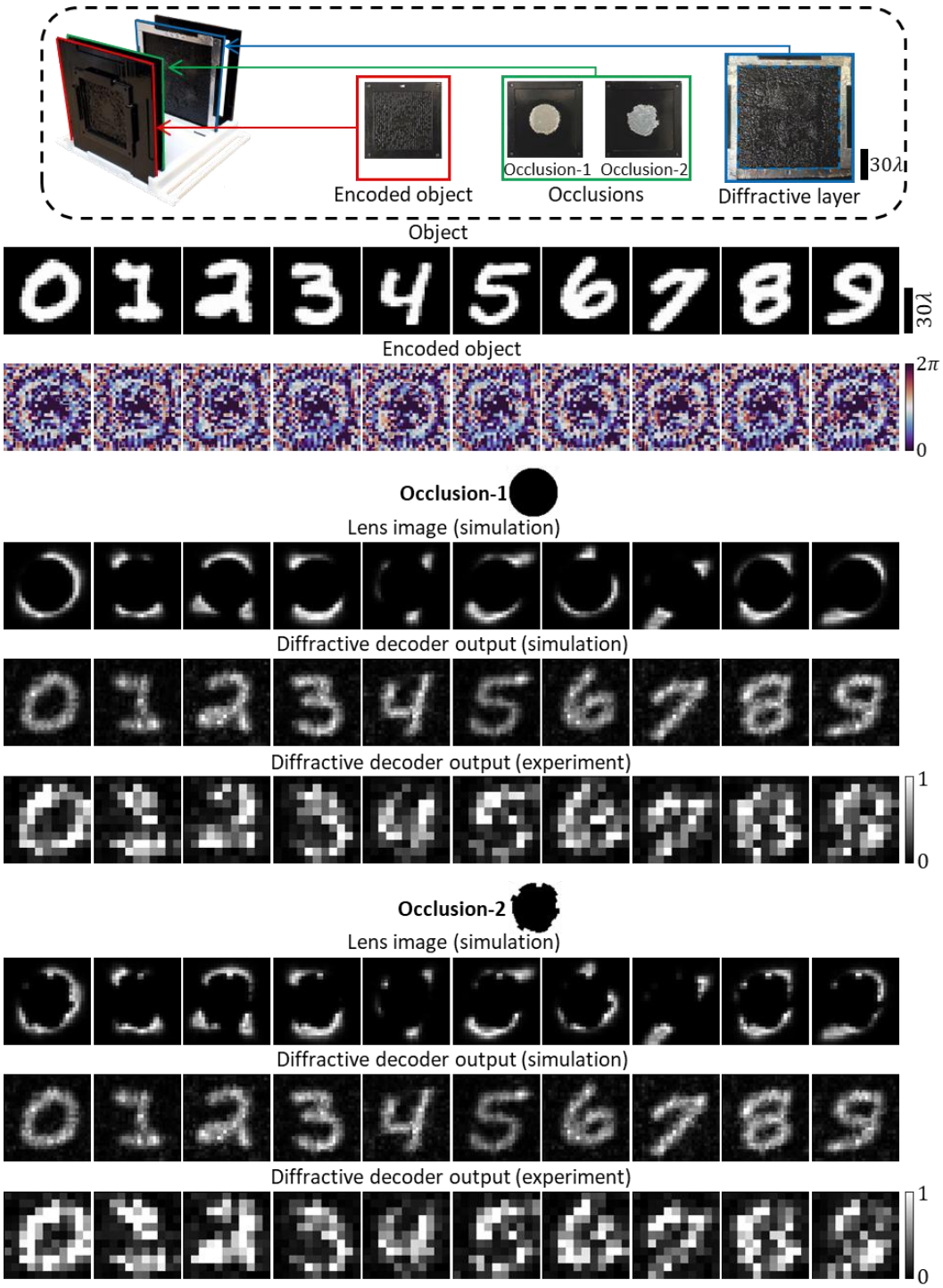


Figure 5.16 10 Experimental results for communication around randomly changing opaque occlusions using the same electronic encoder and diffractive decoder; $\lambda = 0.75$mm. The same electronically encoded objects and the 3D-printed single-layer diffractive decoder were used for communication around two different occlusions; $r_{max} = 18.1\lambda$.

5.3 Discussion

We modeled the scattering of light from the opaque occlusions with 2D cross-sections using the angular spectrum approach, covering a numerical aperture (NA) of 1.0 in air (see the Model subsection in the Methods). In general, any arbitrary fully opaque occlusion volume can be modeled, to a first-order approximation, as a string of scattering edges, forming a 3D loop of secondary waves. In our forward model, these 3D strings of scattering edges for any arbitrary opaque occlusion considered in our analysis were located at a common axial plane for ease of computing, and each point that makes up the scattering edge function communicated with the receiver aperture using traveling waves covering all the modes of free-space wave propagation (NA=1). Using the Huygens–Fresnel principle, one can also extend the joint training of our encoder-decoder pairs to cover non-planar 3D strings of scattering edges representing any arbitrary occlusion volume; such cases, however, would take longer to numerically model and train using deep learning, especially if the axial coverage of the 3D string function that defines the scattering edges of a fully opaque volume is relatively large.

Our optical communication scheme using CNN-based encoding and diffractive all-optical decoding would be useful for the optical communication of information around opaque occlusions caused by existing or evolving structures. In case such occlusions grow in size as a function of time, the same diffractive decoder that is deployed as part of our communication link can still be used with only an update of the digital encoder CNN. To showcase this, in Fig. 5.17, we illustrate an encoder-decoder design with $L = 3$ that was originally trained with an occlusion size of $w_o = 32.0\lambda$ (blue boxes), successfully communicating the input messages between the CNN-based phase transmitter aperture and the output FOV of the diffractive decoder when the occlusion size

remains the same, i.e., $w_o = 32.0\lambda$ (dashed blue box). The same figure also illustrates the failure of this encoder-decode pair once the size of the opaque occlusion grows to $w_o = 40.0\lambda$ (dotted blue box); this failure due to the (unexpectedly) increased occlusion size can be repaired without changing the deployed diffractive decoder layers by just retraining the CNN encoder part; see Fig. 5.17, dashed green box.

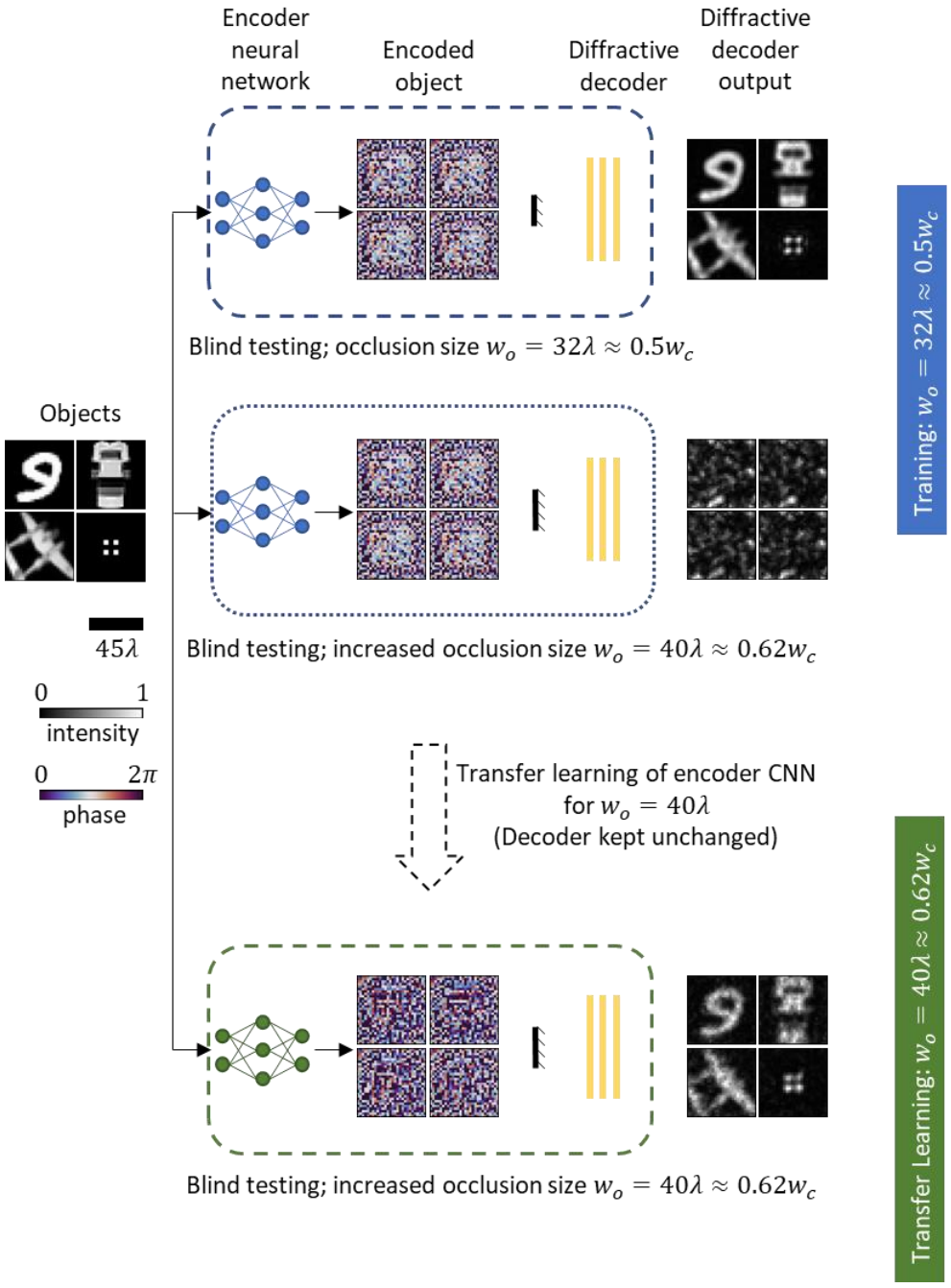


Figure 5.17 Transfer learning of the CNN encoder at the transmitter, while the diffractive decoder at the receiver remains unchanged. This transfer learning is aimed at successful communication in case of an increase/change in the size of the opaque occlusion, obstructing the transmitter field-of-view.

The speed of optical communication through our encoder-decoder pair would be limited by the rate at which the encoded phase patterns (CNN outputs) can be refreshed or by the speed of the output detector-array, whichever is smaller. The transmission and the decoding processes of the desired optical information/message occur at the speed of light propagation through thin diffractive layers and do not consume any external power (except for the illumination light). Therefore, the main power consuming steps in our architecture are the CNN inference, the transmitter of the encoded phase patterns and the detector-array operation.

The communication around occlusions using our scheme works even when the occlusion width is larger than the width of the transmitter aperture since it utilizes CNN-based phase encoding of information to effectively exploit the scattering from the edges of the occlusions. Surprisingly, as the occlusion width surpasses the transmitter aperture width (w_t), the performance of $L = 1$ and $L = 3$ designs slightly improved, as was seen in Fig. 5.4. This relative improvement might be explained by a switch in the mode of operation of our encoder-decoder pair. When the opaque occlusions are smaller than the transmitter aperture, the pixels at the edges of the transmitter can communicate directly to the receiver aperture and therefore, they dominate the power balance. In this operation regime, as the occlusion size gets larger, the effective number of pixels at the transmitter aperture that directly communicates with the receiver/decoder gets smaller, causing a decline in the performance of the diffractive decoder. However, when the occlusion becomes larger than the transmitter aperture, none of the input pixels can dominate the power balance at the receiver end by communicating with it directly; instead, all the pixels of the encoder plane are forced to *indirectly* contribute to the receiver aperture through the edge scattering of the occlusion. This causes the performance to get better for occlusions larger than the transmitter aperture since effectively more pixels of the encoder plane can contribute to the receiver aperture without a major

power imbalance among these secondary wave-based contributions (through edge scattering). This turnaround in performance (i.e., the switching behavior between these two modes of operation) is not observed when the diffractive decoder has a deeper architecture (e.g., $L = 5$) since deeper decoders can effectively balance the ballistic photons that are transmitted from the edge pixels; consequently, edge-pixels of the transmitter aperture do not dominate the output signals even when they can directly 'see' the receiver aperture since multiple layers of a deeper diffractive decoder act as a universal mode processor [32,33,47,56].

We would like to emphasize that the presented framework is also applicable for communication over larger distances (d_{tl}) between the transmitter and the receiver apertures. In Fig. 5.18, we present results for communication over much larger axial distances of $d_{tl} = 600\lambda$ and $d_{tl} = 1200\lambda$ for two different occlusion widths. The diffractive decoder outputs reveal successful communication around the opaque occlusions for these larger distances; note, however, that as the axial distance gets even larger, the optical resolution of the decoder output will deteriorate because of the reduced NA of the communication system.

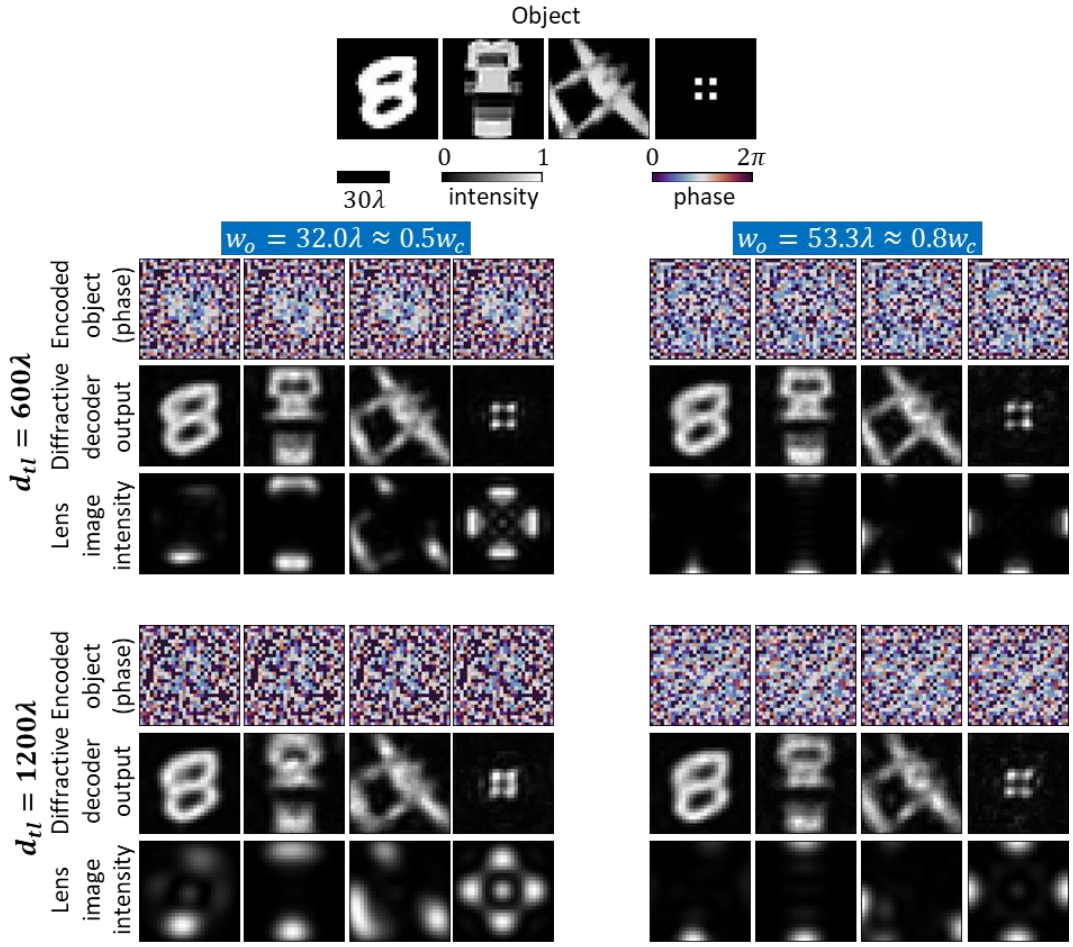


Figure 5.18 Performance of the electronic encoder and diffractive decoder framework for communication around opaque occlusions for large values of the axial distance d_{tl} between the transmitter and the receiver apertures. The occlusion is assumed to be $d_{to} = d_{tl}/9$ distance away from the transmitter aperture.

The success of the simpler decoder designs with $L = 1$ layer, as shown in Figs. 5.3, 5.4, 5.5, 5.6, 5.15 and 5.16, begs the question of whether such an optical communication around opaque occlusions is also feasible with electronic encoding only, i.e., without diffractive decoding (see the 'Free space-only decoder' listed in Fig. 5.19). To address this question, we trained two encoder-only designs, for $w_o = 32.0\lambda$ and $w_o = 53.3\lambda$, and compared their performance against $L = 1$ designs in Fig. 5.20. The encoder-only architecture barely succeeds for $w_o = 32.0\lambda$ and fails

drastically for $w_o = 53.3\lambda$, whereas $L = 1$ designs provide significantly better performance. In the same figure, we also evaluated two additional decoding approaches that do not use any trained diffractive decoders. In one of these approaches, we used a random (untrained) diffusive layer as the all-optical decoder, the phase profile of which was precisely "known" to the electronic encoder. We can see that the untrained diffusive layer fails to perform any meaningful decoding despite the presence of a trained electronic encoder. In the other approach that we used for comparison, we utilized a lens as the all-optical decoder, configured to perform the Fourier transform of the transmitter aperture. Similar to the 'Free space-only decoder', this Fourier lens-based decoder was also not as successful as our presented approach; see Fig. 5.20. All these results and comparative analyses demonstrate the importance of complementing electronic encoding with diffractive decoding for effective communication around opaque occlusions.

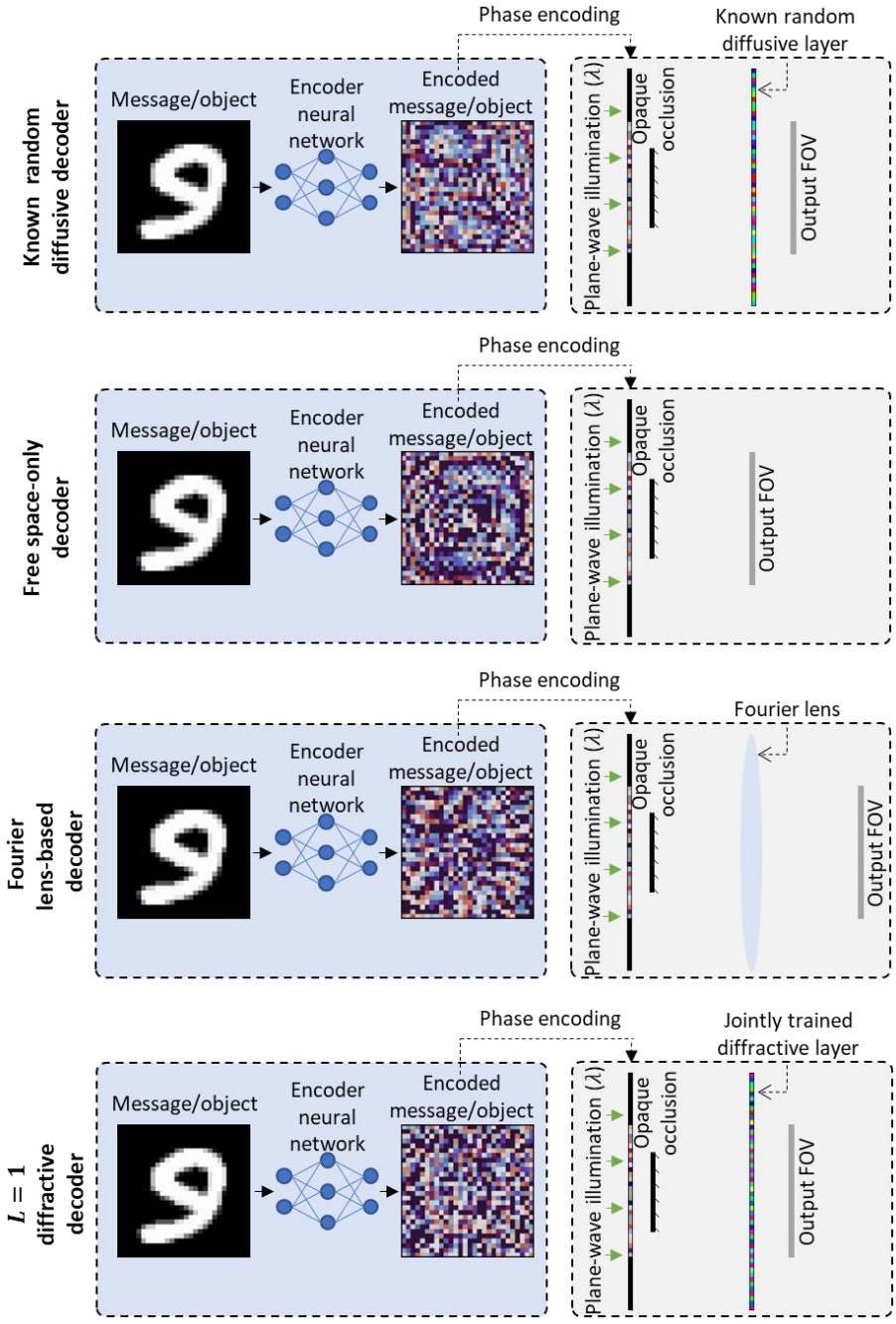


Figure 5.19 Depiction of alternative schemes to diffractive all-optical decoding for communication around opaque occlusions. See Fig. 5.20 for the comparative performance analysis of these different schemes for communication around opaque occlusions.

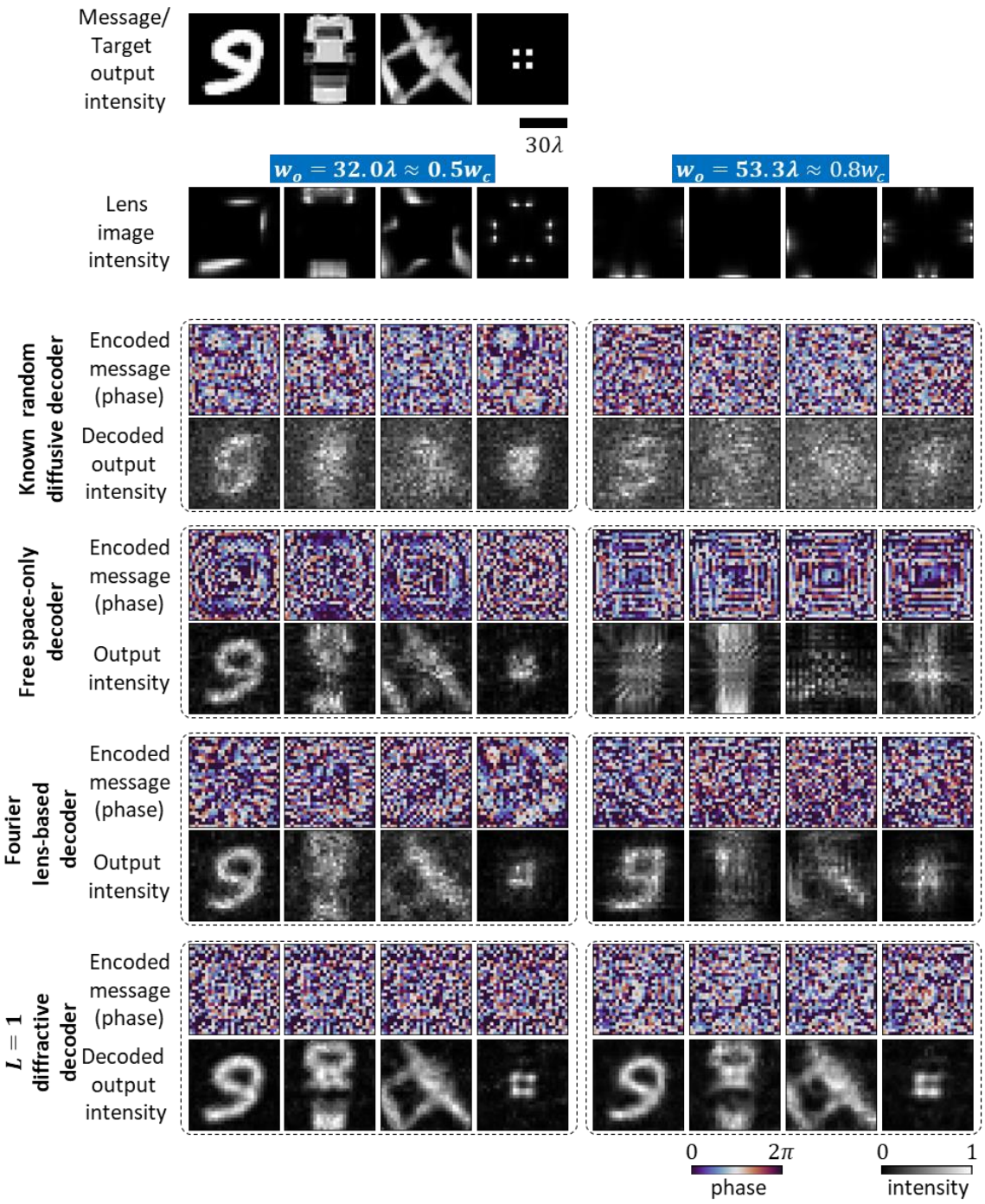


Figure 5.20 Comparative performance analysis of different schemes illustrated in Fig. 5.19. The performances of a known random diffusive decoder, free space-only decoder, and Fourier lens-based decoder (see Fig. 5.19) are compared against the performance of $L = 1$ layer diffractive decoder for two different sizes of opaque occlusion (w_o).

5.4 Methods

5.4.1 Model

In our model, the message/object m that is to be transmitted is fed to a CNN, which yields a phase-encoded representation ψ of the message. The message is assumed to be in the form of an $N_{in} \times N_{in} = 28 \times 28$ pixel image. The coded phase ψ is assumed to have dimension $N_{out} \times N_{out} = 28 \times 28$. The $N_{out} \times N_{out}$ phase elements are distributed over the transmitter aperture of area $w_t \times w_t$, where $w_t \approx 59.73\lambda$ and λ is the illumination wavelength. The lateral width of each phase element/pixel is therefore $w_t/N_{out} \approx 2.12\lambda$. The phase-encoded input wave $\exp(j\psi)$ propagates a distance $d_{to} \approx 13.33\lambda$ to the plane of the opaque occlusion, where its amplitude is modulated by the occlusion function $o(x, y)$ such that:

$$o(x,y) = \begin{cases} 0, & |x| < \frac{w_o}{2}, |y| < \frac{w_o}{2} \\ 1, & \text{otherwise} \end{cases}$$

The encoded wave, after being obstructed and scattered by the occlusion, travels to the receiver through free space. At the receiver, the diffractive decoder all-optically processes and decodes the incoming wave to produce an all-optical reconstruction $\hat{m}'$ of the original message m at its output FOV. We assume the receiver aperture, which coincides with the first layer of the diffractive decoder, to be located at an axial distance of $d_{ol} \approx 106.67\lambda$ away from the plane of the occlusion. The effective size of the independent diffractive features of each transmissive layer is assumed to be $0.53\lambda \times 0.53\lambda$, and each of the L layers comprises 200×200 such diffractive features, resulting in a lateral width of $w_l \approx 106.67\lambda$ for the diffractive layers. The layer-to-layer separation is assumed to be $d_{ll} = 40\lambda$. The output FOV of the diffractive decoder is assumed to be 40λ away from the last diffractive layer and extend over an area $w_d \times w_d$, where $w_d \approx 59.73\lambda$.

The diffractive decoding at the receiver involves consecutive modulation of the received wave by the L diffractive layers, each followed by propagation through the free space. The modulation of the incident optical wave on a diffractive layer is assumed to be realized passively by its height variations. The complex transmittance $\tilde{t}(x,y)$ of a passive diffractive layer is related to its height $h(x,y)$ according to:

$$\tilde{t} = \exp\left(j\frac{2\pi}{\lambda}(n+jk-1)h\right) = \exp\left(-\frac{2\pi k}{\lambda}h\right)\exp\left(j\frac{2\pi}{\lambda}(n-1)h\right) = a\exp(j\varphi) \qquad (5.1)$$

where n and k are the refractive index and the extinction coefficient, respectively, of the diffractive layer material at λ; $a = \exp\left(-\frac{2\pi k}{\lambda}h\right)$ and $\varphi = \frac{2\pi}{\lambda}(n-1)h$ are the amplitude and the phase of the complex field transmittance, respectively. For our numerical simulations, we assume the diffractive layers to be lossless, i.e., $k = 0$, $a = 1$, unless stated otherwise.

The propagation of the optical fields through free space is modeled using the angular spectrum method [15,38], according to which the transformation of an optical field $u(x,y)$ after propagation by an axial distance d can be computed as follows:

$$u(x,y;z = z_0 + d) = \mathcal{F}^{-1}\{\mathcal{F}\{u(x,y;z=z_0)\} \times H(f_x,f_y;d)\} \qquad (5.2)$$

where $\mathcal{F}$ ($\mathcal{F}^{-1}$) is the two-dimensional Fourier (Inverse Fourier) transform operator and $H(f_x,f_y;d)$ is the free-space transfer function for propagation by an axial distance d defined as follows:

$$H(f_x,f_y;d) = \begin{cases} \exp\left(j\frac{2\pi}{\lambda}d\sqrt{1-(\lambda f_x)^2-(\lambda f_y)^2}\right), & f_x^2+f_y^2 < 1/\lambda^2 \\ 0, & \text{otherwise} \end{cases} \qquad (5.3)$$

In our numerical analyses, the optical fields were sampled at an interval of $\delta \approx 0.53\lambda$ along both x and y directions and the Fourier (Inverse Fourier) transforms were implemented using the Fast Fourier Transform (FFT) algorithm.

For the lens-based imaging simulations reported in this work, the plane wave illumination was assumed to be amplitude modulated by the object placed at the transmitter aperture, and the (thin) lens is assumed to be placed at the same plane as the plane of the first diffractive layer in the encoding-decoding scheme, with the diameter of the lens aperture equal to the width of the diffractive layer, i.e., $w_l \approx 106.67\lambda$.

5.4.2 Training

The diffractive decoder features were parameterized using the latent variables h_{latent} such that the feature heights h are related to h_{latent} according to $h = h_{max} \times \frac{1+\sin(h_{\text{latent}})}{2}$, where h_{max} is a hyperparameter denoting the maximum height variation. We used $h_{max} = \frac{\lambda}{n-1}$ so that the corresponding maximum phase modulation was $\varphi_{max} = 2\pi$.

The parameters of the encoder CNN and the diffractive decoder phase features were optimized by minimizing the loss function:

$$\mathcal{L} = \mathcal{L}_{\text{pixel}} + \eta \mathcal{L}_{\text{DE}} \tag{5.4}$$

where $\mathcal{L}_{\text{pixel}}$ is the mean squared error (MSE) between the pixels of the desired message m and the pixels of the (scaled) decoded optical intensity $\hat{m} = \sigma \hat{m}'$, i.e.,

$$\mathcal{L}_{\text{pixel}} = \frac{1}{N_{in} \times N_{in}} \sum_{j=1}^{N_{in}} \sum_{i=1}^{N_{in}} \left(m_{ij} - \hat{m}_{ij}\right)^2 \tag{5.5}$$

The scaling factor σ is defined as:

$$\sigma = \frac{\sum_{j=1}^{N_{in}} \sum_{i=1}^{N_{in}} m_{ij} \widehat{m}'_{ij}}{\sum_{j=1}^{N_{in}} \sum_{i=1}^{N_{in}} \left(\widehat{m}'_{ij}\right)^2} \tag{5.6}$$

The additive loss term $\mathcal{L}_{\mathrm{DE}} = 1 - \mathrm{DE}$, scaled by the weight η, is used to penalize against low diffraction efficiency models. DE is the diffraction efficiency, calculated as:

$$\mathrm{DE} = \frac{1}{N_{in} \times N_{in}} \sum_{j=1}^{N_{in}} \sum_{i=1}^{N_{in}} \widehat{m}'_{ij} \tag{5.7}$$

The training data comprised 110,000 examples: 55,000 images from the MNIST training set and 55,000 custom-prepared images; see Fig. 5.21 for examples. The remaining 5,000 images of the 60,000 MNIST training images, together with 5,000 additional custom-prepared images, i.e., a total of 10,000 images, were used for validation. After the completion of each epoch, the average loss over the validation images was computed, and the model state corresponding to the smallest validation loss was selected as the ultimate design.

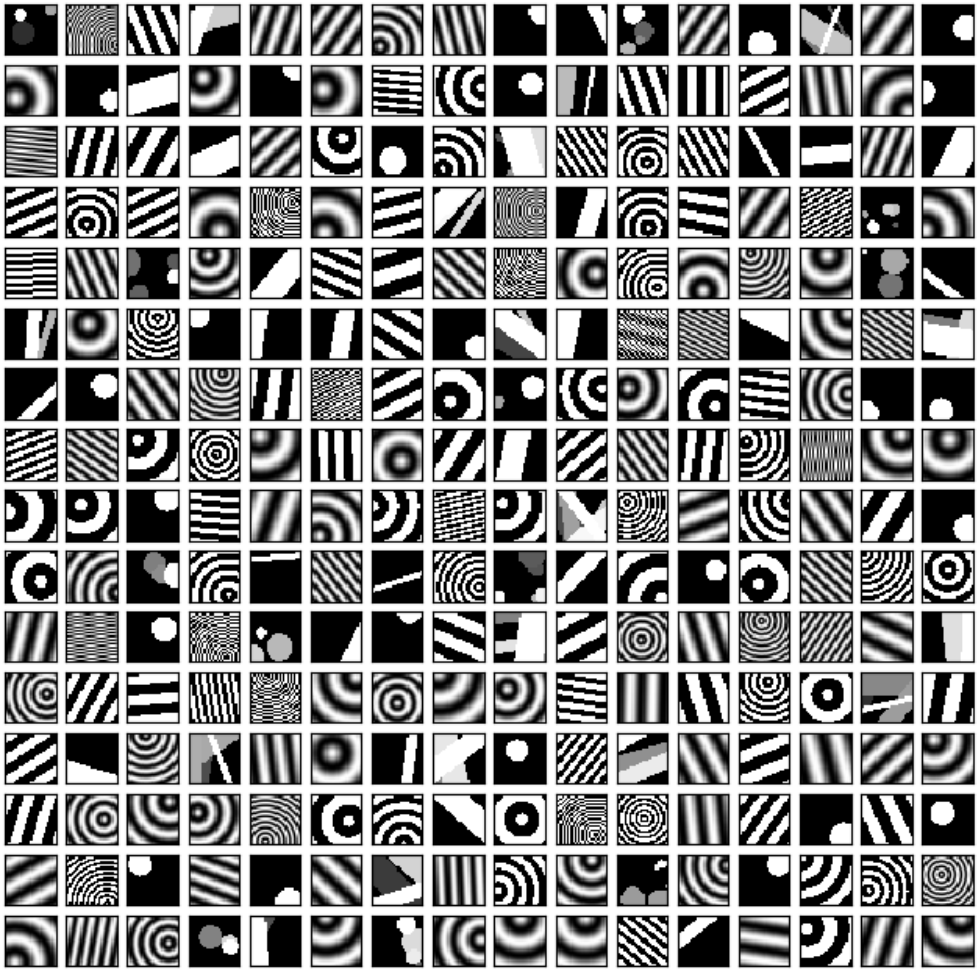


Figure 5.21 Examples of the custom-prepared training images. These images were used to augment the MNIST handwritten digits training set for better generalization.

The electronic encoder-diffractive decoder models were implemented in TensorFlow [128] version 2.4 using the Python programming language and trained on a machine with Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz and NVIDIA GeForce GTX 1080 Ti GPU. The loss function was minimized using the Adam [102] optimizer for 50 epochs with a batch size of 4. The learning rate was initially 1e-3 and it decreased by a factor of 0.99 every 10,000 optimization steps. For the other parameters of the Adam optimizer, the default TensorFlow settings were used. The training

time varied with the model size; for example, training a model with an $L = 3$ diffractive decoder took ~8 hours.

The native TensorFlow implementations of PSNR and SSIM were used for computing these image comparison metrics between the message m and the scaled diffractive decoder output $\widehat{m}$.

5.4.3 Experimental design

In our experiments, the wavelength of operation was $\lambda = 0.75\text{mm}$. We used a single-layer diffractive decoder, i.e., $L = 1$, with $N = 200^2$ independent features and the width of each feature was $\sim 0.53\lambda \approx 0.40\text{mm}$, resulting in an $\sim 80\text{mm} \times 80\text{mm}$ diffractive layer. The width of the transmitter aperture accommodating the encoded phase messages was $w_t \approx 59.73\lambda \approx 44.8\text{mm}$, same as the width of the output FOV w_d. The occlusion width was $w_o \approx 32\lambda \approx 24\text{mm}$. The distance from the transmitter aperture to the occlusion plane was $d_{to} \approx 13.33\lambda \approx 10\text{mm}$, while the diffractive layer was $d_{ol} \approx 106.67\lambda \approx 80\text{ mm}$ away from the occlusion plane. The output FOV was $40\lambda \approx 30\text{mm}$ away from the diffractive layer.

The diffractive layers and the phase-encoded messages (CNN outputs) were fabricated using a 3D printer (Objet30 Pro, Stratasys Ltd). Similar to the implementation of the diffractive layer phase, the phase-encoded messages were implemented by height variations according to $h_o = \psi \frac{\lambda}{2\pi(n-1)}$. The height variations were applied on top of a uniform base thickness of 0.2mm, used for mechanical support. The occlusion was realized by pasting aluminum on a 3D-printed substrate (see Fig. 5.15). The measured complex refractive index $n + jk$ of the 3D-printing material at $\lambda = 0.75\text{mm}$ was $1.6518 + j0.0612$.

While training the experimental model, the weight η of the diffraction efficiency-related loss term was set to be zero. To make the experimental design robust against misalignments, we incorporated random lateral and axial misalignments of the encoded objects, the occlusion and the diffractive layer into the optical forward model during its training [157]. The random misalignments were modeled using the uniformly distributed random variables $\delta_x \sim Uniform(-0.5\lambda, 0.5\lambda)$, $\delta_y \sim Uniform(-0.5\lambda, 0.5\lambda)$ and $\delta_z \sim Uniform(-2\lambda, 2\lambda)$ representing the displacements of the encoded objects, the occlusion and the diffractive layer along x, y and z directions, respectively, from their nominal positions.

5.4.4 Terahertz experimental setup

A WR2.2 modular amplifier/multiplier chain (AMC) in conjunction with a compatible diagonal horn antenna from Virginia Diodes Inc. was used to generate a continuous-wave (CW) radiation at 0.4 THz, by multiplying a 10 dBm RF input signal at f_{RF1} = 11.1111 GHz 36 times. To resolve low-noise output data through lock-in detection, the AMC output was modulated at a rate of f_{MOD} = 1 kHz. The exit aperture of the horn antenna was positioned ~60 cm away from the input (encoded object) plane of the 3D-printed diffractive decoder for the incident THz wavefront to be approximately planar. A single-pixel Mixer/AMC, also from Virginia Diodes Inc., was used to detect the diffracted THz radiation at the output plane. To down-convert the detected signal to 1 GHz, a 10dBm local oscillator signal at f_{RF1} = 11.0833 GHz was fed to the detector. The detector was placed on an X-Y positioning stage consisting of two linear motorized stages from Thorlabs NRT100, and the output FOV was scanned using a 0.5 × 0.1 mm detector with a scanning interval of 2 mm. The down-converted signal was amplified, using cascaded low-noise amplifiers from Mini-Circuits ZRL-1150-LN+, by 40 dB and passed through a 1 GHz (+/-10 MHz) bandpass filter

(KL Electronics 3C40-1000/T10-O/O) to filter out the noise from unwanted frequency bands. The filtered signal was attenuated by a tunable attenuator (HP 8495B) for linear calibration and then detected by a low-noise power detector (Mini-Circuits ZX47-60). The output voltage signal was read out using a lock-in amplifier (Stanford Research SR830), where the f_{MOD} = 1kHz modulation signal served as the reference signal. The lock-in amplifier readings were converted to a linear scale according to the calibration results. To enhance the signal-to-noise ratio (SNR), a 2×2 binning was applied to the THz measurements. We also digitally enhanced the contrast of the measurements by saturating the top 1% and the bottom 1% of the pixel values using the built-in MATLAB function *imadjust* and mapping the resulting image to a dynamic range between 0 and 1.

5.4.5 Dynamic occlusion modeling

The dynamic occlusions were modeled as the union of 24 concentric and disjoint partial circles extending equal angles ($360°/24 = 15°$) at the center. The radii of these partial circles were randomly sampled independently from the distribution $Uniform(9r_{max}/11, r_{max})$, where the parameter r_{max} is a measure of the level of the opaque occlusion. At each training iteration, the occlusion was randomly sampled from the distribution defined by r_{max}. For the experimental demonstration reported in Fig. 5.16, we used $r_{max} \approx 18.1\lambda$.

Chapter 6 Conclusion

Deep learning has revolutionized the landscape of optical design and engineering beyond the scope of traditional design methods. Enabled by deep learning, diffractive networks present a platform for low-power and low-latency optical information processing through light-matter interaction within a thin volume. In this dissertation, I presented novel strategies and applications pertaining to diffractive networks, which bear tremendous potential for unveiling new opportunities and capabilities with regard to optical information processing.

This dissertation revolves around three major application areas of diffractive networks: statistical inference, optical information processing and optical information transfer. In chapters 2 and 3, advancement of statistical inference performance of diffractive networks is addressed, expressly for objects beyond simple handwritten digits and fashion products. Two novel paradigms, namely, ensemble learning and time-lapse image classification, are introduced, which significantly improves the inference of diffractive network on challenging datasets such as grayscale CIFAR-10 images. Chapter 4 discusses optical processing of information stored in holograms with diffractive networks, showcasing all-optical twin-image removal from inline hologram reconstructions. This enables bypassing the power-hungry and high-latency digital processing for phase recovery and inline hologram reconstruction. Chapter 5 discusses the adoption of diffractive network within a hybrid encoding-decoding framework for the transfer of optical information around arbitrarily shaped zero-transmittance occlusions.

In conclusion, this dissertation has greatly contributed to the evolution of diffractive networks as a versatile and reliable tool in optical information processing. By integrating deep learning, these networks unlock numerous opportunities and challenges for future research, with potential

applications in various industries. As deep learning and optical computing continue to advance, diffractive networks are poised to play a pivotal role in the transformation of information processing paradigm.

References

1. P. R. Wiecha, A. Arbouet, C. Girard, and O. L. Muskens, "Deep learning in nano-photonics: inverse design and beyond," Photon. Res., PRJ **9**, B182–B200 (2021).
2. W. Ma, Z. Liu, Z. A. Kudyshev, A. Boltasseva, W. Cai, and Y. Liu, "Deep learning for the design of photonic structures," Nat. Photonics **15**, 77–90 (2021).
3. D. Mengu, M. S. S. Rahman, Y. Luo, J. Li, O. Kulce, and A. Ozcan, "At the intersection of optics and deep learning: statistical inference, computing, and inverse design," Adv. Opt. Photon., AOP **14**, 209–290 (2022).
4. D. D. El-Mosalmy, M. F. O. Hameed, N. F. F. Areed, and S. S. A. Obayya, "Novel neural network based optimization approach for photonic devices," Opt Quant Electron **46**, 439–453 (2014).
5. D. Liu, Y. Tan, E. Khoram, and Z. Yu, "Training Deep Neural Networks for the Inverse Design of Nanophotonic Structures," ACS Photonics **5**, 1365–1369 (2018).
6. Z. Liu, D. Zhu, S. P. Rodrigues, K.-T. Lee, and W. Cai, "Generative Model for the Inverse Design of Metasurfaces," Nano Lett. **18**, 6570–6576 (2018).
7. S. So and J. Rho, "Designing nanophotonic structures using conditional deep convolutional generative adversarial networks," Nanophotonics **8**, 1255–1261 (2019).
8. G. Côté, J.-F. Lalonde, and S. Thibault, "On the Use of Deep Learning for Lens Design," in *OSA Optical Design and Fabrication 2021 (Flat Optics, Freeform, IODC, OFT) (2021), Paper 120781A* (Optica Publishing Group, 2021), p. 120781A.
9. X. Chen, D. Lin, T. Zhang, Y. Zhao, H. Liu, Y. Cui, C. Hou, J. He, and S. Liang, "Grating waveguides by machine learning for augmented reality," Appl. Opt., AO **62**, 2924–2935 (2023).
10. N. H. Farhat, D. Psaltis, A. Prata, and E. Paek, "Optical implementation of the Hopfield model," Appl. Opt., AO **24**, 1469–1475 (1985).
11. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljačić, "Deep learning with coherent nanophotonic circuits," Nature Photonics **11**, 441–446 (2017).
12. N. C. Harris, J. Carolan, D. Bunandar, M. Prabhu, M. Hochberg, T. Baehr-Jones, M. L. Fanto, A. M. Smith, C. C. Tison, P. M. Alsing, and D. Englund, "Linear programmable nanophotonic processors," Optica, OPTICA **5**, 1623–1631 (2018).
13. R. Hamerly, L. Bernstein, A. Sludds, M. Soljačić, and D. Englund, "Large-Scale Optical Neural Networks Based on Photoelectric Multiplication," Phys. Rev. X **9**, 021032 (2019).
14. X. Xu, M. Tan, B. Corcoran, J. Wu, A. Boes, T. G. Nguyen, S. T. Chu, B. E. Little, D. G. Hicks, R. Morandotti, A. Mitchell, and D. J. Moss, "11 TOPS photonic convolutional accelerator for optical neural networks," Nature **589**, 44–51 (2021).
15. J. W. Goodman, *Introduction to Fourier Optics* (Roberts and Company Publishers, 2005).

16. P. L. McMahon, "The physics of optical computing," Nat Rev Phys **5**, 717–734 (2023).

17. J. Hu, D. Mengu, D. C. Tzarouchis, B. Edwards, N. Engheta, and A. Ozcan, "Diffractive optical computing in free space," Nat Commun **15**, 1525 (2024).

18. E. Hasman, N. Davidson, Y. Danziger, and A. A. Friesem, "Diffractive optics: Design, realization, and applications," Fiber and Integrated Optics **16**, 1–25 (1997).

19. L. G. Neto, P. S. P. Cardona, G. A. Cirino, R. D. Mansano, and P. B. Verdonck, "Design, fabrication, and characterization of a full complex-amplitude modulation diffractive optical element," JM3.1 **2**, 96–104 (2003).

20. L. Liu, X. Zhang, M. Kenney, X. Su, N. Xu, C. Ouyang, Y. Shi, J. Han, W. Zhang, and S. Zhang, "Broadband Metasurfaces with Simultaneous Control of Phase and Amplitude," Advanced Materials **26**, 5031–5036 (2014).

21. D. Lin, P. Fan, E. Hasman, and M. L. Brongersma, "Dielectric gradient metasurface optical elements," Science **345**, 298–302 (2014).

22. N. A. Rubin, Z. Shi, and F. Capasso, "Polarization in diffractive optics and metasurfaces," Adv. Opt. Photon., AOP **13**, 836–970 (2021).

23. A. C. Overvig, S. Shrestha, S. C. Malek, M. Lu, A. Stein, C. Zheng, and N. Yu, "Dielectric metasurfaces for complete and independent control of the optical amplitude and phase," Light Sci Appl **8**, 92 (2019).

24. S. K. Saha, D. Wang, V. H. Nguyen, Y. Chang, J. S. Oakdale, and S.-C. Chen, "Scalable submicrometer additive manufacturing," Science **366**, 105–109 (2019).

25. C.-F. Pan, H. Wang, H. Wang, P. N. S, Q. Ruan, S. Wredh, Y. Ke, J. Y. E. Chan, W. Zhang, C.-W. Qiu, and J. K. W. Yang, "3D-printed multilayer structures for high–numerical aperture achromatic metalenses," Science Advances **9**, eadj9262 (2023).

26. X. Luo, "Subwavelength Artificial Structures: Opening a New Era for Engineering Optics," Advanced Materials **31**, 1804680 (2019).

27. Q. Wei, L. Huang, T. Zentgraf, and Y. Wang, "Optical wavefront shaping based on functional metasurfaces," Nanophotonics **9**, 987–1002 (2020).

28. A. Leitis, A. Heßler, S. Wahl, M. Wuttig, T. Taubner, A. Tittl, and H. Altug, "All-Dielectric Programmable Huygens' Metasurfaces," Advanced Functional Materials **30**, 1910259 (2020).

29. X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, "All-optical machine learning using diffractive deep neural networks," Science **361**, 1004–1008 (2018).

30. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**, 436–444 (2015).

31. D. Mengu, Y. Luo, Y. Rivenson, and A. Ozcan, "Analysis of Diffractive Optical Neural Networks and Their Integration With Electronic Neural Networks," IEEE Journal of Selected Topics in Quantum Electronics **26**, 1–14 (2020).

32. O. Kulce, D. Mengu, Y. Rivenson, and A. Ozcan, "All-optical information-processing capacity of diffractive surfaces," Light: Science & Applications **10**, 25 (2021).

33. O. Kulce, D. Mengu, Y. Rivenson, and A. Ozcan, "All-optical synthesis of an arbitrary linear transformation using diffractive surfaces," Light Sci Appl **10**, 196 (2021).

34. D. Mengu and A. Ozcan, "All-Optical Phase Recovery: Diffractive Computing for Quantitative Phase Imaging," Advanced Optical Materials **10**, 2200281 (2022).

35. B. Bai, H. Wei, X. Yang, T. Gan, D. Mengu, M. Jarrahi, and A. Ozcan, "Data-Class-Specific All-Optical Transformations and Encryption," Advanced Materials **35**, 2212091 (2023).

36. C.-Y. Shen, J. Li, T. Gan, Y. Li, M. Jarrahi, and A. Ozcan, "All-optical phase conjugation using diffractive wavefront processing," Nat Commun **15**, 4989 (2024).

37. J. Li, Y. Li, T. Gan, C.-Y. Shen, M. Jarrahi, and A. Ozcan, "All-optical complex field imaging using diffractive processors," Light Sci Appl **13**, 120 (2024).

38. X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, "All-optical machine learning using diffractive deep neural networks," Science **361**, 1004–1008 (2018).

39. J. Li, D. Mengu, Y. Luo, Y. Rivenson, and A. Ozcan, "Class-specific differential detection in diffractive optical neural networks improves inference accuracy," AP **1**, 046001 (2019).

40. M. S. S. Rahman, J. Li, D. Mengu, Y. Rivenson, and A. Ozcan, "Ensemble learning of diffractive optical networks," Light: Science & Applications **10**, 14 (2021).

41. J. Li, D. Mengu, N. T. Yardimci, Y. Luo, X. Li, M. Veli, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Spectrally encoded single-pixel machine vision using diffractive networks," Science Advances **7**, eabd7690 (2021).

42. M. S. S. Rahman and A. Ozcan, "Computer-Free, All-Optical Reconstruction of Holograms Using Diffractive Networks," ACS Photonics **8**, 3375–3384 (2021).

43. Y. Luo, Y. Zhao, J. Li, E. Çetintaş, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Computational imaging without a computer: seeing through random diffusers at the speed of light," eLight **2**, 4 (2022).

44. M. S. S. Rahman and A. Ozcan, "Time-Lapse Image Classification Using a Diffractive Neural Network," Advanced Intelligent Systems **5**, 2200387 (2023).

45. B. Bai, Y. Luo, T. Gan, J. Hu, Y. Li, Y. Zhao, D. Mengu, M. Jarrahi, and A. Ozcan, "To image, or not to image: class-specific diffractive cameras with all-optical erasure of undesired objects," eLight **2**, 14 (2022).

46. D. Mengu, A. Tabassum, M. Jarrahi, and A. Ozcan, "Snapshot multispectral imaging using a diffractive optical network," Light Sci Appl **12**, 86 (2023).

47. J. Li, T. Gan, B. Bai, Y. Luo, M. Jarrahi, and A. Ozcan, "Massively parallel universal linear transformations using a wavelength-multiplexed diffractive optical network," AP **5**, 016003 (2023).

48. J. Li, X. Li, N. T. Yardimci, J. Hu, Y. Li, J. Chen, Y.-C. Hung, M. Jarrahi, and A. Ozcan, "Rapid sensing of hidden objects and defects using a single-pixel diffractive terahertz sensor," Nat Commun **14**, 6791 (2023).

49. C.-Y. Shen, J. Li, D. Mengu, and A. Ozcan, "Multispectral Quantitative Phase Imaging Using a Diffractive Optical Network," Advanced Intelligent Systems 2300300 (2023).

50. B. Bai, X. Yang, T. Gan, J. Li, D. Mengu, M. Jarrahi, and A. Ozcan, "Pyramid diffractive optical networks for unidirectional magnification and demagnification," (2023).

51. J. Li, T. Gan, Y. Zhao, B. Bai, C.-Y. Shen, S. Sun, M. Jarrahi, and A. Ozcan, "Unidirectional imaging using deep learning–designed materials," Sci. Adv. **9**, eadg1505 (2023).

52. B. Bai, R. Lee, Y. Li, T. Gan, Y. Wang, M. Jarrahi, and A. Ozcan, "Information-hiding cameras: Optical concealment of object information into ordinary images," Science Advances **10**, eadn9420 (2024).

53. Y. Luo, D. Mengu, N. T. Yardimci, Y. Rivenson, M. Veli, M. Jarrahi, and A. Ozcan, "Design of task-specific optical systems using broadband diffractive neural networks," Light: Science & Applications **8**, 112 (2019).

54. M. Veli, D. Mengu, N. T. Yardimci, Y. Luo, J. Li, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Terahertz pulse shaping using diffractive surfaces," Nature Communications **12**, 37 (2021).

55. J. Li, T. Gan, B. Bai, Y. Luo, M. Jarrahi, and A. Ozcan, "Massively parallel universal linear transformations using a wavelength-multiplexed diffractive optical network," AP **5**, 016003 (2023).

56. M. S. S. Rahman, X. Yang, J. Li, B. Bai, and A. Ozcan, "Universal linear intensity transformations using spatially incoherent diffractive processors," Light Sci Appl **12**, 195 (2023).

57. X. Yang, M. S. S. Rahman, B. Bai, J. Li, and A. Ozcan, "Complex-valued universal linear transformations and image encryption using spatially incoherent diffractive networks," APN **3**, 016010 (2024).

58. Y. Li, J. Li, Y. Zhao, T. Gan, J. Hu, M. Jarrahi, and A. Ozcan, "Universal Polarization Transformations: Spatial Programming of Polarization Scattering Matrices Using A Deep Learning-Designed Diffractive Polarization Transformer," Advanced Materials 2303395 (2023).

59. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Commun. ACM **60**, 84–90 (2017).

60. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 770–778.

61. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 779–788.

62. R. Collobert and J. Weston, "A unified architecture for natural language processing: deep neural networks with multitask learning," in *Proceedings of the 25th International Conference on Machine Learning*, ICML '08 (Association for Computing Machinery, 2008), pp. 160–167.

63. T. N. Sainath, A. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (2013), pp. 8614–8618.

64. H. Y. Xiong, B. Alipanahi, L. J. Lee, H. Bretschneider, D. Merico, R. K. C. Yuen, Y. Hua, S. Gueroussov, H. S. Najafabadi, T. R. Hughes, Q. Morris, Y. Barash, A. R. Krainer, N. Jojic, S. W. Scherer, B. J. Blencowe, and B. J. Frey, "The human splicing code reveals new insights into the genetic determinants of disease," Science **347**, 1254806 (2015).

65. Y. Rivenson, Z. Göröcs, H. Günaydin, Y. Zhang, H. Wang, and A. Ozcan, "Deep learning microscopy," Optica, OPTICA **4**, 1437–1443 (2017).

66. H. Wang, Y. Rivenson, Y. Jin, Z. Wei, R. Gao, H. Günaydın, L. A. Bentolila, C. Kural, and A. Ozcan, "Deep learning enables cross-modality super-resolution in fluorescence microscopy," Nature Methods **16**, 103–110 (2019).

67. Y. Wu, Y. Rivenson, Y. Zhang, Z. Wei, H. Günaydin, X. Lin, and A. Ozcan, "Extended depth-of-field in holographic imaging using deep-learning-based autofocusing and phase recovery," Optica, OPTICA **5**, 704–710 (2018).

68. Y. Rivenson, Y. Wu, and A. Ozcan, "Deep learning in holography and coherent imaging," Light: Science & Applications **8**, 85 (2019).

69. G. Barbastathis, A. Ozcan, and G. Situ, "On the use of deep learning for computational imaging," Optica, OPTICA **6**, 921–943 (2019).

70. Z. S. Ballard, H.-A. Joung, A. Goncharov, J. Liang, K. Nugroho, D. Di Carlo, O. B. Garner, and A. Ozcan, "Deep learning-enabled point-of-care sensing using multiplexed paper-based sensors," npj Digital Medicine **3**, 1–8 (2020).

71. P. P. Shinde and S. Shah, "A Review of Machine Learning and Deep Learning Applications," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)* (2018), pp. 1–6.

72. D. Psaltis, D. Brady, X.-G. Gu, and S. Lin, "Holography in artificial neural networks," Nature **343**, 325 (1990).

73. A. N. Tait, T. F. de Lima, E. Zhou, A. X. Wu, M. A. Nahmias, B. J. Shastri, and P. R. Prucnal, "Neuromorphic photonic networks using silicon photonic weight banks," Scientific Reports **7**, 7430 (2017).

74. J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein, "Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification," Scientific Reports **8**, (2018).

75. J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, and D. Brunner, "Reinforcement learning in a large-scale photonic recurrent neural network," Optica, OPTICA **5**, 756–760 (2018).

76. T. W. Hughes, M. Minkov, Y. Shi, and S. Fan, "Training of photonic neural networks through in situ backpropagation and gradient measurement," Optica, OPTICA **5**, 864–871 (2018).

77. T. W. Hughes, I. A. D. Williamson, M. Minkov, and S. Fan, "Wave physics as an analog recurrent neural network," Science Advances **5**, eaay6946 (2019).

78. T. Yan, J. Wu, T. Zhou, H. Xie, F. Xu, J. Fan, L. Fang, X. Lin, and Q. Dai, "Fourier-space Diffractive Deep Neural Network," Phys. Rev. Lett. **123**, 023901 (2019).

79. J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, and W. H. P. Pernice, "All-optical spiking neurosynaptic networks with self-learning capabilities," Nature **569**, 208 (2019).

80. D. Mengu, Y. Luo, Y. Rivenson, and A. Ozcan, "Analysis of Diffractive Optical Neural Networks and Their Integration With Electronic Neural Networks," IEEE Journal of Selected Topics in Quantum Electronics **26**, 1–14 (2020).

81. H. Dou, Y. Deng, Y. Deng, T. Yan, T. Yan, H. Wu, H. Wu, X. Lin, X. Lin, X. Lin, X. Lin, Q. Dai, Q. Dai, Q. Dai, and Q. Dai, "Residual D^2NN: training diffractive deep neural networks via learnable light shortcuts," Opt. Lett., OL **45**, 2688–2691 (2020).

82. J. R. Ong, C. C. Ooi, T. Y. L. Ang, S. T. Lim, and C. E. Png, "Photonic Convolutional Neural Networks Using Integrated Diffractive Optics," IEEE Journal of Selected Topics in Quantum Electronics **26**, 1–8 (2020).

83. S. Pai, I. A. D. Williamson, T. W. Hughes, M. Minkov, O. Solgaard, S. Fan, and D. A. B. Miller, "Parallel Programming of an Arbitrary Feedforward Photonic Network," IEEE Journal of Selected Topics in Quantum Electronics **26**, 1–13 (2020).

84. D. Mengu, Y. Zhao, N. T. Yardimci, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Misalignment resilient diffractive optical networks," Nanophotonics **9**, 4207–4219 (2020).

85. J. Li, D. Mengu, N. T. Yardimci, Y. Luo, X. Li, M. Veli, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Spectrally encoded single-pixel machine vision using diffractive networks," Science Advances **7**, eabd7690 (2021).

86. X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, "Supplementary Materials of All-optical machine learning using diffractive deep neural networks," Science **361**, 1004–1008 (2018).

87. "CIFAR-10 and CIFAR-100 datasets," https://www.cs.toronto.edu/~kriz/cifar.html.

88. R. Vilalta and Y. Drissi, "A Perspective View and Survey of Meta-Learning," Artificial Intelligence Review **18**, 77–95 (2002).

89. O. Sagi and L. Rokach, "Ensemble learning: A survey," WIREs Data Mining and Knowledge Discovery **8**, e1249 (2018).

90. D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," arXiv:1409.0473 [cs, stat] (2016).

91. S. Suthaharan, "Support Vector Machine," in *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*, S. Suthaharan, ed., Integrated Series in Information Systems (Springer US, 2016), pp. 207–235.

92. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE **86**, 2278–2324 (1998).

93. V. A. Soifer, "Diffractive nanophotonics and advanced information technologies," Her. Russ. Acad. Sci. **84**, 9–20 (2014).

94. F. Niesler and M. Hermatschweiler, "Two-Photon Polymerization — A Versatile Microfabrication Tool," Laser Technik Journal **12**, 44–47 (2015).

95. Q. Geng, D. Wang, P. Chen, and S.-C. Chen, "Ultrafast multi-focus 3-D nano-fabrication based on two-photon polymerization," Nature Communications **10**, 2179 (2019).

96. P. Yao, G. J. Schneider, B. Miao, J. Murakowski, D. W. Prather, E. D. Wetzel, and D. J. O'Brien, "Multilayer three-dimensional photolithography with traditional planar method," Appl. Phys. Lett. **85**, 3920–3922 (2004).

97. C. Zhang, H. Subbaraman, Q. Li, Z. Pan, J. G. Ok, T. Ling, C.-J. Chung, X. Zhang, X. Lin, R. T. Chen, and L. J. Guo, "Printed photonic elements: nanoimprinting and beyond," J. Mater. Chem. C **4**, 5133–5153 (2016).

98. I.-T. Chen, E. Schappell, X. Zhang, and C.-H. Chang, "Continuous roll-to-roll patterning of three-dimensional periodic nanostructures," Microsystems & Nanoengineering **6**, 1–11 (2020).

99. W. T. Chen, A. Y. Zhu, and F. Capasso, "Flat optics with dispersion-engineered metasurfaces," Nature Reviews Materials **5**, 604–620 (2020).

100. J. Engelberg and U. Levy, "The advantages of metalenses over diffractive lenses," Nature Communications **11**, 1991 (2020).

101. G. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," arXiv:1503.02531 [cs, stat] (2015).

102. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," (2014).

103. M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," Science **349**, 255–260 (2015).

104. W. Knight, "AI Can Do Great Things—if It Doesn't Burn the Planet," Wired (n.d.).

105. R. Toews, "Deep Learning's Carbon Emissions Problem," https://www.forbes.com/sites/robtoews/2020/06/17/deep-learnings-climate-change-problem/.

106. R. Hamerly, "The Future of Deep Learning Is Photonic: Reducing the energy needs of neural networks might require computing with light," IEEE Spectrum **58**, 30–47 (2021).

107. J. Li, Y.-C. Hung, O. Kulce, D. Mengu, and A. Ozcan, "Polarization multiplexed diffractive computing: all-optical implementation of a group of linear transformations through a polarization-encoded diffractive network," Light Sci Appl **11**, 153 (2022).

108. C. Qian, X. Lin, X. Lin, J. Xu, Y. Sun, E. Li, B. Zhang, and H. Chen, "Performing optical logic operations by a diffractive neural network," Light Sci Appl **9**, 59 (2020).

109. Z. Huang, P. Wang, J. Liu, W. Xiong, Y. He, J. Xiao, H. Ye, Y. Li, S. Chen, and D. Fan, "All-Optical Signal Processing of Vortex Beams with Diffractive Deep Neural Networks," Phys. Rev. Applied **15**, 014037 (2021).

110. J. Shi, J. Shi, L. Zhou, L. Zhou, T. Liu, T. Liu, C. Hu, C. Hu, C. Hu, K. Liu, K. Liu, J. Luo, J. Luo, H. Wang, C. Xie, X. Zhang, and X. Zhang, "Multiple-view D^2NNs array: realizing robust 3D object recognition," Opt. Lett., OL **46**, 3388–3391 (2021).

111. T. Zhou, X. Lin, J. Wu, Y. Chen, H. Xie, Y. Li, J. Fan, H. Wu, L. Fang, and Q. Dai, "Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit," Nat. Photonics **15**, 367–373 (2021).

112. Y. Luo, Y. Zhao, J. Li, E. Çetintaş, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Computational imaging without a computer: seeing through random diffusers at the speed of light," eLight **2**, 4 (2022).

113. Ç. Işıl, D. Mengu, Y. Zhao, A. Tabassum, J. Li, Y. Luo, M. Jarrahi, and A. Ozcan, "Super-resolution image display using diffractive decoders," Science Advances **8**, eadd3433 (2022).

114. Y. Luo, D. Mengu, and A. Ozcan, "Cascadable all-optical NAND gates using diffractive networks," Sci Rep **12**, 7121 (2022).

115. Z. Huang, Y. He, P. Wang, W. Xiong, H. Wu, J. Liu, H. Ye, Y. Li, D. Fan, and S. Chen, "Orbital angular momentum deep multiplexing holography via an optical diffractive neural network," Opt. Express, OE **30**, 5569–5584 (2022).

116. J. Zhang, J. Zhang, Z. Ye, J. Yin, L. Lang, L. Lang, S. Jiao, and S. Jiao, "Polarized deep diffractive neural network for sorting, generation, multiplexing, and de-multiplexing of orbital angular momentum modes," Opt. Express, OE **30**, 26728–26741 (2022).

117. P. Buske, A. Völl, M. Eisebitt, J. Stollenwerk, J. Stollenwerk, and C. Holly, "Advanced beam shaping for laser materials processing based on diffractive neural networks," Opt. Express, OE **30**, 22798–22816 (2022).

118. C. Liu, Q. Ma, Z. J. Luo, Q. R. Hong, Q. Xiao, H. C. Zhang, L. Miao, W. M. Yu, Q. Cheng, L. Li, and T. J. Cui, "A programmable diffractive deep neural network based on a digital-coding metasurface array," Nat Electron **5**, 113–122 (2022).

119. X. Luo, Y. Hu, X. Ou, X. Li, J. Lai, N. Liu, X. Cheng, A. Pan, and H. Duan, "Metasurface-enabled on-chip multiplexed diffractive neural networks in the visible," Light Sci Appl **11**, 158 (2022).

120. J. Shi, J. Shi, Y. Chen, X. Zhang, and X. Zhang, "Broad-spectrum diffractive network via ensemble learning," Opt. Lett., OL **47**, 605–608 (2022).

121. M. S. S. Rahman, J. Li, D. Mengu, Y. Rivenson, and A. Ozcan, "Ensemble learning of diffractive optical networks," Light: Science & Applications **10**, 14 (2021).

122. W. Bishara, T.-W. Su, A. F. Coskun, and A. Ozcan, "Lensfree on-chip microscopy over a wide field-of-view using pixel super-resolution," Opt. Express, OE **18**, 11181–11191 (2010).

123. A. Greenbaum, W. Luo, B. Khademhosseinieh, T.-W. Su, A. F. Coskun, and A. Ozcan, "Increased space-bandwidth product in pixel super-resolved lensfree on-chip microscopy," Sci Rep **3**, 1717 (2013).

124. A. Greenbaum, A. Feizi, N. Akbari, and A. Ozcan, "Wide-field computational color imaging using pixel super-resolved on-chip microscopy," Opt. Express, OE **21**, 12469–12483 (2013).

125. R. Dahl, M. Norouzi, and J. Shlens, "Pixel Recursive Super Resolution," in (2017), pp. 5439–5448.

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

127. W. Luo, Y. Zhang, A. Feizi, Z. Göröcs, and A. Ozcan, "Pixel super-resolution using wavelength scanning," Light Sci Appl **5**, e16060–e16060 (2016).

128. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: a system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI'16 (USENIX Association, 2016), pp. 265–283.

129. R. K. Kostuk, *Holography: Principles and Applications* (CRC Press, 2019).

130. D. Gabor, "A New Microscopic Principle," Nature **161**, 777–778 (1948).

131. E. N. Leith and J. Upatnieks, "Wavefront Reconstruction with Continuous-Tone Objects*," J. Opt. Soc. Am., JOSA **53**, 1377–1381 (1963).

132. A. Ozcan and E. McLeod, "Lensless Imaging and Sensing," Annual Review of Biomedical Engineering **18**, 77–102 (2016).

133. J. R. Fienup, "Phase retrieval algorithms: a comparison," Appl. Opt., AO **21**, 2758–2769 (1982).

134. M. R. Teague, "Deterministic phase retrieval: a Green's function solution," J. Opt. Soc. Am., JOSA **73**, 1434–1441 (1983).

135. N. Streibl, "Phase imaging by the transport equation of intensity," Optics Communications **49**, 6–10 (1984).

136. T. Latychevskaia and H.-W. Fink, "Solution to the Twin Image Problem in Holography," Phys. Rev. Lett. **98**, 233901 (2007).

137. L. Waller, L. Tian, and G. Barbastathis, "Transport of Intensity phase-amplitude imaging with higher order intensity derivatives," Opt. Express, OE **18**, 12552–12561 (2010).

138. A. Greenbaum and A. Ozcan, "Maskless imaging of dense samples using pixel super-resolution based multi-height lensfree on-chip microscopy," Opt. Express, OE **20**, 3129–3143 (2012).

139. T. Latychevskaia and H.-W. Fink, "Practical algorithms for simulation and reconstruction of digital in-line holograms," Appl. Opt., AO **54**, 2424–2434 (2015).

140. Y. Rivenson, Y. Wu, H. Wang, Y. Zhang, A. Feizi, and A. Ozcan, "Sparsity-based multi-height phase recovery in holographic microscopy," Scientific Reports **6**, 37862 (2016).

141. Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng, and A. Ozcan, "Phase recovery and holographic image reconstruction using deep learning in neural networks," Light: Science & Applications **7**, 17141–17141 (2018).

142. H. Wang, M. Lyu, and G. Situ, "eHoloNet: a learning-based end-to-end approach for in-line digital holographic reconstruction," Opt. Express, OE **26**, 22603–22614 (2018).

143. A. Goy, K. Arthur, S. Li, and G. Barbastathis, "Low Photon Count Phase Retrieval Using Deep Learning," Phys. Rev. Lett. **121**, 243902 (2018).

144. Z. Ren, Z. Xu, and E. Y. M. Lam, "End-to-end deep learning framework for digital holographic reconstruction," AP **1**, 016004 (2019).

145. G. Barbastathis, A. Ozcan, and G. Situ, "On the use of deep learning for computational imaging," Optica, OPTICA **6**, 921–943 (2019).

146. Y. Rivenson, Y. Wu, and A. Ozcan, "Deep learning in holography and coherent imaging," Light Sci Appl **8**, 85 (2019).

147. K. Wang, J. Dou, Q. Kemao, Q. Kemao, J. Di, J. Di, and J. Zhao, "Y-Net: a one-to-two deep learning framework for digital holographic reconstruction," Opt. Lett., OL **44**, 4765–4768 (2019).

148. T. Zeng, H. K.-H. So, and E. Y. Lam, "RedCap: residual encoder-decoder capsule network for holographic image reconstruction," Opt. Express, OE **28**, 4876–4887 (2020).

149. M. Deng, S. Li, A. Goy, I. Kang, and G. Barbastathis, "Learning to synthesize: robust phase retrieval at low photon counts," Light Sci Appl **9**, 36 (2020).

150. L. Huang, T. Liu, X. Yang, Y. Luo, Y. Rivenson, and A. Ozcan, "Holographic Image Reconstruction with Phase Recovery and Autofocusing Using Recurrent Neural Networks," ACS Photonics **8**, 1763–1774 (2021).

151. Y. Wu, Y. Rivenson, Y. Zhang, Z. Wei, H. Günaydin, X. Lin, and A. Ozcan, "Extended depth-of-field in holographic imaging using deep-learning-based autofocusing and phase recovery," Optica, OPTICA **5**, 704–710 (2018).

152. Y. Wu, Y. Luo, G. Chaudhari, Y. Rivenson, A. Calis, K. de Haan, and A. Ozcan, "Bright-field holography: cross-modality deep learning enables snapshot 3D imaging with bright-field contrast using a single hologram," Light Sci Appl **8**, 25 (2019).

153. Y. Rivenson, T. Liu, Z. Wei, Y. Zhang, K. de Haan, and A. Ozcan, "PhaseStain: the digital staining of label-free quantitative phase microscopy images using deep learning," Light Sci Appl **8**, 23 (2019).

154. A. Greenbaum, W. Luo, T.-W. Su, Z. Göröcs, L. Xue, S. O. Isikman, A. F. Coskun, O. Mudanyali, and A. Ozcan, "Imaging without lenses: achievements and remaining challenges of wide-field on-chip microscopy," Nat Methods **9**, 889–895 (2012).

155. O. Mudanyali, D. Tseng, C. Oh, S. O. Isikman, I. Sencan, W. Bishara, C. Oztoprak, S. Seo, B. Khademhosseini, and A. Ozcan, "Compact, Light-weight and Cost-effective Microscope based on Lensless Incoherent Holography for Telemedicine Applications," Lab Chip **10**, 1417–1428 (2010).

156. H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," arXiv:1708.07747 [cs, stat] (2017).

157. D. Mengu, Y. Zhao, N. T. Yardimci, Y. Rivenson, M. Jarrahi, and A. Ozcan, "Misalignment resilient diffractive optical networks," Nanophotonics **1**, (2020).

158. D. Mengu, Y. Rivenson, and A. Ozcan, "Scale-, Shift-, and Rotation-Invariant Diffractive Optical Networks," ACS Photonics **8**, 324–334 (2021).

159. K. Du, I. Wathuthanthri, and C.-H. Choi, "The Rise of Scalable Micro/Nanopatterning," Micromachines (Basel) **8**, 275 (2017).

160. K. Cooper, "Scalable Nanomanufacturing—A Review," Micromachines **8**, 20 (2017).

161. Y. Huang, M. C. Leu, J. Mazumder, and A. Donmez, "Additive Manufacturing: Current State, Future Potential, Gaps and Needs, and Recommendations," Journal of Manufacturing Science and Engineering **137**, (2015).

162. D. O'Brien, G. Parry, and P. Stavrinou, "Optical hotspots speed up wireless communication," Nature Photon **1**, 245–247 (2007).

163. P. Yang, Y. Xiao, M. Xiao, and S. Li, "6G Wireless Communications: Vision and Potential Techniques," IEEE Network **33**, 70–75 (2019).

164. Z. Xu and B. M. Sadler, "Ultraviolet Communications: Potential and State-Of-The-Art," IEEE Communications Magazine **46**, 67–73 (2008).

165. X. He, E. Xie, M. S. Islim, A. A. Purwita, J. J. D. McKendry, E. Gu, H. Haas, and M. D. Dawson, "1 Gbps free-space deep-ultraviolet communications based on III-nitride micro-LEDs emitting at 262 nm," Photon. Res., PRJ **7**, B41–B47 (2019).

166. C. H. Kang, I. Dursun, G. Liu, L. Sinatra, X. Sun, M. Kong, J. Pan, P. Maity, E.-N. Ooi, T. K. Ng, O. F. Mohammed, O. M. Bakr, and B. S. Ooi, "High-speed colour-converting photodetector with all-inorganic CsPbBr3 perovskite nanocrystals for ultraviolet light communication," Light Sci Appl **8**, 94 (2019).

167. S. Arnon and D. Kedar, "Non-line-of-sight underwater optical wireless communication network," J. Opt. Soc. Am. A, JOSAA **26**, 530–539 (2009).

168. L. Wang, Z. Xu, and B. M. Sadler, "An approximate closed-form link loss model for non-line-of-sight ultraviolet communication in noncoplanar geometry," Opt. Lett., OL **36**, 1224–1226 (2011).

169. H. Xiao, Y. Zuo, J. Wu, Y. Li, and J. Lin, "Non-line-of-sight ultraviolet single-scatter propagation model in random turbulent medium," Opt. Lett., OL **38**, 3366–3369 (2013).

170. Z. Cao, X. Zhang, G. Osnabrugge, J. Li, I. M. Vellekoop, and A. M. J. Koonen, "Reconfigurable beam system for non-line-of-sight free-space optical communication," Light Sci Appl **8**, 69 (2019).

171. Z. Liu, Y. Huang, H. Liu, and X. Chen, "Non-line-of-sight optical communication based on orbital angular momentum," Opt. Lett., OL **46**, 5112–5115 (2021).

172. A. Velten, T. Willwacher, O. Gupta, A. Veeraraghavan, M. G. Bawendi, and R. Raskar, "Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging," Nat Commun **3**, 745 (2012).

173. G. Gariepy, F. Tonolini, R. Henderson, J. Leach, and D. Faccio, "Detection and tracking of moving objects hidden from view," Nature Photon **10**, 23–26 (2016).

174. M. O'Toole, D. B. Lindell, and G. Wetzstein, "Confocal non-line-of-sight imaging based on the light-cone transform," Nature **555**, 338–341 (2018).

175. C. Saunders, J. Murray-Bruce, and V. K. Goyal, "Computational periscopy with an ordinary digital camera," Nature **565**, 472–475 (2019).

176. T. Maeda, Y. Wang, R. Raskar, and A. Kadambi, "Thermal Non-Line-of-Sight Imaging," in *2019 IEEE International Conference on Computational Photography (ICCP)* (2019), pp. 1–11.

177. F. Heide, M. O'Toole, K. Zang, D. B. Lindell, S. Diamond, and G. Wetzstein, "Non-line-of-sight Imaging with Partial Occluders and Surface Normals," ACM Trans. Graph. **38**, 22:1-22:10 (2019).

178. D. B. Lindell, G. Wetzstein, and M. O'Toole, "Wave-based non-line-of-sight imaging using fast f-k migration," ACM Trans. Graph. **38**, 116:1-116:13 (2019).

179. J. Boger-Lombard and O. Katz, "Passive optical time-of-flight for non line-of-sight localization," Nat Commun **10**, 3343 (2019).

180. X. Liu, I. Guillén, M. La Manna, J. H. Nam, S. A. Reza, T. Huu Le, A. Jarabo, D. Gutierrez, and A. Velten, "Non-line-of-sight imaging using phasor-field virtual wave optics," Nature **572**, 620–623 (2019).

181. M. Kaga, T. Kushida, T. Takatani, K. Tanaka, T. Funatomi, and Y. Mukaigawa, "Thermal non-line-of-sight imaging from specular and diffuse reflections," IPSJ Transactions on Computer Vision and Applications **11**, 8 (2019).

182. C. A. Metzler, F. Heide, P. Rangarajan, M. M. Balaji, A. Viswanath, A. Veeraraghavan, and R. G. Baraniuk, "Deep-inverse correlography: towards real-time high-resolution non-line-of-sight imaging," Optica, OPTICA **7**, 63–71 (2020).

183. X. Liu, S. Bauer, and A. Velten, "Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems," Nat Commun **11**, 1645 (2020).

184. C. Wu, J. Liu, X. Huang, Z.-P. Li, C. Yu, J.-T. Ye, J. Zhang, Q. Zhang, X. Dou, V. K. Goyal, F. Xu, and J.-W. Pan, "Non–line-of-sight imaging over 1.43 km," Proceedings of the National Academy of Sciences **118**, e2024468118 (2021).

185. B. Wang, M.-Y. Zheng, J.-J. Han, X. Huang, X.-P. Xie, F. Xu, Q. Zhang, and J.-W. Pan, "Non-Line-of-Sight Imaging with Picosecond Temporal Resolution," Phys. Rev. Lett. **127**, 053602 (2021).

186. J. Maycock, C. P. McElhinney, B. M. Hennelly, T. J. Naughton, J. B. McDonald, and B. Javidi, "Reconstruction of partially occluded objects encoded in three-dimensional scenes by using digital holograms," Appl. Opt., AO **45**, 2975–2985 (2006).

187. Z. Yaqoob, D. Psaltis, M. S. Feld, and C. Yang, "Optical phase conjugation for turbidity suppression in biological samples," Nature Photon **2**, 110–115 (2008).

188. Y. Rivenson, A. Rot, S. Balber, A. Stern, and J. Rosen, "Recovery of partially occluded objects by applying compressive Fresnel holography," Opt. Lett., OL **37**, 1757–1759 (2012).

189. Y. Xiao, L. Zhou, and W. Chen, "Wavefront control through multi-layer scattering media using single-pixel detector for high-PSNR optical transmission," Optics and Lasers in Engineering **139**, 106453 (2021).

190. Y. Xiao, L. Zhou, Z. Pan, Y. Cao, and W. Chen, "Physically-secured high-fidelity free-space optical data transmission through scattering media using dynamic scaling factors," Opt. Express, OE **30**, 8186–8198 (2022).

191. Z. Pan, Y. Xiao, Y. Cao, L. Zhou, and W. Chen, "Accurate optical information transmission through thick tissues using zero-frequency modulation and single-pixel detection," Optics and Lasers in Engineering **158**, 107133 (2022).

192. S. Popoff, G. Lerosey, M. Fink, A. C. Boccara, and S. Gigan, "Image transmission through an opaque material," Nat Commun **1**, 81 (2010).

193. L. Zhu, F. Soldevila, C. Moretti, A. d'Arco, A. Boniface, X. Shao, H. B. de Aguiar, and S. Gigan, "Large field-of-view non-invasive imaging through scattering layers using fluctuating random illumination," Nat Commun **13**, 1447 (2022).

194. "MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges," http://yann.lecun.com/exdb/mnist/.

195. Y. Li, S. Cheng, Y. Xue, and L. Tian, "Displacement-agnostic coherent imaging through scatter with an interpretable deep neural network," Opt. Express, OE **29**, 2244–2257 (2021).

196. L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," Journal of machine learning research **9**, (2008).

197. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Transactions on Image Processing **13**, 600–612 (2004).